

Львівський державний університет внутрішніх справ

Тетяна МАГЕРОВСЬКА

Володимир СЕНИК

Дмитро МАГЕРОВСЬКИЙ

Проектування користувацьких інтерфейсів та комп'ютерної графіки

Навчальний посібник

Львів

2026

Рекомендовано до розміщення в електронних сервісах ЛьвДУВС
Вченою радою Львівського державного університету внутрішніх справ
(протокол від 25 листопада 2025 року № 5)

Рекомендовано до опублікування Науково-методичною радою
Національного університету «Львівська політехніка»
(протокол від 19 січня 2026 року № 1/2026)

Р е ц е н з е н т и:

Демків І. І. – доктор фізико-математичних наук, професор
(Національний університет «Львівська політехніка»);

Огірко О. І. – кандидат технічних наук, доцент
(Львівський державний університет внутрішніх справ)

Магеровська Т. В., Сеник В. В., Магеровський Д. В.

М 12 Проектування користувацьких інтерфейсів та комп'ютерної графіки :
навчальний посібник. Львівський державний університет внутрішніх справ.
Львів, 2026. 408 с.

Видання є складовою навчально-методичного забезпечення таких дисциплін, як «Інформатика», «Комп'ютерна графіка», «Вебдизайн та веброзробка» для здобувачів першого (бакалаврського) рівня вищої освіти спеціальностей: F3 «Комп'ютерні науки», F6 «Інформаційні системи та технології», B2 «Дизайн».

Також навчальний посібник можна використовувати під час підготування фахівців інших галузей знань і спеціальностей, які вивчають технології комп'ютерної графіки.

The textbook «Fundamentals of Computer Graphics and Web Design» is part of the educational and methodological support for the academic courses «Informatics», «Computer Graphics», and «Web Design and Web Development» intended for students pursuing a bachelor's degree in the following specialities: F3 «Computer Science», F6 «Information Systems and Technologies», and B2 «Design».

It may also be used in the training of professionals in other fields and specialities that study computer graphics technologies.

ЗМІСТ

ЗМІСТ	3
ВСТУП.....	7
РОЗДІЛ 1. ОСНОВИ КОМП'ЮТЕРНОЇ ГРАФІКИ ТА ВЕБДИЗАЙНУ	8
1.1. Основні поняття	8
1.2. Використання комп'ютерної графіки у вебдизайні.....	9
1.3. Основи UI/UX-дизайну	10
1.4. Принципи та сучасні тренди вебдизайну.....	21
1.5. Вплив психології на вебдизайн	25
1.6. Колір у цифровому дизайні	26
Характеристики кольору	26
Колірні моделі та простори	28
Побудова палітр і дизайн-токени.....	32
Рольові токени і шкали	35
Колір та емоції.....	37
1.7. Графічні формати для вебдизайну	38
Растрові формати.....	38
Векторні формати.....	40
3D формати	40
Мультимедійні формати	41
Анімаційні формати	42
1.8. Типи комп'ютерної графіки.....	43
За способом подання зображення.....	43
За типом тривимірності	44
За методом формування.....	44
За призначенням	44

За використанням освітлення та тіней	45
За технологіями рендерингу	45
1.9. Огляд популярних графічних редакторів	45
Програмне забезпечення растрової графіки	45
Програмне забезпечення векторної графіки	46
Програмне забезпечення 3D графіки	47
Огляд AI-інструментів у дизайні	48
РОЗДІЛ 2. ВЕКТОРНА ГРАФІКА У ВЕБДИЗАЙНІ	50
2.1. Основні поняття векторної графіки	50
Переваги та недоліки векторної графіки	50
Графічні формати векторної графіки	51
Програмне забезпечення векторної графіки	52
2.2. Основні інструменти векторних редакторів	53
Встановлення Figma	54
Робочий простір Figma	54
Інтерфейс користувача Figma	58
Примітиви	74
Фрейми	89
Перо	101
Текст	111
Заливка і градієнт	117
Межі	146
Булеві операції	151
Режим Outline: векторні контури	160
Маски	161
Плагіни	164
Стили у Figma	165
Компоненти у Figma	168

2.3. Створення іконок, логотипів та ілюстрацій	174
РОЗДІЛ 3. РАСТРОВА ГРАФІКА У ВЕБДИЗАЙНІ	178
3.1. Основні поняття растрової графіки	178
3.2. Основні інструменти растрових редакторів	186
Вступ до Paint.NET	186
Робота з файлом зображення.....	200
Шари.....	204
Основні інструменти та функції	223
Редагування зображень.....	272
Плагіни та розширення	321
РОЗДІЛ 4.ІНТЕРАКТИВНІСТЬ, ПРОТОТИПУВАННЯ ТА ОПТИМІЗАЦІЯ ВЕБДИЗАЙНУ.....	325
4.1. Основи прототипування інтерфейсів	325
4.2. Анімація у Figma	327
Принципи Smart Animate	327
Перехід між кадрами (Frames) і затримка.....	328
Компонентні стани та мікроанімації	329
Анімація при взаємодії (On Click, While Hovering, After Delay).....	330
Прототипування переходів у Figma.....	330
Експорт інтерактивних прототипів	331
4.3. Макети для вебсайтів і мобільних застосунків	331
4.4. Компоненти вебсторінок.....	337
4.5. Інтерактивність та анімація у вебдизайні	
(GIF, SVG, CSS, JavaScript)	344
GIF-Анімація	345
SVG-анімація	346
Анімація засобами CSS.....	346
Анімація за допомогою JavaScript	349
Практичні поради	351

4.6. Аналіз та оптимізація досвіду користувача	352
Інструменти для аналізу досвіду користувача Google Analytics та Hotjar	353
Тестування користувацьких інтерфейсів	355
Показники ефективності UX/UI-дизайну.....	357
Оптимізація інтерфейсу за результатами аналітики та тестування.....	358
Оптимізація графіки і handoff	359
4.7. Доступність (WCAG 2.2)	369
Контраст	370
Клавіатурна навігація.....	374
ARIA в макетах.....	374
Reduce Motion	374
Інструменти і процес.....	374
РОЗДІЛ 5. АДАПТИВНИЙ ДИЗАЙН	376
5.1. Основи адаптивного дизайну.....	376
5.2. Робота з адаптивністю у Figma	378
Auto Layout – гнучка структура елементів	378
Constraints – закріплення позиції елементів	387
5.3. Типографіка та шрифти в адаптивному дизайні	390
Основні принципи адаптивної типографіки	390
UX-Writing та контент-дизайн	396
5.4. Тестування адаптивності.....	397
5.5. Робота з онлайн-ресурсами: шрифти, іконки, шаблони	398
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	403

ВСТУП

Нині складно уявити будь-яку сферу діяльності без використання комп'ютерної графіки та вебдизайну, адже вони є основою цифрового середовища, в якому ми щодня працюємо, навчаємося, шукаємо інформацію чи просто спілкуємося. Саме завдяки цим напрямам звичні для нас інтерфейси набувають зрозумілої форми, зручності й естетичної привабливості.

Комп'ютерна графіка – це спосіб візуально передати інформацію за допомогою технологій, створюючи зображення, ілюстрації, схеми чи 3D-моделі. Вебдизайн, своєю чергою, поєднує художнє бачення та технічні інструменти для створення зручних і привабливих вебсторінок, сайтів і додатків. Обидві ці галузі тісно пов'язані з поняттями **UI (User Interface)** та **UX (User Experience)**, які визначають, наскільки інтуїтивною і простою є взаємодія користувача із цифровим продуктом.

Цей навчальний посібник має допомогти здобувачам вищої освіти опанувати і теоретичні засади, і практичні навички, необхідні для сучасного дизайнера. В ньому розглядаються основи роботи з растровою та векторною графікою, принципи побудови композиції, гармонії кольорів, типографіки, а також аналізується специфіка створення вебмакетів і прототипів.

Метою цього навчального посібника є не лише дати уявлення про основні поняття комп'ютерної графіки та вебдизайну, а й сформувати у здобувачів вищої освіти вміння застосовувати отримані знання у своїх проєктах. Після опанування матеріалу здобувач вищої освіти зможе самостійно розробляти дизайн для сайтів, презентацій чи мобільних застосунків, а також ефективно співпрацювати у команді над створенням сучасних цифрових продуктів.

РОЗДІЛ 1

ОСНОВИ КОМП'ЮТЕРНОЇ ГРАФІКИ ТА ВЕБДИЗАЙНУ

1.1. ОСНОВНІ ПОНЯТТЯ

Комп'ютерна графіка (Computer Graphics) – це розділ інформатики, який вивчає методи створення та опрацювання візуальних зображень за допомогою ІТ-технологій.

Також можна сказати, що це вид творчої діяльності, який працює із зображеннями, що створюються, перетворюються, оцифровуються, обробляються і виводяться за допомогою засобів обчислювальної техніки.

Дизайн – це результат проектування об'єктів із метою виконання обраних завдань: технічних, соціальних, споживчих тощо.

Комп'ютерний дизайн – це процес створення візуальних зображень, інтерфейсів і моделей за допомогою комп'ютерних програм. Він охоплює широкий спектр напрямів, таких як графічний дизайн, вебдизайн, 3D-моделювання, UX/UI дизайн, моушн-дизайн тощо.

Вебдизайн – це процес планування, проектування та розроблення інтерфейсу вебсайтів або вебдодатків. Він охоплює розроблення макетів, вибір кольорових схем, шрифтів, графічних елементів, інтерактивних функцій, що дають змогу користувачам зручно взаємодіяти із сайтом.

UI/UX дизайн – важлива складова створення цифрових продуктів, яка забезпечує зручність, естетику та ефективність у взаємодії користувачів із програмним забезпеченням.

Юзабіліті – це зручність і простота використання інтерфейсу. Мета юзабіліті – зменшення кількості дій, необхідних для досягнення цілей користувачів.

Інтерфейс – це засіб взаємодії між користувачем і системою, наприклад, програмою, пристроєм чи апаратним забезпеченням. Інтерфейс може бути **графічним** (GUI) із візуальними елементами, такими як кнопки та меню, або **командним** (CLI), де взаємодія здійснюється через текстові команди.

Інтерфейси також можуть охоплювати сенсорні екрани, голосове керування, нейроінтерфейси й інші форми взаємодії.

Основна мета інтерфейсу – зробити роботу із програмою зрозумілою та зручною для користувача.

UI-дизайн (User Interface Design), **або дизайн інтерфейсу користувача** – це процес створення інтерфейсів програмного забезпечення та вебдодатків із акцентом на стиль і оформлення. Тобто цим терміном описують те, як продукт виглядає: кольори, текстури, анімація, стиль механічних елементів (меню, кнопки, поля введення тощо), стиль іконок, написів.

UX-дизайн (User Experience Design) – це підхід, який охоплює не лише візуальні елементи, а й досвід користувача під час взаємодії з продуктом. UX-дизайнер вивчає поведінку та потреби користувачів для створення інтуїтивно зрозумілих і зручних у роботі інтерфейсів.

1.2. ВИКОРИСТАННЯ КОМП'ЮТЕРНОЇ ГРАФІКИ У ВЕБДИЗАЙНІ

Комп'ютерна графіка посідає ключове місце у створенні сучасного вебдизайну. Вона визначає не лише зовнішній вигляд сайту, а й те, як користувач сприймає і взаємодіє із контентом. Саме завдяки продуманим візуальним рішенням вебсторінки стають зрозумілими і зручними у користуванні.

У процесі розроблення сайтів графіка застосовується на всіх етапах – від створення логотипів і піктограм до повноцінного оформлення сторінок, рекламних банерів, ілюстрацій чи фонових зображень. Кожен із цих елементів виконує не лише декоративну, а й змістову роль: допомагає сформувати цілісний образ бренду, передати настрій і підтримати структуру інформації.

Наприклад, **логотип** є візуальним обличчям компанії й відображає її характер; **іконки** спрощують навігацію, даючи змогу швидше орієнтуватися на сайті; **банери** привертають увагу до важливих повідомлень чи акцій; **ілюстрації** створюють емоційний зв'язок із користувачем.

Окрім статичних елементів, важливу роль виконує **анімація**. Завдяки їй сторінки стають живішими: плавні переходи, рухомі кнопки чи мікроанімації допомагають користувачу інтуїтивно розуміти, як працює інтерфейс. Це робить взаємодію природнішою і зручнішою.

Комп'ютерна графіка також широко використовується під час створення **реklamних матеріалів, мультимедійних проєктів, інтерактивних презентацій і навчальних платформ**. Усе це сприяє формуванню впізнаваності бренду та підвищує рівень залучення користувачів.

Зрештою, саме продумане використання графіки у вебдизайні перетворює сайт на ефективний засіб комунікації. Візуальні елементи допомагають передати зміст швидше, ніж текст, викликають емоції та створюють довіру до продукту. Успішний дизайнер поєднує художній смак із технічними знаннями, перетворюючи графіку на інструмент, який робить інформацію доступною, а цифрове середовище – естетично довершеним.

1.3. ОСНОВИ UI/UX-ДИЗАЙНУ

Користувачі повинні отримувати позитивні емоції від використання продукту. Це може бути досягнуто через естетику, анімації, звуки або інші елементи дизайну. Також сайт або додаток мають бути структуровані так, щоб користувачі могли легко та швидко навчитися ними користуватися без додаткових інструкцій і знайти потрібну їм інформацію. Це важливо, оскільки користувачі схильні віддавати перевагу програмам, у яких зрозумілі інтерфейси та логіка роботи. Важливе значення має і комфортне використання програмних продуктів на різних пристроях, незалежно від розміру екрану чи типу пристрою.

Якщо програма має естетичний і зручний інтерфейс, користувачі також залишаються більш задоволеними і схильними продовжувати її використовувати, що може сприяти популяризації продукту.

Окрім того, добре продуманий UX-дизайн допомагає запобігати помилкам, які користувачі можуть зробити під час використання програми, та зменшить кількість звернень до служби підтримки, що також поліпшує загальне враження від продукту.

Загалом UX є важливим аспектом у розробленні програмного забезпечення, оскільки від нього залежить, наскільки успішним буде продукт на ринку та як довго користувачі залишатимуться його прихильниками.

UX/UI дизайнери – це фахівці, які займаються проєктуванням зручних і естетичних інтерфейсів цифрових продуктів, таких як вебсайти та мобільні додатки. Вони фокусуються на тому, як користувачі взаємодіють із продуктами та намагаються покращити цей досвід, застосовуючи принципи дизайну та досліджуючи поведінку користувачів.

UI-дизайнер створює зовнішній вигляд продукту: кнопки, іконки, шрифти, кольорову палітру й інші графічні елементи так, щоб вони були не лише привабливими, а й зручними для використання.

UX-дизайнер працює над тим, щоб створити логічний і зручний для користувача шлях, починаючи з першої точки взаємодії і до досягнення бажаного результату. Прикладом може бути аналіз того, як користувач проходить реєстрацію у додатку, наскільки інтуїтивно це відбувається і чи задоволений він процесом.

До основних обов'язків UX/UI дизайнерів належать:

- збір даних про потреби, бажання та проблеми цільової аудиторії;
- створення ескізів, wireframe-ів та інтерактивних прототипів для візуалізації ідеї;
- апробація дизайну реальними користувачами для виявлення проблем;
- розроблення естетично привабливих інтерфейсів із використанням кольорів, шрифтів, графіки тощо;
- взаємодія з розробниками, маркетологами й іншими учасниками проєкту для реалізації дизайну;
- пошук сценаріїв використання продукту на основі ролей користувачів.

Вебсайт можна назвати **вебзастосунком**, якщо він має не тільки статичний набір сторінок, а й включає тісну взаємодію із користувачем і реакцію на його кроки для вирішення додаткових завдань (окрім надання інформації).

Прикладом простого вебсайту є персональний блог, а вебзастосунок – це вебмагазин або чат-GTP.

Що складніший проєкт, то більше фахівців потрібно задіяти і то більше етапів роботи потрібно пройти. Залежно від розміру та призначення вебзастосунку етапи розроблення можуть змінюватися, але зазвичай вони охоплюють: аналітику, збір інформації, написання технічного завдання, пошук референсів, створення мудбордів, розроблення UX та UI-дизайну, розроблення презентації проєкту, експорт ресурсів, верстку, програмування, тестування та деплой.

Графічні дизайнери беруть активну участь у **перших шести етапах** розроблення вебдодатку, забезпечуючи естетичність і функціональність його графічних компонентів.

Розглянемо кожен етап докладніше.

Етап 1. Аналітика, збір інформації, написання технічного завдання

Перед тим, як починати будь-який проєкт у сфері UX/UI чи вебдизайну, потрібно закласти «фундамент» – провести аналітику, зібрати всю необхідну інформацію й сформулювати технічне завдання. Без цього навіть найталановитіший дизайнер ризикує створити красивий, але неефективний продукт, який не вирішує потреб користувача.

Аналітичний етап важливий для проєктів із великими бюджетами або високим рівнем ризику – наприклад, для стартапів чи корпоративних систем. Саме на цьому етапі формується розуміння, **для кого створюється продукт, які завдання він має вирішувати і як це зробити якнайкраще.**

Для отримання повної картини зазвичай виконують такі дії:

Визначають цільову аудиторію – вік, стать, соціальний статус, професію, місце проживання, звички тощо. Це допомагає зрозуміти, хто буде користуватися продуктом і які його очікування.

Проводять аналіз конкурентів. Вивчають їхні сильні сторони, помилки, стиль комунікації та дизайн, щоб знайти можливості для кращого рішення.

Оцінюють економічну складову. Прогнозують витрати, потенційний прибуток або вигоди для бізнесу.

Збирають контент. Готують тексти, графіку, фото, відео, які будуть використані на сторінках.

Формулюють функціональні вимоги. Визначають, які можливості має мати сайт або застосунок: інтеграції із сервісами, система керування контентом (CMS), структура бази даних, вимоги до хостингу та безпеки.

Створюють карту сайту (sitemap) – схему розміщення сторінок і логіку переходів між ними.

Таку роботу можуть виконувати **бізнес-аналітики, консалтингові компанії** або **UX-дизайнер**, якщо проєкт невеликий. На основі отриманих даних складається **технічне завдання (ТЗ)** – документ, що чітко описує вимоги, цілі, структуру, функції та очікувані результати майбутнього продукту.

Принципи юзабіліті, які слід враховувати під час створення інтерфейсів:

1. **Закон Якоба Нільсена** (Heuristics of Jakob Nielsen) – користувачі взаємодіють із вебсайтами й додатками на основі попереднього досвіду, отже відомі елементи мають бути передбачуваними. Наприклад, більшість користувачів очікує побачити логотип сайту у верхньому лівому куті для повернення на головну сторінку.
2. **Закон Міллера** (Miller's Law) – людина може тримати в короткочасній пам'яті в середньому 7 елементів одночасно. Отже, інтерфейс має бути простим, а інформація – структурована.
3. **Закон Фіттса** (Fitts's Law) – час, що необхідний для наведення та натискання на елемент інтерфейсу, залежить від його розміру та відстані до нього. Тому важливі елементи повинні бути великими та розташованими на доступних місцях.
4. **Закон Хікка** (Hick's Law) – що більше варіантів вибору має користувач, то більше часу йому потрібно для прийняття рішення. Це необхідно враховувати під час розроблення меню та списків.
5. **Закон Теслера** (Tesler's Law) – складність будь-якої системи не можна усунути повністю, але деякі процеси можна зробити автоматичними. Наприклад, автоматичне заповнення форм полегшує роботу користувача.

6. **Закон Погружки (Aesthetic Usability Effect)** – користувачі часто сприймають візуально привабливі інтерфейси як більш функціональні, що підкреслює важливість естетики.
7. **Принцип зворотного зв'язку** – користувачі повинні отримувати зворотний зв'язок від системи після будь-якої дії. Наприклад, підтвердження після надсилання форми або візуалізація процесу завантаження.

Етап 2. Пошук референсів, створення мудбордів

Після того як концепцію проєкту погоджено та технічне завдання готове, настає один із найтворчіших етапів роботи – пошук референсів і формування мудборду. Саме тут дизайнер починає візуально уявляти майбутній продукт.

UX/UI-дизайнер аналізує сайти, мобільні додатки та проєкти на професійних платформах, таких як **Behance**, **Dribbble**, **Awwwards** чи **Pinterest**. Його завдання – знайти приклади, які відповідають задуму, відображають потрібний настрій, колірну гаму, типографіку, стиль і характер майбутнього інтерфейсу. Водночас важливо не копіювати чужі рішення, а лише надихатися ними, відбираючи найкращі ідеї для подальшої роботи.

Зібрані матеріали – скріншоти сайтів, зразки шрифтів, відтінки, текстури, приклади інтерфейсів чи ілюстрацій – дизайнер систематизує в одну колекцію, яку називають **moodboard** (мудборд). Це своєрідна візуальна дошка настрою, що відображає атмосферу й стиль майбутнього проєкту.

Мудборд допомагає не лише дизайнеру, а й замовнику краще зрозуміти напрям, у якому рухається розробка. Після презентації клієнт може висловити свої вподобання: що йому близьке, а що не відповідає очікуванням. Такий підхід дає змогу ще на ранньому етапі узгодити бачення, уникнути непорозумінь і значно скоротити час на подальші вдосконалення.

Завдяки грамотно створеному мудборду команда отримує спільне візуальне бачення проєкту, яке буде відправною точкою для наступних кроків – побудови прототипу та розроблення дизайну інтерфейсу.

Етап 3. UX-дизайн

Після того, як зібрано інформацію, узгоджено концепцію та створено мудборд, починається один із ключових етапів – UX-дизайн. Саме на цьому етапі визначається логіка взаємодії користувача з майбутнім вебдодатком або сайтом.

Основне завдання UX-дизайнера – **зрозуміти, хто користуватиметься продуктом і як саме він діятиме у кожній ситуації**. Дизайнер продумує всі сценарії: від першого входу користувача до завершення цільової дії – наприклад, заповнення форми, оформлення замовлення чи перегляду контенту.

Щоб це реалізувати, сайт розбивають на **окремі екрани** – логічні блоки або сторінки, які відповідають певним діям користувача. Для кожного екрану створюють **схеми (wireframes)** або **ескізи (скетчі)**, на яких позначають розташування елементів інтерфейсу: кнопок, меню, форм, зображень, текстових блоків.

На цьому етапі дизайн не потребує кольорів, шрифтів чи декоративних деталей – головне, щоб була зрозуміла **структура**. Тому скетчі зазвичай виконують у **чорно-білих тонах**, використовуючи прості лінії, блоки й умовні позначки. Такий підхід дає змогу зосередитися на логіці користувацького шляху, а не на візуальних ефектах (Рис. 1).

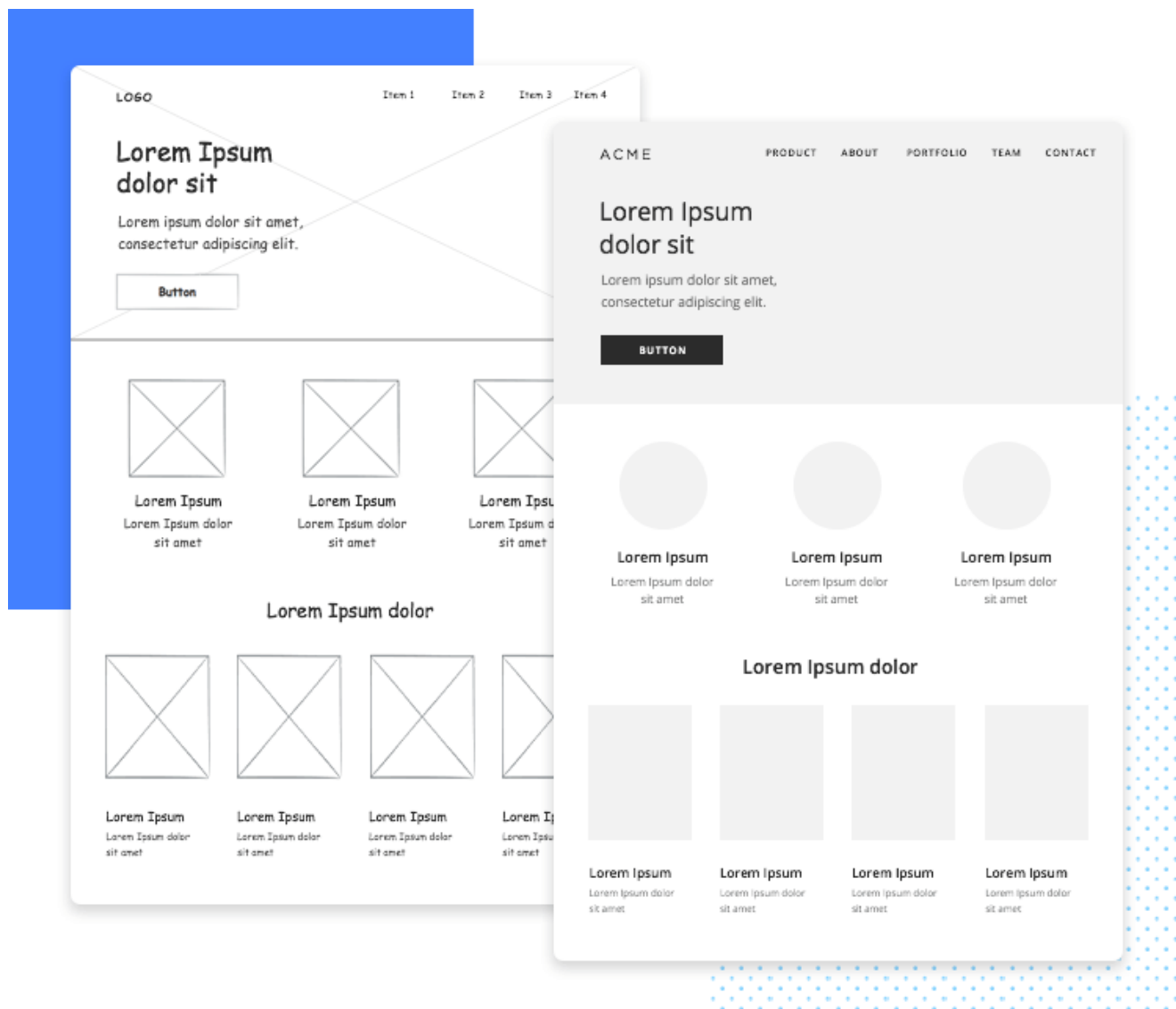


Рис. 1. Скетчі екранів

Після створення початкових начерків дизайнер переходить до **інтерактивного прототипу** – макета, який можна «проклікати», ніби справжній сайт. Це дає змогу перевірити, наскільки інтуїтивною є навігація, чи зрозуміло користувачу, куди натискати далі та чи немає зайвих кроків.

Для цього найчастіше використовують **векторні графічні редактори**, зокрема **Figma**, що дає змогу швидко з'єднувати екрани між собою, імітуючи поведінку реального вебдодатку. Прототип можна протестувати безпосередньо у браузері, отримати зворотний зв'язок від користувачів і внести корективи ще до початку фінальної розробки.

Етап 4. UI-дизайн

Коли структура сайту визначена, а всі екрани мають свої скетчі та логіку взаємодії, розпочинається наступний етап – **UI-дизайн (User Interface Design)**, тобто розроблення зовнішнього вигляду майбутнього продукту.

UI-дизайнер відповідає за **візуальну мову інтерфейсу** – те, як користувач сприйматиме сайт на рівні кольору, форми, типографіки й естетики. На цьому етапі з'являється індивідуальність проєкту: вибирається **кольорова гама, шрифти, іконки**, визначаються стилі кнопок, відступів і загальна композиція.

Спершу дизайнер створює **кілька варіантів візуального стилю**, експериментуючи з палітрами, текстурами, контрастами та поєднанням шрифтів. Потім ці варіанти презентуються замовнику, щоб спільно обрати той, який найкраще передає характер бренду чи ідею продукту.

Після затвердження обраного стилю формується **візуальна система** – єдина сітка, за якою вирівнюються всі елементи інтерфейсу. Вона забезпечує гармонійність, узгодженість і зручність сприйняття контенту. На основі цієї системи дизайнер відмальовує **ключові екрани**, де відпрацьовуються всі деталі – від кнопок і форм до навігації та підказок.

Далі на основі затвердженого стилю створюються **решта екранів**. На цьому етапі визначається остаточний вигляд сайту чи додатку – те, як саме користувач його бачитиме після завершення розробки. Якщо UX-дизайн відповідає на питання *«як це працює?»*, то UI-дизайн дає відповідь *«як це виглядає?»*.

Вдалий інтерфейс поєднує естетику з функціональністю: кожен колір, розмір шрифту чи тінь мають своє значення. Завдяки цьому користувач не лише орієнтується без зусиль, а й отримує позитивний емоційний досвід від взаємодії з продуктом.

Етап 5. Презентація проєкту

Коли всі екрани створені й узгоджені між собою, слід показати результат клієнту або команді – це етап **презентації проєкту**. Саме тут дизайнер показує, як виглядатиме й працюватиме майбутній продукт, пояснюючи логіку, акценти та рішення, прийняті під час розробки.

Презентація може мати два формати:

- **Статичний** – коли демонструються готові зображення екранів із коментарями, описами кольорової гами, типографіки та стилістики. Такий варіант підходить для попереднього узгодження або коли не потрібно показувати інтерактивність.
- **Інтерактивний** – коли дизайнер створює клікабельний прототип у Figma чи іншому інструменті, і замовник може самостійно переглянути, як працюють переходи між сторінками, які ефекти з'являються під час натискання та як поводить інтерфейс.

Грамотно проведена презентація допомагає замовнику краще зрозуміти задум і оцінити не лише візуальну частину, а й зручність користування. Водночас дизайнер отримує можливість пояснити логіку своїх рішень і показати, як кожен елемент сприяє досягненню цілей проєкту.

Після презентації можуть бути внесені незначні правки – уточнення кольорів, позицій або дрібних деталей. Коли все узгоджено, проєкт переходить до фінальної стадії – **підготовки до розроблення**.

Етап 6. Експорт ресурсів

Після затвердження дизайну настає практичний етап – **експорт ресурсів**. Це підготовка всіх графічних матеріалів, необхідних для верстки та програмування.

Frontend-розробники не працюють безпосередньо з дизайнерським файлом, тому їм потрібні **окремі графічні елементи** – іконки, ілюстрації, логотипи, фони, фотосвітлини, кнопки, декоративні елементи тощо. Всі ці матеріали мають бути оптимізовані за розміром, форматом і якістю, щоб сайт працював швидко та коректно на різних пристроях.

Підготовкою й експортом займається **UX/UI-дизайнер**, який створював макети. Він визначає, які ресурси потрібні розробникам, експортує їх у відповідних форматах (наприклад, **SVG, PNG, JPG, WebP, AVIF**), перевіряє масштабування та додає специфікації: відступи, розміри, кольори, шрифти.

Для цього часто використовують **Figma Inspect** або спеціальні платформи для передачі макетів – **Zeplin, Avocode, Adobe XD Share** тощо. Це допомагає уникнути помилок під час верстки й гарантує, що кінцевий результат буде максимально близьким до оригіналу.

Етап 7. Верстка (HTML/CSS/JS)

Верстка – це написання коду, що дає змогу отримати статичні сторінки без переходів між екранами. Такі сторінки мають усі елементи інтерфейсу, візуально все виглядає, як готовий продукт, але під час натискання на кнопки або на посилання – нічого не відбувається.

Верстальники і frontend-розробники використовують такі технології, як:

- **HTML** (HyperText Markup Language) – мова гіпертекстової розмітки, за допомогою якої задаються елементи вебсторінки та її вміст;
- **CSS** (Cascading Style Sheets) – каскадні таблиці стилів, за допомогою яких задається зовнішній вигляд елементів, які створюються за допомогою HTML;
- **JS** (JavaScript) – мова програмування, за допомогою якої можна додати анімацію, змусити працювати слайдер тощо;
- **Frontend-фреймворки** – набір інструментів і готових до використання компонентів, які працюють на основі HTML, CSS та JS. Це значно зменшує час розробки. Наприклад, фреймворк **Bootstrap** охоплює шаблони оформлення для вебформ, кнопок, міток, типографіки, блоків навігації та інших компонентів.

У багатьох графічних редакторах вбудована мова програмування, яка є схожою на JavaScript. Це дає змогу автоматизувати рутинні дії.

Етап 8. Програмування

Створенням функціональної частини сайту займаються програмісти або backend-розробники. Вся ця система невидима і захована всередині, але саме вона дає змогу вебсайту реагувати на дії користувачів.

Backend-розробники використовують широкий спектр мов програмування та технологій. Наприклад:

- **Node.js** – серверна платформа для виконання JavaScript-коду, яка використовується для розроблення вебдодатків. Вона дозволяє створювати сервери для оброблення запитів і динамічно генерувати контент, що відповідає на дії користувачів;
- **PHP** (Hypertext Preprocessor) – мова програмування, що використовується для розроблення вебдодатків. Вона дає змогу динамічно генерувати HTML-код, у відповідь на дії користувачів;
- **MySQL** – система керування базами даних, що дозволяє вебсайтам зберігати й обробляти інформацію. Наприклад, облікові записи користувачів та їх повідомлення у соціальній мережі.

Етап 9. Тестування

Після завершення основної розробки жоден вебсайт не може бути запущений без перевірки. Етап **тестування (Testing)** – це момент, коли фахівці переконуються, що сайт працює стабільно, без збоїв і відповідає всім технічним вимогам.

У процесі тестування беруть участь **тестувальники** та **QA-фахівці (Quality Assurance)**, які перевіряють кожну функцію, сторінку та сценарій взаємодії користувача. Мета цього етапу – знайти і виправити помилки до того, як із сайтом почнуть працювати реальні відвідувачі.

Є два основні підходи до перевірки:

- **вручну** – тестувальник запускає вебсайт і перевіряє всі сценарії роботи користувачів, намагаючись виявити помилки;
- **автоматично** – QA-спеціаліст пише скрипт, який запускає браузер і виконує всі дії замість людини.

Іноді для тестування залучаються звичайні користувачі, але їх попереджають, що сайт ще не готовий і може містити помилки. Цей процес називається відкритим **бета-тестуванням**.

Етап 10. Розгортання вебсайту

Коли всі перевірки успішно пройдено, проєкт переходить до фінальної стадії – **розгортання (Deployment)**. Це процес, під час якого сайт переноситься

із середовища розробки на реальний сервер у мережі «Інтернет», щоб ним могли користуватися всі відвідувачі.

На цьому етапі зазвичай працюють **системні адміністратори** та **DevOps-інженери**. Вони налаштовують сервери, бази даних, сертифікати безпеки (SSL), доменні імена, а також контролюють швидкість завантаження та стабільність роботи ресурсу.

Після розгортання сайт стає доступним за своєю адресою (URL). У цей момент проводиться фінальна перевірка: тестуються форми, посилання, зображення, адаптивність і коректність відображення у різних браузерах.

Завдання команди DevOps – забезпечити **безперебійну роботу сайту**. Вони стежать за навантаженням на сервер, оновленнями системи та захистом від можливих збоїв або хакерських атак.

Розгортання – це не лише технічна дія, а й момент, коли проєкт уперше «оживає» у публічному просторі. Саме з цього моменту дизайн, логіка й код перетворюються на повноцінний вебпродукт, готовий до взаємодії із користувачами.

1.4. ПРИНЦИПИ ТА СУЧАСНІ ТРЕНДИ ВЕБДИЗАЙНУ

За статистикою, 75% користувачів формують думку про сайт на основі його зовнішнього вигляду, тому розробник мусить відстежувати всі тенденції у вебдизайні.

Наявність застарілого сайту з поганою продуктивністю може призвести до втрати величезної кількості клієнтів і як наслідок – до втрати бізнесу. Користувачі одразу відрізняють сайт, який не оновлювався 10 років, і сайт, який відповідає сучасним тенденціям вебдизайну.

Які ж сучасні тенденції у вебдизайні?

1. Адаптивний дизайн, орієнтований на мобільні пристрої (Mobile-first design). Понад 60% світового вебтрафіку (середні дані за останні роки) припадає на мобільні пристрої, тому сайти, які виглядають добре і на великих моніторах, і на смартфонах, є найпопулярнішими. Отже, варто розпочати проєктування

та розроблення сайту для мобільних екранів, а пізніше переходити до настільних комп'ютерів. Такий підхід забезпечить коректну роботу сайту на всіх пристроях.

2. Використання заголовків для привернення уваги до всієї сторінки.

Перше враження про вебдодаток створюють повносторінкові заголовки або розділи hero, які займають весь екран яскравим зображенням або відео із коротким і зрозумілим анімованим текстом і чітким заклик до дії. Ці заголовки призначені для негайного привернення уваги та надання індивідуальності бренду.

3. Екологічний вебдизайн. Оскільки мережа «Інтернет» спричиняє приблизно 3,7% світових викидів вуглецю, кожен сайт залишає свій слід. Оптимізація сайту скорочує обсяг даних, що зберігається і передається, водночас зменшується його вплив на навколишнє середовище. Щоб зменшити навантаження на дані необхідно:

- використовувати оптимізовані зображення меншого розміру;
- за можливості стискати зображення та інші великі файли;
- обмежити кількість вебшрифтів, щоб заощадити трафік;
- скоротити кількість непотрібних плагінів і скриптів;
- регулярно перевіряти продуктивність сайту, щоби підтримувати його енергоефективність.

4. Доступний та інклюзивний дизайн. Інклюзивний дизайн – це підхід до проєктування, який враховує потреби всіх користувачів, включаючи особливі групи, такі як люди з обмеженими можливостями, літні люди, діти та інші. Основна мета – створення продуктів, послуг і середовищ, які є доступними, зрозумілими та зручними для максимально широкого кола користувачів.

Для створення таких дизайнів необхідно розуміти потреби різних груп користувачів, активно залучати їх до процесу проєктування, щоб продукти були адаптовані до вирішення специфічних проблем. Інтерфейси та функції повинні бути інтуїтивно зрозумілими та не вимагати спеціальних навичок або знань. Важливо також зважати на різноманіття в способах сприйняття інформації, а також на те, що в середовищі всі користувачі повинні відчувати себе важливими.

Доступні сайти мають задовольняти такі вимоги:

- достатній колірний контраст між кольорами тексту і фону;

- відмова від використання тільки кольору як єдиного способу передачі ключової інформації;
- використання альтернативного тексту для зображень;
- забезпечення навігації за допомогою клавіатури;
- використання зрозумілих шрифтів достатнього розміру;
- транскрипції або субтитри для відео.

У світі розроблені відповідні настанови з доступності вебвмісту **WCAG (Веб Content Accessibility Guidelines)**, з якими можна ознайомитися за посиланням <https://www.w3.org/Translations/WCAG21-ua/>, та які містять деталі щодо того, як зробити вебконтент доступним для людей із обмеженими можливостями.

5. Використання сіток і двоколонкових макетів. Використання сіток і колонок – ефективний спосіб упорядкувати вміст візуально насиченого вебсайту, надавши йому чистого та структурованого вигляду. Сітки використовують для демонстрації зображень продуктів або вибраного контенту, а двоколонкове компонування – для збалансованого оповідання. Асиметричні сітки додають візуальну цікавість, зберігаючи структурованість контенту. Але важливо забезпечувати однаковість інтервалів та вирівнювання для естетичного вигляду.

6. Мінімалізм. Чистий, простий дизайн із акцентом на важливій інформації дає змогу вебсайту здаватися відкритим і витонченим. Мінімальне використання декоративних елементів і кольорів, креативна типографіка надають індивідуальність і стиль, не перевантажуючи сторінку.

7. Використання зображень та ілюстрацій. Ілюстрації та візуалізація даних можуть допомогти пояснити складні концепції або дані таким чином, щоб їх було легко зрозуміти. Візуальні матеріали мають бути чіткі та зрозумілі. Однак необхідно уникати використання занадто великої кількості візуальних матеріалів одночасно. Це може створити гнітючий ефект. Візуальні елементи мають мати сенс у контексті решти сайту.

8. Креативна типографіка. Креативна типографіка – це експерименти зі шрифтами: унікальні шрифти, інноваційні інтервали, перетворення тексту на візуальне мистецтво, анімація текстів і текстові ефекти. Смілива типографіка

та креативні текстові макети домінують як основний тренд вебдизайну, допомагаючи виділити важливий контент.

9. Інтерактивність. Інтерактивність робить вебсайт функціональнішим, забезпечуючи взаємодію із користувачами. До основних елементів інтерактивності належать:

- **Анімований курсор.** Такий курсор додає особливості вебсайту та привертає увагу користувача. Наприклад, під час наведення на інтерактивний об'єкт, такий курсор може змінювати форму або колір. Однак потрібно бути обережними і не надто захоплюватися анімацією, інакше це може виявитися радше відволікаючим, аніж корисним, а також пам'ятати, що анімований курсор має бути сумісним із усіма браузерами.
- **Анімація.** Анімації варто використовувати з функціональною метою. Наприклад, кнопка може змінювати колір під час натискання. Також анімації можуть бути адаптовані до вподобань користувача, враховуючи його вік та інші фактори.
- **Паралакс. Паралакс-скролінг** – це техніка, яка створює ілюзію глибини на 2D-сайті, прокручуючи фон з іншою швидкістю, ніж передній план. Це створює візуальний ефект, який робить вебсайт більш тривимірним. Під час використання паралакс-прокручування на сайті слід пам'ятати, що прокручування має бути плавне й однакове на всьому сайті.
- **Горизонтальне прокручування.** Горизонтальне скролювання – прокручування, яке рухається зліва направо, а не вгору або вниз. Використовується для відображення довгих списків елементів, таких як меню або галерея продуктів.
- **Мікровзаємодії.** Мікровзаємодії – це крихітні функціональні анімації, які реагують на дії користувача. Наприклад, вони можуть підтверджувати успішне відправлення форми або надавати користувачам інформацію про їх прогрес. Загалом вони полегшують виконання завдань, забезпечують зворотний зв'язок із користувачами та додають дизайну індивідуальності.

- **Інтерактивні елементи навігації.** Вони сприяють простоті використання сайту, допомагаючи користувачам швидше знаходити потрібну інформацію. Наприклад, меню, що розгортаються, інтерактивні карти тощо.

10. Використання штучного інтелекту. Дедалі частіше вебсайти використовують графіку та відео, створені за допомогою ШІ. Це дає змогу автоматизувати рутинні завдання та зекономити ресурси.

1.5. ВПЛИВ ПСИХОЛОГІЇ НА ВЕБДИЗАЙН

Успішний вебдизайн ґрунтується не лише на технічних рішеннях чи візуальній гармонії, а й на розумінні **психології користувача**. Дизайнери, які знають, як люди сприймають інформацію, приймають рішення та реагують на стимули, можуть створювати інтерфейси, що не просто приваблюють, а й утримують увагу, формують довіру та спонукають до дій.

Одним із важливих аспектів є когнітивні упередження – автоматичні та часто несвідомі моделі мислення, які впливають на процес прийняття рішень:

- **Ефект якоря** – це упередження, коли перша інформація, яку отримує людина, надалі сильно впливає на її рішення. Наприклад, якщо на сторінці сайту з цінами відображено дорогий товар першим, інші варіанти здаватимуться вигіднішими, навіть якщо це не відповідає дійсності, враховуючи різні характеристики.
- **Упередження доступності** – люди частіше згадують просту інформацію. У UX-дизайні це можна використовувати для акцентування на важливій інформації або частішого її показу, щоб така інформація залишалася у пам'яті користувача та впливала на його вибір.
- **Ефект підтвердження** – людина має схильність шукати й інтерпретувати інформацію, що підтверджує її переконання. Тобто користувачі можуть нехтувати інформацією, що суперечить їхньому початковому враженню про продукт. Отже, важливо створювати інтерфейси, які відповідають очікуванням користувачів.

- **Ефект статус-кво** – користувачі часто уникають змін і віддають перевагу знайомому. У ВЕБ це виявляється у схильності користувачів залишатися на звичних сайтах або використовувати знайомі елементи інтерфейсу. Отже, дизайн повинен бути інтуїтивним і не створювати зайвих ускладнень.
- **Парадокс вибору** – занадто багато варіантів можуть призвести до того, що користувач просто не може прийняти рішення. Цей ефект можна зменшити, групуючи варіанти або надаючи рекомендації.
- **Ефект кінця** – користувачі зазвичай запам'ятовують останній етап процесу, тому важливо, щоб він був позитивним. Наприклад, після здійснення покупки або заповнення форми користувачі мають отримувати повідомлення про успіх.

Отже, для того, щоб використати когнітивні упередження, необхідно зменшити кількість опцій, що допоможе користувачам швидше прийняти рішення, використовувати великі кнопки або підкреслення певних елементів для спрямування їх до бажаних дій і зробити процес взаємодії максимально простим і зрозумілим.

1.6. КОЛІР У ЦИФРОВОМУ ДИЗАЙНІ

Колір у цифровому дизайні – це властивість, що залежить від спектру світла, яке поверхня відбиває / пропускає, або яке дисплей випромінює.

Теорія кольору – це сукупність знань про фізику / фізіологію / психологію сприйняття й практичні методи організації кольору в середовищі. Виокремлюють **кольорознавство** (систематизація знань) і **колористика** (принципи, методи і прийоми колірної організації).

У вебдизайні теорія кольору використовується з урахуванням **колірних просторів екранів, контрастності за WCAG і дизайн-токенів** для забезпечення читабельності, доступності й узгодженості стилю.

ХАРАКТЕРИСТИКИ КОЛЬОРУ

Кольори поділяються на дві категорії – хроматичні й ахроматичні.

Хроматичними називаються всі спектральні кольори. Ці кольори отримуються шляхом розкладання світла на спектр (Рис. 2) і їх ще називають **спектральними**

кольорами. До **хроматичних** кольорів належать: червоний, жовтий, помаранчевий, зелений, синій, фіолетовий (кольори веселки) та їх комбінації.

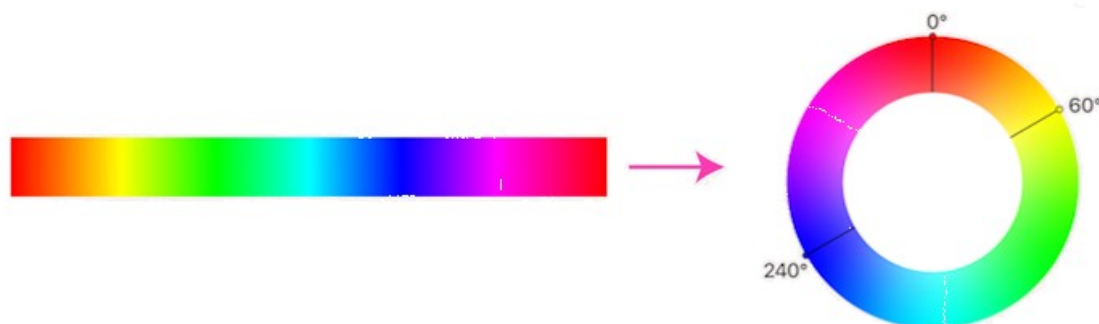


Рис. 2. Хроматичні спектральні кольори

До **ахроматичних** (ті, що не мають кольору) належать: білий, чорний і всі відтінки сірого. Людське око здатне відрізнити приблизно 400 відтінків сірого. У теорії кольору ці кольори є надзвичайно важливими, оскільки за їхньою допомогою створюються відтінки і тони.

Окрім спектральних кольорів, є ще три колірні групи: **світлі** (спектральні кольори, змішані з білим кольором), **темні** (спектральні кольори, змішані з чорним кольором) та **пастельні** кольори (спектральні кольори, змішані з різними відтінками сірого кольору).

До основних характеристик кольору належать: колірний тон, насиченість та яскравість.

Колірний тон (Hue) – це основний параметр, який визначає тип кольору (червоний, зелений, синій тощо). Він вимірюється в градусах (від 0° до 360°) на колірному колі. Наприклад, червоний відповідає 0°, зелений – 120°, синій – 240°.

Насиченість (Saturation) – це параметр, який визначає виразність (інтенсивність) кольору. Найвиразніші (насичені) кольори – «чисті спектральні». А у разі зменшення насиченості колір поступово знебарвлюється.

Насиченість вимірюється у відсотках, де 0% означає відсутність кольору (відтінки сірого), а 100% – повна насиченість (Рис. 3).



Рис. 3. На першому рисунку інтенсивність 0%, а на останньому 100%

Колірний тон представляється як координата на колі. Чисті спектральні кольори – це кольори, які знаходяться безпосередньо на контурі кола. У разі віддалення від краю до центру кола кольори тьмяніють.

Наприклад, рожевий отримується підмішуванням у червоний білого, отже, знижена насиченість. Навпаки, чистий червоний свідчить про відсутність білого (Рис. 4).



Рис. 4. Зміна колірного тону

Яскравість (Brightness) визначає освітленість або затемненість кольору. Це також відсотковий показник, де 0% означає чорний (абсолютна темрява), а 100% – максимальна яскравість (Рис. 5).



Рис. 5. Зміна яскравості

Наприклад, бордовий отримується підмішуванням до червоного чорного, отже, знижена яскравість. Навпаки, чистий червоний свідчить про відсутність чорного.

КОЛІРНІ МОДЕЛІ ТА ПРОСТОРИ

Розглянемо колірні моделі, що використовуються в комп'ютерній графіці для вебдизайну.

HEX: ШІСТНАДЦЯТКОВИЙ ЗАПИС

Це популярний формат кодування кольорів для вебдизайну. Він представлений у виді #RRGGBB, де RR, GG, BB – це двозначні шістнадцяткові числа, що представляють червоний, зелений і синій канали відповідно. Наприклад: #FF5733

00 – мінімальне значення каналу, а FF – максимальне.

# 000000		відсутність кольору за усіма каналами, чорний
# FFFFFFFF		максимум за усіма каналами, білий
# FF0000		максимум за червоним каналом, яскраво-червоний
# 00FF00		максимум за зеленим, яскраво-зелений
# 0000FF		максимум за синім каналом, яскраво-синій

Якщо за всіма трьома каналами стоїть однакове значення, наприклад, **#808080**, то колір буде монохромним, тобто не матиме відтінку. Це відбувається тому, що канали один одного компенсують.

HEX-коди – це популярний формат кодування кольорів для вебдизайну. Превага цих кодів у тому, що їх зручно копіювати та передавати.

RGB-КОЛЬОРИ

RGB – базова колірна модель у веб. RGB дає змогу встановлювати кольори на основі числових значень Red (Червоний), Green (Зелений) та Blue (Синій). Кожен колір складається з трьох значень, де кожне значення може варіюватися від 0 до 255.

RGB-кольори використовують для створення різних відтінків і кольорів у дизайнах. Наприклад, **градієнти** та **переходи** кольорів можна створювати шляхом зміни значень кожного з трьох каналів.

rgb(0, 0, 0)		відсутність кольору за всіма каналами, чорний
rgb(255, 255, 255)		максимум за всіма каналами, білий
rgb(255, 0, 0)		максимум за червоним каналом, яскраво-червоний
rgb(0, 255, 0)		максимум за зеленим, яскраво-зелений
rgb(0, 0, 255)		максимум за синім каналом, яскраво-синій

sRGB – це стандарт того, як саме ці числові значення перетворюються у колір на екрані (яка «біла точка», яка крива яскравості тощо). Тобто sRGB – це конкретний «вид» RGB. У веб, за замовчуванням, усі #hex/rgb() сприймаються

як sRGB, тому саме sRGB, а не «просто RGB», забезпечує передбачуваність відтворення.

СМΥК (Cyan, Magenta, Yellow, Black) – колірна модель, що використовується у поліграфії; у вебі використовується лише для конвертації документів, які підуть до друку. У вебі СМΥК не застосовується.

RGBA-КОДИ

Як альтернативу неінтуїтивним HEX-кодам розроблено систему кодування, яка адаптована для використання у коді вебсторінок та інших програмах. Вона містить в основі RGB, а також додає до трьох каналів додатковий, який дає змогу задати прозорість. Такий канал називається **альфа-каналом**. Перші три канали задаються від 0 до 255, а альфа-канал від 0 до 1.

Значення **0,5** відповідає 50% прозорості.

RGBA-код формується так: **rgba(червоний, зелений, синій, альфа)**.

Приклади:

rgba(0, 0, 0, 0.5) – чорний, з прозорістю 50%;

rgba(255, 255, 255, 1) – непрозорий білий.

DISPLAY P3

Display P3 – колірний простір RGB із первинними кольорами DCI-P3, білою точкою D65 і гаммою як у sRGB. Цей простір дає ширшу колірну гаму (колірна гама – діапазон кольорів, які може відтворити або записати певний пристрій) на сучасних дисплеях. Використовується як прогресивне покращення, обов’язково з **fallback** у sRGB.

Приклад: `color(display-p3 0.8 0.2 0.1 / 0.9)`, де:

- 0.8 0.2 0.1 – це частки R G B в діапазоні 0...1: R = 0.8 (80%), G = 0.2 (20%), B = 0.1 (10%) – насичений червоно-помаранчевий;
- / 0.9 – прозорість (alpha) – 90% непрозорий (10% прозорий).

На екранах із підтримкою **P3** це виглядає як дуже насичений червоно-помаранчевий із 90% непрозорістю.

HSL/HSB-КОЛЬОРИ

HSL/HSB (Hue, Saturation, Brightness) – це одна з моделей кольорового простору, яка використовується для опису кольорів на основі трьох параметрів:

- **Hue** – колірний тон (хроматичний колір);
- **Saturation** – насиченість;
- **Brightness** – яскравість / **Lightness** – (світлість).

Головна різниця між **B** і **L**: зменшує **Brightness** – усе рівномірно темнішає до чорного, **Lightness** – змішує колір із білим або чорним (50% L – «чистий» тон, нижче – темнішає (чорніє), вище – світлішає (вибілюється).

На Рис. 6 продемонстровано, як змінюються колірний тон, насиченість та яскравість із використанням просторової моделі.



Рис. 6. 3d-демонстрація моделі HSB

LAB / LCH / OKLAB / OKLCH

У HSL/HSB кольори іноді «стрибають». LAB/LCh/OKLCH розробили для того, щоб кроки зміни кольорів виглядали рівномірно для користувача.

Lab. Уявіть тривимірний простір. Вісь **L** показує, наскільки світло / темно. Вісь **a** – чи більше в кольорі зеленого, чи червоного. Вісь **b** – чи більше синього, чи жовтого. Перемістилися по осях – змінили колір.

LCh – це той самий Lab, але у «полярних» координатах:

- **L** – світлість;
- **C (Chroma)** – насиченість;
- **h° (Hue)** – відтінок.

Легко пояснювати завдання під час дизайну: «зроби **світліше** на 6 пунктів **L**», «зменш **C** для спокійнішого вигляду», «поверни **h°** на +10°, щоб відійти від занадто фіолетового».

OKLab/OKLCH – сучасні версії, які ще краще збігаються з людським сприйняттям **на дисплеях**. Тому градієнти без «провалів», а шкали 50...900 виходять рівномірнішими. **OKLCH** зручний для побудови палітр і токенів.

Приклад: OKLCH(62% 0.12 240):

- **L** = 62% – світлість (наскільки світлий / темний колір). 62% – середньо-світлий;
- **C** = 0,12 – хрома (наскільки «кольоровий» / насичений). 0.12 → помірна насиченість;
- **h** = 240° – відтінок у градусах на колірному колі. 240° = синій (без зміщення у фіолет).

Візуально – це середньо-світлий, помірно насичений синій.

ПОБУДОВА ПАЛІТР І ДИЗАЙН-ТОКЕНИ

Використання кольорів та їх комбінацій у комерційних і бізнес-цілях ґрунтується на теорії кольору або психологічному й емоційному сприйнятті кольору людиною. Грамотне кольорове оформлення здатне зіграти вирішальну роль у досягненні поставлених перед вебзастосунком цілей шляхом корекції сприйняття користувачами візуальної інформації.

Одним із інструментів, що допомагають вебдизайнеру зрозуміти, як взаємодіють кольори, є колірне коло, за допомогою якого можна створювати різні типи палітр.

Монохроматична палітра. Використовує різні відтінки одного кольору з різною насиченістю та яскравістю (Рис. 7). Створює елегантний і професійний ефект із акцентом на формі та змісті, а не на кольорових контрастах. Підходить для дизайну в мінімалістичному стилі.

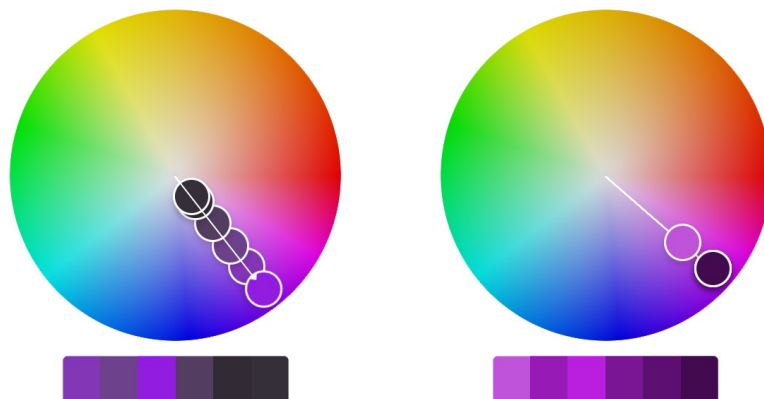


Рис. 7. Монохроматична палітра

Аналогова палітра. Використовує кольори, які знаходяться поруч на колі й не конфліктують між собою, а підкреслюють ефект один одного (Рис. 8). Створює гармонійний і спокійний ефект. Наприклад, якщо основним кольором обрано фіолетовий, то доповнюючими його кольорами будуть кольори, які розташовані по обидва боки від нього: рожевий і блакитний.

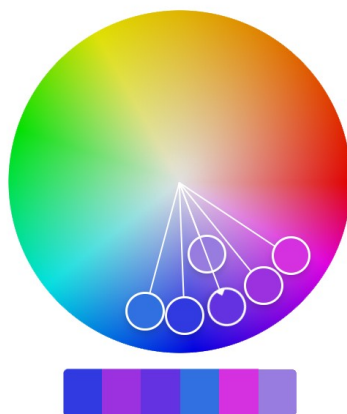


Рис. 8. Аналогова палітра

Комплементарна палітра. Використання кольорів і відтінків, що різко контрастують один із одним, дає змогу підкреслити важливість окремих елементів

чи інформації. Контрастні відтінки, тобто діаметрально протилежні, вибираються за допомогою колірного кола. Щоб підібрати контрастну пару для того чи іншого відтінку, достатньо вибрати відповідний сегмент на колірному колі (Рис. 9), після чого вибрати відповідний сегмент на зворотному боці.

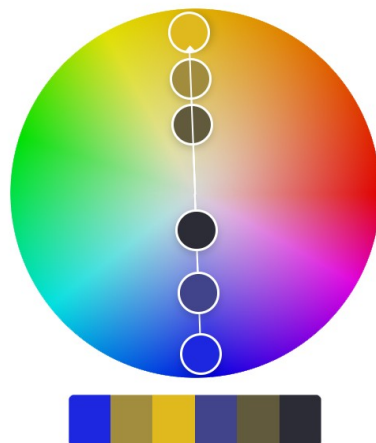


Рис. 9. Комплементарні кольори

Триадна палітра. Використовує три кольори на колі, що рівновіддалені один від одного (Рис. 10) (наприклад, червоний, жовтий і синій). Забезпечує яскравий і збалансований контраст, що дає змогу зробити інтерфейс додатку динамічним і живим.

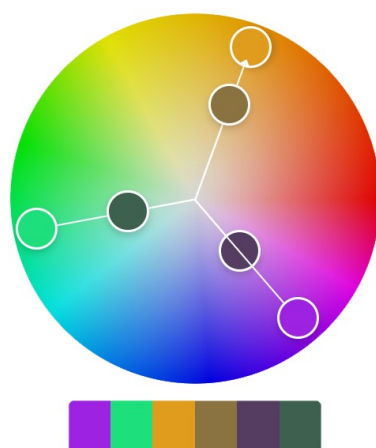


Рис. 10. Триадна палітра

Тетраїдна палітра. Охоплює чотири кольори, які утворюють прямокутник на колі (Рис. 11). Ця схема охоплює дві групи комплементарних кольорів і підходить

для складного дизайну із багатьма елементами, створюючи багатий і збалансований контраст.

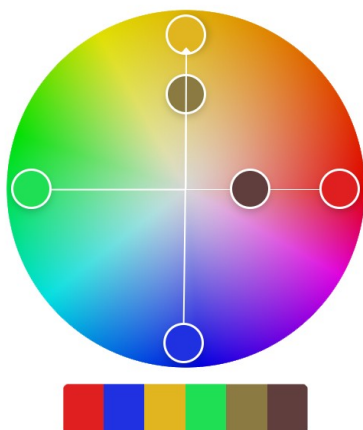


Рис. 11. Тетраїдна палітра

РОЛЬОВІ ТОКЕНИ І ШКАЛИ

Рольові токени – це кольори, які описують не «відтінок заради відтінку», а його призначення в інтерфейсі. Наприклад, замість призначення кнопці випадкового синього, їй призначається роль **primary**. Якщо пізніше бренд змінить синій на м'ятний, достатньо підмінити значення токена – весь продукт оновиться без ручного пошуку кольорів.

Зазвичай визначають шість базових ролей. **Primary** – головний акцент і колір основної дії; **Secondary** – другорядні акценти; **Neutral** – все «беземоційне»: тло, текст, бордери; **Success**, **Warning**, **Danger** – службові стани для успіху, попереджень і помилок. Ролі однакові по всьому продукту, тож користувач бачить послідовну мову кольору.

Кожна роль має шкалу від найсвітлішого до найтемнішого. Зручно робити 8–10 кроків із номерами від 50 до 900, де 500 – базове значення. Найкраще будувати шкали в OKLCH, змінюючи передусім L (світлість): від дуже світлих значень приблизно 98% для кроку 50 до приблизно 36% для кроку 900. Точні числа підганяють під бренд і перевіряють на контраст за WCAG, щоб текст на кнопках і картках був читабельним.

Нейтральна шкала має бути без кольоровості: насиченість **C** утримується в межах 0...0,02, щоб сірі не «зеленіли» чи «синіли». Для **Primary** і **Secondary** помірна насиченість **C** (приблизно 0,08–0,14) дає живий, але контрольований акцент. Для статусів добирають прості відтінки: зелений для успіху, бурштиновий для попередження, червоний для помилок; насиченість теж помірна, щоб не «світилося».

У темній темі ж фонові кольори роблять у діапазоні низької світлості **L** (приблизно 8–26%) для збереження різниці рівнів поверхонь, а акцентам трохи піднімають світлість і зменшують насиченість **C** – так колір не перетворюється на неон і не втомлює очі. Тобто **Primary-500** у темній темі зазвичай світліший, ніж у світлій, і трохи менш насичений.

Окрім шкал, кожна роль використовується у станах компонентів. Так, наприклад, у кнопки є фон, колір тексту чи іконки поверх нього, рамка та реакція на взаємодію. Ці стани не вигадують «з нуля» – вони посилаються на кроки рольової шкали. Наприклад, фон базової кнопки може бути **Primary-500**, наведення – **Primary-600**, натискання – **Primary-700**, а контур фокусу – світліший відтінок на кшталт **Primary-400** з невеликою прозорістю. Для вимкненого стану беруть світліший фон і темніший текст, але з достатнім контрастом.

Щоб усім цим було легко керувати на глобальному рівні називають роль і крок шкали: **color-primary-50 ... color-primary-900** тощо. Паралельно заводять «**on-***» токени, наприклад, **on-primary** – це кольори тексту та піктограм поверх відповідного тла. А на рівні компонентів оголошують «розмовні» імена, наприклад, **btn-primary-bg**, **btn-primary-fg**, **btn-primary-hover-bg**. Ці компонентні токени просто посилаються на рольові, тому редизайн робиться в одному місці.

Найчастіші помилки трапляються тоді, коли кроки шкали обирають навмання, у темній темі залишають занадто високу насиченість, або забувають про «**on-***» токени. У результаті десь обов’язково провалюється читабельність. Усе це легко виправляється, якщо тримати шкали в **OKLCH**, будувати їх рівномірно за світлістю та перевіряти контраст на ключових парах.

КОЛІР ТА ЕМОЦІЇ

Колір – один із найсильніших інструментів вебдизайну. Він не просто прикрашає сторінку, а й формує **емоційний настрій**, допомагає передати характер бренду й впливає на рішення користувача, навіть коли той цього не усвідомлює. Правильно підібрана палітра здатна зробити сайт динамічним і натхненним або, навпаки, спокійним і довірливим.

Завдання дизайнера – не просто «гарно пофарбувати» інтерфейс, а **створити атмосферу**, яка відповідає змісту та меті ресурсу. Колір допомагає розставити акценти, підкреслити важливі елементи, а також забезпечує зручне сприйняття контенту.

ЕМОЦІЙНЕ СПРИЙНЯТТЯ КОЛЬОРІВ

Теплі кольори – червоний, помаранчевий, жовтий – асоціюються з енергією, активністю, теплом і позитивом. Їх часто використовують у маркетингових кампаніях, щоб привернути увагу, стимулювати дію або створити відчуття радості й руху.

Холодні кольори – синій, блакитний, зелений, фіолетовий – навіюють спокій, стабільність і довіру. Вони підходять для корпоративних, фінансових чи медичних сайтів, де важливо викликати впевненість і відчуття безпеки.

Нейтральні кольори – білий, сірий, чорний, бежевий – виконують роль тла, допомагаючи іншим відтінкам розкритися. Вони створюють відчуття простору, елегантності та сучасності, тому часто використовуються у мінімалістичних інтерфейсах.

ТЕМНІ ТЕМИ ЯК СУЧАСНА ТЕНДЕНЦІЯ

Недавнім часом особливої популярності набули **темні теми оформлення**. Вони не лише виглядають стильно й сучасно, а й **зменшують навантаження на очі**, насамперед під час роботи у темних приміщеннях.

У темному дизайні важливо дотримуватися балансу:

- текст рідко роблять чисто білим (#FFF), адже такий контраст створює ефект сляйва;
- поверхні поділяють на **декілька рівнів глибини** з м'якими переходами відтінків;
- контраст між елементами помірний, щоб уникнути «засвічених» ділянок і зберегти комфорт для зору.

Темна палітра часто поєднується з **акцентними кольорами** – золотим, бірюзовим, фіолетовим або мідним, що додає візуального багатства без перевантаження.

Отже, колір у вебдизайні – це **мова емоцій**. Він може розповісти про бренд без слів, викликати довіру, стимулювати дію або створити відчуття спокою. Від уміння дизайнера керувати цією мовою залежить не лише естетика, а й ефективність усього вебпродукту.

1.7. ГРАФІЧНІ ФОРМАТИ ДЛЯ ВЕБДИЗАЙНУ

Графічні формати – це формати файлів, які використовуються для зберігання зображень. Є чимало різних графічних форматів, кожен із яких має свої особливості, переваги та недоліки. Деяка частина з них стала стандартами і використовується у більшості графічних програм.

Розглянемо графічні формати й особливості їх використання у вебдодатках.

РАСТРОВІ ФОРМАТИ

Растрові формати – призначені для зберігання зображень, що складаються із пікселів. Найпоширенішими типами растрових файлів у вебдодатках є:

PNG (Portable Network Graphics). Файли цього формату мають розширення .png, .apng і можуть зберігати метадані, такі як інформація про авторські права або коментарі.

PNG підтримує колірні глибини до 48 біт на піксель для кольорових зображень і до 16 біт на піксель для відтінків сірого. PNG підтримує не лише колірну модель RGB/RGBA, а й grayscale та індексовану палітру (PNG-8).

Використовує стиснення даних без втрат. Це означає, що зображення не втрачає якість під час зберігання й завантаження, але файли можуть бути доволі великими порівняно із форматами зі стисненням із втратами, такими як JPEG.

Підтримує альфа-канали, що дає змогу використовувати прозорість у зображеннях.

Не підтримує шари й анімацію. Для цього є розширення формату PNG - APNG (Animated PNG).

Цей формат широко використовується у вебдизайні, графічному дизайні та для збереження зображень із різкими контурами, іконок із прозорістю, скрінів інтерфейсів, коли потрібне стиснення без втрат. Фотографії краще зберігати в AVIF/WebP.

JPEG (Joint Photographic Experts Group). Формат, що часто використовується як формат даних у цифрових камерах. У веб формат JPEG застосовується для відображення напівтонових ілюстрацій та графічної інформації з плавним переходом тонів. Формат добре підходить для фотосвітлин і складних зображень, якщо формати AVIF/WebP недоступні. Розширення файлів: .jpg, .jpeg.

JPEG не підтримує прозорість, багатошаровість й анімації. Під час детального розгляду і сильно стисненого зображення помітно розмиття меж і характерний **муар** поблизу них.

WebP (Веб Picture). Растровий формат, який забезпечує оптимальне співвідношення між розміром файлу і якістю зображення, що робить його популярним на вебсайтах, онлайн-платформах, мобільних додатках і соціальних мережах.

Цей формат підтримує анімацію, подібну до GIF, але з меншим розміром файлу та кращою якістю, а також прозорість.

Підтримує як стиснення із втратами (розмір **WebP**-файлу менший ніж JPEG приблизно на 30%), так і без втрат (розмір **WebP**-файлу менший, ніж PNG на 26%).

WebP підтримується більшістю сучасних браузерів (Chrome, Firefox, Edge, Opera).

AVIF (AV1 Image File Format). Сучасний формат растрових зображень, що зберігає картинку майже без втрати якості. За однакової візуальної якості AVIF зазвичай має менший розмір, аніж JPEG і навіть WebP. На великих фото це добре помітно: банери й галереї завантажуються швидше. Водночас деталізація, плавні градієнти та дрібні переходи кольору зберігаються краще.

AVIF підтримує прозорість (як PNG), тому зручно робити накладні елементи без фону. Підтримує просту анімацію, а також дружить із «широкими» кольорами та HDR, тож яскраві фотосвітлина на сучасних екранах виглядають більш насиченими.

ВЕКТОРНІ ФОРМАТИ

Векторні формати – призначені для зберігання двовимірної векторної графіки. Найпоширенішим векторним форматом у вебдодатках є **SVG (Scalable Vector Graphics)**. Цей формат підтримується більшістю веббраузерів і графічних редакторів. Зберігає зображення у форматі .xml, що дає змогу їх масштабувати без втрати якості та легко редагувати. Також **SVG** підтримує анімацію й інтерактивність. Підходить для створення іконок, логотипів, схем, графіків, ілюстрацій.

3D ФОРМАТИ

Тривимірні формати – містять опис форми, текстури та кольору об'ємних моделей.

glTF (GL Transmission Format) – це формат для 3D-моделей, створений спеціально «під веб». Зазначений формат добре стискує тривимірні сцени та мінімізує обробку під час виконання додатків, що використовують ВебGL та інші API.

Із-поміж переваг glTF можна зазначити чітку ієрархію об'єктів у структурі 3D-сцени, зберігання інформації про джерела світла та камери сцени, підтримку скелетної анімації та сучасних матеріалів із фізично коректним відтворенням

(можна задавати базовий колір, шорсткість, металевість, нормалі, оклюзію та емісію). Працюють анімації, деформації та морф-таргети, тож моделі можуть бути не лише статичними, а й рухомими.

OBJ (Wavefront OBJ) – старий, але дуже поширений формат файлів опису статичної 3D-геометрії та текстур (позиція вершин, зв'язок координат текстури із вершиною, нормаль для кожної вершини, параметри, що описують полігони).

USDZ – формат, що дозволяє миттєвий перегляд у доповненій реальності через Safari («AR Quick Look»): користувач відкриває посилання і бачить об'єкт у себе в кімнаті без додаткових застосунків.

USDZ не має внутрішнього стиснення, тому модель має мати небагато полігонів, помірні текстури.

МУЛЬТИМЕДІЙНІ ФОРМАТИ

Мультимедійні формати – призначені для зберігання даних різних типів (графіки, звуку, відео) в одному файлі. Під час роботи з мультимедіа потрібно розрізняти поняття контейнеру й кодеку.

Контейнер – це файл, у який складені потоки відео, аудіо, субтитри й метадані. Він задає **розширення** файлу та спосіб зберігання вмісту.

Кодек – це спосіб стиснення, яким закодовані самі відео- й аудіодані всередині контейнера.

Щоб відео відтворилось, і контейнер, і кодеки всередині повинні підтримуватися браузером / пристроєм.

До найпопулярніших мультимедійних контейнерів для веб належать:

MP4 (контейнер) – універсальний контейнер для відео на сайтах, що сумісний із усіма браузерами: **H.264/AVC** плюс аудіо **AAC** (інколи MP3).

WebM (контейнер) – контейнер для використання на вебсайтах та у додатках, оптимізований для передавання відео високої якості при меншому розмірі файлів. Сумісний майже з усіма браузерами. Файл .webm (контейнер) зазвичай містить відео VP9/AV1 і аудіо Opus.

Розглянемо окремі кодеки:

MP3 (аудіо-кодек/контейнер) – формат аудіо в інтернеті, підтримується всіма браузерами та пристроями. Використовується для музики, аудіокниг та інших типів аудіоконтенту. Дозволяє додавати метадані (назва, виконавець, альбом тощо). Застарілий порівняно з AAC.

AAC (аудіо-кодек) – сучасний аудіоформат, що забезпечує кращу якість звуку за меншого розміру файлу. Є стандартним форматом аудіо для MP4.

Opus (аудіо-кодек). Рекомендований для вебу кодек із дуже хорошою якістю на низьких бітрейтах (стріми, голос, довгі відео) із широкою підтримкою браузерів.

АНІМАЦІЙНІ ФОРМАТИ

Анімаційні формати використовуються для створення динамічного та привабливого контенту для вебсайтів. Формат обирають за двома критеріями – якість / розмір і сумісність із браузерами.

Якщо анімація довга або «важка», краще зробити її відеороликом – так іде апаратне відтворення й економиться живлення батареї.

У продакшені зручно давати кілька форматів: спершу найефективніший (AVIF/WebP), нижче – запасний (APNG або навіть GIF). Так анімація відображатиметься всюди, просто в когось – у кращій якості та з меншою «вагою».

GIF (Graphics Interchange Format). Класичний растровий формат, що використовує палітру з 256 кольорів на кадр і без напівпрозорості. Дає змогу створювати анімовані зображення за допомогою кількох шарів або кадрів. Кожен кадр зображення зберігається як окремий шар, і ці кадри відтворюються послідовно, повторюючись по колу, створюючи ефект анімації.

Цей формат популярний в інтернеті, передусім для створення мемів, анімованих зображень і простих графічних елементів через свою малу «вагу» та сумісність із більшістю браузерів і платформ.

WebP з анімацією. Сучасна заміна GIF із меншим розміром за кращої якості. Підтримує прозорість і сумісний із сучасними браузерами (Chrome, Firefox, Edge). Практичний вибір для анімованих растрових елементів на сайті.

AVIF з анімацією. Ще ефективніше стискання, можливість прозорості й підтримка нових дисплеїв. Формат швидко набирає популярності, але для надійності все одно корисно мати запасний варіант – WebP чи APNG.

SVG – формат, який дає змогу створювати векторні зображення та анімацію. Зображення у цьому форматі зберігають якість під час масштабування та легко редагуються й інтегруються у HTML. Ідеально підходить для логотипів, піктограм та інтерактивної інфографіки.

Lottie (JSON-анімації). Це не «картинка», а опис анімації у форматі JSON, який відтворює спеціальна бібліотека у браузері. Дає невеликі за розміром файли і зручне керування анімацією в коді. Добре підходить для мікровзаємодій у інтерфейсі.

1.8. ТИПИ КОМП'ЮТЕРНОЇ ГРАФІКИ

Є декілька класифікацій комп'ютерної графіки за різними критеріями, залежно від типу зображень, технологій або застосувань. Основні види класифікації:

ЗА СПОСОБОМ ПОДАННЯ ЗОБРАЖЕННЯ

Растрова графіка. Зображення складається з **пікселів**, кожен з яких має свій колір. **Піксель** (англ. pixel) – найдрібніша одиниця цифрового зображення у растровій графіці. Він є неподільним об'єктом прямокутної (зазвичай квадратної) форми, що має певний колір. **Растр** – сукупність пікселів.

Векторна графіка. Зображення складається з геометричних **примітивів**. **Примітив** – найпростіший геометричний об'єкт, який може бути опрацьований (накреслений, збережений) системою: точки, лінії, криві та багатокутники.

Фрактальна графіка. Зображення створюються за допомогою математичних формул, що описують **фрактали**. **Фрактали** (від лат. fractus – подрібнений) – це математичні об’єкти, що мають самоподібність у різних масштабах.

ЗА ТИПОМ ТРИВИМІРНОСТІ

Двовимірна (2D) графіка. Зображення створюються у двовимірному просторі. Для створення зображень (моделювання об’єктів) використовуються **полігони**. Полігон – це плоска геометрична фігура, обмежена відрізками ліній (сторонами), що має кілька вершин, де ці відрізки з’єднуються. Полігону в графіці зазвичай надається трикутна або чотирикутна форма, хоча можуть бути й складніші полігони. Полігони використовуються для проєктування інтерфейсів, ілюстрацій, світлин, а також у вебдизайні.

Тривимірна (3D) графіка. Зображення створюються у тривимірному просторі. Для створення зображень (моделювання об’єктів) також використовуються **полігони**. Використовується в анімації, архітектурі, кінематографії, віртуальній реальності, іграх.

ЗА МЕТОДОМ ФОРМУВАННЯ

Статична графіка. Зображення не змінюється у часі, тобто залишається постійним після створення. Використовується у дизайні інтерфейсів, вебдизайні, друкованих матеріалах.

Динамічна графіка. Зображення змінюється у часі, здійснює рух, трансформується або змінює колір. Це може бути анімація, відео чи інтерактивні елементи. Використовується у відеоіграх, кіно, анімації, рекламних роликах.

ЗА ПРИЗНАЧЕННЯМ

Інженерна (CAD). Застосовується для створення точних технічних креслень, схем і моделей в архітектурі, будівництві, машинобудуванні.

Наукова графіка. Використовується для візуалізації наукових даних і процесів, таких, наприклад, як моделювання кліматичних змін або візуалізація фізичних процесів.

Анімаційна графіка. Створює рухомі зображення для кіноіндустрії, реклами та відеоігор. Може бути як 2D, так і 3D.

Комп'ютерна графіка для ігор. Використовується для створення реалістичних або стилізованих ігрових середовищ, персонажів і спецефектів.

ЗА ВИКОРИСТАННЯМ ОСВІТЛЕННЯ ТА ТІНЕЙ

Плюска (Flat) графіка. Об'єкти не мають тіней (або спрощені) і об'ємів, виглядають плоскими. Може імітувати освітлення у стилі малюнків або картин, із акцентом на стилізацію, а не на реалістичність.

Фотореалістична графіка. Використовує тіні, освітлення, текстури для створення зображень, максимально схожих на реальні об'єкти. Використовується у кінематографі, архітектурній візуалізації та рекламі.

ЗА ТЕХНОЛОГІЯМИ РЕНДЕРИНГУ

Растрезація. Перетворення 3D-сцени на 2D-зображення шляхом обчислення кольорів пікселів. Растрезація є дуже швидким і популярним методом рендерингу в реальному часі, наприклад, у відеоіграх. Однак вона менш точна порівняно з методом трасування променів, оскільки не завжди враховує всі ефекти взаємодії світла.

Рейтрейсинг (трасування променів). Складніший метод рендерингу, що враховує фізику світла для створення реалістичних тіней, відбиттів і прозорості. Використовується для створення фотореалістичних зображень і візуальних ефектів.

1.9. ОГЛЯД ПОПУЛЯРНИХ ГРАФІЧНИХ РЕДАКТОРІВ

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РАСТРОВОЇ ГРАФІКИ

Платне:

- **Adobe Photoshop** – один із надпотужних редакторів для роботи з растровою графікою. Має величезну кількість інструментів і можливостей для редагування зображень.

- **Corel PaintShop Pro** – інший популярний редактор, який пропонує різноманітні інструменти для обробки фотосвітлин і створення графіки.
- **Affinity Photo** – професійний редактор із великою кількістю функцій. Має меншу вартість порівняно з Adobe Photoshop.

Безкоштовне:

- **GIMP** – безкоштовний і відкритий редактор, який пропонує чимало функцій, схожих на Photoshop.
- **Krita** – безкоштовний редактор, орієнтований на цифрових художників. Можливість створення плоскої анімації.
- **Paint.NET** – безкоштовний растровий графічний редактор для Microsoft Windows, створений на основі .NET Framework. У деяких випадках можна розглядати як полегшену безкоштовну альтернативу Adobe Photoshop.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ВЕКТОРНОЇ ГРАФІКИ

Платне:

- **Adobe Illustrator**. Один із найвідоміших і потужних інструментів для створення векторної графіки. Він широко використовується дизайнерами для створення логотипів, іконок, ілюстрацій, вебдизайну й інших видів графіки;
- **CorelDRAW**. Ще один популярний інструмент, який має чималу кількість функцій для створення та редагування векторних зображень. CorelDRAW часто використовується для графічного дизайну, оформлення друкованих матеріалів і вебдизайну;
- **Adobe XD**. Додаток, інтегрований з іншими продуктами Adobe. Має можливості прототипування та дизайну, командної роботи у режимі реального часу;
- **Affinity Designer**. Програмне забезпечення, яке є конкурентом Adobe Illustrator. Підтримує одночасну роботу з растровими та векторними зображеннями;
- **Sketch**. Потужний інструмент для UI/UX дизайну з великим набором плагінів;

- **InVision Studio.** Прототипування, анімації та командна робота в режимі реального часу;
- **Axure RP.** Потужні інструменти для створення прототипів із можливістю складної логіки;
- **Xara Designer Pro.** Професійний набір інструментів для створення векторної графіки, що підходить для дизайну реклами, логотипів і вебграфіки.

Безкоштовне:

- **Figma.** Має безкоштовну та платну версії. Онлайн-інструмент для дизайну інтерфейсів і спільної роботи, який підтримує створення векторної графіки. Figma дає змогу командам працювати над проєктами у режимі реального часу;
- **Penpot.** Відкритий вихідний код, дозволяє спільну роботу в режимі реального часу. Є веббазованим інструментом;
- **Gravit Designer.** Інструмент для векторної графіки, вебверсія та десктопний додаток. Має безкоштовну та платну версії;
- **Lunacy.** Редактор графіки з підтримкою Sketch-файлів, підтримує інтеграцію з хмарними сервісами;
- **Vectr.** Простий у використанні інструмент для створення векторної графіки;
- **Inkscape.** Вільне програмне забезпечення з відкритим кодом для створення векторної графіки.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ 3D ГРАФІКИ

У вебдизайні 3D графіка переважно використовується для створення інтерактивного контенту, візуалізацій і анімацій. Є чимало інструментів, що дають змогу створювати 3D-рішення для вебдодатків.

Платні:

- **Unity WebGL.** Платформа для експорту 3D-додатків, створених в Unity та для роботи у веббраузерах;

- **PlayCanvas.** Хмарна платформа для розробки 3D-графіки й ігор, орієнтована на командну роботу. Наявна безкоштовна версія, що пропонує базові можливості;
- **Sketchfab.** Хмарна платформа для публікації та візуалізації 3D моделей у веббраузері. Безкоштовна версія використовується дизайнерами для демонстрації робіт, а платна підписка – для комерційних цілей і завантаження моделей.

Безкоштовні:

- **Three.js** – Бібліотека JavaScript призначена для створення об'єктів 3D графіки. Підтримує створення інтерактивних 3D-сцен із тінями, світлом та анімаціями;
- **Babylon.js** – Фреймворк, що пропонує розширений функціонал для створення складних сцен, AR/VR-додатків та інтерактивних 3D-презентацій;
- **Blender.** Редактор для створення 3D-моделей і анімацій, які експортуються у формати glTF або OBJ для використання у вебпроектах.

ОГЛЯД AI-ІНСТРУМЕНТІВ У ДИЗАЙНІ

Нині штучний інтелект став невід'ємною частиною творчого процесу. Якщо раніше дизайнер покладався лише на власні навички, то тепер у нього є інструменти, які допомагають швидше візуалізувати ідеї, створювати прототипи й опрацьовувати графіку.

Нижче розглянемо чотири найкорисніші платформи для вебдизайнера: **Midjourney, Runway, ChatGPT і Uizard.**

Midjourney – генератор візуальних концепцій для вебу. Він використовується для створення референсів і стилістичних концепцій. Дизайнер може ввести опис типу: «Сучасний мінімалістичний лендинг у стилі Japandi з білим фоном і нейтральною типографікою» – і вже через хвилину отримати десятки варіантів дизайну. Такий підхід допомагає швидко визначити напрям майбутнього інтерфейсу, підібрати атмосферу сайту, стиль фотосвітлин або структуру блоку.

Midjourney не замінює Figma, але чудово працює як етап натхнення – коли потрібно знайти естетику проєкту ще до створення першого кадру.

Runway – це платформа для створення анімованого або відеоконтенту без зйомок. За допомогою AI-функцій можна видаляти фони з відео, змінювати об'єкти у кадрі або навіть створювати ролики лише з текстового опису (функція **Text-to-Video**). Для вебдизайнера це корисно під час підготовки **героїв-банерів, промо-анімацій, бекграундів** чи відеозаставок для лендингів. Замість того, щоб шукати готові кліпи або замовляти відеозйомку, дизайнер може самостійно створити матеріал, який ідеально відповідає стилю сайту.

ChatGPT – універсальний помічник для вебдизайнерів. Він допомагає:

- писати тексти для сторінок, кнопок, банерів, підписів до форм;
- формулювати технічні завдання, пояснення для клієнта або розробника;
- структурувати контент, генерувати назви блоків і call-to-action;
- аналізувати UX-рішення і підказувати, як покращити логіку сайту.

Наприклад, якщо дизайнер запитає: «Як зробити головну сторінку зручнішою для користувача?» – ChatGPT може запропонувати покращення у структурі, навігації чи формулюванні текстів. Це не лише економить час, а й допомагає дивитися на дизайн з погляду користувача, а не лише візуальної форми.

Uizard – це інструмент, який автоматично перетворює текстовий опис або ескіз у готовий вебмакет. Достатньо ввести запит: «Лендинг для кав'ярні з меню, відгуками та кнопкою бронювання» – і програма створить робочий прототип із базовою структурою. Uizard зручний на етапі швидких презентацій ідей або тестування сценаріїв користувача. Після цього макет можна доопрацювати у певному графічному редакторі – замінити шрифти, кольори, компоненти й адаптувати під реальний стиль бренду.

AI-інструменти поступово стають стандартом у роботі вебдизайнера. Вони не замінюють професійні навички, але дають змогу працювати стратегічніше: більше часу витратити на ідеї, композицію та UX, а не на рутину. Справжня майстерність вебдизайнера тепер полягає не лише у створенні пікселів, а й у вмінні керувати інтелектом, який допомагає їх розставити.

РОЗДІЛ 2

ВЕКТОРНА ГРАФІКА У ВЕБДИЗАЙНІ

2.1. ОСНОВНІ ПОНЯТТЯ ВЕКТОРНОЇ ГРАФІКИ

Векторна графіка (також геометричне моделювання або об'єктно-орієнтована графіка) – це створення зображення в комп'ютерній графіці зі сукупності геометричних примітивів: точок, ліній, кривих, полігонів, тобто об'єктів, які можна описати математичними виразами.

Багатокутники (Полігони) – фігури, які складаються з кількох ліній, з'єднаних між собою, що утворюють замкнений контур. Полігони можуть бути простими (трикутники, чотирикутники) або складними.

Контури – серії ліній і кривих, що можуть бути відкритими або замкненими. Використовуються для створення складних форм і фігур.

Заливка (Фон) – визначає, як кольори заповнюють багатокутники чи контури. Заливка може бути однотонною, градієнтною чи візерунковою.

Обведення (Межі) – визначає властивості країв форм, таких як товщина, колір і стиль (суцільна, пунктирна, штрихова).

Трансформації – операції, що змінюють положення, розмір і орієнтацію об'єктів. Основні типи трансформацій охоплюють масштабування, обертання, переміщення та відображення.

ПЕРЕВАГИ ТА НЕДОЛІКИ ВЕКТОРНОЇ ГРАФІКИ

Переваги:

- **Масштабованість без втрати якості.** Оскільки векторні зображення ґрунтуються на математичних формулах, а не на пікселях, рисунок матиме однакову роздільну здатність на носіях різного розміру.
- **Малий розмір файлів.** Через відсутність пікселів векторні файли зазвичай менші, ніж растрові зображення.

- **Редагування та модифікації.** Векторні зображення легко редагувати, змінюючи окремі елементи, кольори, контури тощо.
- **Простота анімації.** Векторна графіка часто використовується в анімації, оскільки об'єкти легко трансформувати й анімувати.

Недоліки:

- **Обмеженість у деталях.** Векторна графіка може не застосовуватися для зображення складних і детальних зображень, таких як світлини або текстури, де потрібно багато дрібних деталей і плавних переходів кольорів.
- **Складність створення.** Створення векторних зображень може бути складнішим і вимагати більше часу, особливо якщо йдеться про створення складних ілюстрацій.
- **Обмежена підтримка ефектів.** Векторна графіка може мати обмеження щодо використання певних ефектів, таких як градієнти й тіні, які можна легко створити у растровій графіці.
- **Сумісність.** Щоб відкривати векторні файли та працювати з ними, знадобиться програма для роботи з векторними зображеннями.
- **Труднощі з конвертуванням.** Зазвичай значно складніше перетворити растрове зображення у векторне, ніж навпаки, оскільки для цього потрібна вища обчислювальна потужність.

ГРАФІЧНІ ФОРМАТИ ВЕКТОРНОЇ ГРАФІКИ

Найпоширенішими типами векторних файлів є:

- **SVG (Scalable Vector Graphics).** Використовується для описування двовимірної векторної графіки. Підтримується більшістю веббраузерів і графічних редакторів. Зберігає зображення у форматі .xml, що робить його легко читабельним і редагованим. Підтримує анімацію й інтерактивність;
- **EPS (Encapsulated PostScript).** Формат файлу для зберігання векторної графіки, тексту та растрових зображень. Зазвичай використовується у видавничій справі та для обміну графічними файлами між різними додатками;

- **PDF (Portable Document Format).** Широко використовується для публікацій та обміну документами. Підтримує векторну графіку, тексти і растрові зображення. Має підтримку багатошарової графіки;
- **AI (Adobe Illustrator), CDR (CorelDRAW).** Використовуються у графічному дизайні, ілюстраціях і верстці. Підтримують велику кількість функцій: шари, ефекти, стилі. Мають проблеми із сумісністю з іншими програмами.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ВЕКТОРНОЇ ГРАФІКИ

Платне:

- **Adobe Illustrator.** Один із найвідоміших і потужних інструментів для створення векторної графіки. Він широко використовується дизайнерами для створення логотипів, іконок, ілюстрацій, вебдизайну й інших видів графіки;
- **CorelDRAW.** Ще один популярний інструмент, який має чималу кількість функцій для створення та редагування векторних зображень. CorelDRAW часто використовується для графічного дизайну, оформлення друкованих матеріалів і вебдизайну;
- **Adobe XD.** Додаток інтегрований з іншими продуктами Adobe. Має можливості прототипування та дизайну, командної роботи в режимі реального часу;
- **Affinity Designer.** Програмне забезпечення, яке є конкурентом Adobe Illustrator. Підтримує одночасну роботу з растровими та векторними зображеннями;
- **Sketch.** Потужний інструмент для UI/UX дизайну з великим набором плагінів;
- **InVision Studio.** Прототипування, анімації та командна робота в режимі реального часу;
- **Axure RP.** Потужні інструменти для створення прототипів із можливістю складної логіки;
- **Xara Designer Pro.** Професійний набір інструментів для створення векторної графіки, що підходить для дизайну реклами, логотипів та вебграфіки.

Безкоштовне:

- **Figma**. Має безкоштовну та платну версії. Онлайн-інструмент для дизайну інтерфейсів і спільної роботи, який підтримує створення векторної графіки. Figma дозволяє командам працювати над проектами у режимі реального часу;
- **Penpot**. Відкритий вихідний код, дозволяє спільну роботу в режимі реального часу. Є веббазованим інструментом;
- **Gravit Designer**. Інструмент для векторної графіки, вебверсія та десктопний додаток. Має безкоштовну та платну версії;
- **Lunacy**. Редактор графіки з підтримкою Sketch-файлів, підтримує інтеграцію з хмарними сервісами;
- **Vectr**. Простий у використанні інструмент для створення векторної графіки;
- **Inkscape**. Вільне програмне забезпечення з відкритим кодом для створення векторної графіки.

Розглянемо принципи та засоби побудови векторних зображень на прикладі безкоштовного програмного забезпечення Figma.

2.2. ОСНОВНІ ІНСТРУМЕНТИ ВЕКТОРНИХ РЕДАКТОРІВ

Векторна графіка є одним із основних інструментів для створення та редагування графічних зображень у сучасних комп'ютерних програмах.

Графічні редактори, що працюють із векторною графікою, надають потужні інструменти для створення і редагування векторних зображень. Такі редактори дають змогу створювати чіткі, детальні й якісні ілюстрації: від логотипів і рекламних матеріалів до іконок та інтерфейсів програм.

Розглянемо основні концепції векторної графіки, зокрема основні функції та можливості графічних редакторів, на прикладі такого інструменту, як Figma.

Figma — це онлайн-інструмент векторної графіки для дизайну інтерфейсів і спільної роботи над проектами. Він дає змогу створювати макети, прототипи, інші елементи дизайну, а також співпрацювати з іншими користувачами і розробниками у режимі реального часу. Figma популярний інструмент завдяки доступності

через веббраузер, що дає змогу користувачам працювати з будь-якого пристрою без необхідності встановлення додаткового програмного забезпечення.

Більшість функцій Figma є у вільному доступі, хоча існують і платні додатки для великих команд і проєктів. Все, що потрібно починаючому користувачеві, йде у комплекті. Цей графічний редактор невимогливий до ресурсів комп'ютера і працюватиме на будь-якому пристрої (потрібен лише браузер). Також немає значення, яка операційна система використовується на пристроях (Windows, MacOS або Linux).

Figma забезпечує найширші можливості: від векторного редактора до створення інтерактивного прототипу (прототипування), який можна миттєво протестувати, наприклад, у смартфоні. Окрім того, цей програмний засіб забезпечує спільне редагування дизайну, зберігання історії версій тощо. Увесь згаданий інструментарій розглядатиметься нижче.

ВСТАНОВЛЕННЯ FIGMA

Figma має версії для роботи в операційних системах Mac і Windows, а також безкоштовну програму **Figma Mirror** для iOS і Android. Усіх їх можна завантажити на офіційному сайті: figma.com/downloads.

Для початку роботи потрібно зареєструватися на сайті Figma <https://www.figma.com/> за допомогою настільного комп'ютера або ноутбука. Смартфон для реєстрації використовувати не можна (з нього можна лише переглядати раніше створений дизайн). Для реєстрації користувачу знадобиться робоча електронна поштова скринька (email). Вона використовуватиметься як логін.

РОБОЧИЙ ПРОСТІР FIGMA

ПРОФІЛЬ

У налаштуваннях профілю можна завантажити аватарку, змінити ім'я, пароль і електронну пошту, на які приходитимуть повідомлення. Там же можна оновити тариф і видалити обліковий запис.

ПРОСТІР ІЗ ФАЙЛАМИ

Простір із файлами у Figma, або, так званий, **Workspace** дозволяє організовувати файли проєктів, ділитися ними з іншими користувачами та керувати правами доступу (Рис. 12).

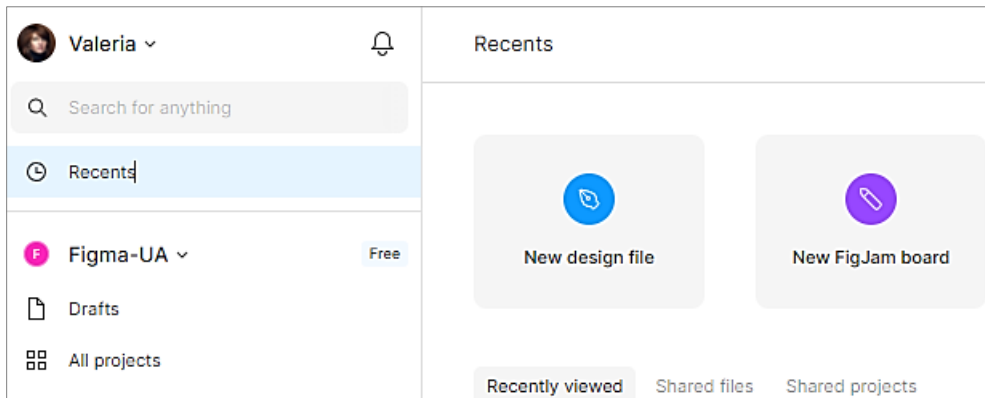


Рис. 12. Простір з файлами у Figma (Workspace)

Розділ **Drafts** (Чернетки) – це особистий простір користувача для зберігання файлів, над якими він працює. Чернетки доступні лише користувачу, поки він не вирішить поділитися ними або перемістити в інший простір, наприклад, у командний проєкт. Тобто файли у цьому розділі є приватними.

Розділ **Recents** (Нещодавні) показує всі файли, з якими користувач нещодавно працював або відкривав, незалежно від того, де вони зберігаються (у чернетках, у командних проєктах чи спільних папках). У цьому розділі можуть бути і особисті файли користувача, і файли, якими поділилися інші користувачі. Це зручний спосіб швидко знайти та повернутися до роботи над останніми проєктами, над якими велася робота.

КОМАНДИ

Команди – потрібні для того, щоб організувати роботу групи людей, які працюють над кількома проєктами.

Після встановлення Figma завжди створена одна команда за замовчуванням. На сторінці команди доступні проєкти, які до неї належать, а також налаштування. У безкоштовному обліковому записі немає можливості працювати у команді.

Щоб створити нову команду, натискаємо **Create new** (Рис. 13).

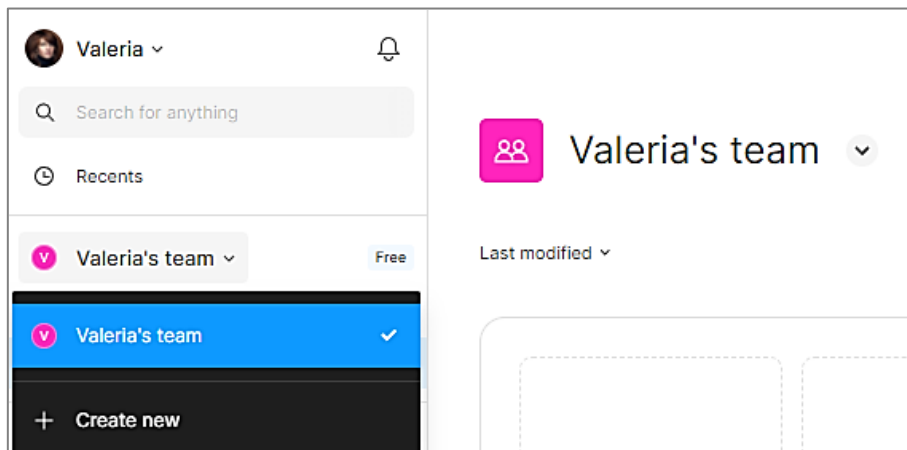


Рис. 13. Створення команди у Figma

Після створення команди можна запросити інших користувачів приєднатися до проєкту. В запрошеного користувача у вікні Figma у верхньому правому куті з'явиться повідомлення, яке можна прийняти або відхилити (Рис. 14).

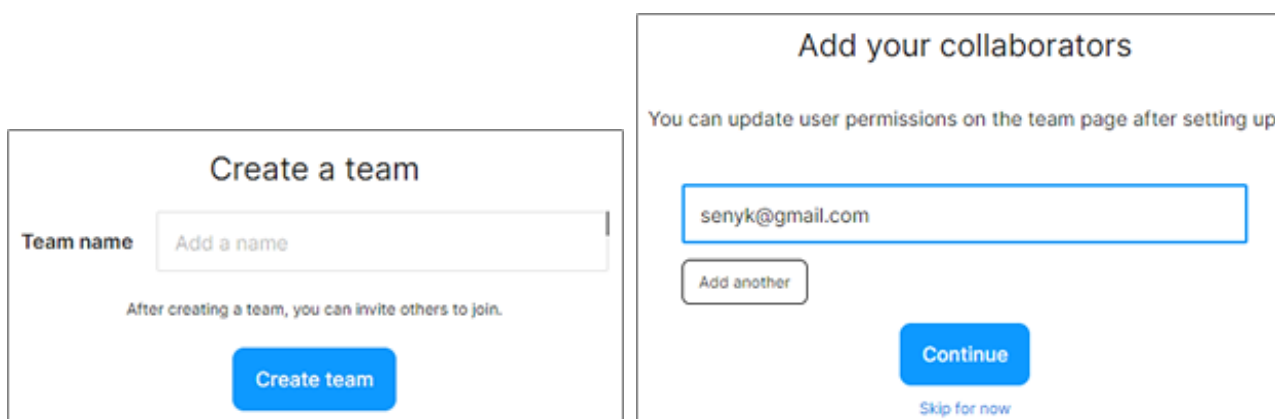


Рис. 14. Запрошення користувачів до командного проєкту

Функція **Enable Libraries** дозволяє зробити доступними бібліотеки для всіх файлів команди.

Функція **Enable Slack OAuth** дозволяє авторизуватися у Figma та приєднатися до команди, якщо авторизований у Slack.

ПРОЄКТИ

Проекти групують файли. Також файли можна створювати одразу в проєктах. У безкоштовному обліковому записі може бути до трьох проєктів. Після встановлення Figma завжди створено один проєкт **Team project** за замовчуванням.

Створення проєкту

Для створення проєкту на сторінці команди потрібно натиснути кнопку **+Project** (Рис. 15).

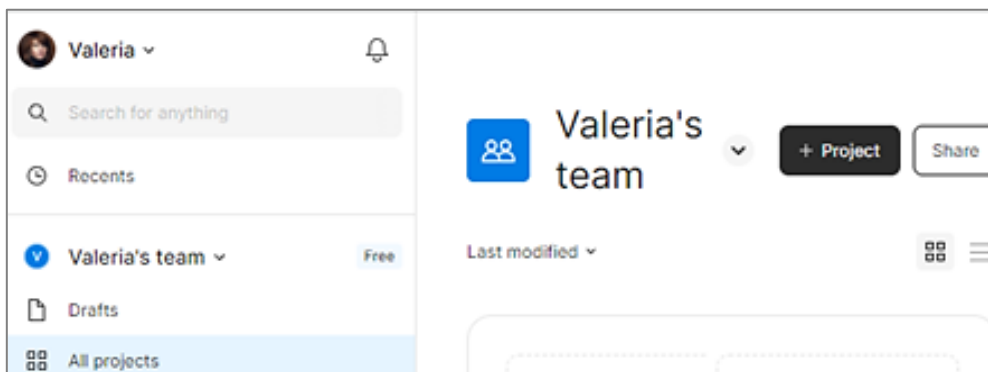


Рис. 15. Створення командного проєкту

Під час створення проєкту можна визначити **рівень доступу**:

- **Everyone at Team can edit** – будь-який член команди може редагувати файли всередині нього;
- **Everyone at Team can view** – усі можуть лише переглядати файли;
- **Invite-only** – доступ до проєктів за індивідуальними запрошеннями.

У безкоштовній версії можна створити лише один проєкт.

ФАЙЛИ

Файли – аналог скетч-проєктів. У них можна створювати макети. За замовчуванням макети створюються у загальній категорії **Drafts**. На файли можна давати гіперпосилання. Також файли можна вивантажувати з Figma у форматі **.fig**. Програма надає можливість створювати необмежену кількість файлів. Файли Figma можуть мати різні рівні доступу – лише перегляд, коментування, повний доступ до редагування.

Створення файлу

У разі, якщо файл буде створюватись за межами будь-якого проєкту, слід натиснути кнопку **Create new** у розділі **Drafts** (Рис. 16). Відкриється нове полотно, де користувач може розпочати створювати свій дизайн.

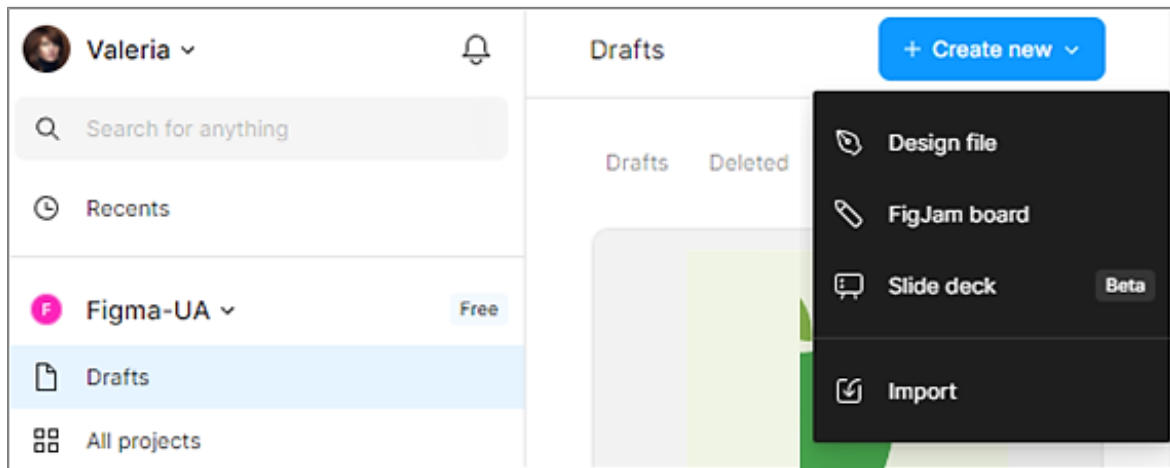


Рис. 16. Створення файлу за межами проєкту

Для того, щоб повернення на сторінку проєкту, необхідно у верхньому лівому куті натиснути на кнопку головного меню програми і виконати команду **Back to files** (Рис. 17).

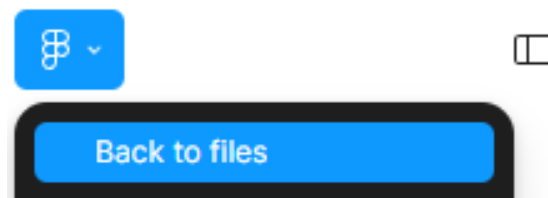


Рис. 17. Повернення до проєкту

Якщо файл створюватиметься у проєкті, то слід зайти у проєкт і натиснути на кнопку **Create new**. Відкриється нове полотно для роботи. Зліва від назви відкритого файлу буде видно кнопку повернення на сторінку активного проєкту.

ІМПОРТ

У Figma можна імпортувати файли з розширеннями **.fig** або **.sketch**. Щоб здійснити імпорт файлу, слід натиснути на кнопку **Import** у вікні **Workspace**.

ІНТЕРФЕЙС КОРИСТУВАЧА FIGMA

Інтерфейс будь-якого графічного редактора можна умовно розділити на кілька ключових елементів. Кожен із них виконує специфічну функцію й активно використовується під час роботи. Інтерфейс користувача Figma показано на Рис. 18. Розглянемо кожну його частину.

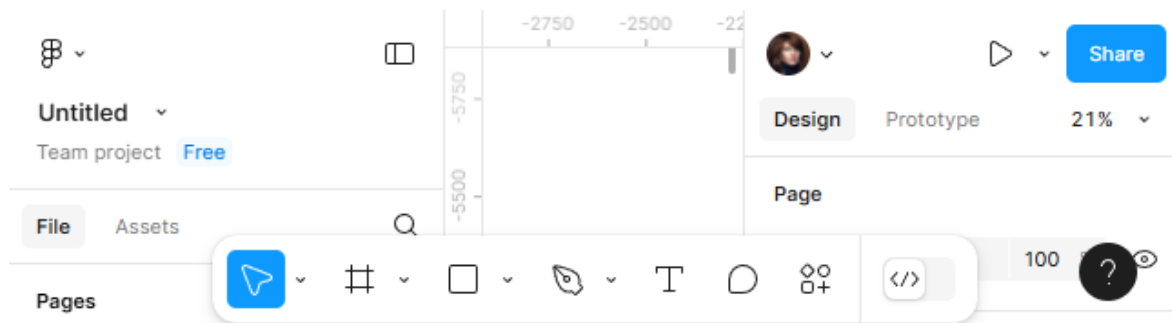


Рис. 18. Інтерфейс користувача Figma

ГОЛОВНЕ МЕНЮ І ПАНЕЛЬ ІНСТРУМЕНТІВ

ГОЛОВНЕ МЕНЮ

Головне меню – складається зі списку команд і розкривається синьою кнопкою у верхньому лівому куті екрану (Рис. 19). У меню є пошук за пунктами. Пошук **Quick actions** шукає пункти меню, а не елементи, які користувач створював. Ця опція стане у нагоді, якщо користувачу потрібна конкретна команда, але він не пам'ятає, де саме вона знаходиться.

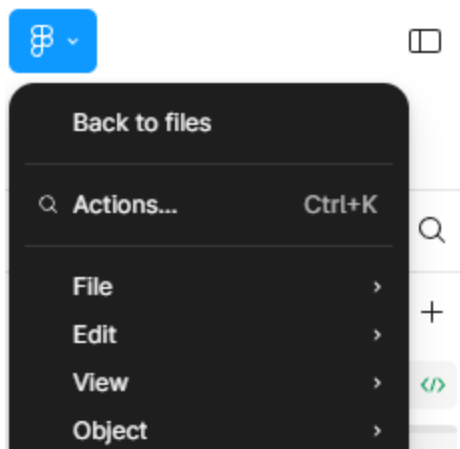


Рис. 19. Головне меню Figma

Підменю **File** допомагає застосувати дію до файлу, з яким користувач працює. Наприклад, за допомогою **New from Sketch file** можна імпортувати файли, створені у графічному редакторі **Sketch**.

Figma – це інструмент для спільної роботи, вона автоматично зберігає зміни файлу. Користувач також може зберегти зміни негайно, застосувавши команду **Save to Version History**, або переглянути історію змін, які внесли інші користувачі до проєкту, активувавши команду **Show Version History**.

Щоб зберегти файл Figma у PDF використовується команда **Export Frames to PDF**. Ця опція стане у нагоді, наприклад, під час підготовки презентацій.

Підменю **Edit** (Рис. 20) дозволяє працювати з елементами дизайну в файлі. Щоб застосувати дію до елемента, його потрібно виділити. Наприклад, об'єкт можна скопіювати як код CSS, SVG або PNG-файл за допомогою команди **Copy As**.

Функція **Paste Over Selection** дозволяє розмістити скопійований елемент у верхньому лівому куті іншого об'єкта.

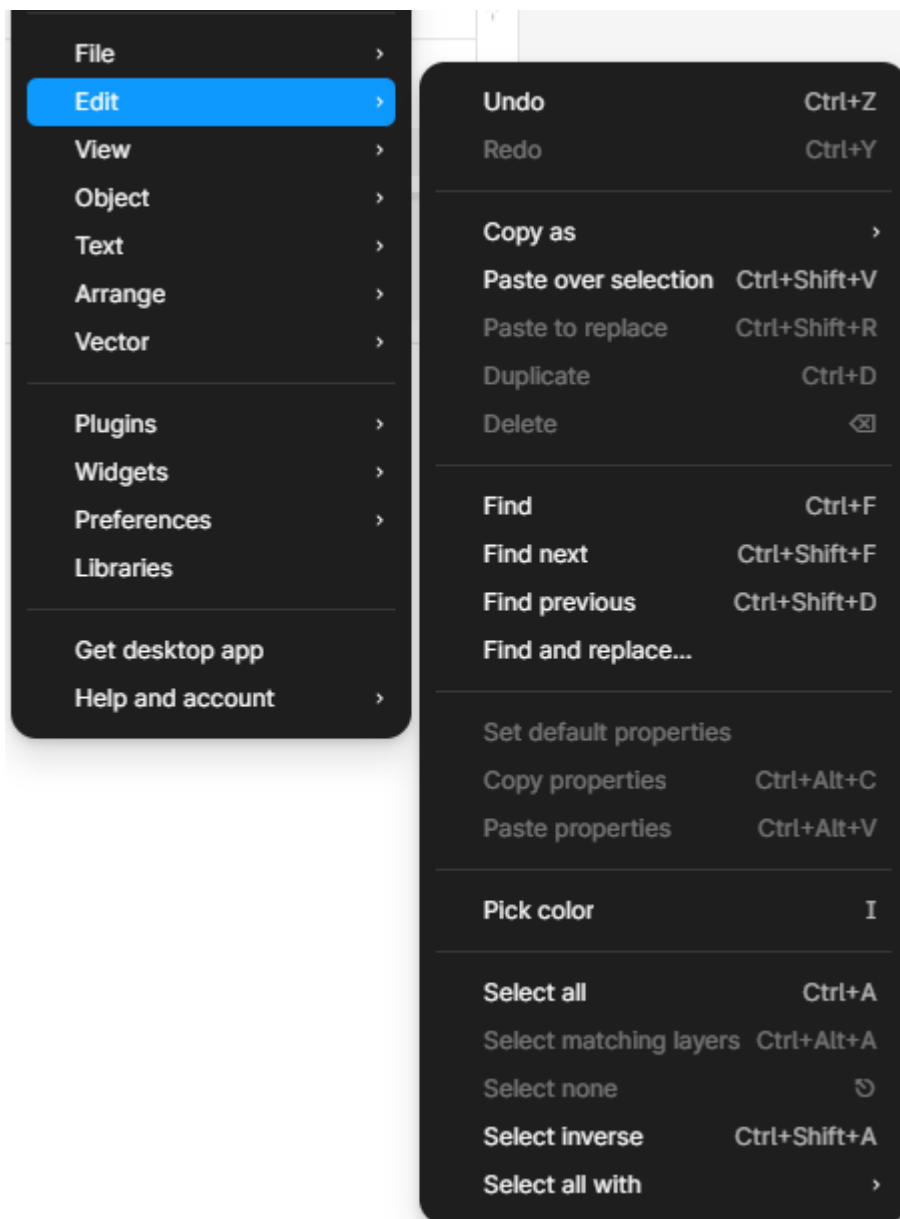


Рис. 20. Підменю Edit

Усі нові об'єкти, створені у файлі, за замовчуванням відображаються сірим кольором. Виникають ситуації, коли потрібно створити безліч об'єктів з іншим

кольором. У такому разі потрібно застосувати команду **Set Default Properties**: скопіювати колір потрібного об'єкта один раз і всі наступні об'єкти будуть створюватися із заданими властивостями. Також можна зафарбувати об'єкт у потрібний колір за допомогою піпетки **Pick Color**.

Група команд **Select All with** підсвічує всі схожі об'єкти у файлі: з однаковими властивостями, заливкою, шрифтами, ефектами тощо.

Підменю **View** (Рис. 21, Рис. 22) знаходиться ліворуч у меню редактора (нижче підменю **Edit**) та дублюється праворуч. Ця команда відповідає за масштабування макета і навігацію по ньому.

Із-поміж значної кількості опцій слід зупинитися на функції **Pixel Preview**. Оскільки Figma – векторний редактор, тож у такому випадку користувач під час роботи бачить рівні векторні лінії. Використавши перегляд об'єкта у пікселях, можна розглянути, як елемент виглядатиме на екрані.

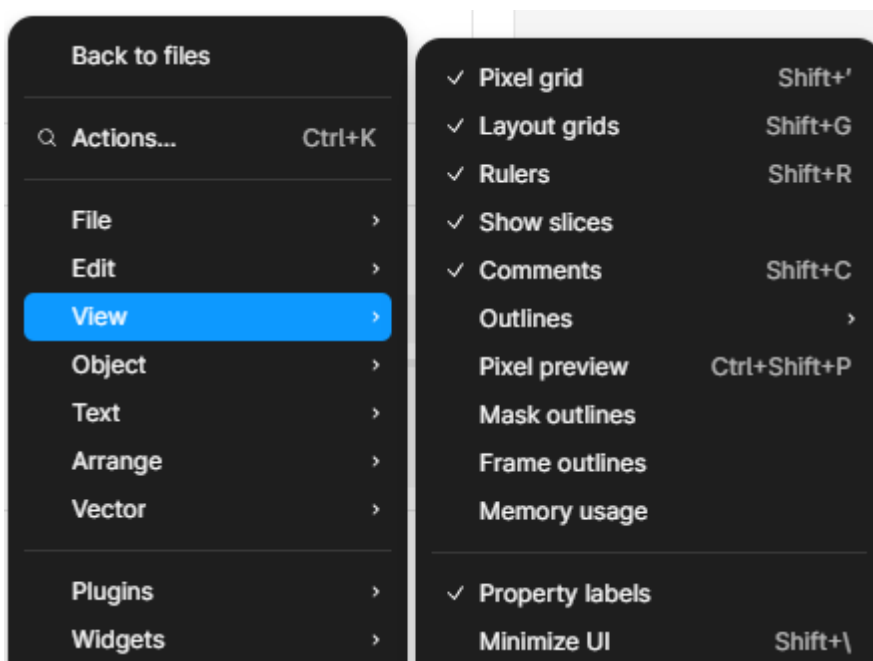


Рис. 21. Підменю View

Масштабувати макет можна функціями **Zoom In/Zoom Out**. Щоб показати лінійки та витягнути напрямні слід використати команду **Rules**. Для розширення робочої області можна приховати панель шарів та інтерфейс команд **Layers Panel, Show/Hide UI**.

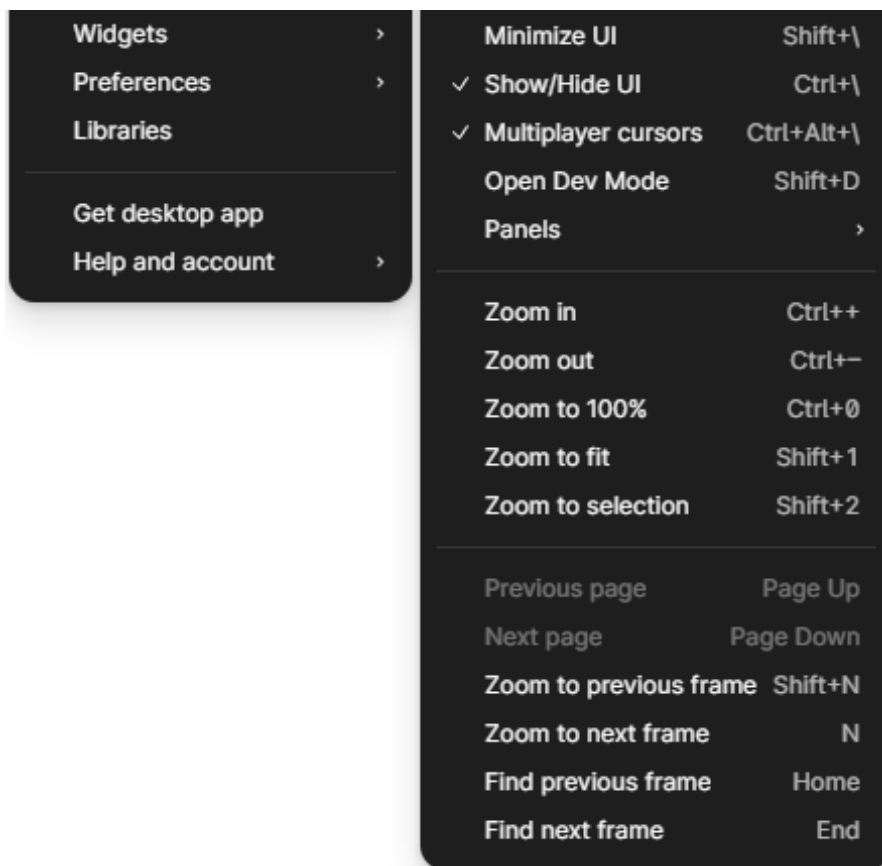


Рис. 22. Підменю View

Підменю **Preferences** (Рис. 23) відповідає за навігацію у файлі.

Перші три пункти – **Snap to Geometry**, **Snap to Pixel Grid**, **Snap to Objects** – забезпечують налаштування прив'язки. За їх допомогою можна злегка «примагнітити» елементи й уникнути «кривої» сітки під час швидкого креслення.

Highlight Layers on Hover – команда, яку краще тримати ввімкненою завжди. Ця функція підсвічує дрібні елементи дизайну (кнопки, літери, іконки, значки) під час наведення на них курсора.

Увімкнена функція **Keyboard Zooms into Selection** наближатиме / вилучатиме макет відносно елемента, вибраного на екрані.

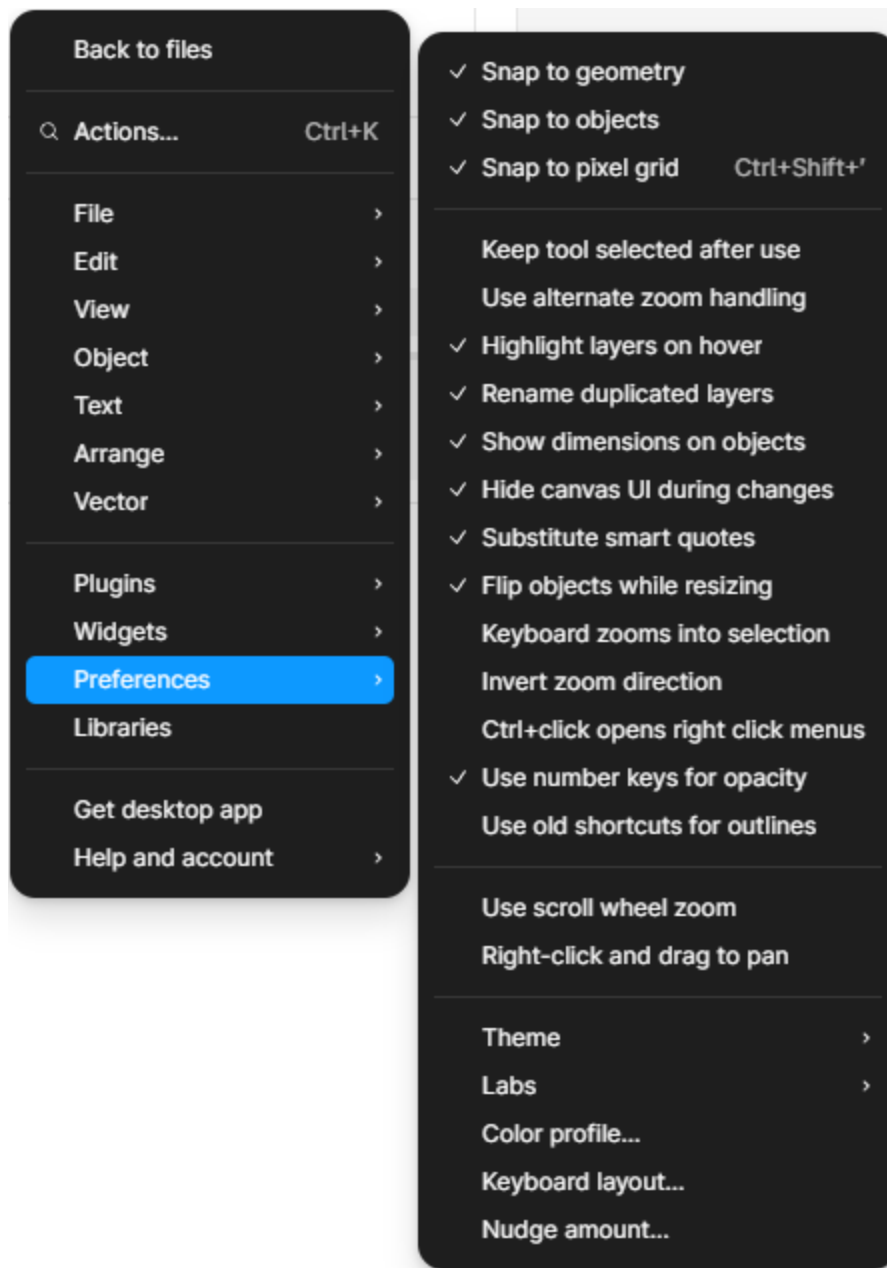


Рис. 23. Підменю Preferences

ПРИХОВУВАННЯ ІНТЕРФЕЙСУ

Для приховування інтерфейсу в Figma для Windows використовується комбінація клавіш **Ctrl** + ****.

ПЕРЕЙМЕНУВАННЯ ФАЙЛУ

Під кнопкою головного меню вказано назву **проекту / ім'я поточного файлу**. За замовчуванням файл матиме назву **Untitled**, яку можна змінити, вибравши у контекстному меню команду **Rename** (Рис. 24).

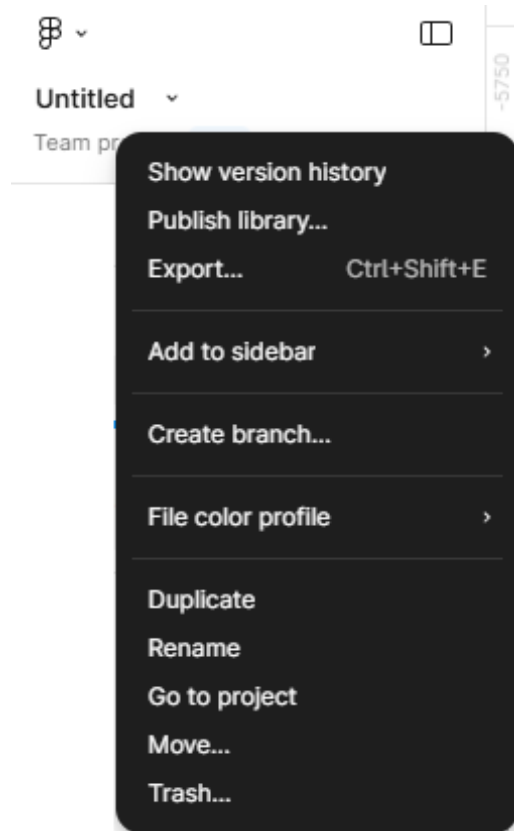
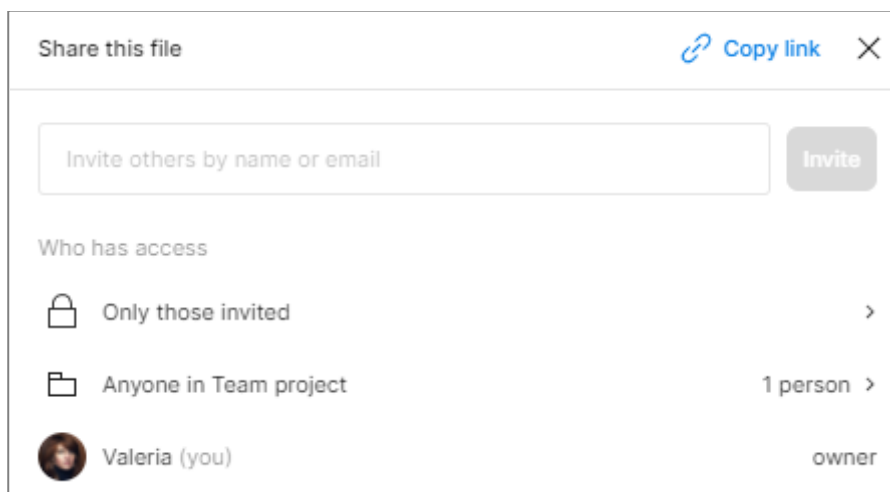


Рис. 24. Перейменування проєкту в Figma

НАДАННЯ ДОСТУПУ ЗА ПОСИЛАННЯМ

Блакитна кнопка **Share** у правій частині панелі інструментів дає змогу налаштувати рівень доступу до файлу (лише для перегляду або для редагування). Для цього слід вибрати опцію **Anyone with the link** (будь-хто за посиланням), щоб надати доступ за посиланням, натиснути кнопку **Copy link** і поділитися цим посиланням із тими, кому потрібно надати доступ (Рис. 25).



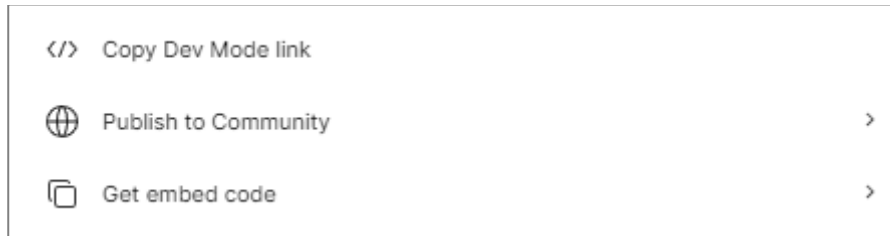


Рис. 25. Надання доступу до файлу в Figma

Надаючи доступ за посиланням, потрібно пам'ятати, що той, у кого є таке посилання, має можливість зберегти вихідний проєкт у себе. Це може бути небажано у разі, якщо користувач турбується про безпеку проєкту. Нині Figma працює над вирішенням цієї проблеми.

ПАНЕЛЬ ІНСТРУМЕНТІВ (TOOLBAR)

Панель інструментів у Figma працює як і в інших редакторах. Надалі в інших підрозділах буде розглянуто кожен інструмент детально.

РОБОЧА ОБЛАСТЬ

У центрі екрану розташовано сіре безрозмірне полотно. Воно може бути величезним і його достатньо для створення проєкту будь-якого розміру. По полотну можна ковзати, затиснувши **пробіл** і перетягуючи простір лівою кнопкою мишки. Також можна затискати **колесо мишки**. На полотні існують нульові координати: **x: 0**, **y: 0**. Це центр, відносно якого позиціонуються всі шари, в яких немає батьківських шарів. Шар будь-якого типу має базові властивості: координати **x**, **y**, **ширину**, **висоту** та **кут нахилу**.

ФОН РОБОЧОЇ ОБЛАСТІ

Колір фону робочої області можна задати на панелі **Design**, що розташована справа (Рис. 26).

Figma може писати назви фреймів блакитним, а компонентів фіолетовим кольором. Тому під час установлення (зміни) кольору фону робочої області варто враховувати, що він має бути таким, щоб дрібні написи у заголовках екранів читалися.

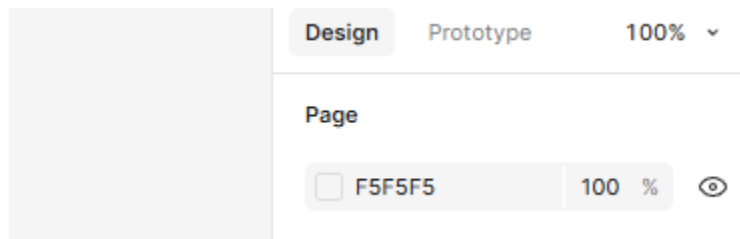


Рис. 26. Встановлення кольору фону робочої області

Око навпроти рядка із кольором дозволяє робити прозоре тло робочої області з канонічними шашечками (Рис. 27).



Рис. 27. Прозоре тло робочої області з канонічними шашечками

Для темного фонового кольору зручно використовувати сірий #333333, а для світлого – #F2F2F2. Щоб встановити ці кольори, достатньо ввести у поле **Background** з клавіатури **33** або **F2**.

Встановлювати тло темним зручно, коли робота над проектом є тривалою, особливо у темній кімнаті. Що менше світла потрапляє у вічі, то комфортніше.

Налаштування кольору на певній сторінці є глобальним для всіх учасників команди, тому слід домовитися про той колір, який комфортний для всіх.

Не варто робити полотно білим, інакше кадри будуть недостатньо контрастними, навіть якщо їх контури видно.

РОБОТА З МАСШТАБОМ

У вікні **Zoom** у верхньому правому куті вікна програми відображається значення масштабу у відсотках (Рис. 28).

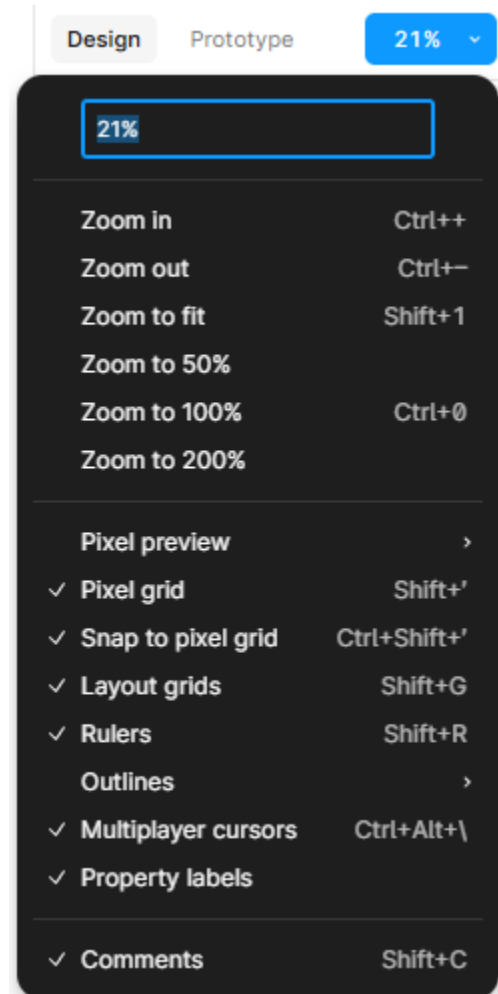


Рис. 28. Вікно налаштування масштабу відображення

Для зміни масштабу потрібно натиснути його і вибрати або ввести комбінацією клавіш потрібне значення:

- **Zoom in, Ctrl + Plus** – наближає у двічі;
- **Zoom out, Ctrl + Minus** – віддаляє у двічі;
- **Zoom To Fit, Shift + 1** – виставляє таке значення масштабування та кадрування, щоб увесь контент сторінки помістився у робочій області (зручна команда, щоб оглянути всю сторінку);
- **Zoom To Selection, Shift + 2** – виставляє такий розмір і кадрування, щоб виділений об'єкт зайняв максимум простору робочої області (команда, яка спрощує налаштування масштабу вікна під певний шар на полотні). Якщо виділено маленьку іконку або групу екранів, ця команда покаже виділене на весь доступний розмір робочої області;
- **Zoom to 100%, Shift + 0** – відображає фактичний масштаб проєкту.

РЕЖИМИ ВІДОБРАЖЕННЯ РОБОЧОЇ ОБЛАСТІ

Режими відображення робочої області доступні у вікні **Zoom**.

Pixel Preview, **Ctrl + Shift + P** – це режим, який допомагає побачити, як дизайн виглядатиме на реальних екранах, показуючи, як елементи вирівнюються за піксельною сіткою (Рис. 29).

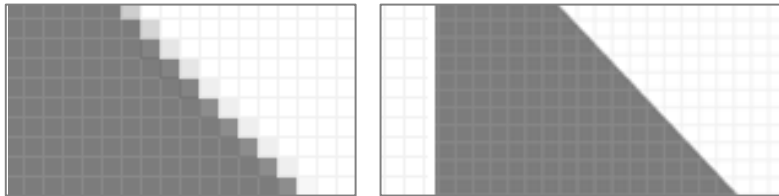


Рис. 29. Режим відображення *Pixel Preview*: зліва: вимкнено, праворуч: увімкнено

Pixel Grid, **Shift + ‘** – тогл-режим¹, в якому показується піксельна сітка (Рис. 30).

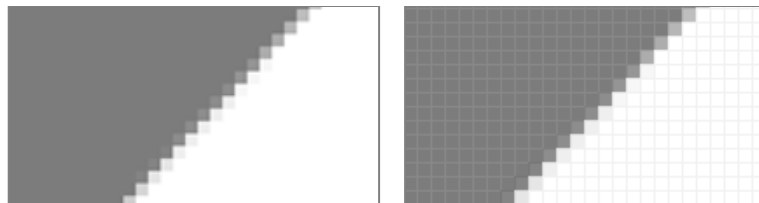


Рис. 30. Режим відображення *Pixel Grid*: зліва: вимкнено, праворуч: увімкнено

Snap To Pixel Grid, **Ctrl + Shift + ‘** – тогл-режим прив’язування шарів до сітки під час створення або переміщення. Якщо цей режим увімкнений, то всі нові або рухомі шари ставляться на рівні пікселів. Якщо вимкнений, то вони можуть потрапити на дробові значення пікселів (напівпікселі). У цьому випадку координати матимуть числа після крапки, наприклад: 2.29. При щільності пікселів 1x контури такої фігури будуть відображатися нечітко (Рис. 31).

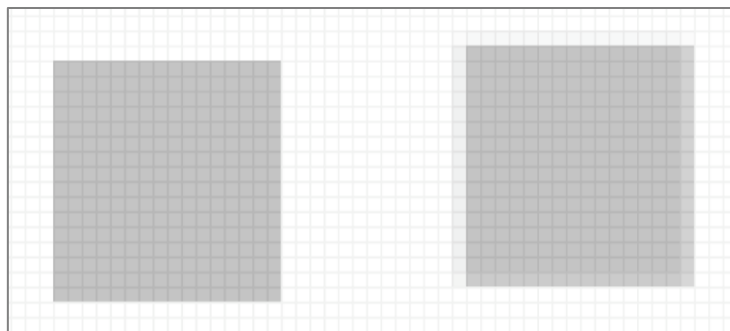


Рис. 31. Приклад застосування режиму відображення *Snap To Pixel Grid*

¹ **Тогл-режим** – режим роботи, який може бути ввімкнений або вимкнений.

² **Blur** – графічний ефект, який використовується на фотографіях, відео та відеоіграх, що розмиває зображення.

Layout Grids, Shift + G – тогл-режим, який вмикає та вимикає сітки для всіх кадрів одночасно. Це зручний спосіб швидко переключатися між режимами зі сіткою і без неї під час роботи над макетами.

**Multiplayer Cursors, Ctrl + Alt + ** – відключення курсорів інших користувачів, які одночасно працюють у файлі.

ЛІНІЙКИ

Для увімкнення лінійки використовується команда **Rulers, Shift + R (Main menu → View → Rulers)**. Відключити їх можна таким самим способом.

Напрявні (Guides) – це віртуальні лінії, які допомагають створювати макети, забезпечуючи точність вирівнювання елементів. Напрявні витягуються з горизонтальної та вертикальної лінійки (Рис. 32). Щоб увімкнути / вимкнути напрямні у Figma, потрібно натиснути ту саму комбінацію клавіш **Shift + R**. Це приховує / показує і лінійки, і напрямні.

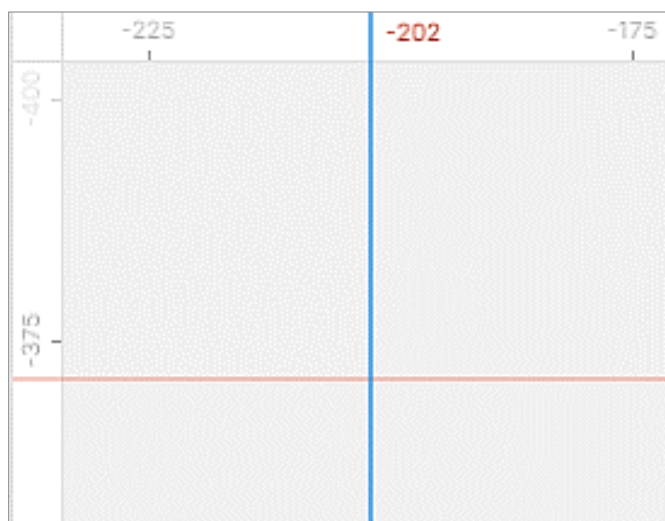


Рис. 32. Використання лінійки (напрявних)

ПАНЕЛЬ ШАРІВ

Панель шарів – зберігає список усіх створених об'єктів. Тут визначається, в якій послідовності їх відображати. На панелі шарів можна вибирати об'єкти так, як на робочій області. Це лише інший спосіб виділити об'єкт, щоб відредагувати або змінити його властивості.

Панель шарів **Layers** розташована зліва від робочої області (Рис. 33).

На вкладці **Layers** вказується список усіх шарів, які існують у робочій області. У кожного шару є контекстне меню (Рис. 34), де можна переглянути більшість гарячих клавіш.



Рис. 33. Панель шарів. Вкладка *Layers*

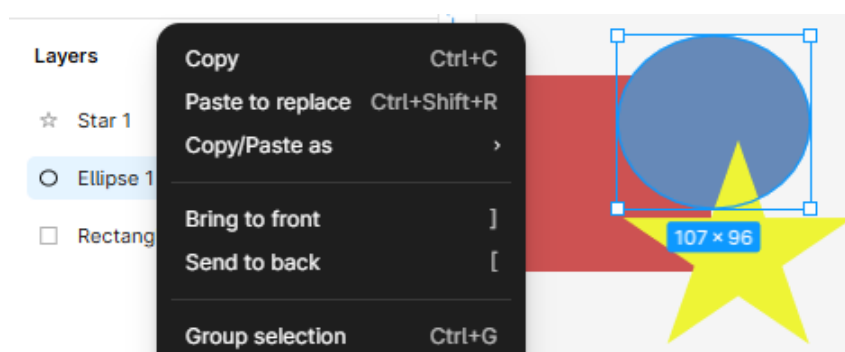


Рис. 34. Контекстне меню активного шару

КЕРУВАННЯ РЕСУРСАМИ ТА КОМПОНЕНТАМИ

Для керування ресурсами та компонентами, які користувач може використовувати у проєкті, використовується вкладка **Assets** (Рис. 35). Тут можна переглядати та використовувати встановлені ресурси; отримувати доступ до бібліотек компонентів, які можна додати до проєкту; керувати власними ресурсами та компонентами; здійснювати пошук необхідних елементів за ключовими словами.

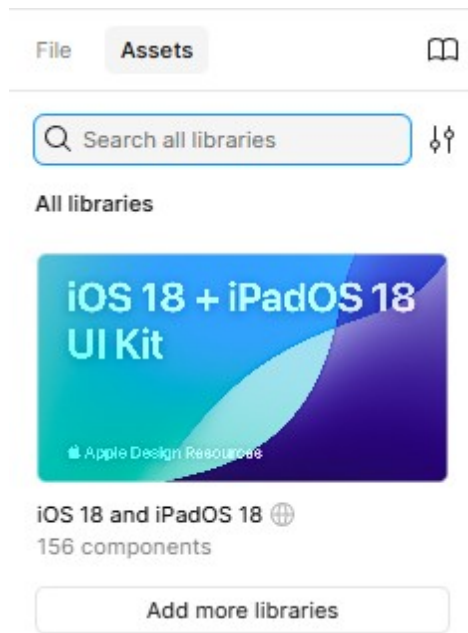


Рис. 35. Панель шарів. Вкладка Assets

Таблиця 1. Найпоширеніші комбінації клавіш під час роботи із шарами

Комбінація клавіш	Застосування
Ctrl + R	Перейменувати шар. Клавіша Enter використовується для збереження після перейменування.
Tab	Виділити наступний.
Ctrl + D	Дублювати вибраний шар. Новий шар з'явиться у списку шарів нижче першого. На полотні він буде правіший за свого «батька». Якщо натискати кілька разів, шари дублюватимуться на тому ж відступі, який був між оригіналом і першим дубляжем. Можна також затиснути Alt та тягнути шар, який потрібно дублювати.
Shift + Tab	Виділити попередній.
Enter	Виділити об'єкти, що знаходяться в активному фреймі, компоненті або групі.
Shift + Enter	Виділити батьківський шар.
Ctrl + A	Виділити всі шари у групі.

Ctrl + Shift + A	Виділити всі шари в артборді.
Shift + Ctrl + L	Заблокувати або розблокувати шар. Заблокований шар залишається видимим, але його не можна рухати та виділяти.
Shift + Ctrl + H	Приховати / показати шар.
Ctrl +] або Ctrl + [	Переміщення шарів вгору / вниз.
Ctrl + Shift +] або Ctrl + Shift + [	Переміщення шарів на передній / задній план.
Ctrl + G	Групування шарів.
Ctrl + Shift + G	Розгрупування шарів.
Ctrl + C	Копіювання шарів.
Ctrl + V	Вставлення шарів.
Ctrl + Alt + V	Вставлення шарів на те ж місце.
Ctrl + Alt + L	Вирівнювання шарів за лівим краєм.
Ctrl + Alt + C	Вирівнювання шарів по центру.
Ctrl + Alt + R	Вирівнювання шарів за правим краєм.
Ctrl + Alt + T	Вирівнювання шарів за верхнім краєм.
Ctrl + Alt + B	Вирівнювання шарів за нижнім краєм.
Ctrl + Shift + > , Ctrl + Shift + <	Збільшити / зменшити шар.

СТОРІНКИ

У Figma кожен проєкт складається зі сторінок, на яких розміщуються кадри (Frames), компоненти й інші елементи дизайну. В Figma можна створити кілька сторінок у межах одного файлу, натиснувши **+** поруч із заголовком **Pages** (Рис. 36).

Між сторінками можна переключатися клавішами **PageUp**, **PageDn**. Кожна сторінка містить свою робочу область. Сторінки можна перейменовувати, перетягувати у потрібному порядку або використовувати їх для організації проєкту, наприклад, мати окремі сторінки для дизайну на різних пристроях або для різних етапів розробки.

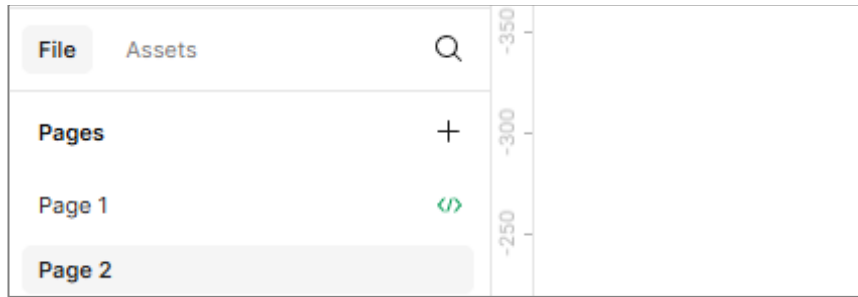


Рис. 36. Створення сторінок у межах одного файлу

Для швидкого створення нової сторінки можна використовувати комбінацію клавіш **Shift + P**.

ПЕРЕМІЩЕННЯ ШАРІВ МІЖ СТОРІНКАМИ

У Figma не доступна функція перетягування кадру з однієї сторінки до іншої. Для переміщення слід використовувати команду **Move To Page** у контекстному меню.

ПАНЕЛЬ ВЛАСТИВОСТЕЙ

У правій частині інтерфейсу розташована велика біла панель на всю висоту вікна – **панель властивостей**. Ця панель має три вкладки: **Design**, **Prototype** та **Inspect**. Вкладка **Inspect** є тільки в платній версії (Рис. 37).

Вкладка **Design** – основний розділ інтерфейсу, де користувачі створюють і редагують свої проекти.

Вкладка **Prototype** – призначена для створення інтерактивних прототипів на основі макетів користувача.

Вкладка **Inspect** – призначена для розробників і дозволяє отримувати необхідні дані з макетів для реалізації у коді.

Залежно від виділеного типу об'єкта у панелі властивостей може знаходитись різний набір полів і кнопок. Якщо жоден зі шарів не обрано, то на цій панелі відображатимуться налаштування робочої області. Зокрема, у такій панелі відображатиметься уже розглянуте нами, налаштування кольору робочої області та порожні блоки: **Local variables** для створення багаторазових чисел, тексту та кольорів, які можна застосовувати до файлів і прототипів користувача, **Local Styles** – для палітри, **Export** – для експорту.

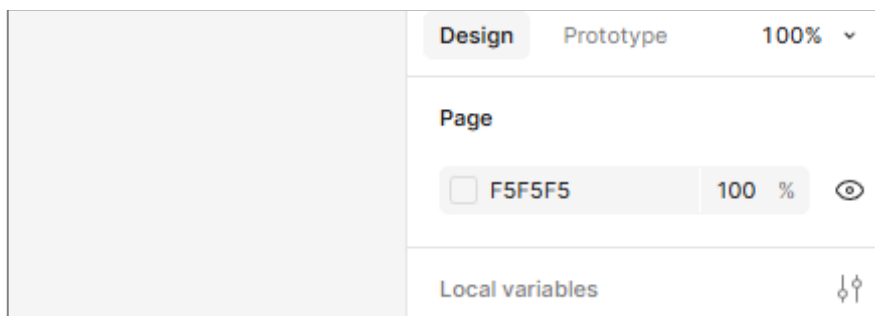


Рис. 37. Панель властивостей

Якщо виділено будь-який тип шарів, бічна панель матиме характерні елементи саме для цього типу шару.

ЗАГАЛЬНІ ВЛАСТИВОСТІ ШАРІВ

Шар будь-якого типу має базові властивості:

- **X, Y** – координати у пікселях. Якщо шар на нескінченному полотні, то координати абсолютні. Якщо всередині фрейма – відносні;
- **Width, Height** – ширина та висота у пікселях;
- **Constrain Proportions** (кнопка правіше Width та Height) – дозволяє зберегти пропорції ширини та висоти, якщо вони змінюються в одному із полів;
- **Rotation** – кут нахилу. Будь-який шар можна нахиляти на цей кут;
- **Corner Radius** – радіус заокруглення кутів у примітивів;
- **Layer** – режим накладання. За замовчуванням **Pass Through**. Будь-який шар, включаючи фрейми, може бути напівпрозорим (10%, 50%, 100%).

Усі векторні типи: примітиви, текстові шари та векторні мережі мають такі властивості, як обведення **Stroke** та заливання **Fill**.

ПРИМІТИВИ

Примітиви – базові векторні фігури із закритим периметром. Якщо периметр розірваний, примітив перетворюється на **шлях** (path). Примітиви ще називають **шейпами** (shapes).

Figma дозволяє створювати такі типи примітивів і шляхів:

- **Rectangle** (Прямокутник), **R** – для створення прямокутників;
- **Ellipse** (Коло/Овал), **O** – для створення кіл та еліпсів;

- **Polygon** (Багатокутник) – для створення фігур із великою кількістю кутів;
- **Star** (Зірка) – для створення зірок;
- **Line** (Лінія), **L** – для створення прямих ліній;
- **Arrow** (Стрілка), **Shift + L** – для створення стрілок.

Для створення примітиву необхідно:

1. На панелі інструментів (Рис. 38) обрати потрібний інструмент або натиснути комбінацію клавіш, що йому відповідає. Кнопка має підсвічуватися синім.
2. Клікнути та затиснути ліву кнопку мишки у будь-якому місці робочої області, щоб задати початок створення фігури. Не відпускаючи ліву кнопку мишки, перемістити курсор у будь-якому напрямку до утворення фігури потрібного розміру.

Якщо вибрано інструмент будь-якого примітиву, під час простого клацання мишкою по робочій області буде створена фігура розміром 100×100. Якщо затиснутий **Shift**, під час перетягування фігура зберігатиме рівну ширину та висоту (наприклад, квадрат, коло, рівносторонній трикутник). Якщо натиснути **Alt**, об'єкт розтягується, починаючи з центру.

Під час створення та після відображення примітиву навколо нього з'явиться синя рамка. Це ознака **виділення об'єкта**. Цю рамку часто називають обмежувальною (**bounding box**). Під нею відображається розмір об'єкта у **пікселях**.

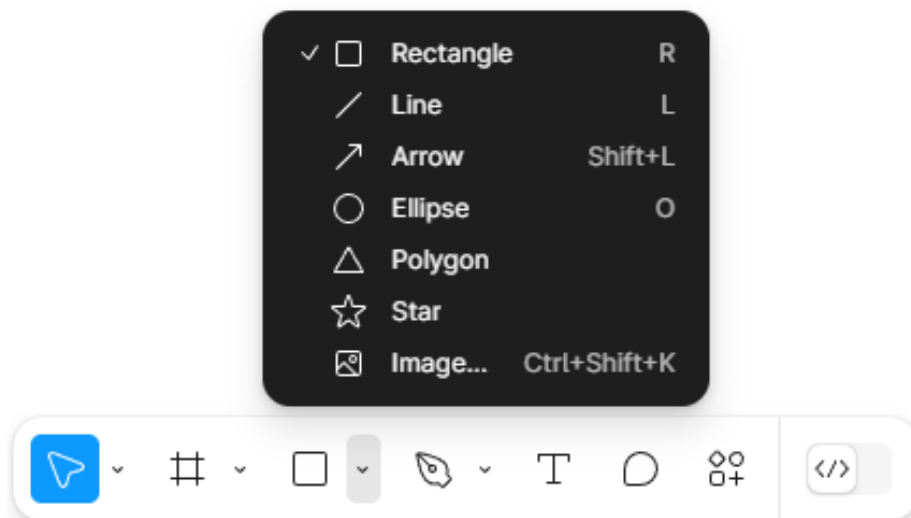


Рис. 38. Панель для створення примітивів

ПРЯМОКУТНИК

Прямокутник створюється командою **Rectangle**, **R**.

На панелі шарів зліва з'явиться рядок **Rectangle 1** з іконкою прямокутника (Рис. 39). Круглі маркери у кутах прямокутника дозволяють заокруглити фігуру.

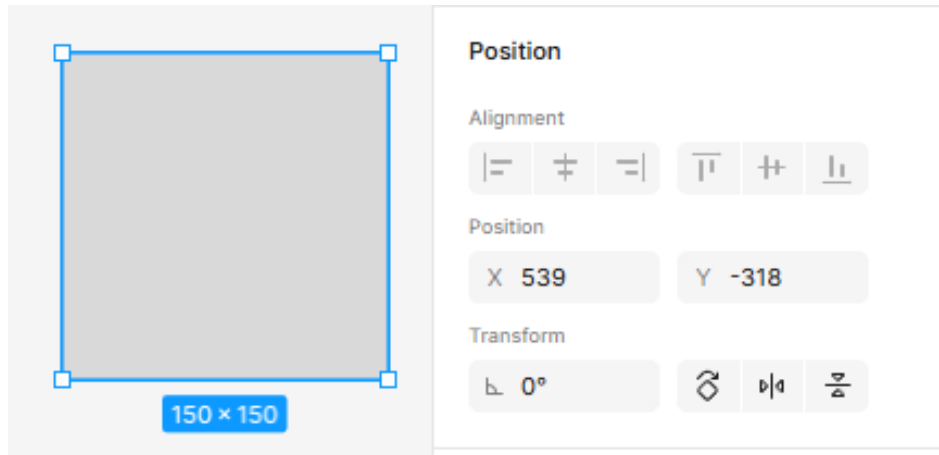


Рис. 39. Створення прямокутника

КОЛО, ОВАЛ І ПАЙЧАРТ

Коло створюється аналогічно прямокутнику командою **Ellipse**, **O**. На панелі шарів зліва з'явиться рядок **Ellipse 1** з іконкою кола (Рис. 40).



Рис. 40. Відображення іконки кола на панелі шарів

У Figma кола й овали можна легко розмикати. Для цього слід вхопити мишкою круглий маркер, який рошташовано у правій частині, та протягнути вгору або вниз. Так змінюється властивість **Sweep** і отримується пайчарт. Пайчарт має два радіуси з круглими маркерами, які можна підлаштовувати до потрібних значень, а також центральну точку, яка дозволяє вирізати серцевину (Рис. 41).

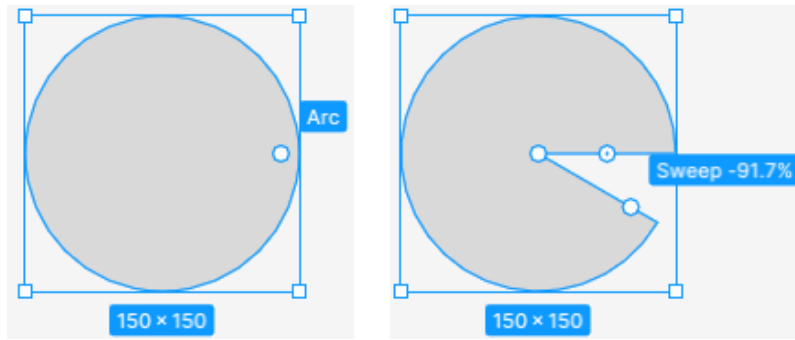


Рис. 41. Створення кола та пайчарту

Щоб вирізати з кола середину необхідно перетягнути центр униз.

За роботу з пайчартами відповідають три властивості, які розташовуються на панелі властивостей виділеного кола (Рис. 42):

- **Start** – кут початку в градусах. Початкове значення – 0. Вихідний градус – на 3 години;
- **Sweep** – відсоток заповнення від 0 до 100%;
- **Ratio** – розмір порожнечі в центрі у відсотках від радіусу. При 0% пайчарт буде заповнений по центру, при 100% заливки видно не буде, залишиться тільки stroke (контур).

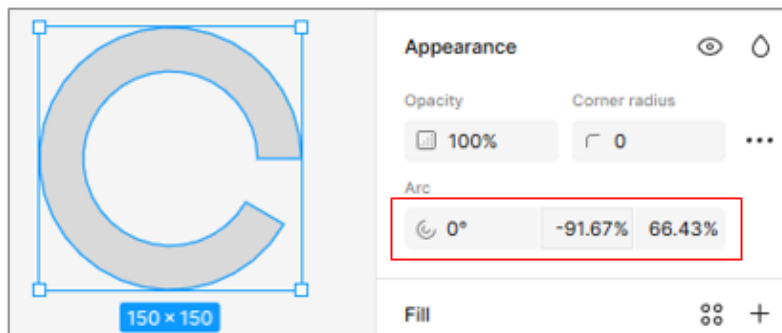


Рис. 42. Створення кола та пайчарту

ПОЛІГОН

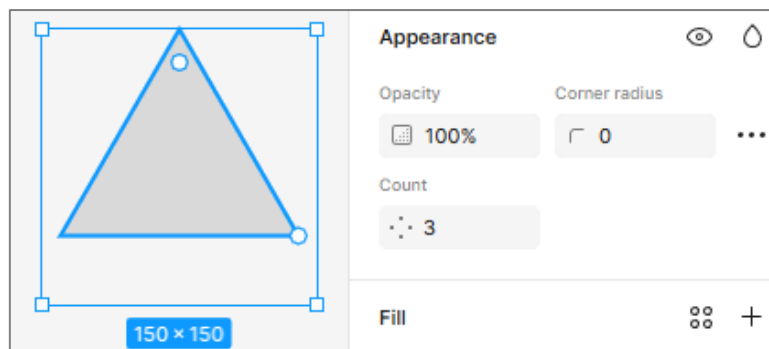


Рис. 43. Створення полігону

Інструмент **Polygon** дозволяє створювати багатокутники. Цей інструмент не має «рідної» клавіші. Його можна викликати через Панель інструментів. У панелі шарів з'явиться рядок **Polygon 1** з іконкою трикутника (див. Рис. 43).

Односторонній трикутник, що з'явився, має два круглі маркери: всередині зверху та праворуч на кутку. Кутовий маркер дозволяє **задавати потрібну кількість кутів** в інтервалі від 3 до 60. При 60 кутах багатокутник мало відрізняється від кола. Верхній маркер дозволяє **заокруглювати кути** трикутника.

ЗІРКА

Інструмент **Star** дозволяє малювати променеві багатокутники. Цей інструмент також не має «рідної клавіші». Його можна викликати за допомогою Панелі інструментів. У панелі шарів з'явиться рядок **Star 1** з іконкою зірки (Рис. 44).

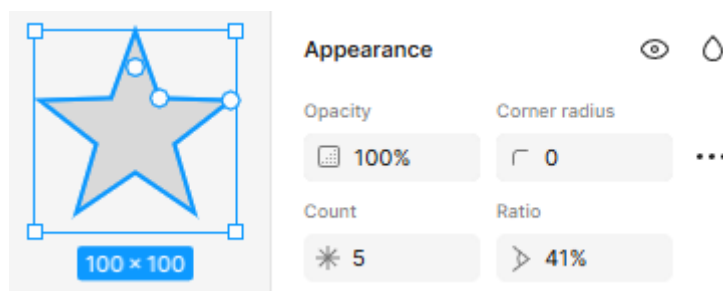


Рис. 44. Створення зірки

Окрім стандартних, зірка має 3 параметри:

- **Corner Radius** – ступінь заокруглення кутів;
- **Count** – кількість променів;
- **Ratio** – ступінь гостроти променів (%).

Маркер, який розташовано у верхньому промені, дозволяє заокруглити кути всіх променів. Маркер, який розташовано у кутку між верхнім і правим променями, дозволяє задавати **ratio** (встановлює, чи буде зірка «гострою», чи «пузатою») (Рис. 45).

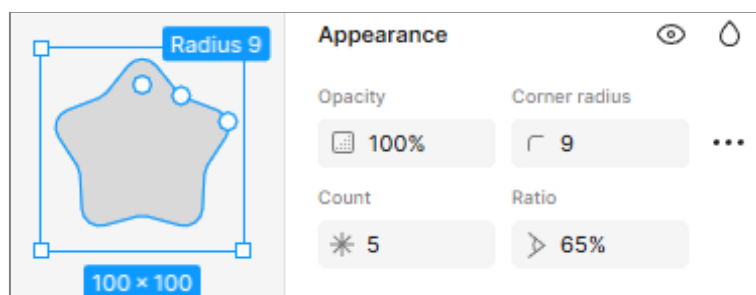


Рис. 45. Встановлення «гостроти» зірки

ЛІНІЯ

Лінія створюється командою **Line**, **L**. Інструмент дозволяє за допомогою перетягування створити векторний відрізок. У панелі шарів з'явиться рядок **Line 1** з іконкою відрізка (Рис. 46).

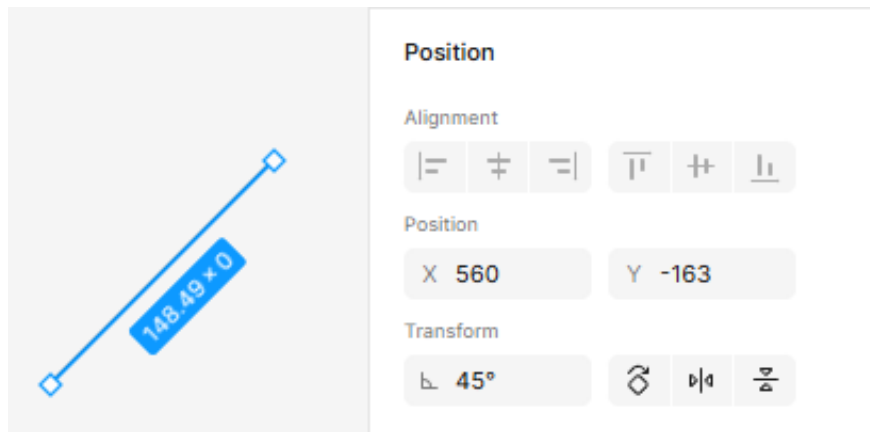


Рис. 46. Створення лінії

Є лише одна пара координат для початку лінії. У полі **Width** лінії задається її довжина, а у полі **Angle** кут нахилу.

СТРІЛКА

Стрілка створюється командою **Arrow**, **Shift + L**. Інструмент працює аналогічно **Line**. Єдина різниця – на кінцевій точці створюється стрілка (Рис. 47).

Наконечник можна відключати, перетворюючи стрілку на звичайну лінію, або навпаки. Можна намалювати криву та включити їй стрілку на будь-якому кінці.

Для стрілок можна налаштувати різні параметри: товщину **Weight**, колір, стиль наконечника **Arrowheads** (**Line arrow**, **Circle arrow**, **Diamond arrow**).

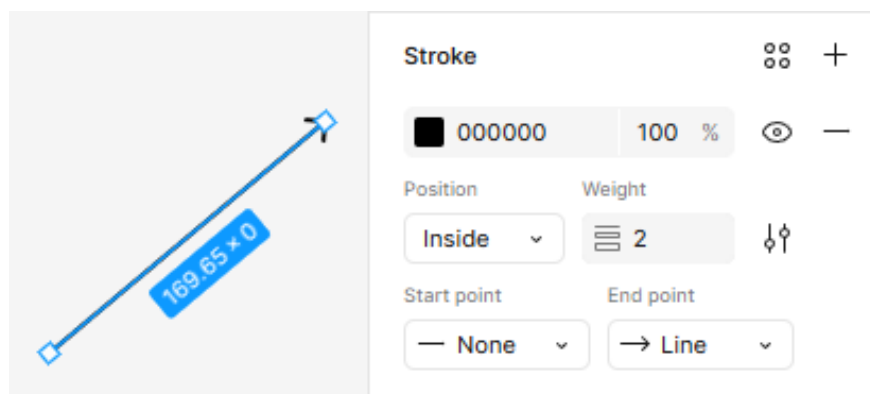


Рис. 47. Створення стрілки

Щоб задати додаткові налаштування стрілки, потрібно розгорнути меню **Advanced Stroke**. Це меню виглядає як блакитні повзунки у блоці **Stroke** (Рис. 48).

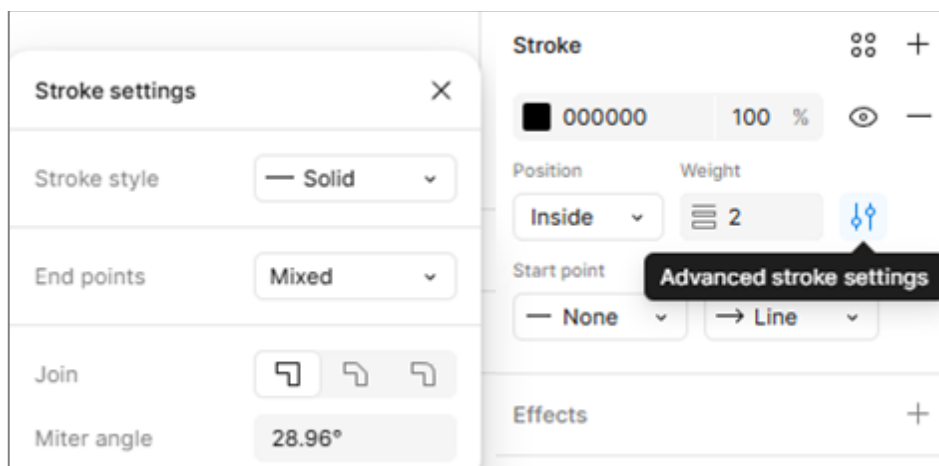


Рис. 48. Налаштування параметрів стрілки

У панелі **Advanced stroke** можна змінювати стиль лінії **Stroke style** (суцільна, пунктирна чи штрихова лінія), кінці лінії **Join** (**Butt cap** (прямий кінець), **Round cap** (округлений кінець), **Square cap** (прямокутний кінець із продовженням)).

ОПЕРАЦІЇ НАД ПРИМІТИВАМИ

ЗСУВ

Для переміщення примітиву його потрібно активувати та перетягнути у потрібне місце. Якщо під час руху затиснути **Shift**, буде зафіксовано горизонтальну або вертикальну вісь, відносно яких примітив можна переміщати. Також можна використовувати клавіші стрілок на клавіатурі для точного зсуву попіксельно. Під час утримування клавіші **Shift** – зсув здійснюється на 10 пікселів.

Щоб точно змінити позицію примітиву можна задати координати **X** і **Y** вручну на панелі властивостей.

У Figma можна здійснювати зсув елементів за допомогою клавіатури, використовуючи стрілки (вліво, вправо, вгору, вниз). Водночас зсув на 1 піксель відповідає одному натисканню стрілки на клавіатурі.

МАСШТАБУВАННЯ

Для зміни масштабу примітиву слід його активувати, захопити мишкою один із кутів або бокових маркерів, і перетягнути так, щоб змінити розміри до необхідного

значення. Товщина меж у цьому разі змінюватися не буде. Якщо потрібно зберегти пропорції під час масштабування, слід утримувати клавішу **Shift**.

Можна також змінити розміри примітиву, ввівши нові значення ширини **W** і висоти **H** у панелі властивостей.

Для збільшення або зменшення примітиву так, щоб лінії обведення теж змінилися, використовується інструмент **Scale**, **K** (Рис. 49). Режим **Scale** небезпечний тим, що лінії можуть отримувати дробові значення (наприклад, ширина 1073,1 пікселі (Рис. 50)). Якщо такі значення зустрічаються в іконках, вони матимуть «замилений» вигляд.

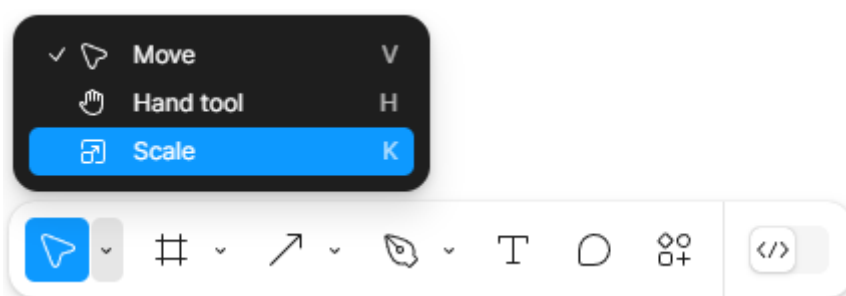


Рис. 49. Встановлення масштабу примітиву

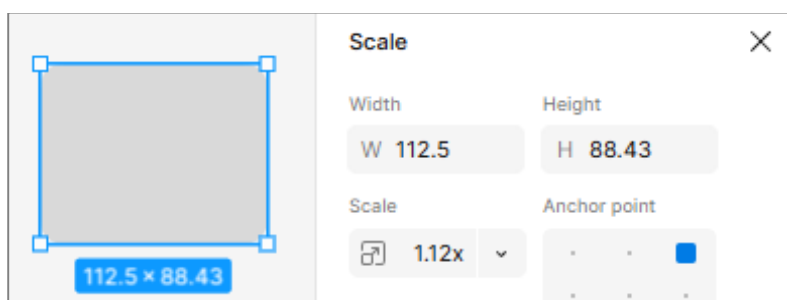


Рис. 50. Приклад примітиву з дробовим значенням кількості пікселів

ЗАЛИВКА Й ОБВЕДЕННЯ: ПЕРШИЙ ПОГЛЯД

Як і у будь-якому векторному редакторі, у Figma є можливість заливати векторні фігури кольором, а також накладати колір на їх контур. Для роботи із заливкою використовується блок **Fill**. Обведення фігур контуром здійснюється з використанням блоку **Stroke**.

У Figma для шару примітиву можна налаштовувати необмежену кількість обведень і заливок. Водночас кольори накладаються відповідно до свого відсотку

прозорості. Наприклад, якщо взяти 50% чистого синього кольору і накласти 50% чистого червоного, то отримаємо колір, наближений до кольору фуксії (Рис. 51).

У **Fill** та **Stroke** колір накладається на фігуру за принципами, які детально розглядатимуться у розділі [Заливка та градієнт](#). Налаштування обведення фігур розглядатимуться у розділі [Межі](#).

Щоб забрати заливку чи обведення потрібно натиснути «мінус» біля зайвого кольору.

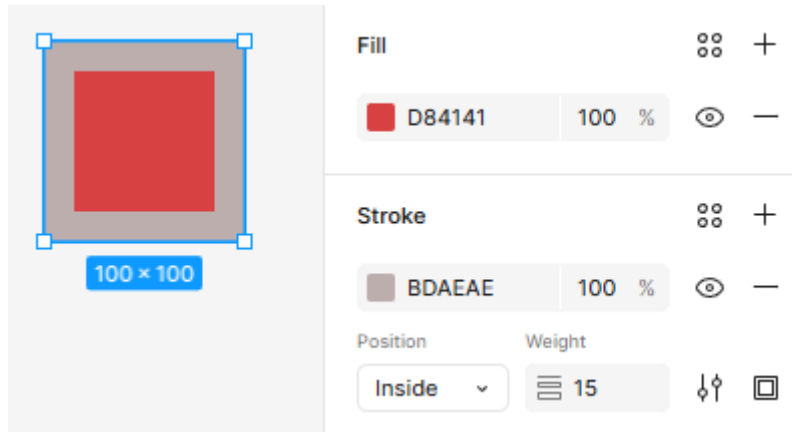


Рис. 51. Приклад накладання кольорів у Figma

ЗАОКРУГЛЕННЯ

Заокруглення кутів у Figma можна змінювати.

Для заокруглення всіх кутів одночасно необхідно:

1. Виділити об'єкт.
2. У правій панелі знайти параметр заокруглення **Corner Radius** (Радіус кутів) (Рис. 52). За замовчуванням він може бути встановлений на 0.
3. Змінити значення радіусу. Що більше число, то більш заокругленим буде кут.

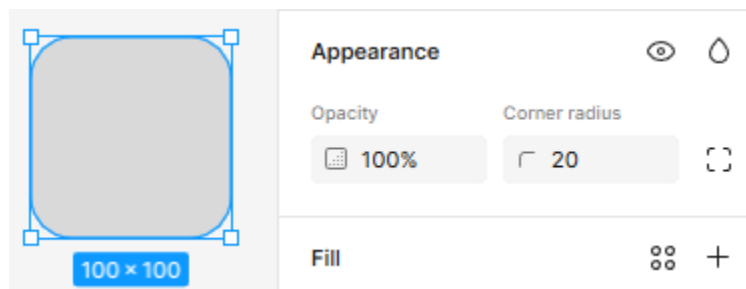


Рис. 52. Заокруглення всіх кутів у примітивів

Щоб прибрати заокруглення, потрібно виставити **Corner Radius** на 0 або відсунути маркери від центру примітиву.

Округлення одного кута прямокутника. Для прямокутника можна заокруглити кожен кут окремо. Для цього потрібно виділити примітив. Включити режим редагування клавішею Enter. Виділити потрібну вершину та задати її радіус у полі **Corner Radius** (Рис. 53).

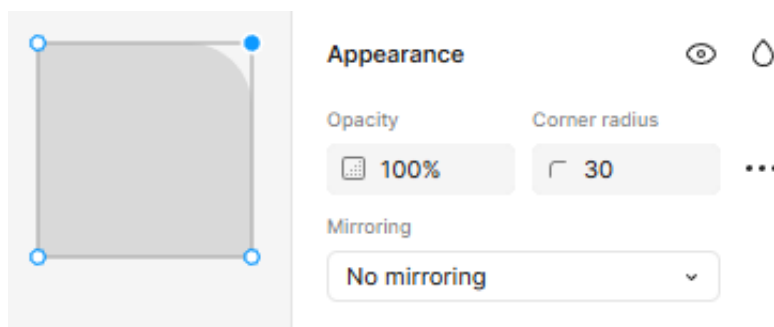


Рис. 53. Заокруглення окремих кутів примітивів

ЛИПКІСТЬ (SNAPPING)

Липкість (Snapping) дозволяє автоматично вирівнювати об'єкти відносно інших елементів на сторінці (Рис. 54).

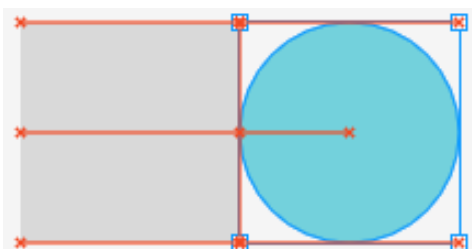


Рис. 54. Розміщення об'єктів на полотні

Коли користувач переміщує об'єкт, він автоматично прилипає до меж, центрів або відстаней інших об'єктів. Водночас, навіть працюючи мишкою, користувач може розставляти об'єкти полотна з точністю до пікселя.

Такий режим включається автоматично, але, якщо користувачу потрібно його вимкнути, слід зняти відповідний прапорець у головному меню **Preferences** (Рис. 55).

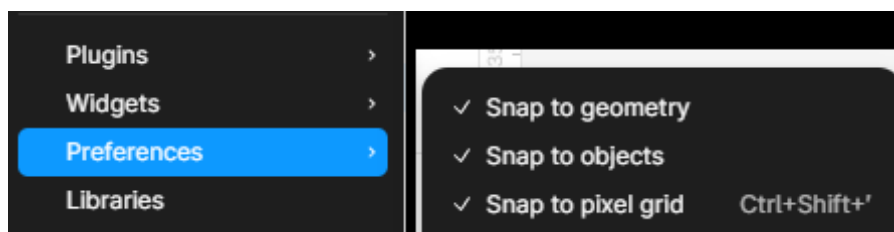


Рис. 55. Головне меню Preferences

КОПІЮВАННЯ

Figma має дуже зручний інструмент для точного позиціонування, який показує відстань між об'єктами під час копіювання. Щоб ним скористатися, потрібно виділити об'єкт, затиснути **Alt** і потягнути донизу, аби дублювати його. Під час перетягування будуть видимі червоні лінії та відстані у пікселях, які підказуватимуть, як його можна вирівнювати відносно батьківського об'єкта (Рис. 56).

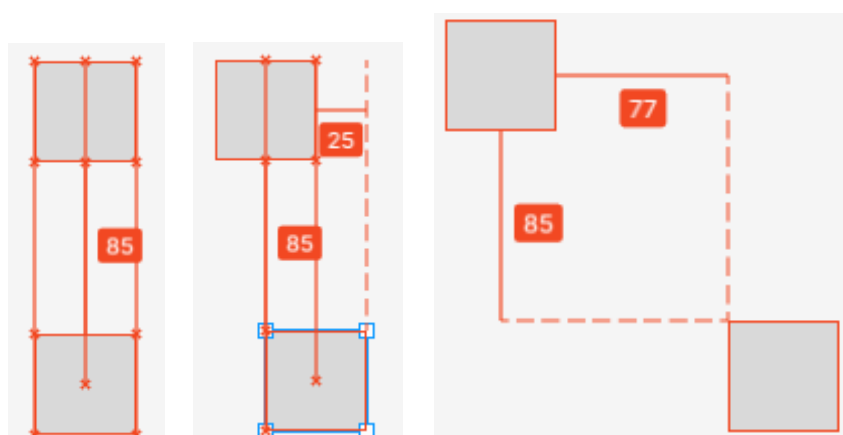


Рис. 56. Вирівнювання при копіюванні об'єктів у Figma

ВИЗНАЧЕННЯ ВІДСТАНЕЙ

ВИЗНАЧЕННЯ ВІДСТАНЕЙ ЗА ДОПОМОГОЮ ALT

Як визначити відстані у Figma зручно продемонструвати на наступному прикладі:

1. Накреслимо фрейм, **F**.
2. Усередині нього створимо квадрат, **R**.

3. Виділимо фрейм. Затиснемо **Alt** і наведемо курсор на квадрат. Доки курсор знаходитиметься всередині квадрата, будуть відображатися відстані у пікселях від кордонів квадрата до кордонів фрейму (Рис. 57).

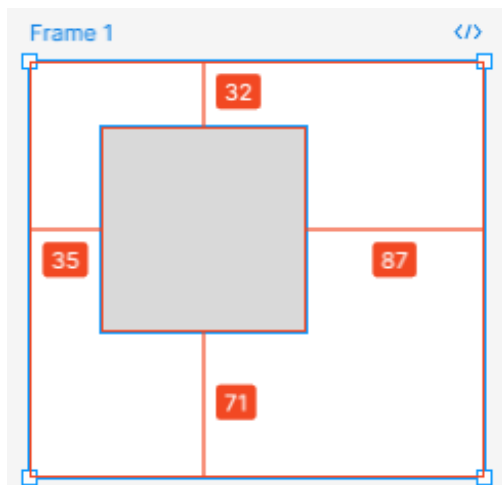


Рис. 57. Визначення відстаней у Figma (приклад 1)

4. Усередині фрейму створим коло, **O**.
5. Виділимо квадрат. Змістимо курсор із квадрата і затиснемо **Alt**. Наведемо курсор на коло. Доки буде натиснута клавіша **Alt** і курсор знаходитиметься всередині кадру, будуть відображатися відстані у пікселях від меж квадрата до меж кола і фрейму (Рис. 58).

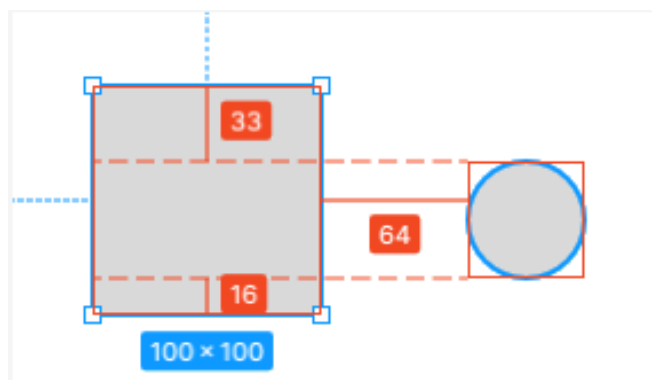


Рис. 58. Визначення відстаней у Figma (приклад 2)

ВИЗНАЧЕННЯ ВІДСТАНЕЙ ЗА ДОПОМОГОЮ ПРЯМОКУТНИКІВ

Є ще один спосіб виміряти відстані. Оскільки прямокутники самі собою – ідеальні вимірники, їм можна надавати потрібну форму, а потім вимірювати ними відстані.

Наприклад, найшвидший спосіб задати відступ у 64 пікселі – зробити допоміжні прямокутники і за ними налаштувати ширину основної фігури (Рис. 59).

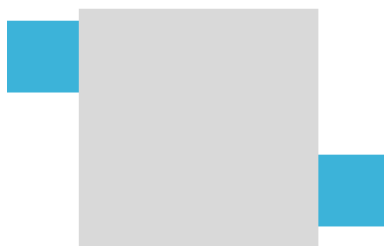


Рис. 59. Визначення відстаней за допомогою прямокутників

Потім ці прямокутники можна видалити або винести за межі кадру.

РЕЖИМ РЕДАГУВАННЯ

У Figma режим редагування (**Edit mode**) дозволяє вносити зміни у дизайн, змінювати положення об'єктів, редагувати текст, змінювати властивості шарів, додавати нові елементи, застосовувати ефекти тощо.

Для переходу до режиму редагування потрібно виділити фігуру та натиснути **Enter**. Водночас кутові квадратні маркери змінять вигляд на кружечки (точки) (Рис. 60).



Рис. 60. Перехід у режим редагування

Для виходу із режиму редагування використовуються клавіші **Esc** або **Enter**.

Під час наведення на фігуру Figma наочно показує штрихуванням, що користувач знаходиться у цьому режимі (Рис. 61).



Рис. 61. Робота користувача у режимі редагування

Кружечки можна перетягувати по одному за допомогою мишки, або декілька одночасно, попередньо виділивши їх за допомогою клавіші **Shift**. Виділена точка відображатиметься залитою синім кольором. Якщо точка виділена, її можна рухати

також за допомогою **стрілок** на клавіатурі або використовувати комбінацію клавіш **Shift + стрілки** (Рис. 62).

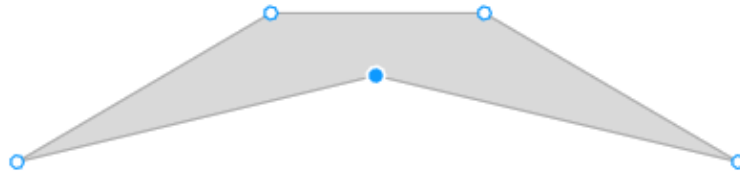


Рис. 62. Зміна об'єкта у режимі редагування

Так, прямокутник можна перетворити на іншу фігуру, наприклад, на трапецію. Для цього пересуваємо його точки у режимі редагування. В цьому випадку в нього безповоротно зникнуть маркери заокруглення. Це відбувається тому, що Figma перестає сприймати його як об'єкт типу **Rectangle**, а вважатиме його векторною фігурою довільної форми.

У режимі редагування можна додати фігурі нових точок і виліпити з неї нову фігуру.

Також фігуру можна розірвати. Для цього потрібно видалити будь-яку точку. Разом із нею видаляються прилеглі до неї відрізки (Рис. 63).

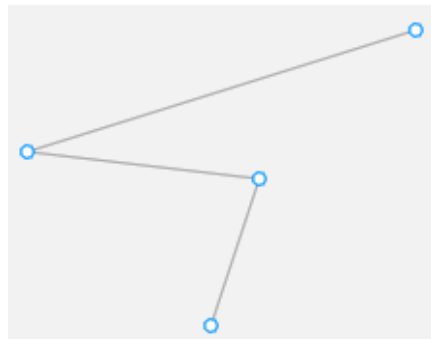


Рис. 63. Приклад розривання фігури

Якщо потрібно видалити лише одну сторону, створюється точка на її середині (Figma під час наведення курсору показує, де знаходиться центр відрізка). Видаляємо точку клавішею **Delete**. Залишається векторний скелет із точок і відрізків. Водночас у лівій верхній точці залишилося заокруглення, якого не видно через те, що векторна фігура не має обведення. Так здійснюється перетворення фігури типу **Rectangle** на розімкнутий шлях (Рис. 64). Надалі потрібно задати **stroke** (обведення), щоб відобразилось заокруглення (Рис. 65).

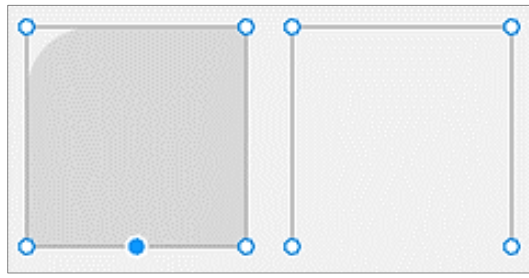


Рис. 64. Приклад видалення сторони багатокутника

Заокруглений кут залишився. Це означає, що кожна точка векторного шляху зберігає інформацію про ступінь заокруглення навіть у разі, якщо примітив розірваний.

Що стосується заливки, то вона є у налаштуваннях, але її не видно, оскільки фігура розімкнена.

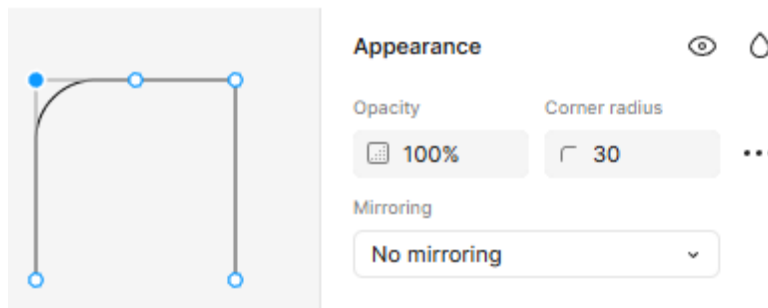


Рис. 65. Встановлення заокруглення після видалення сторони багатокутника

Тепер Figma сприймає такий об'єкт як векторний шлях або векторну мережу. В шляху немає поняття внутрішнього та зовнішнього, тому обведення може бути лише у режимі **Center**, незалежно від того, який режим виставлений у налаштуваннях.

Якщо взяти **Pen**, **P**, навести на одну з кінцевих точок, то в курсорі з'явиться коло. Це означає, що під час натискання користувач продовжуватиме малювати лінію з цієї точки. Клацаємо мишкою по точці, клацаємо по іншій навпроти і так замикаємо фігуру (Рис. 66).

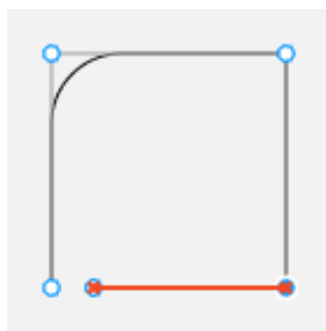


Рис. 66. Створення замкнутої фігури з векторного шляху

ФРЕЙМИ

У третьому розділі нами розглядалося питання роботи з панеллю шарів. Зокрема у ньому було зауважено, що кожен тип шарів має унікальні налаштування у панелі властивостей. Подальше вивчення роботи із шарами почнемо із фреймів.

У Figma термін **фрейм** означає основний контейнер для створення дизайну, аналогічний **артбордам** (artboard) в інших графічних редакторах. Фрейми використовуються для організації та структурування елементів дизайну, створення різних екранів у пропорціях популярних пристроїв або сторінок в одному проєкті.

Фрейми створюються клавішею **F** або інструментом **Frame** на панелі інструментів. Якщо виділити створений фрейм, на панелі праворуч відобразяться його властивості (Рис. 67).

Коли користувач створює дизайн усередині фрейму, до панелі властивостей зліва будуть додаватися шари. **Шари** у Figma – це вміст фрейму: об'єкти, текст, світлини тощо.

Зверху в блоці **Position** відображаються налаштування вирівнювання. У порожньому фреймі такого відображення немає, тому що немає чого вирівнювати.

Розмір фрейму можна змінювати, потягнувши за край або поставивши точний розмір у полях **W** (Width) та **H** (Height) у блоці **Layout** панелі властивостей. Також можна зі списку **Frame** у лівому верхньому куті панелі властивостей вибрати потрібний шаблон розмірів, що відповідає конкретному пристрою.

У Figma фрейми мають власні обмеження за розміром. Тому елементи всередині фрейму можуть розміщуватися за його межами, але їх видимість буде обмежена межами фрейму. Прапорець **Clip Content** визначає, чи буде видно шар, частини якого виходять за межі фрейма (Рис. 68).

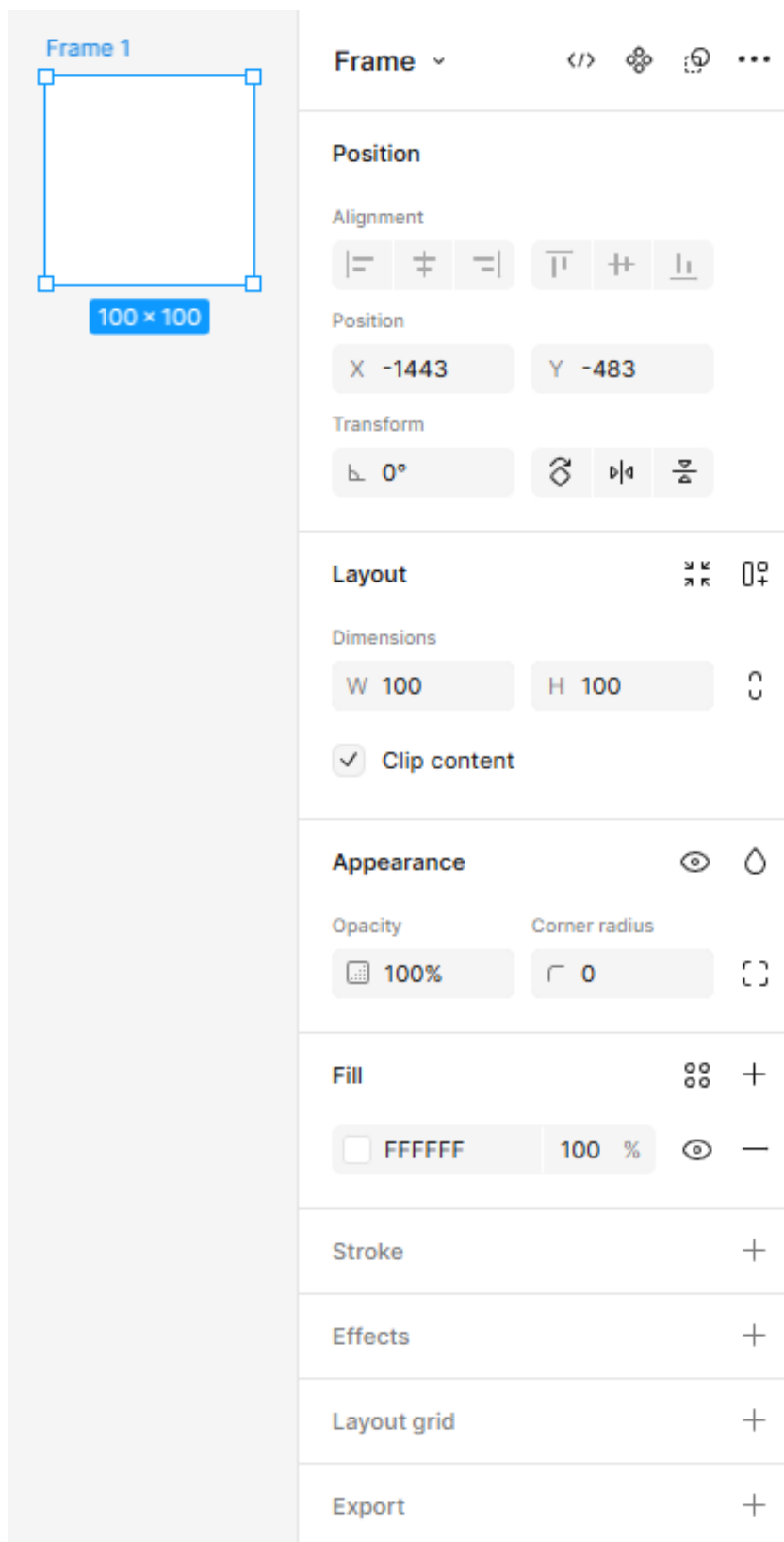


Рис. 67. Створення фрейму

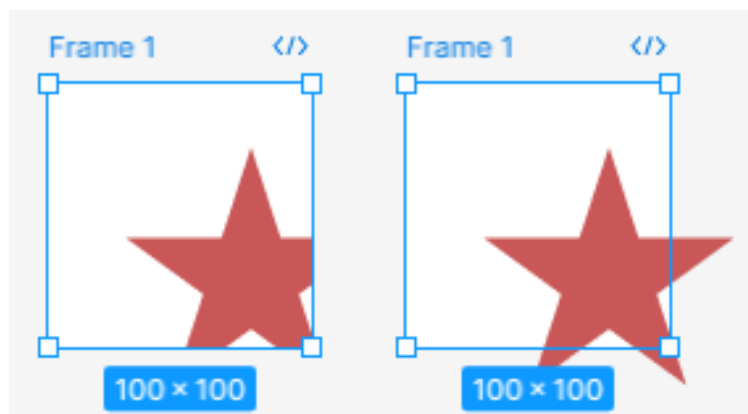


Рис. 68. Використання інструменту *Clip Content*

У блоці **Srtoke** налаштовується межа фрейму, в блоці **Layout Grid** – сітка, в **Fill** – фон, а в **Effects** – тіні та блюр². У блоці **Export** можна налаштувати, в які формати шар експортуватиметься.

Розглянемо налаштування параметрів фрейму більш детально.

ФОНОВИЙ КОЛІР ФРЕЙМУ

Фрейми можуть мати фоновий колір (за замовченням білий), який задається у блоці **Fill**. Спершу заливка фону стоїть у режимі **Solid**, що дає рівномірне накладання кольору (Рис. 69).

Щоб змінити фоновий колір фрейму в Figma, потрібно виконати такі кроки:

1. Вибрати фрейм, фоновий колір якого потрібно змінити.
2. У панелі властивостей активувати розділ **Fill** (заливка).
3. Якщо фрейм має колір заливки, натиснути на кольорове поле під назвою розділу **Fill**. Якщо заливки немає, то натиснути кнопку + поруч із **Fill**, щоб додати новий шар заливки.
4. У кольоровій панелі, що з'явиться, вибрати потрібний колір або ввести код кольору вручну.

Зміни автоматично застосуються до обраного фрейму.

² **Blur** — графічний ефект, який використовується на фотографіях, відео та відеоіграх, що розмиває зображення.

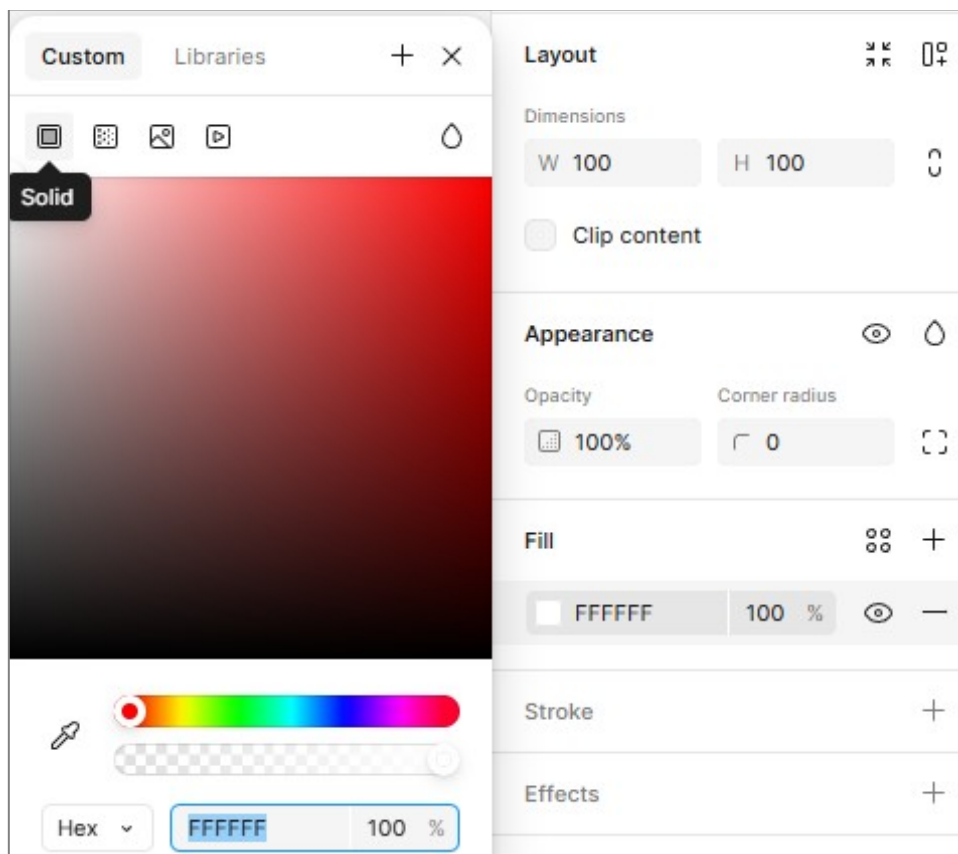


Рис. 69. Налаштування фонового кольору фрейму

У Figma діє клавіша **I**, яка викликає **Color Picker** – піпетку. Піпетка дає змогу зчитати потрібний колір у будь-якому місці проєкту.

МЕЖІ ФРЕЙМІВ

Фрейми можуть мати уявну межу, яка допомагає розуміти їх розмір у ситуації, коли фрейм не має фону. Для задання цього режиму використовується послідовність команд налаштування: **View** → **Outlines Show** → **Outlines**, **Ctrl + Shift + O** (Рис. 70).



Рис. 70. Налаштування меж фрейму

ОБЕРТАННЯ ФРЕЙМОМ

Будь-який об'єкт можна за один клік обернути фреймом. Це іноді буває потрібно, щоб згодом його експортувати або перетворити на компонент. Для цього

використовується команда **Frame Selection**, **Ctrl + Alt + G**, що знаходиться у контекстному³ меню об'єкта (Рис. 71). Зворотна дія – **Shift + Ctrl + G**.

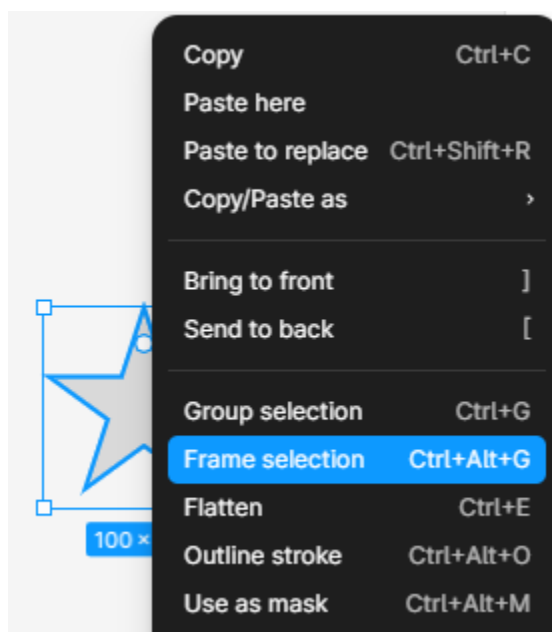


Рис. 71. Контекстне меню об'єкта

Якщо фрейм навколо шару вже є і потрібно стиснути його до розмірів цього шару, використовується команда **Resize To Fit**, **Shift + Ctrl + Alt + R** у панелі властивостей  (Рис. 72).

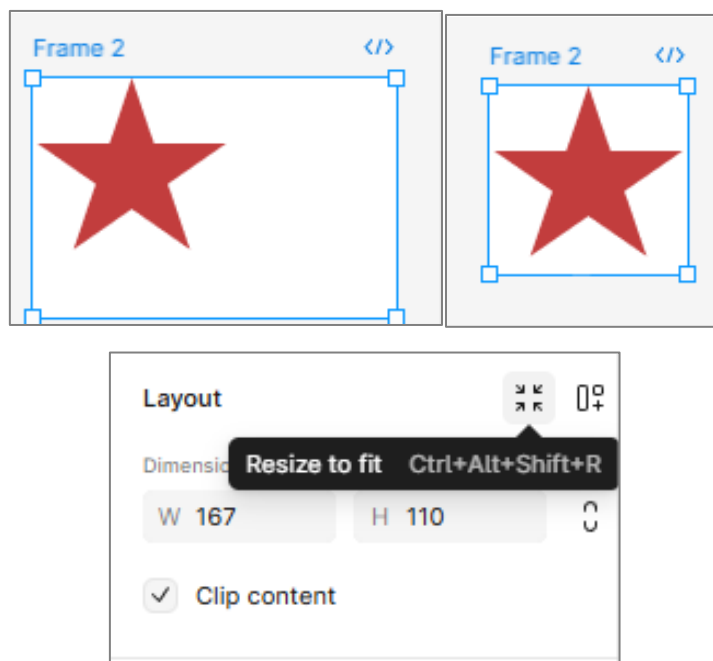


Рис. 72. Стискання фрейму до розмірів шару

³ Контекстне меню – меню, що викликається при клацанні правою клавішею мишки по об'єкту.

Якщо елементи виходять за межі фрейму, їх можна обітнути. Для цього потрібно увімкнути опцію **Clip content** у панелі властивостей фрейму. Це означає, що будь-який вміст, який виходить за межі фрейму, не буде видно.

РОБОТА З СІТКОЮ

У Figma **Grid** (сітка) – допомагає структурувати елементи на екрані, забезпечуючи послідовність та естетичну гармонію. Сітки використовуються для створення макетів, вирівнювання елементів і встановлення модульності.

ДОДАВАННЯ СІТКИ

Для того, щоб додати сітку до фрейму необхідно:

1. Вибрати фрейм, до якого потрібно додати сітку.
2. У правій панелі у розділі **Layout Grid** натиснути на +.
3. Якщо це потрібно для складнішої структури, можна додати кілька сіток до одного фрейму, натискаючи +.

Ctrl + G – показує або приховує сітку у всіх кадрах проекту.

ТИПИ СІТОК

Сітка може мати один із трьох типів: **Grid**, **Columns**, **Rows**. Вони можуть накладатися одна на одну і впорядковуватись у блоці **Layout Grid** подібно до шарів. Можна вмикати / вимикати потрібні сітки, натиснувши на око. Щоб налаштувати тип сітки і його параметри, потрібно натиснути на значок **Layout grid settings** (Рис. 73).

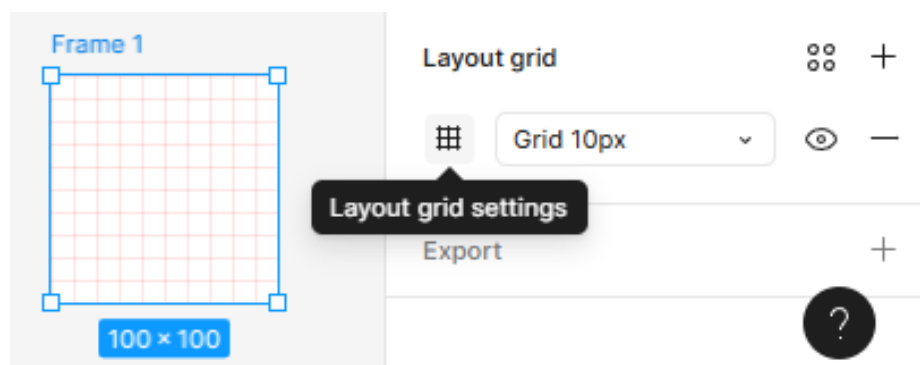


Рис. 73. Створення сітки

Grid (сітка) – проста сітка, яка ділить фрейм на рівні квадрати. Використовується для базового вирівнювання (Рис. 74).

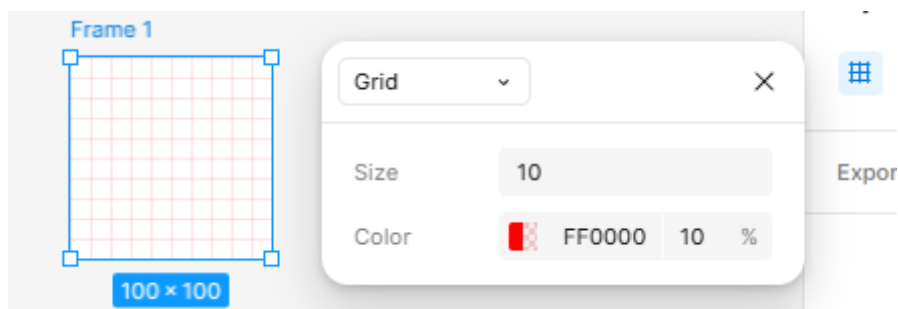


Рис. 74. Налаштування сітки типу *Grid*

Columns (стовпці) – вертикальна сітка, яка поділяє фрейм на стовпці. Часто використовується для вебдизайну.

Rows (рядки) – горизонтальна сітка, яка поділяє фрейм на рядки. Може бути корисною для мобільних макетів або специфічних компонентів.

Для сітки **Columns** та **Rows** можна встановити такі параметри (Рис. 75):

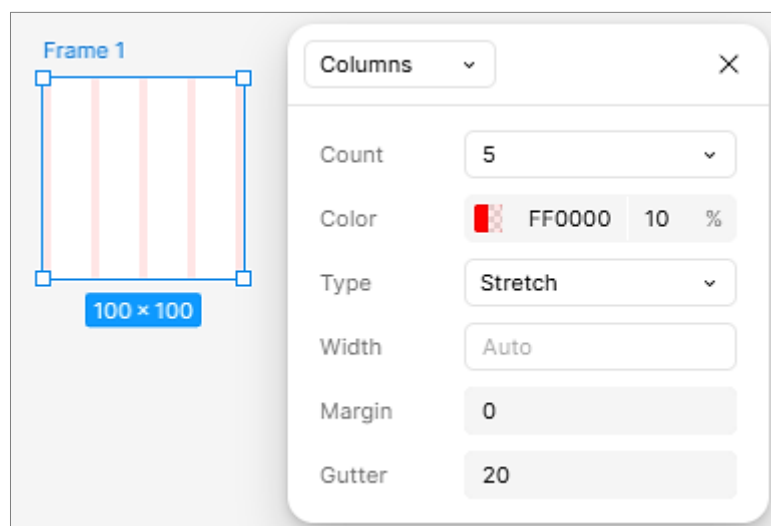


Рис. 75. Налаштування сіток типу *Columns* та *Rows*

- **Count** (кількість) – кількість стовпчиків або рядків;
- **Color** (колір) – колір сітки;
- **Type** (тип) – для адаптації до ширини фрейму використовується тип **Stretch**, а для фіксованих сіток – **Left/Center/Right**;
- **Width** (ширина) та **Height** (висота) – визначає розмір кожного стовчика або рядка;
- **Margin** (відступ) – відступ сітки від країв фрейму;

- **Gutter** (проміжок) – відстань між стовпцями або рядками. За замовчуванням стоїть 10. Для того, щоб змінити значення, необхідно виконати послідовність команд головного меню: **Preferences** → **Nudge Amount**.

КРОК ЗСУВУ ПО СІТЦІ

Щоб перемістити виділений об'єкт на певний крок сітки, можна використати клавіші зі стрілками на клавіатурі. За замовчуванням крок сітки за допомогою стрілок – 1 піксель.

Щоб збільшити крок переміщення (наприклад, до 10 пікселів), потрібно утримувати клавішу **Shift** та натискати стрілки. Це дозволить переміщувати об'єкт на більший крок відповідно до встановленої сітки або за замовчуванням на 10 пікселів.

СЕКЦІЇ

У Figma є два основні поняття, які часто використовуються у дизайні: **Section** (секції, **Shift+S**) та **Frame** (фрейм) (Рис. 76). Вони мають різне призначення і функціональність.

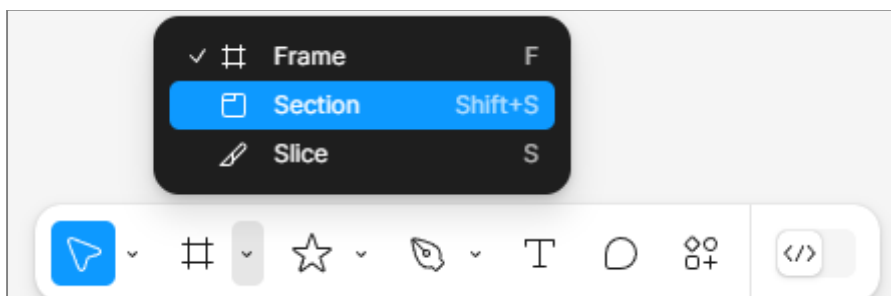


Рис. 76. Створення секцій у Figma

Секція – це тип контейнера у Figma, який використовується для групування декількох фреймів або інших об'єктів у робочій області.

Секції корисні для групування пов'язаних між собою елементів дизайну, наприклад, коли користувач хоче об'єднати різні екрани у рамках одного проєкту або розділити великі макети на логічні частини, а також для спрощення навігації у проєкті, особливо коли в одному файлі міститься велика кількість елементів.

Секції не мають стільки опцій налаштувань, як фрейми, наприклад, не мають можливості змінювати власні розміри автоматично від контенту.

ЗРІЗ

У Figma інструмент **Slice** (Зріз, **S**) використовується для створення окремих частин або областей у проєкті, які можна експортувати як зображення. За допомогою **Slice** можна виділити певну частину макета, яку потім буде експортовано у форматі PNG, JPG або у інший графічний формат (Рис. 77).

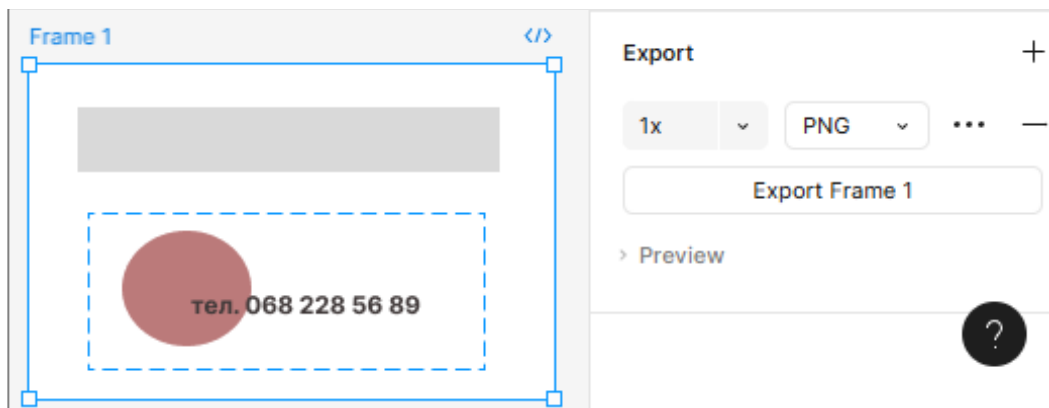


Рис. 77. Створення зрізу в Figma

На відміну від використання стандартного вибору й експорту фреймів або груп, **Slice** дає змогу виділити будь-яку прямокутну область незалежно від структури шарів.

ГРУПУВАННЯ

Під час створення складних об'єктів можна зіштовхнутися із проблемою управління шарами. Наприклад, будь-яка кнопка складається з прямокутника (визначає межі) та тексту (який пояснює, для чого ця кнопка призначена). Деколи кнопка може містити ще й іконку і розташовуватися на інших складних об'єктах. Щоб перемістити таку кнопку, користувачу доведеться спершу виділити кожен її складову. А якщо він захоче перемістити верхню частину сайту з навігаційним меню? Це вже займе значно більше часу. З такими мікрозавданнями користувач зустрічатиметься постійно.

Щоб вирішити таку проблему, є групи, які дозволяють працювати з кількома об'єктами як з одним шаром. Наприклад, прямокутник і текст можна об'єднати

в одну групу за змістом і це вже буде кнопка. Такий підхід є дуже зручним і значно пришвидшує роботу.

Група – це спеціальний шар, який вміщує інші шари і призначений для об'єднання та спрощення подальшої роботи з ними. Група працює з великою кількістю шарів як з одним.

СТВОРЕННЯ ТА ВИДАЛЕННЯ ГРУПИ

Створення та видалення групи зручно продемонструвати на такому прикладі. Припустимо, що користувачу потрібно об'єднати декілька примітивів, які складають кнопку, в групу. Для цього слід:

1. Виділити всі примітиви, з яких побудована кнопка (затиснути ліву кнопку мишки зверху ліворуч і протягнувши курсор до нижнього правого кута. На панелі зліва підсвітяться виділені шари).
2. Натиснути праву кнопку мишки над будь-яким із компонентів кнопки і з меню обирати команду **Group Selection**. Доступна також комбінація клавіш для швидкого об'єднання: **Ctrl + Alt + G**.

Потрібно зауважити, що у панелі шарів замість двох виділених примітивів з'явився один шар із новою іконкою та назвою **Group 1** – це і є новостворена кнопка.

Отже, користувач може працювати із кнопкою як зі звичайним шаром: переміщувати, масштабувати, дублювати тощо без необхідності виділяти кожен шар окремо.

Щоб розбити кнопку назад на складові, потрібно виділити групу і у контекстному меню вибрати команду **Ungroup** або натиснути **Ctrl + Shift + G**.

Об'єкти всередині фрейму можна також групувати. Припустимо, користувач створює кілька сторінок сайту. На кожній сторінці внизу екрану будуть контакти: телефон і електронна пошта. Можна згрупувати об'єкти послідовністю команд **Object → Group Selection** або клавішами **Ctrl + G**. У результаті відбудуться зміни в шарах об'єктів – вони об'єднуються у групу (Рис. 78). Тепер можна переміщати всі об'єкти разом.

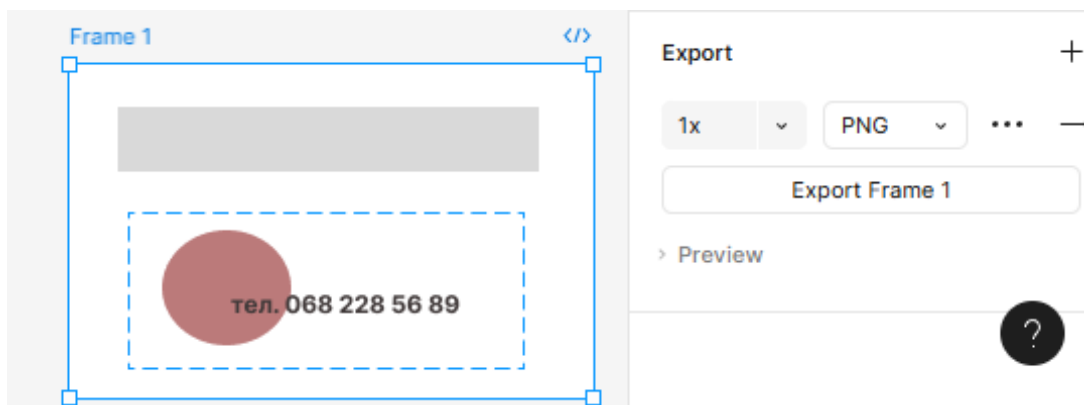


Рис. 78. Групування об'єктів у Figma

ВИРІВНЮВАННЯ ТА РОЗПОДІЛ

ALIGN (ВИРІВНЮВАННЯ)

Align (Вирівнювання) – це інструмент для вирівнювання елементів у макетах. Він дозволяє швидко і точно розташовувати об'єкти, забезпечуючи злагоджений і професійний вигляд проєкту користувача.

Щоб вирівняти об'єкти у фреймі за осями X/Y або зробити однаковою відстань між елементами, необхідно виділити фрейм і скористатися панеллю властивостей праворуч (Рис. 79). Приклад застосування вирівнювання по горизонталі показано на Рис. 80.

Для виконання вирівнювання можна також використовувати комбінації клавіш:

Align Left, **Alt + A** – вирівнювання за лівим краєм;

Align Horizontal Centers, **Alt + H** – за горизонтальним центром;

Align Right, **Alt + D** – за правим краєм;

Align Top, **Alt + W** – за верхнім краєм;

Align Vertical Centers, **Alt + V** – за вертикальним центром;

Align Bottom, **Alt + S** – за нижнім краєм.

Також можна використовувати комбінації клавіш **Alt + H**, **Alt + V**, які дозволяють вирівнювати шар по центру фрейму.

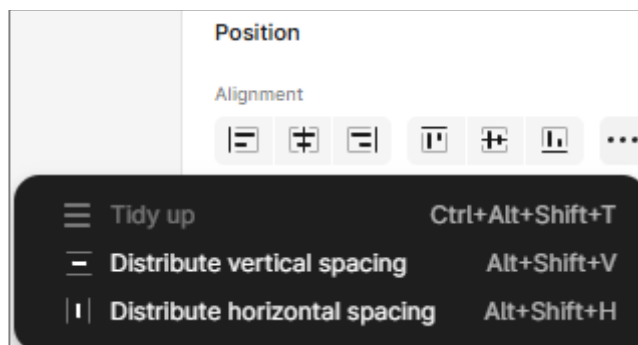


Рис. 79. Панель для вирівнювання елементів

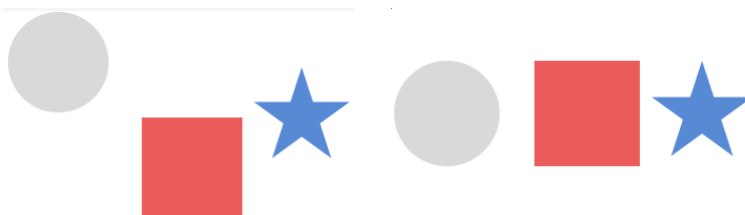


Рис. 80. Приклад застосування вирівнювання у Figma

РОЗПОДІЛ

Distribute (Розподіл) у Figma використовується для рівномірного розподілу об'єктів у проєкті. Розподіл допомагає підтримувати симетрію та акуратність, насамперед під час роботи з великими масивами об'єктів. Це корисний інструмент у разі, коли потрібно впорядкувати елементи так, щоб вони мали однакові інтервали між собою.

Для того, щоб розподілити об'єкти, їх потрібно виділити (або виділити фрейм, у якому вони знаходяться), а потім вибрати опцію розподілу:

- **Distribute Horizontal Spacing** (Горизонтальний розподіл) – розподілить об'єкти рівномірно за горизонталлю.
- **Distribute Vertical Spacing** (Вертикальний розподіл) – розподілить об'єкти рівномірно за вертикаллю.

Після вибору відповідного варіанту об'єкти розташуються так, щоб проміжки між ними були рівними (Рис. 81).

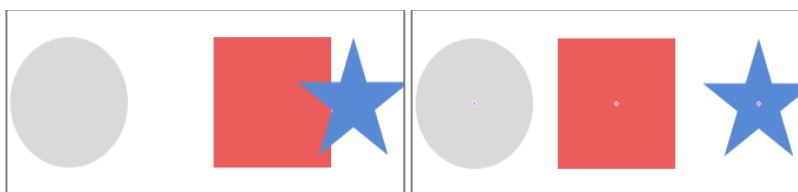


Рис. 81. Приклад застосування розподілу

Можна також використати комбінації клавіш для проведення операції розподілу:

Tidy Up, **Ctrl + Alt + Shift + T** – розумний розподіл. Дозволяє задати однакову відстань між фігурами, який зручно змінювати у полі відступу **Spacing** (Рис. 82).

Distribute Vertical Spacing, **Shift + Alt + V** – дозволяє вирівняти об'єкти по вертикалі з однаковим інтервалом між ними.

Distribute Horizontal Spacing, **Shift + Alt + H** – працює аналогічно **Vertical Spacing**.

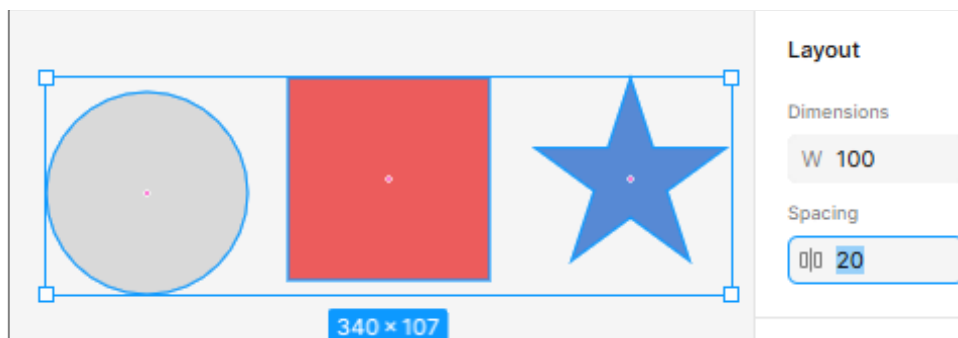


Рис. 82. Зміна відступу в розподілі

ПЕРО

Перо – найважливіший інструмент для малювання кривих ліній і складних непередбачуваних форм за допомогою кривих Безьє.

Криві Безьє – це математичний апарат для опису кривих, який використовував П'єр Безьє у 1960-х роках для проєктування дизайну кузовів автомобілів компанії Renault.

Згодом це відкриття відіграло ключову роль у проєктуванні та комп'ютерній графіці у 2d та 3d форматах. Фігури на основі кривих Безьє називають по-різному, залежно від програми, але суть не змінюється. Користувачі графічних редакторів можуть зустріти такі терміни: **сплайни** (spline) та **шляхи** (path).

Усі криві Безьє можна умовно поділити на декілька видів залежно від кількості точок, з яких вони складені, що визначає їх складність (див. Рис. 83 зліва направо): лінійні, квадратичні, кубічні, вищих порядків.

Зазвичай у графічних додатках використовуються послідовності кубічних кривих Безьє для побудови будь-яких форм.

У Figma інструмент **Pen** (Перо) дозволяє створювати різноманітні фігури та контури шляхом додавання вузлів і кривих.

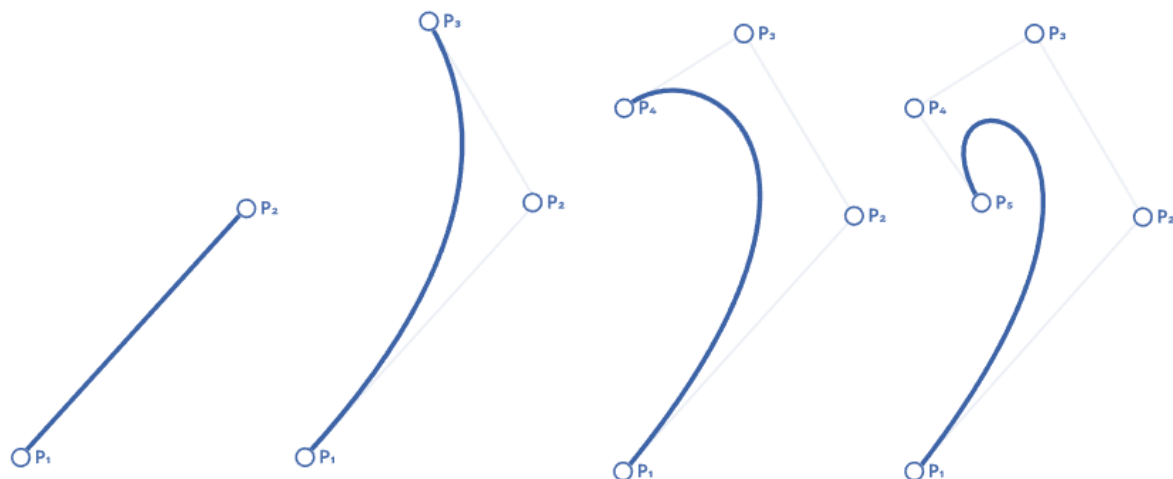


Рис. 83. Види кривих Безьє

Для малювання за допомогою інструменту **Pen**, **P** його потрібно вибрати на панелі інструментів, а потім виконати операції, які описано нижче.

ДОДАВАННЯ ВУЗЛІВ (ТОЧОК)

Для створення першого вузла потрібно клацнути мишкою у будь-якому місці на полотні. Кожне наступне клацання створює новий вузол, з'єднуючи його з попереднім. Якщо клацнути мишкою, утримуючи клавішу **Shift**, можна додавати вузли, вирівнюючи їх горизонтально, вертикально або під кутом 45° (Рис. 84).



Рис. 84. Додавання вузлів

СТВОРЕННЯ КРИВИХ

Для створення кривої потрібно клацнути і перетягнути мишку після додавання вузла. Вузли з'являться з керуючими лініями (handles), які можна перетягувати, змінюючи форму кривої (Рис. 85).



Рис. 85. Створення кривих

ЗАВЕРШЕННЯ МАЛЮВАННЯ

Для завершення створення фігури достатньо натиснути на клавішу **Enter**.

Щоб закрити контур потрібно клацнути мишкою на початковий вузол. Це створить замкнену форму.

РЕДАГУВАННЯ ВУЗЛІВ І КРИВИХ

Як і з примітивами, користувач може змінити пропорції векторних об'єктів, виділивши їх і потягнувши за будь-який із кутів або сторін. Але на цьому можливості редагування не завершується. Окрім того, користувач може:

- додати ще лінії;
- змінити або перемістити точки;
- налаштувати кривизну;
- додати заливку замкненим контурам.

Для проведення таких операцій спершу потрібно увійти в режим редагування. Для цього слід виділити об'єкт і натиснути клавішу **Enter** або двічі клікнути мишкою у векторному об'єкті. У такому випадку Панель інструментів зміниться. Ліворуч з'являться нові кнопки для роботи з інструментами редагування сплайнів.

ІНСТРУМЕНТ MOVE, V. ПЕРЕМІЩЕННЯ ТОЧОК

Інструмент **Move**, **V** (Рис. 86) дає змогу виділити і перемістити одну або кілька точок векторного об'єкта мишкою. Тут потрібно затиснути клавішу **Shift**, як і під час побудови нової кривої. Також для переміщення точок можна використати клавіатуру (клавіші зі стрілками). Цей інструмент активується за замовчуванням, коли користувач переходить у режим редагування.

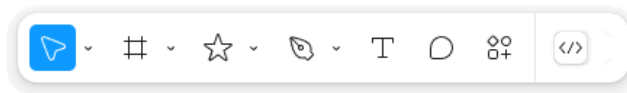


Рис. 86. Інструмент Move

ІНСТРУМЕНТИ BEND, CTRL. УПРАВЛІННЯ КРИВИЗНОЮ

Інструмент **Bend**, **Ctrl** (Рис. 87) використовується для керування та згладжування ліній.

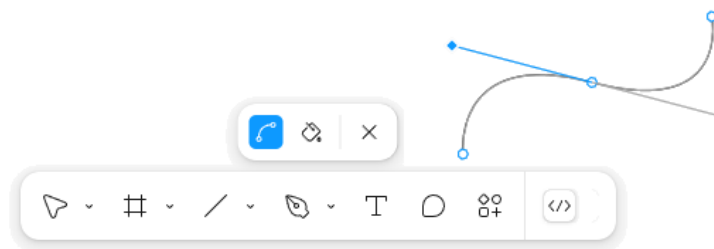


Рис. 87. Інструмент Bend

Для виконання таких операцій слід:

1. Перейти в режим редагування об'єкта.
2. Розмістити курсор між двома точками будь-якої лінії. У нижньому правому куті курсору з'явиться іконка зігнутої лінії.
3. Затиснути ліву кнопку мишки та потягнути. Лінія розпочне повторювати рухи користувача, намагаючись підлаштуватися під положення курсору.
4. Відпустити ліву кнопку мишки, щоб зафіксувати кривизну (Рис. 88).

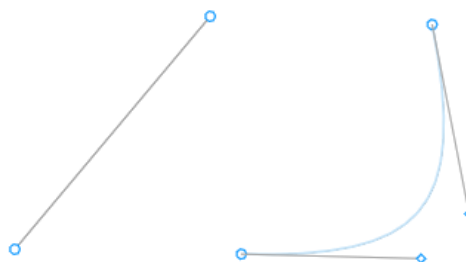


Рис. 88. Приклад проведення згладжування лінії

Керувати згладжуванням можна також через точку кривої.

Для цього слід виконати такі дії:

- Перейти в режим редагування об'єкта.
- Підвести курсор мишки до будь-якої точки кривої. У правому нижньому куті курсору з'явиться іконка з точкою та двома напрямними.
- Затиснути ліву кнопку мишки та протягнути. З точки з'являться дві симетричні напрямні, а лінії з боків точки згладяться. Щоб керувати лише однією напрямною, слід натиснути кнопку **Alt**. Щоб увімкнути прив'язку до кута нахилу (кратна 45 градусів), слід натиснути кнопку **Shift**.
- Відпустити ліву кнопку мишки, щоб зафіксувати створену кривизну (Рис. 89).

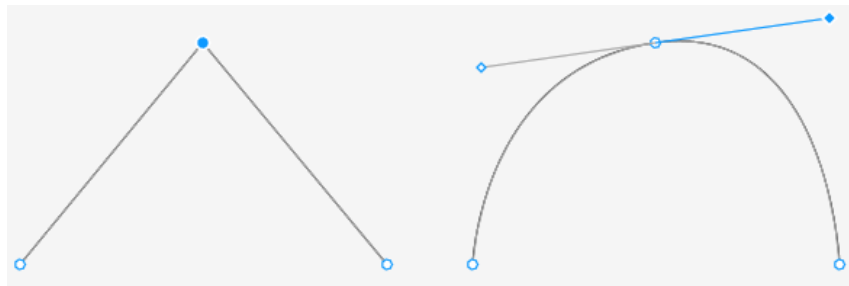


Рис. 89. Приклад проведення згладжування лінії через точку кривої

Нарешті можна керувати згладжуванням через направляючі точки.

Для керування згладжуванням через направляючі точки потрібно виконати такі дії:

1. Перейти в режим редагування об'єкта.
2. Навести курсор до будь-якої напрямної точки кривої. У правому нижньому куті курсору з'явиться іконка з точкою та двома напрямними.
3. Клацнути на ній лівою кнопкою мишки, щоб з'явилися направляючі лінії.
4. Перетягнути кінець направляючої лінії (направляючу точку) в потрібному напрямку (Рис. 90).

Якщо клацнути лівою кнопкою мишки по направляючій точці та відпустити, відповідна напрямна лінія буде видалена (Рис. 91).

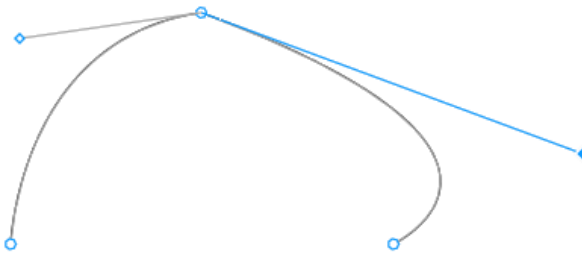


Рис. 90. Приклад проведення згладжування лінії через направляючі точки

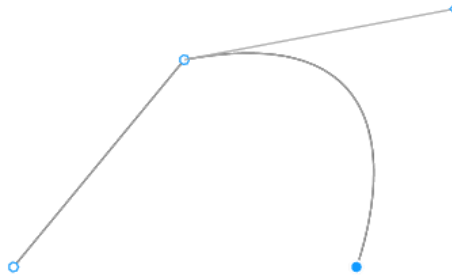


Рис. 91. Приклад видалення прямої

Часто трапляється, що направляючі ліній не відображаються, але за формою лінії помітно, що вони там є. У таких випадках для відображення напрямних потрібно перейти на інструмент **Move**, **V** і виділити потрібну точку. З'являться направляючі лінії, після чого потрібно перейти назад до режиму **Bend**, **Ctrl**.

ІНСТРУМЕНТ PAINT BUCKET.

ЗАЛИВАННЯ ЗАМКНУТИХ КОНТУРІВ

Інструмент **Paint Bucket** дає змогу **вибірково** зафарбовувати замкнені контури векторної фігури.

Налаштування заливки доступні у розділі **Fill** і повністю повторюють можливості звичайних примітивів. Для того, щоб частково зафарбувати контури фігури, слід виконати такі дії:

1. Вибрати інструмент **Paint Bucket**, **B**.
2. Клікнути мишкою всередині замкнутого контуру, який потрібно заповнити кольором.
3. Перейти до Панелі властивостей і у розділі **Fill** натиснути на прямокутник кольору, щоб відкрити меню налаштувань заливки.
4. Вибрати варіант заливки (Рис. 92).

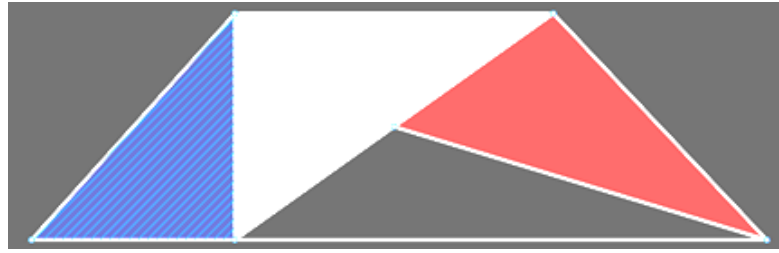


Рис. 92. Приклад заливки замкнених контурів

Щоб забрати заливку, потрібно повторно клікнути мишкою всередині зафарбованого контуру.

НАЛАШТУВАННЯ ОБВЕДЕННЯ

Налаштування обведення детально розглядатиметься у розділі [Обведення](#).

ЗАОКРУГЛЕННЯ КУТІВ

Для рівномірного заокруглення кутів кривої користувачу потрібно перейти до режиму редагування, виділити потрібні точки та вказати ступінь заокруглення у полі **Corner radius** на панелі властивостей (Рис. 93).

Такий спосіб заокруглення не впливає на направляючі точок і працює тільки з кутами контурів. Якщо користувач задав кривизну ліній за допомогою інструмента **Bend**, заокруглення працювати не буде.

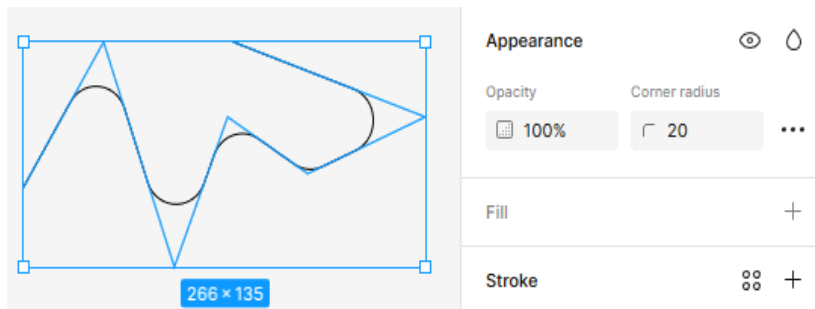


Рис. 93. Встановлення параметрів заокруглення кривих

ДОДАВАННЯ ТА ВИДАЛЕННЯ ТОЧОК

Як зазначалося, додати та видалити точки можна у будь-який момент, попередньо ввійшовши в режим редагування.

Для **копіювання точки** потрібно:

1. Вибрати інструмент **Move**, **V**.
2. Навести курсор на точку, яку потрібно скопіювати.

3. Затиснути кнопку **Alt**. Курсор зміниться на подвійний, підказуючи, що все готово до копіювання.
4. Затиснути ліву клавішу мишки та перетягнути курсор. Нова точка слідуватиме за курсором.

Для **додавання точки** всередині наявної лінії потрібно:

1. Вибирати інструмент **Pen, P**.
2. Навести курсор на середину лінії туди, куди потрібно додати точку (Figma підказуватиме нам середину лінії). До курсора у правому нижньому кутку додасться плюс.
3. Клікнути лівою кнопкою мишки. Точку додано.

Для **видалення точки** потрібно:

1. Вибирати інструмент **Pen, P**.
2. Навести курсор на точку, яку потрібно видалити.
3. Затиснути клавішу **Alt**. До курсора додасться мінус у нижньому правому кутку.
4. Клікнути лівою кнопкою мишки. Точку видалено (Рис. 94).

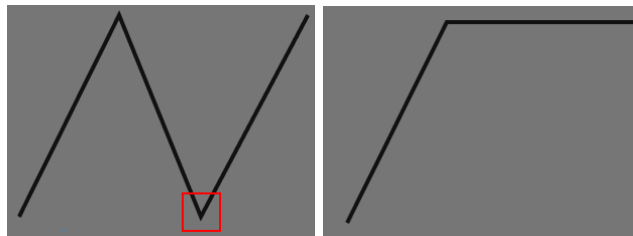


Рис. 94. Видалення точок

Увага: якщо виділити точку інструментом **Move** і видалити її кнопкою **Delete**, то Figma не з'єднуватиме сусідні точки і може утворитися окремий контур усередині шару. Цей спосіб ідеальний, якщо користувач видаляє окремі контури, наприклад, зайві елементи завантаженої іконки, щоб спростити її (рРис. 95).

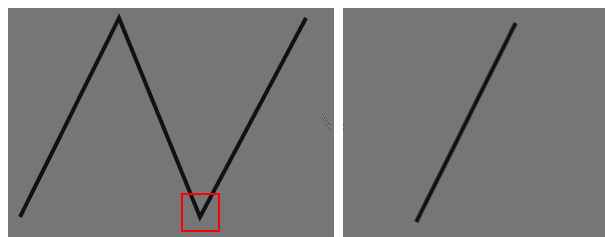


Рис. 95. Видалення зайвих елементів зображення

Додавати точки можна і поза наявними лініями. У такому разі векторний об'єкт складатиметься з кількох контурів.

ПЕРЕТВОРЕННЯ ОБВЕДЕННЯ У КРИВІ

На деяких етапах роботи, насамперед перед експортом іконки у формат SVG, може знадобитися перетворення всіх ліній у контури. Тобто, буде потрібно **конвертувати обведення у закриті контури із заливкою**. Це необхідно зробити для того, щоб іконки коректно відображалися у браузері. Після цієї операції користувач не зможе налаштувати товщину обведення або будь-які інші її властивості, крім кольору. Тому виконувати цю операцію рекомендується останньою чергою. Щоб конвертувати обведення у закриті контури із заливкою, потрібно викликати контекстне меню векторного об'єкта та виконати команду **Outline stroke, Ctrl+Shift+O**.

ВЕКТОРНІ МЕРЕЖІ

Векторна мережа – це одношарова фігура, що складається з векторних точок і ліній.

У більшості інструментів, включаючи Sketch та Illustrator, кожна векторна точка може мати максимум два відрізки, які до неї прилягають.

У Figma у векторну точку може сходитись будь-яка кількість ліній. Це надає великі творчі можливості для користувачів. Наприклад, не потрібно накладати дві точки різних об'єктів одна на одну, щоб імітувати один об'єкт (Рис. 96).

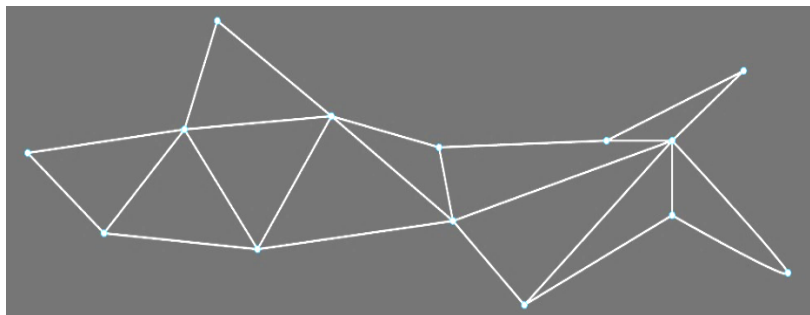


Рис. 96. Векторна мережа

Така можливість дає змогу користувачу швидко створювати складні фігури, які легко редагувати: налаштовувати товщину всіх ліній одним кліком мишки; рухати

всі точки одночасно або по одній (разом із точкою пересунуться і всі лінії, які до неї тягнуться) (Рис. 97).

Для створення векторної мережі потрібно вибрати перо **Pen**, **P**, намалювати два відрізки та клацнути мишкою у центральну точку між відрізками. Шлейф продовжує тягнутися із цієї точки (Рис. 98).

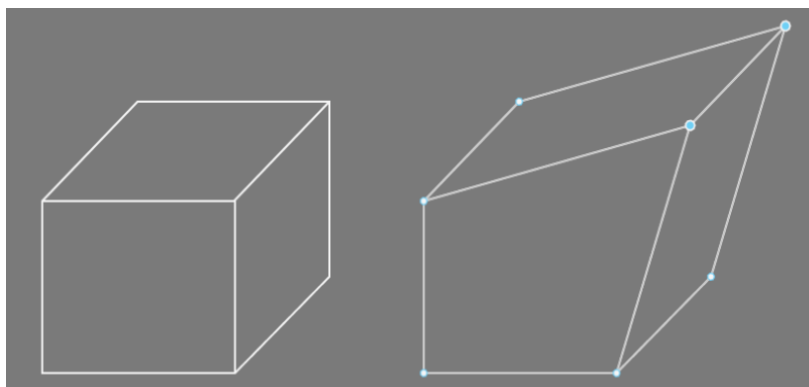


Рис. 97. Приклад редагування фігур за допомогою векторних мереж

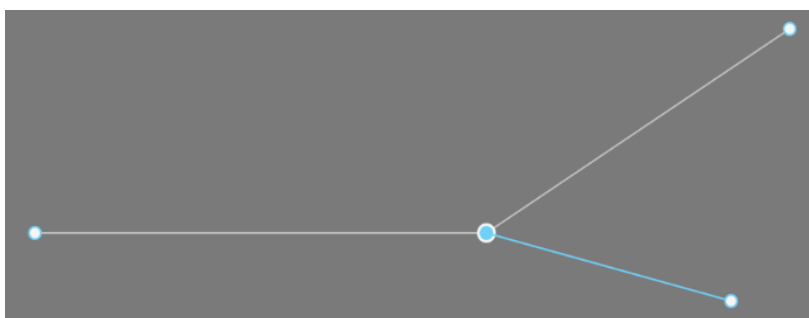


Рис. 98. Приклад створення векторної мережі

У результаті утворилася рогатка, в якій друга опорна точка має три лінії, що входять до неї. Завершуємо редагування (**Enter**). Таким чином спроектовано найпростішу векторну мережу.

Описані особливості Figma надають значні можливості під час роботи з вектором. Зокрема вони забезпечують гнучкість у роботі з лініями. Наприклад, щоб намалювати іконку за допомогою інструменту **Line** доведеться використати декілька ліній. І якщо змінити товщину ліній такої іконки, вона «розсиплеться» (Рис. 99).

Якщо ж намалювати фігуру як векторну мережу, то товщину ліній можна змінювати як завгодно (Рис. 100).

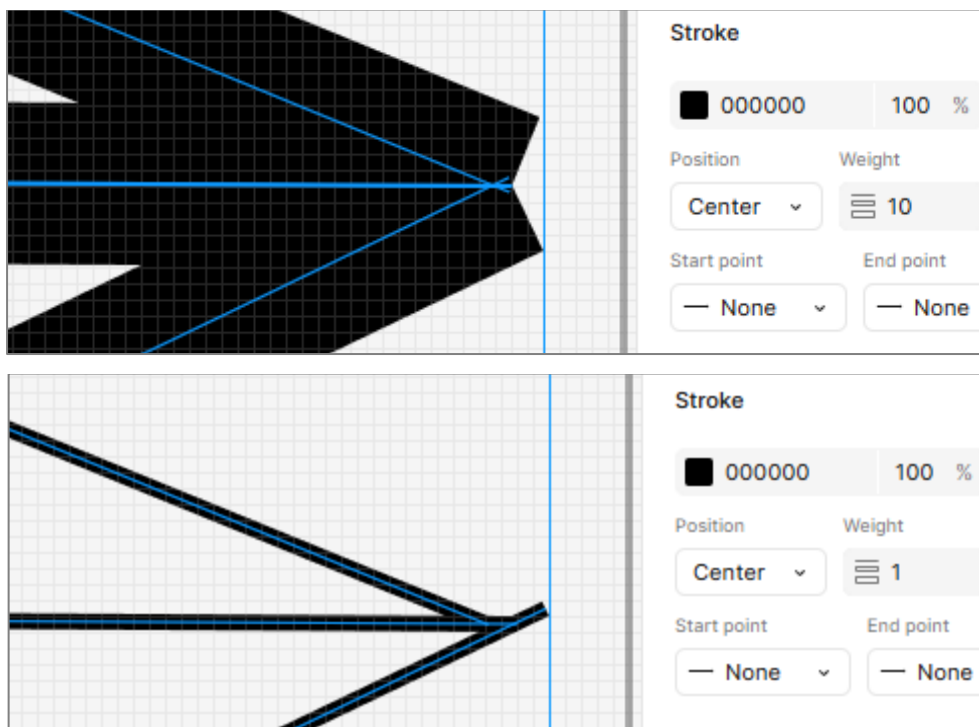


Рис. 99. Приклад редагування фігури, створеної інструментом Line

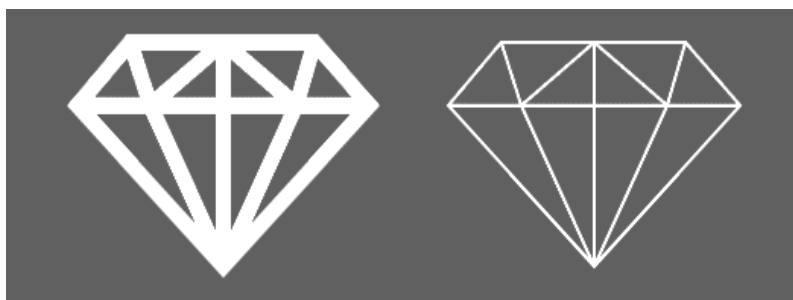


Рис. 100. Приклад редагування фігури, створеної векторною мережею

ТЕКСТ

80% дизайну інтерфейсів – це робота з типографікою. Figma має хороші можливості для роботи з текстом через спеціальний тип шарів.

СТВОРЕННЯ ТЕКСТУ

Щоб створити текстовий шар на панелі інструментів вибираємо інструмент **Text**, **T** і клацаємо мишкою на потрібне місце у робочій області або розтягуємо блок як прямокутник або фрейм (Рис. 101).

Figma підтримує системні шрифти пристрою та шрифти з **Google Fonts**, які зручно використовувати у веброзробці. Також можна встановити **Figma Font Helper** для доступу до локальних шрифтів системи.

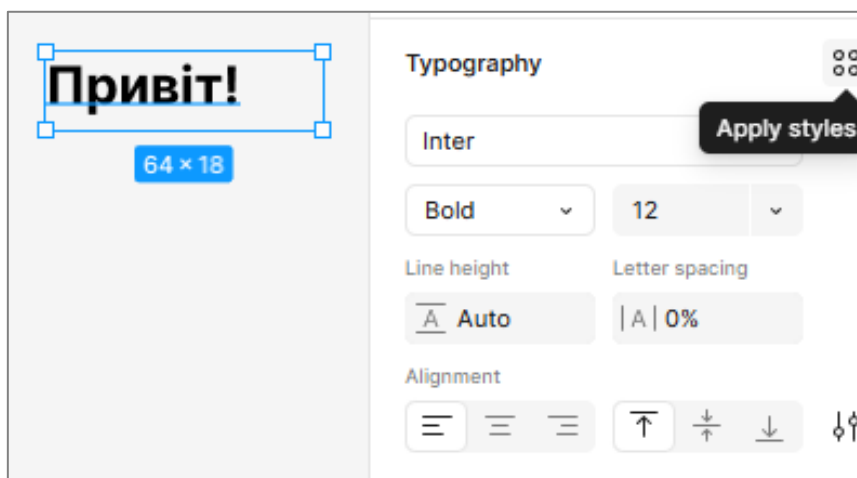


Рис. 101. Приклад створення тексту

Якщо виділити текстовий шар, то справа відобразиться панель роботи з текстом. Справа вгорі блоку розташовується іконка **Apply styles** – спливаюче меню стилів (Рис. 102), з якого можна вибрати готовий стиль тексту.

Далі йдуть поля:

- **Fonts** (шрифти);
- **Variable** (стилі шрифтів) – **Bold** (жирний), **Italic** (курсив), **Underline** (підкреслений) тощо;
- **Size** (розмір);
- **Line Height** (висота рядка) – регулювання міжрядкового інтервалу для поліпшення читабельності тексту A Auto;
- **Letter Spacing** (розрив між символами) – регулювання відстані між символами у тексті у % |A| 0%;
- **Alignment** (вирівнювання).

Режими вирівнювання по горизонталі:

Text Align Left (за лівим краєм), **Ctrl + Alt + L**;

Text Align Center (по центру), **Ctrl + Alt + T**;

Text Align Right (за правим краєм), **Ctrl + Alt + R**.

Режими вирівнювання по вертикалі:

Align Top (за верхнім краєм);

Align middle (по центру);

Align bottom (за нижнім краєм).

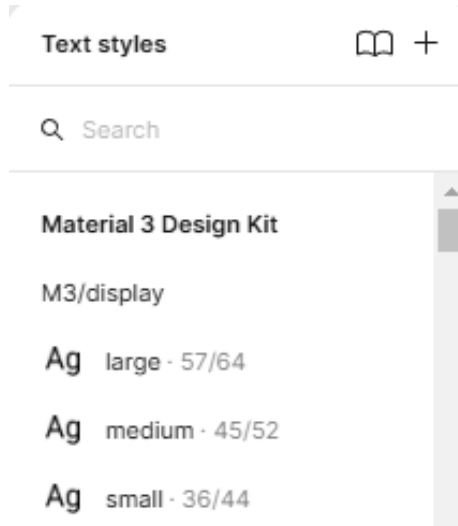


Рис. 102. Спливаюче меню стилів Apply styles

Опція **Type settings** (налаштування) (Рис. 103) – надає доступ до розширених можливостей роботи з текстом (Рис. 104).

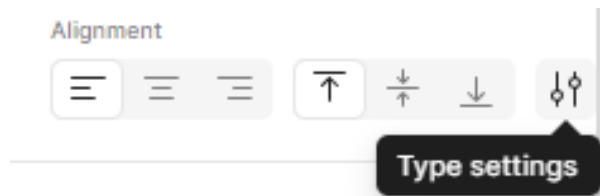


Рис. 103. Опція Type settings

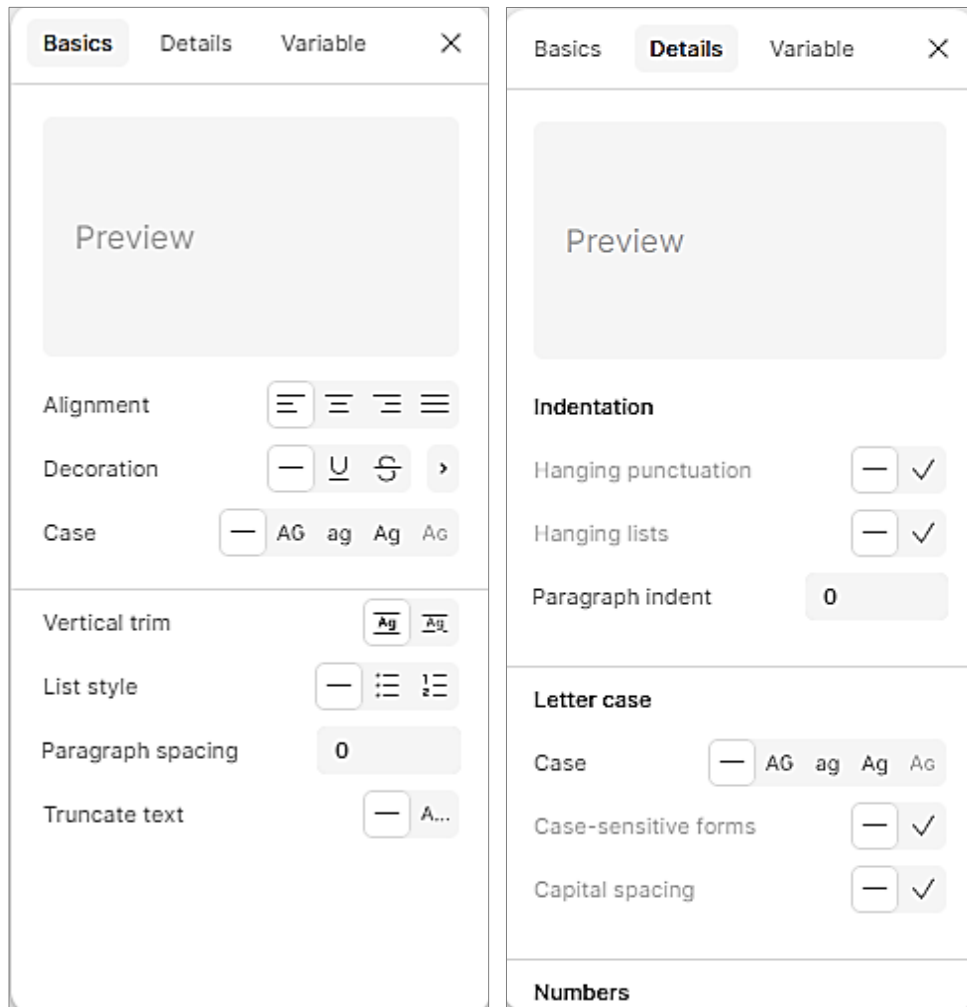


Рис. 104. Розширені параметри роботи з текстом

Тут можна додатково задати:

- **Text Align Justify** (за шириною), **Ctrl + Alt + J** – вирівнювання тексту по лівому і правому краю одночасно.
- **Decoration** (оформлення) – підкреслений, закреслений.
- **Case** (регістр) – всі літери великі, всі літери маленькі, кожне слово з великої літери, як у реченні.
- **Vertical trim** (вертикальна обрізка) – це опція керує відступами навколо тексту (зверху та знизу) й оптимізує висоту контейнера, зменшуючи висоту текстового блоку до фактичного тексту (Рис. 105).

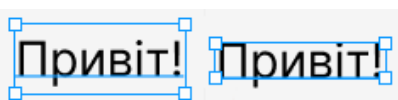


Рис. 105. Приклад вертикальної обрізки тексту

- **List style** (списки) – налаштування стилів для маркованих (bulleted) або нумерованих (numbered) списків.
- **Paragraph Spacing** (інтервал між абзацами) – відступ між абзацами, розділеними клавішею **Enter**.
- **Truncate text** (обрізати текст) – використовується для обмеження відображення тексту, коли він перевищує розміри текстового поля. Можливі два режими: **скорочення тексту** (якщо текст не поміщається у встановлене поле, він обрізається, а в кінці додаються три крапки (...)) та **збереження структури** (навіть якщо текст не відображається повністю, його вміст залишається в кадрі та його можна побачити при редагуванні) (Рис. 106).

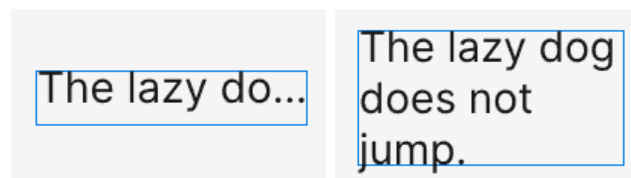


Рис. 106. Приклад скорочення тексту та збереження структури

У розділі **Layout** зібрані режими, які управляють розміщенням і форматуванням тексту всередині фреймів або компонентів (Рис. 107):

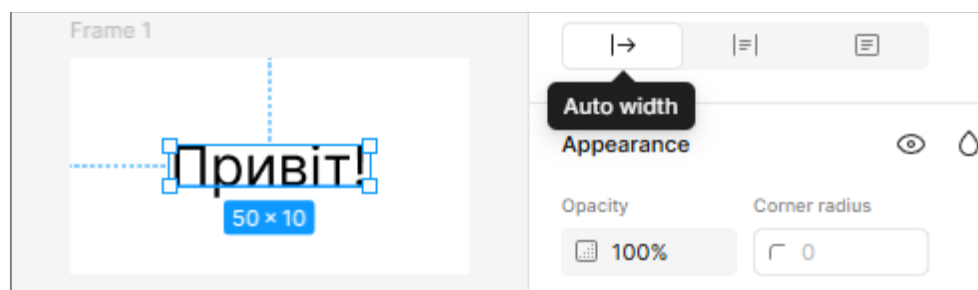


Рис. 107. Управління розміщенням і форматуванням тексту всередині фреймів або компонентів

- **Auto Width** $\rightarrow$ (автоширина) – ширина текстового контейнера (автоматично підлаштовується під довжину тексту, але водночас висота контейнера є фіксованою). Цей режим зручний у разі, якщо текст повинен розміщуватись в один рядок без перенесення (Рис. 108).

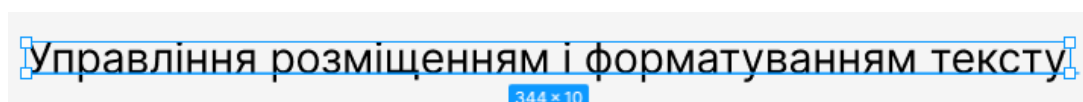


Рис. 108. Приклад застосування режиму *Auto Width*

- **Auto Height** (автовисота) – висота контейнера (автоматично підлаштовується під кількість рядків тексту, водночас ширина контейнера є фіксованою). Цей режим зручний у разі, якщо текст може займати кілька рядків, але ширина контейнера повинна бути фіксованою (Рис. 109).

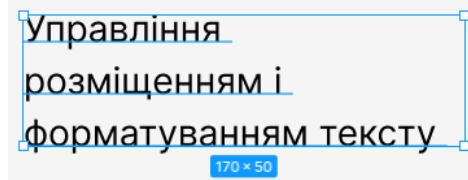


Рис. 109. Приклад застосування режиму *Auto Height*

- **Fixed Size** (фіксований розмір) – контейнер має фіксований розмір і текст або буде обмежено цим розміром, або, якщо він перевищує межі контейнера, буде обрізаний. Цей режим зручний під час створення блоків тексту з чітко визначеними розмірами (Рис. 110).

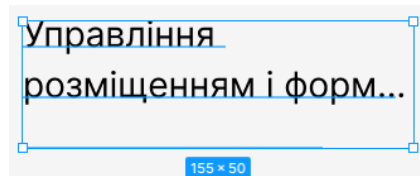


Рис. 110. Приклад застосування режиму *Fixed Size*

Якщо текстовий шар створити клацанням мишки, він матиме налаштування **Auto Width**. Якщо шар створений перетягуванням, текстовий блок одразу матиме фіксовану ширину **Fixed Size**.

ПЕРЕТВОРЕННЯ ТЕКСТУ У КОНТУР (OUTLINE TEXT)

Для перетворення тексту на криву необхідно викликати контекстне меню текстового елементу та вибрати команду **Outline stroke**, **Shift + Ctrl + O**. Після цього текст буде перетворений у векторний контур і користувач зможе редагувати кожну літеру як окремий об'єкт.

Перетворення тексту в контур доцільно здійснювати, якщо потрібно змінити форму літер або, якщо потрібно, щоб текст мав однаковий вигляд на всіх пристроях, незалежно від наявності потрібного шрифту (Рис. 111).

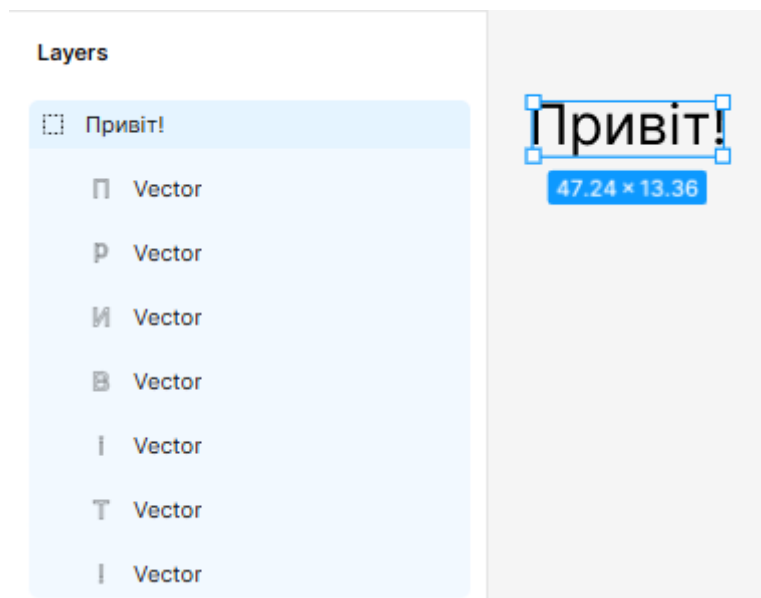


Рис. 111. Приклад перетворення тексту в контур

ЗАЛИВКА І ГРАДІЄНТ

Заливка **Fill** може бути у фрейма, примітива чи текстового шару. В цьому розділі розглянемо, які режими заливки бувають і як їх застосовувати на практиці.

Для того, щоб додати заливку до об'єкта, його потрібно виділити і на Панелі властивостей у групі **Fill** натиснути квадрат кольору. Якщо потрібно до одного об'єкта застосувати декілька заливок одночасно (при створенні складних композицій напівпрозорих кольорів) необхідно натиснути на значок **+**.

У верхньому лівому куті розгорнутої палітри відобразиться режим заливки фону. За замовчуванням обрано режим **Solid** – однорідна заливка. Можуть бути також градієнти, зображення та відео (Рис. 112).

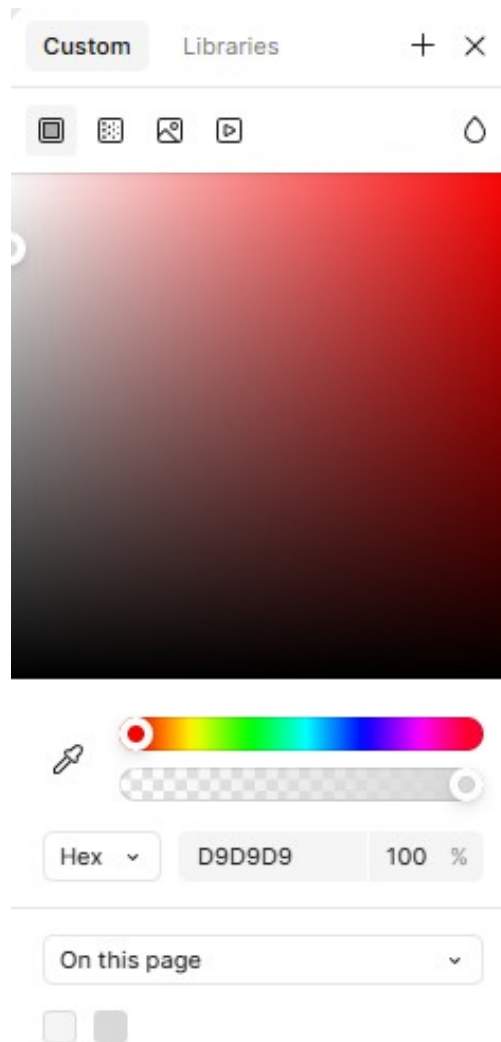


Рис. 112. Задання заливки та градієнта

Ліворуч від райдужного фейдера розміщена кнопка з піпеткою, **I**, що дає змогу знімати кольори у будь-якому місці полотна.

РЕЖИМ ЗАЛИВКИ SOLID: РІВНИЙ КОЛІР

Розглянемо найпростіший режим заливання суцільним кольором. Спершу колір відображається у виді HEX-коду. Такі коди не несуть інформації про те, що колір може бути прозорим.

Фейдер нижче райдужного дає змогу налаштувати **Opacity** (мутність, прозорість) заливки від 0 до 100. Також його можна задати у полі праворуч від HEX-коду у %. У прикладі (Рис. 113) сірий квадрат прозорий на 50%.

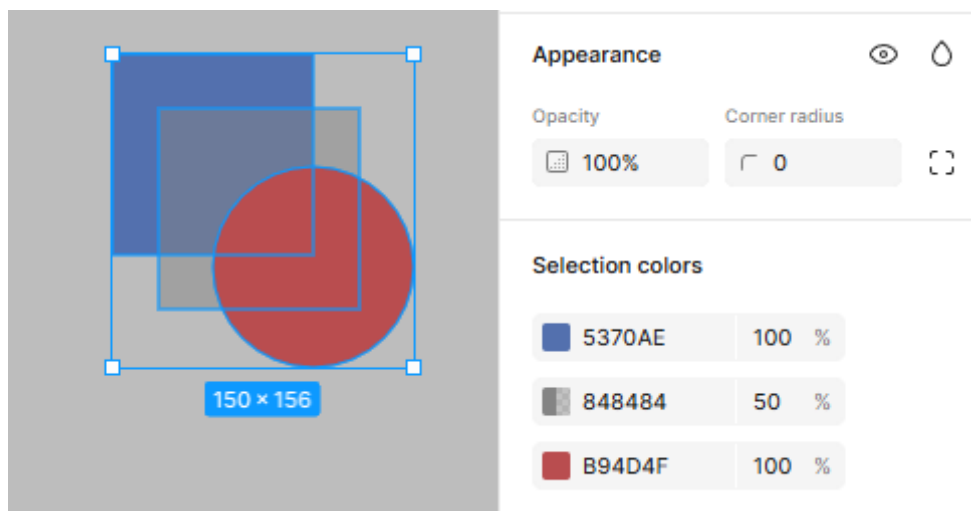


Рис. 113. Встановлення параметрів заливки об'єктів

Прозорість шару налаштовується у схожому полі в блоці **Opacity** розділу **Appearance** (Рис. 114).

Іконка краплі у верхньому правому куті цього розділу відкриває перелік режимів накладання кольорів. У наведеному прикладі встановлено режим за замовчуванням – **Pass Through**.

Необхідно пам'ятати, що режим прозорості шару та режим прозорості примітиву – не одне й те саме.

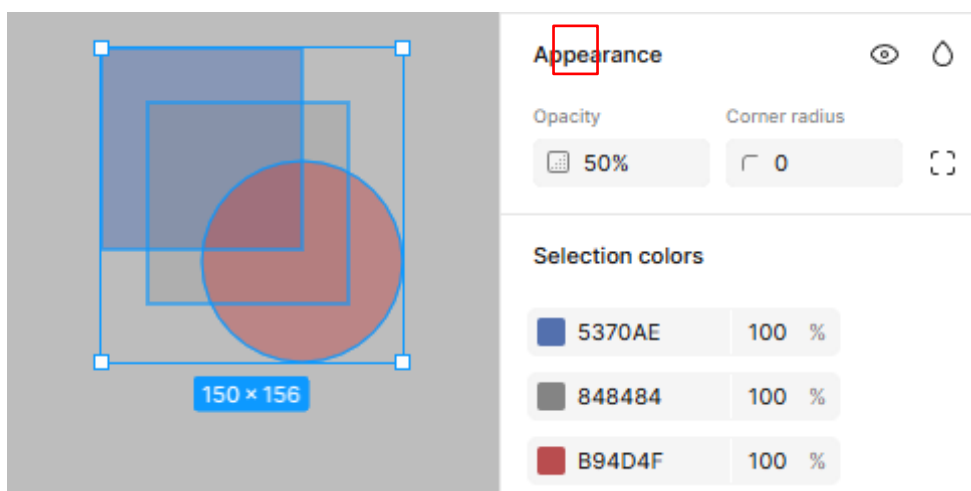


Рис. 114. Налаштування прозорості шару (заливки фігур повністю непрозорі, а прозорість шару 50%)

У режимі **Solid** можна переключити спосіб кодування кольору (Рис. 115). У режимі **RGB** відображається три поля для кожного каналу та четверте поле для альфа-каналу (Рис. 116).

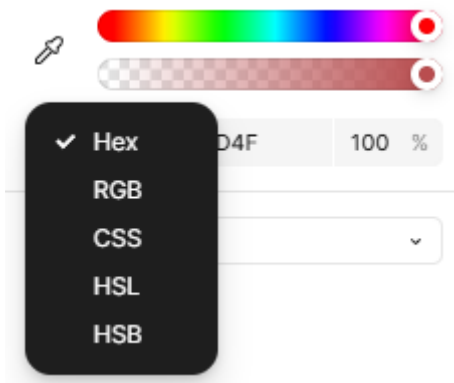


Рис. 115. Переключення способів кодування кольору

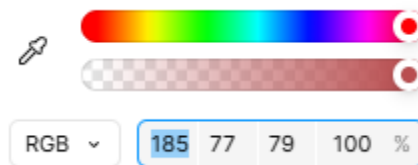


Рис. 116. Встановлення кольору заливки у режимі RGB

РЕЖИМ ЗАЛИВКИ GRADIENT: ГРАДІЄНТ

Градієнти – плавні переходи від одного кольору до іншого.

У Figma є кілька режимів градієнтів, які дають змогу створювати різноманітні візуальні ефекти:

- **Linear Gradient** (лінійний градієнт) – градієнт, який змінюється в одному напрямку (по прямій лінії) між двома або більше кольорами. Напрямок і кут можна налаштувати, перетягуючи маркери на полотні або змінюючи значення вручну.
- **Radial Gradient** (радіальний градієнт) – градієнт, який виходить з однієї точки (центру) і розходиться до країв елемента. Це дає змогу створювати ефекти, схожі на сяйво або світло, яке розходиться з певної точки.
- **Angular Gradient** (кутовий градієнт) – градієнт, який обертається навколо певної точки, створюючи ефект кольорового кільця або спіралі. Цей тип градієнта найкраще підходить для створення кольорових кругових або спіральних ефектів.
- **Diamond Gradient** (ромбовий градієнт) – градієнт, який розходиться з центру, але має форму ромба. Кольори змінюються від центру до кутів.

Кожен із цих градієнтів можна налаштовувати, змінюючи кольори, їхні позиції, прозорість та інші параметри для досягнення бажаного візуального ефекту.

На розгорнутій панелі градієнтів не дуже зручно оцінювати вибраний колір, тому що віконця контрольних точок градієнта маленькі. У такому разі перехід кольору виходить недостатньо точним. Якщо користувач заздалегідь визначить кожен колір, а потім встановить їх у градієнті, результат буде більш передбачуваним.

Для цього можна, наприклад, створити кілька прямокутників, **R** необхідних кольорів та під час налаштування градієнта знімати з цих пробників кольори піпеткою (Рис. 117). Насамперед це зручно під час роботи зі складними градієнтами, що складаються з кількох кольорів.

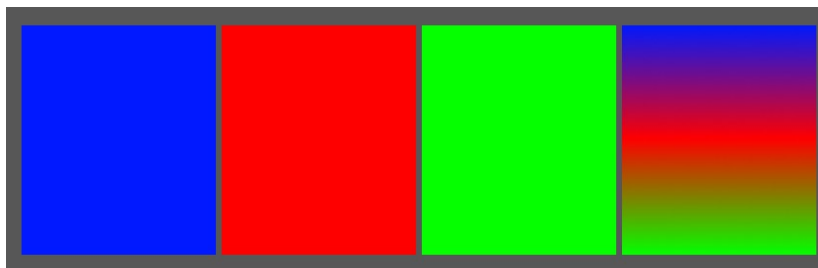


Рис. 117. Приклад створення градієнтної заливки

РЕЖИМ ЗАЛИВКИ LINEAR: ЛІНІЙНИЙ ГРАДІЄНТ

Лінійний градієнт має вісь, уздовж якої змінюється колір.

Задати кольори, які беруть участь у градієнті, можна за допомогою контрольних точок на його осі. На наведеному найпростішому прикладі (Рис. 118) їх дві: чорна та біла.



Рис. 118. Приклад створення лінійного градієнта

Створення градієнта

Розглянемо порядок створення градієнтної заливки. Для цього:

1. Створимо примітив квадрата, **R**. Розгорнемо вікно заливки. За замовчуванням його колір сірий #D9D9D9 (Рис. 119).

2. У лівому верхньому куті змінюємо режим заливки з **Solid** на **Gradient**.

У центрі фігури з'явилася вертикальна вісь і дві контрольні точки кольорів. Верхня точка виділена (Рис. 120).

Якщо прозорість однієї із контрольних точок установити на 0%, то виникне ефект плавного зникнення квадрата. Нижній квадратний кольоровий пробник розділений на дві частини. У лівій його половині відображається колір з урахуванням поточного налаштування прозорості (Рис. 121).

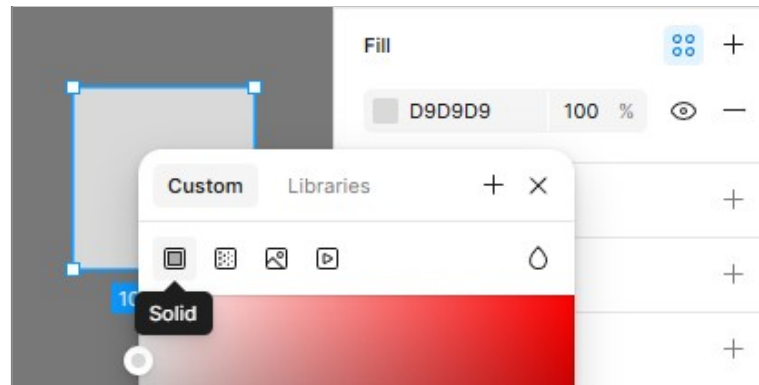


Рис. 119. Примітив квадрата з вікном заливки

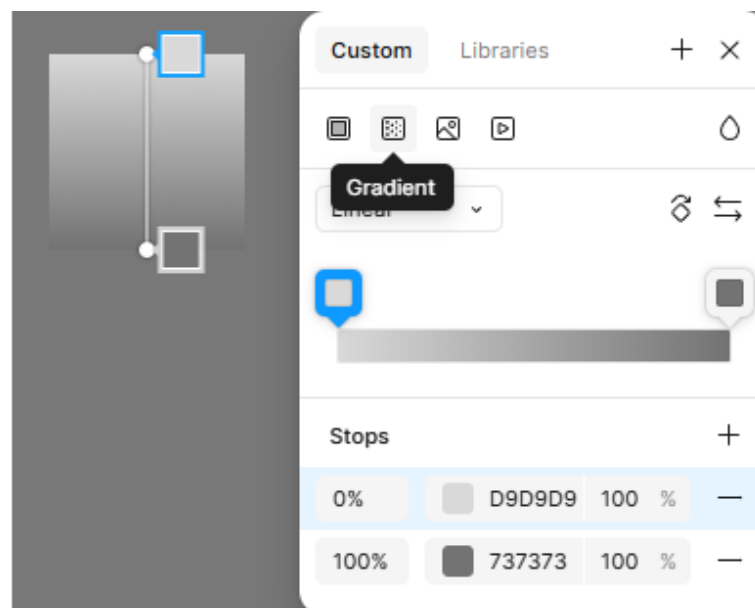


Рис. 120. Створення осі для градієнтної заливки

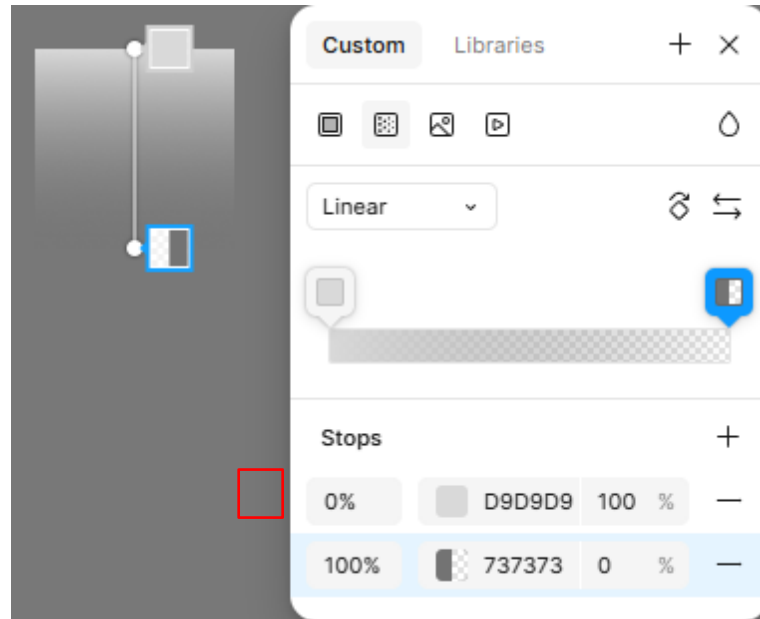


Рис. 121. Налаштування прозорості градієнтної заливки

Якщо у режимі **Solid** спершу обрано інший колір, то під час переходу в **Gradient** він буде використаний у градієнті (Рис. 122).

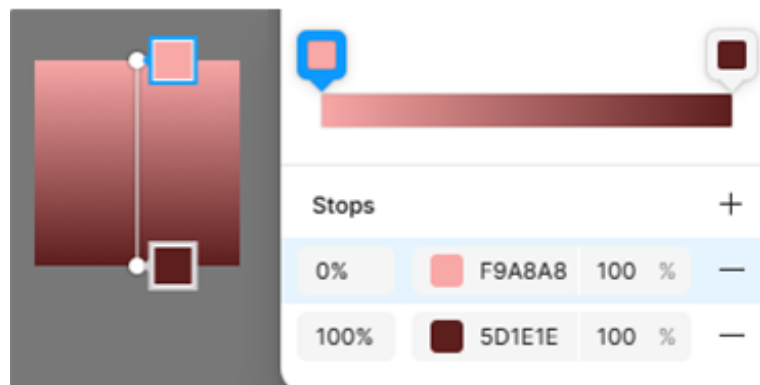


Рис. 122. Використання кольору в режимі *Solid* під час переходу в режим *Gradient*

1. Виділимо нижню точку, змінимо прозорість і змінимо колір. Перехід між кольорами буде явно помітним.
2. По осі можна клацнути мишкою і таким чином додати ще одну контрольну точку кольору (Рис. 123). Вона відобразиться у колонці **Stops** і у відсотковому еквіваленті буде відображено її місцерозташування. Якщо потрібно буде її видалити, натискаємо **Delete**.

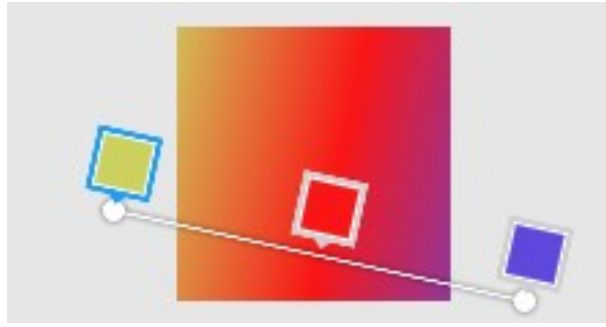


Рис. 123. Проведення маніпуляцій з віссю градієнтної заливки

Контрольну точку можна пересувати по осі. Також можна перетягувати кінці осі та задавати їй потрібний кут.

Корисні клавіші для роботи з градієнтами

Щоб зафіксувати градієнт та приховати панель заливки, потрібно натиснути клавішу **Enter** або **Esc**.

Якщо контрольну точку активовано, її можна зміщувати **стрілками** по осі градієнта. Затиснутий **Shift** пришвидшує рух **стрілки** на осі.

Під час зміни кута затиснутий **Shift** дає змогу перейти у режим **Snap To Geometry** і зафіксувати вісь під рівними кутами з кроком у 45° аналогічно, як було описано у розділі **Перо**.

Якщо схопити за один кінець лінії та затиснути **пробіл**, то всю лінію разом із її контрольними точками можна буде зсувати, не змінюючи кут. Це дає змогу швидко і точно налаштовувати положення градієнта, не рухаючи його кінці окремо.

Перпендикуляр градієнта

Ще одна унікальна особливість лінійних градієнтів у Figma – це можливість повертати кут градієнта відносно основної осі (Рис. 124).

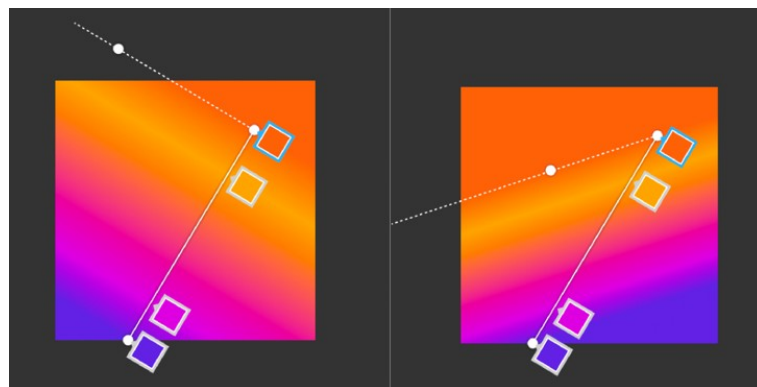


Рис. 124. Приклад кута повертання градієнта відносно основної осі

Гradient має третю непримітну круглу керуючу точку, яка з'являється під час затискання клавіші **Alt**. Якщо умовно провести пряму лінію до одного з кінців осі, то вийде кут 90°. Це додаткова перпендикулярна ручка осі gradienta. Якщо затиснути **Alt** і потягнути за неї, перехід кольорів буде розташовуватися перпендикулярно до неї.

РЕЖИМ ЗАЛИВКИ RADIAL: РАДІАЛЬНИЙ ГРАДІЄНТ

Такий gradient формується на основі еліпса. Він має центр і зовнішню межу. Вони з'єднані радіусом, де розміщуються контрольні точки кольору.

Приклад із двох накладених один на одного радіальних gradientів наведено на Рис. 125.

Для посилення ефекту сяяння застосовано режим накладання **Color Dodge** у верхньому шарі.

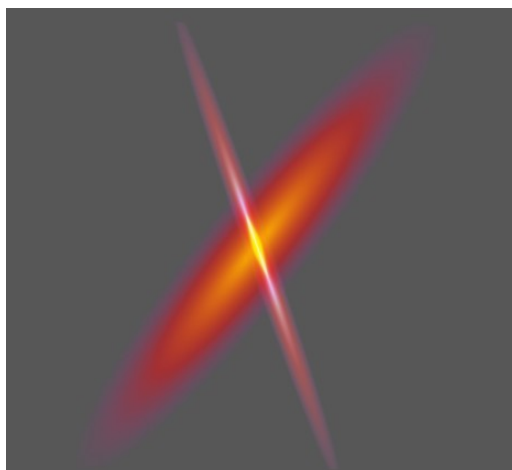


Рис. 125. Приклад накладення двох радіальних gradientів

Color Dodge – це один із режимів змішування (blending modes), який робить світліші області зображення ще світлішими, а темні – менш виразними. Доступ до режимів змішування можна отримати, натиснувши кнопку **Apple blend mode** (Рис. 126) в розділі **Appearance**.

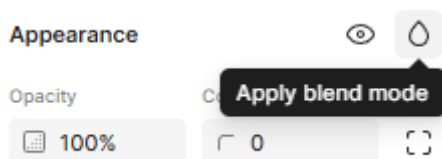


Рис. 126. Відкривання панелі режимів змішування

Створення найпростішого радіального градієнта

Для створення найпростішого радіального градієнта необхідно виконати такі дії:

1. Утворити новий квадрат, **R** та розгорнути панель **Fill**.
2. Перевести заливку з режиму **Solid** у режим **Radial**.
3. Аналогічно лінійним градієнтам утвориться дві контрольні точки сірого кольору, одній із яких надамо повну прозорість – 0%. На рисунку також відображатиметься радіальна вісь. Результат виконання цих операцій показано на Рис. 127.

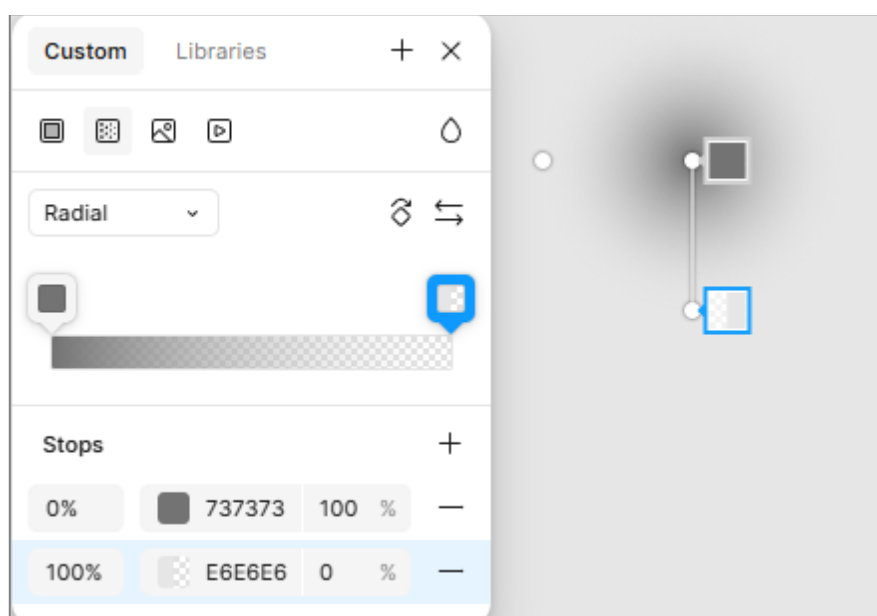


Рис. 127. Приклад творення найпростішого радіального градієнта (кроки 1–3)

4. Для того, щоб створювати радіальні градієнти не лише на основі кола, можна скористатися керуючою точкою зліва. Наприклад, щоб сплюснути коло в еліпс, тягнемо керуючу точку до центру (Рис. 128).

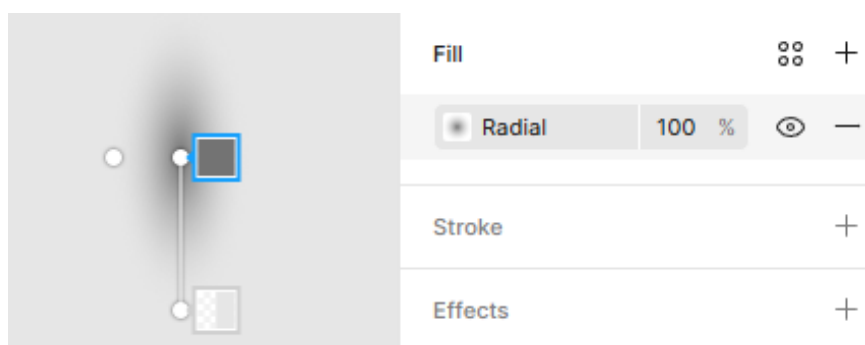


Рис. 128. Приклад створення найпростішого радіального градієнта (крок 4)

5. Можна також використовувати клавішу **Alt**, як і у лінійних градієнтах. Вона дає змогу зміщувати керуючу точку так, щоб еліпс спотворювався. Це зручніше, ніж налаштовувати загальний кут нахилу радіусу, а потім утискати еліпс за малим радіусом. У прикладі з затиснутим **Alt** потягнули керуючу точку вниз і в ліво (Рис. 129).

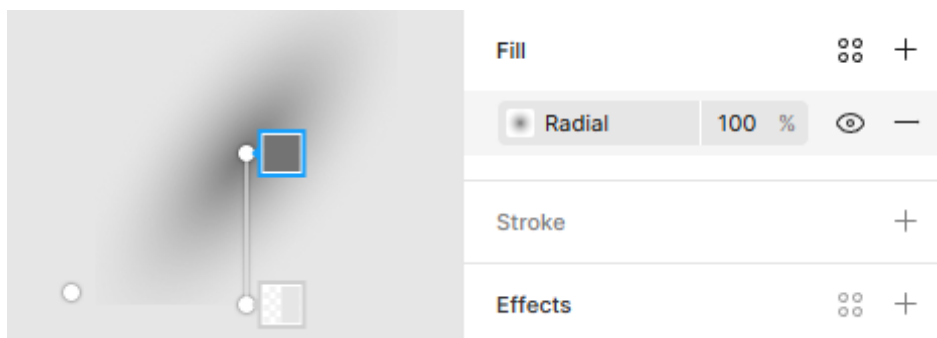


Рис. 129. Приклад створення найпростішого радіального градієнта (крок 5)

Повторимо приклад зі саявом (див. Рис. 125):

6. Створимо еліпс, нахилимо його та накладемо градієнт (Рис. 130).

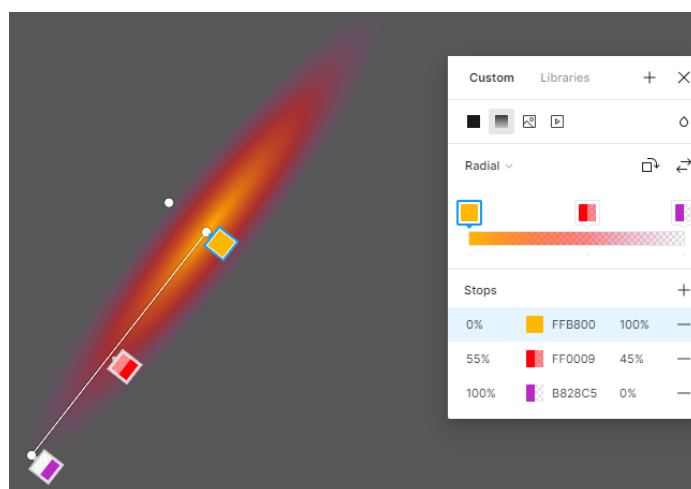


Рис. 130. Приклад накладання ефекту «саяво»

7. Зробимо копію першого градієнта. Для цього потрібно виділити створений градієнт у списку **Fill** і натиснути комбінацію клавіш **Ctrl + D**. Таким чином, на фігурі квадрата утвориться два шари заливки. Схоже працюють шари і для обведення **Stroke**.
8. Змінюємо скопійованому Градієнту кут і переміщаємо перпендикуляр якомога ближче до центру.

9. Після створення градієнта, у тій же панелі властивостей необхідно вибрати опцію **Apple blend mode** (режим накладання) позначене краплею.
10. Вибираємо режим **Color Dodge** зі списку режимів накладання.

За допомогою радіальних градієнтів можна промальовувати округлі поверхні, імітуючи об'єм (Рис. 131).

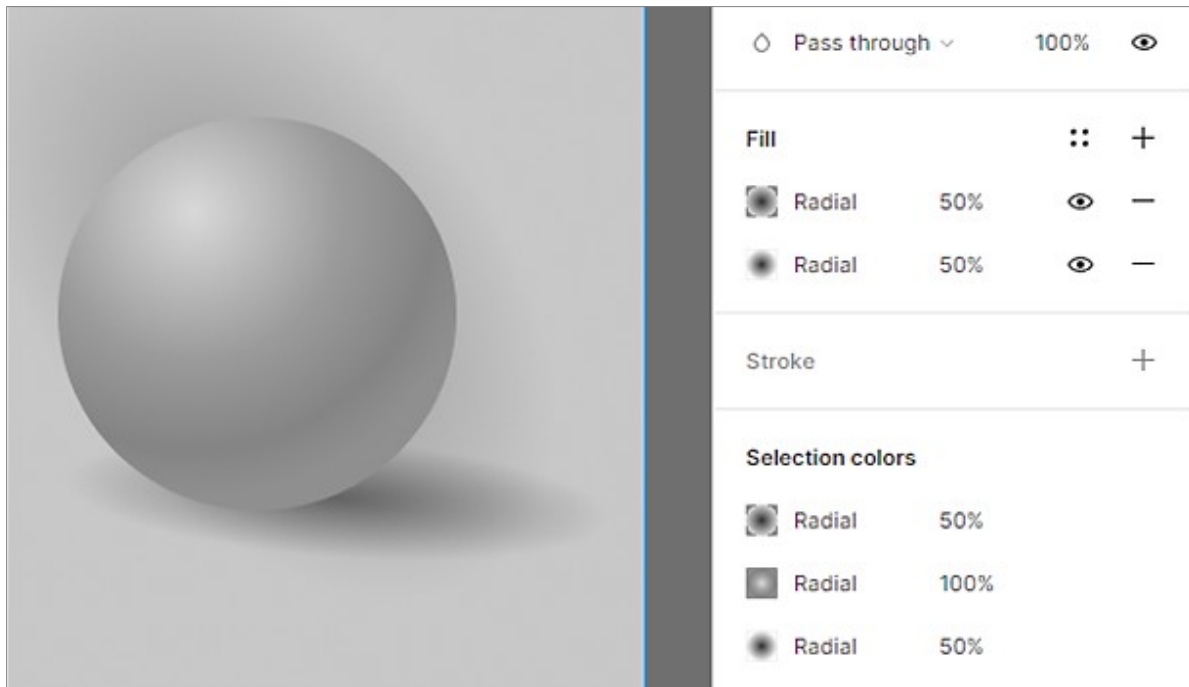


Рис. 131. Імітація об'єму за допомогою радіального градієнта

РЕЖИМ ЗАЛИВКИ ANGULAR:

КУТОВИЙ ГРАДІЄНТ

Такий градієнт формується на основі кута у 360° , в якому перехід кольору йде від початкової точки до кінцевої. Оскільки цей кут складає повне коло, для керування таким градієнтом також використовується еліпс. Однак перехід кольору в ньому відбувається за зовнішнім контуром, а не радіусом.

Кутові градієнти не підтримуються у форматі SVG, тому такі фігури доведеться експортувати у растр.

Для створення простого радіального градієнту потрібно.

1. Створити, наприклад, квадрат, **R**.
2. Перейти до режиму **Angular**. Відобразиться коло, діаметр якого відповідає довжині ребра квадрата.

Початкова та кінцева точка розміщуються на 6 годину, а перехід здійснюється за годинниковою стрілкою. Як і у будь-якому іншому режимі, градієнт спершу складається з одного кольору з різним відтінками (Рис. 132). Починається з кольору, що був вибраний у лінійному режимі, а закінчується темнішим або світлішим.

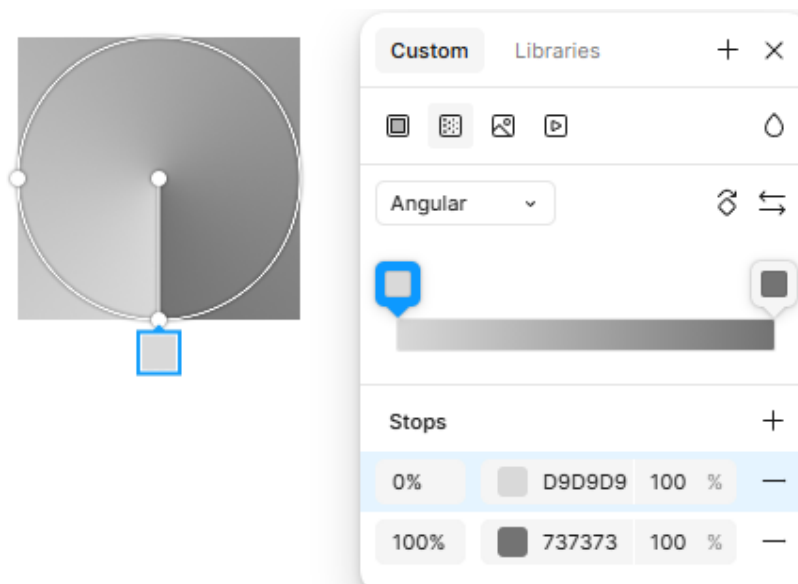


Рис. 132. Створення кутового градієнта (приклад 1)

Щоб принцип побудови кутового градієнта став наочнішим, перетягнемо початкову контрольну точку кольору праворуч, на 3 годину. Кінцева точка так і залишиться на 6 годину. Тепер вони не накладаються одна на одну (Рис. 133). Тепер видно, як у обидві сторони від світлої точки колір поступово переходить від світлого до темного.

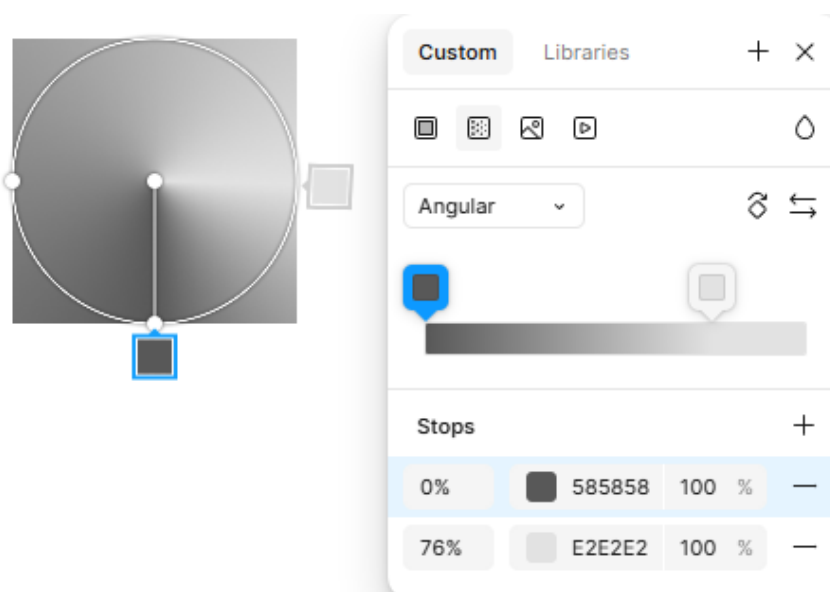


Рис. 133. Створення кутового градієнта (приклад 2)

Так само, як і в інших типах градієнтів, кутовий градієнт можна змінювати за допомогою контрольної точки зліва, повертаючи кут нахилу.

Використовуючи кутовий градієнт, можна зробити круговий спектр, розділивши коло на 6 рівних відрізків (Рис. 134).

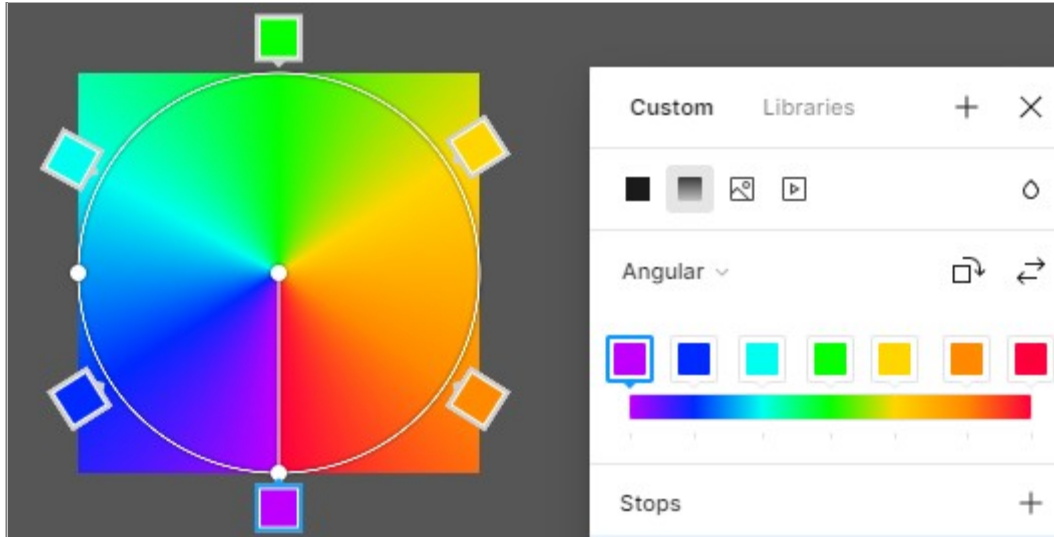


*Рис. 134. Поділ фігури на відрізки
під час створення кутового градієнта*

Також за допомогою кутового градієнта можна намалювати конічні поверхні. Форму конуса можна намалювати за допомогою еліпса та пера. Потім об'єднати їх в одну фігуру, розірвавши верхню точку еліпса (Рис. 135).

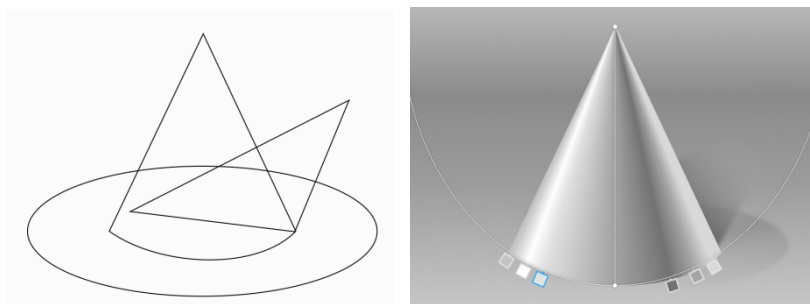


*Рис. 135. Створення конічної поверхні
за допомогою кутового градієнта*

Ще одне застосування радіальних градієнтів – круглі поверхні, схожі на хромовані. З них можна зробити ручки, кнопки й інший UI (Рис. 136).

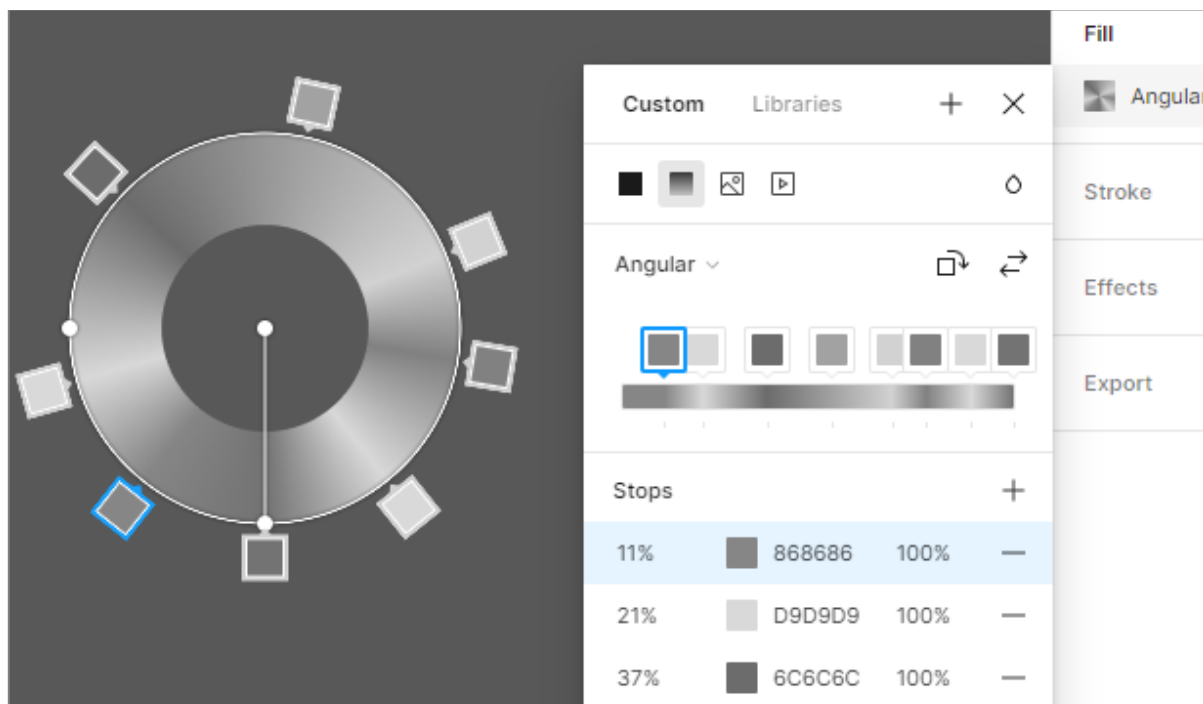


*Рис. 136. Створення круглих «хромованих» поверхонь
за допомогою кутового градієнта*

РЕЖИМ ЗАЛИВКИ DIAMOND

Цей режим дає змогу створювати градієнти, схожі на радіальні, з тією відмінністю, що їх зовнішній край переходу має форму ромба.

Цей режим застосовується доволі рідко, наприклад, для створення відблисків на хромованих ретро-футуристичних логотипах у стилі 80-х. Під час створення таких відблисків слід бути обережними із силою відблиску, інакше вийде нереалістично. Використання такого режиму заливки продемонструємо на такому прикладі:

1. Створимо квадрат, **R**.
2. Перемикнемо його заливку **Fill** у режим **Diamond**.

Як і у радіальному градієнті, у цьому типі вісь переходу кольору починається від центру фігури та закінчується на зовнішній межі (Рис. 137).

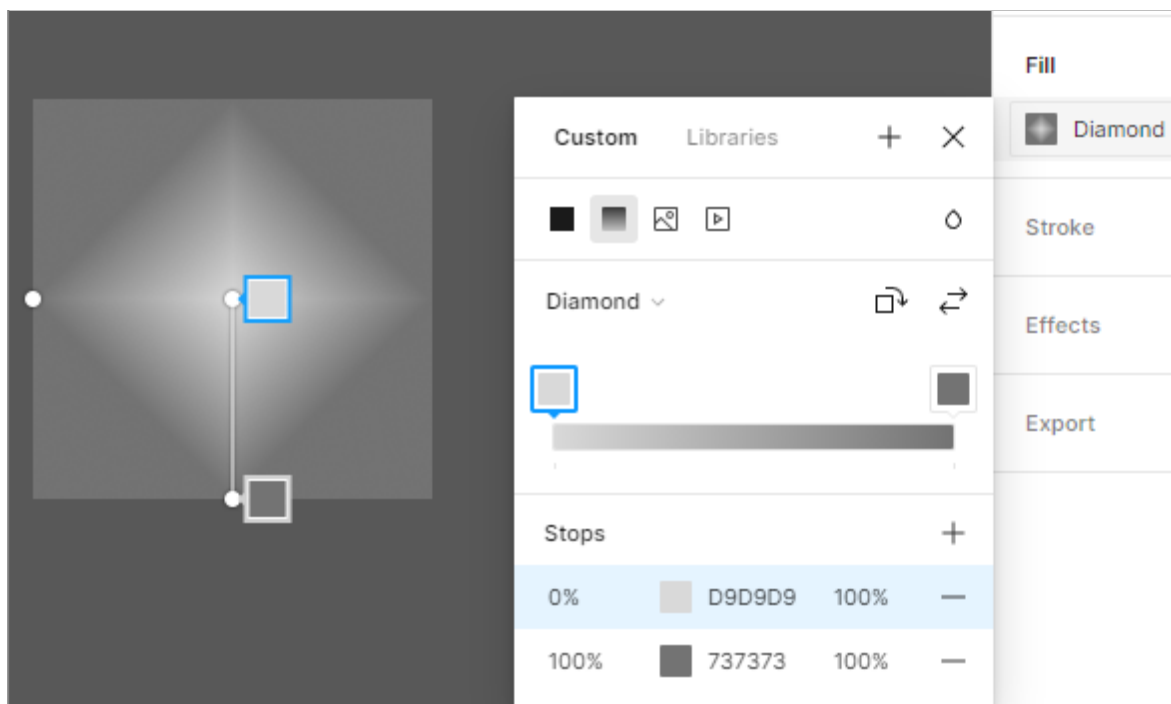


Рис. 137. Приклад створення заливки *Diamond* (крок 1, 2)

3. Виберемо першим кольором білий, посередині шкали – сірий, а наприкінці – колір робочої області. Зробимо його повністю прозорим (Рис. 138).
4. Переключимо градієнт у режим накладання (меню з краплею) **Color Dodge**. Цей режим підсилює білий у центрі та приглушує сірий на краях, даючи відчуття просвіту (Рис. 139).

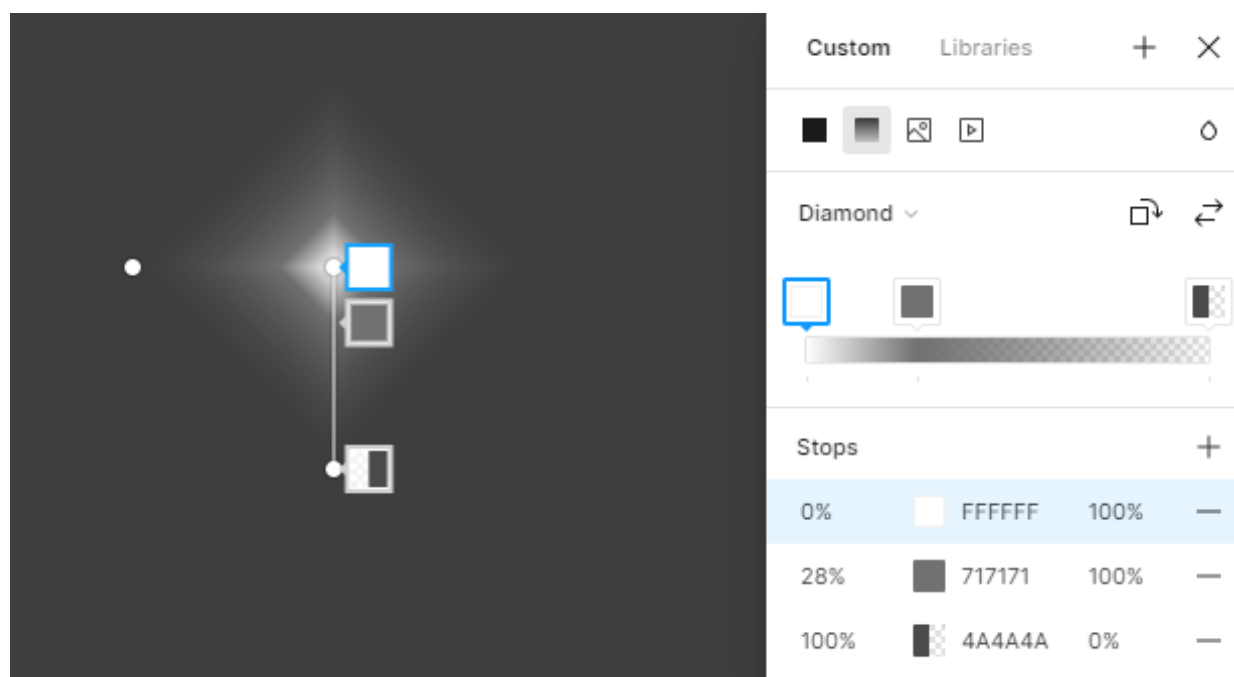


Рис. 138. Приклад створення заливки *Diamond* (крок 3)

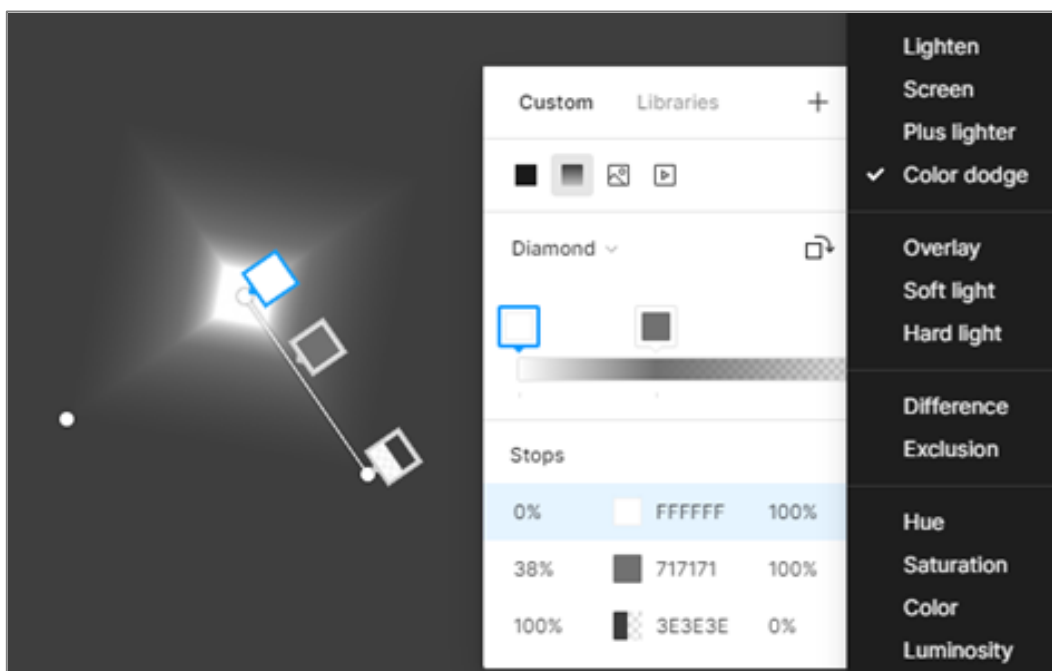


Рис. 139. Приклад створення заливки Diamond: режим накладання Color Dodge (крок 4)

5. Переміщаючи сіру точку, можна налаштувати бажаний розмір білого засвіченого ромба. Бажано, щоб форма зовнішнього ромба взагалі була розмита. У такому випадку залишиться лише внутрішній білий ромб і чотири промені.
6. Загнемо кут, утримуючись за нижню точку.
7. Вийдемо з режиму редагування градієнта, **Enter**.
8. Встановимо відблиск на край фігури, де могло б відбиватися найяскравіше джерело світла (Рис. 140).

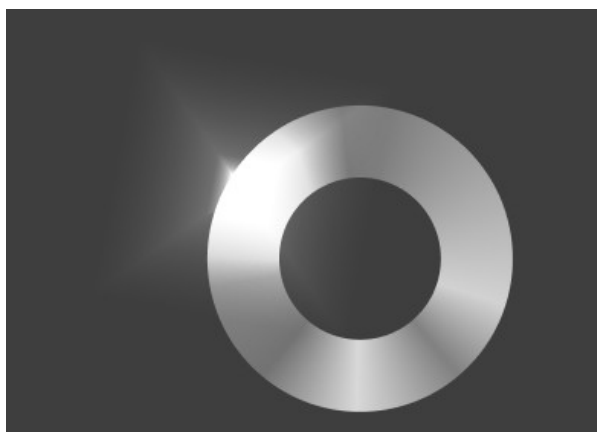


Рис. 140. Приклад створення заливки Diamond (результат)

РЕЖИМ ЗАЛИВКИ IMAGE

Режим **Image** – дає змогу накладати на фігури растрову заливку: світлини, патерни і текстури.

У разі вставлення у Figma будь-якого JPG або PNG-файлу, він з'явиться на полотні як прямокутний вектор. Растрове зображення накладатиметься на нього як Image.

Для того, щоб вставити растровий файл у Figma, можна використати перетягування або команду **File → Place Image**, **Shift + Ctrl + K**. На відміну від інших режимів заливок, якщо виділити шар з растром і натиснути **Enter**, користувач перейде у панель роботи із заливкою, а не в режим редагування векторної фігури. Таким чином інтерфейс акцентує роботу на тому, що у режимі **Image** користувачеві важливіше працювати із заливкою, а не з примітивом. На Рис. 141 слід звернути увагу на іконку зліва **Image 1** і вікно вибору режиму заливання, розташованого справа.

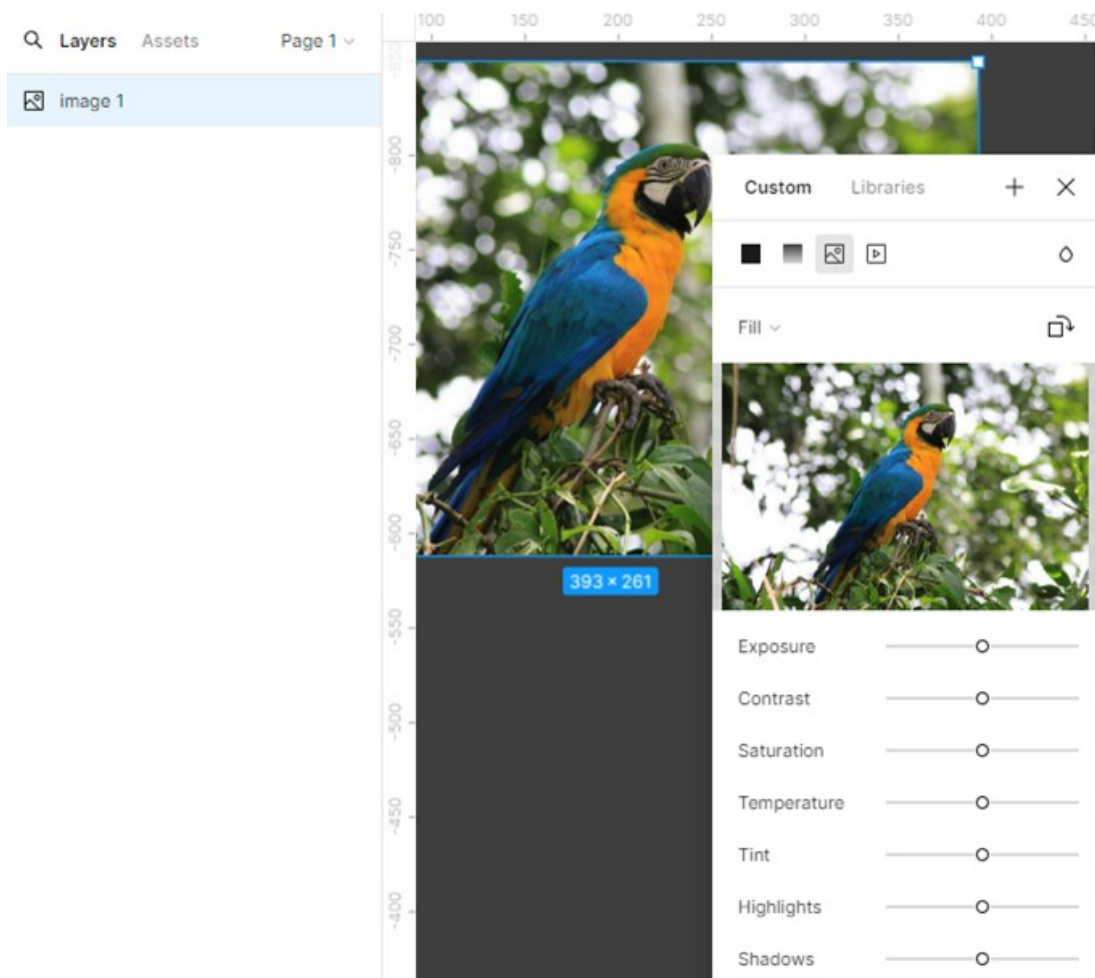


Рис. 141. Перехід до режиму заливки Image

Власне режим заливки **Image** може мати 4 варіанти (Рис. 142):

Fill – уся площа векторної рамки заповнюється растром.

Fit – растр вписується у векторну рамку.

Crop – режим, в якому можна точно налаштувати, як потрібно кадрувати растр, змінюючи форму векторної рамки, нахиляючи чи спотворюючи пропорції растру.

Tile – режим, в якому растр використовується як патерн і заповнює собою площу фігури «плиткою».

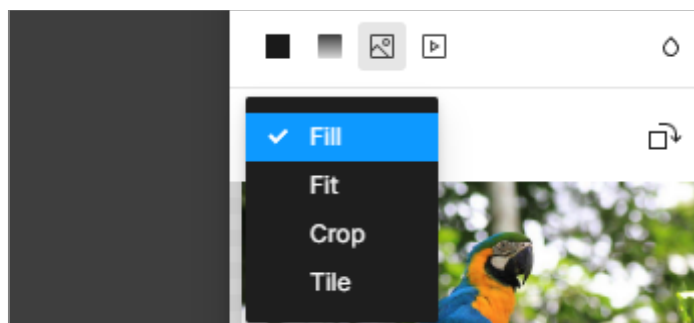


Рис. 142. Варіанти режиму заливки Image

РЕЖИМ IMAGE / FILL

Режим Image / Fill встановлюється за замовчуванням. Растрове зображення максимально заповнює площу фігури. Центр растру в тій самій точці, що й центр фігури-рамки. Якщо розтягнути прямокутник, то растр підлаштується під більшу сторону. Приклад розтягування у ширину показано на Рис. 143.

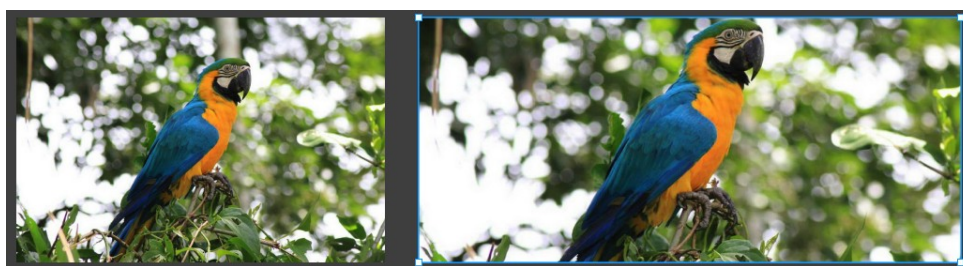


Рис. 143. Застосування режиму Image / Fill під час розтягування у ширину

Якщо стискати ширину так, щоб вона стала менша за висоту, то растр заповнюватиме всю доступну висоту (Рис. 144).

Центр растру і прямокутника, як і раніше, залишаються в одній точці. Зсунути зображення всередині прямокутника, змістивши центральну точку, не можна. Для цього використовується режим **Crop**.

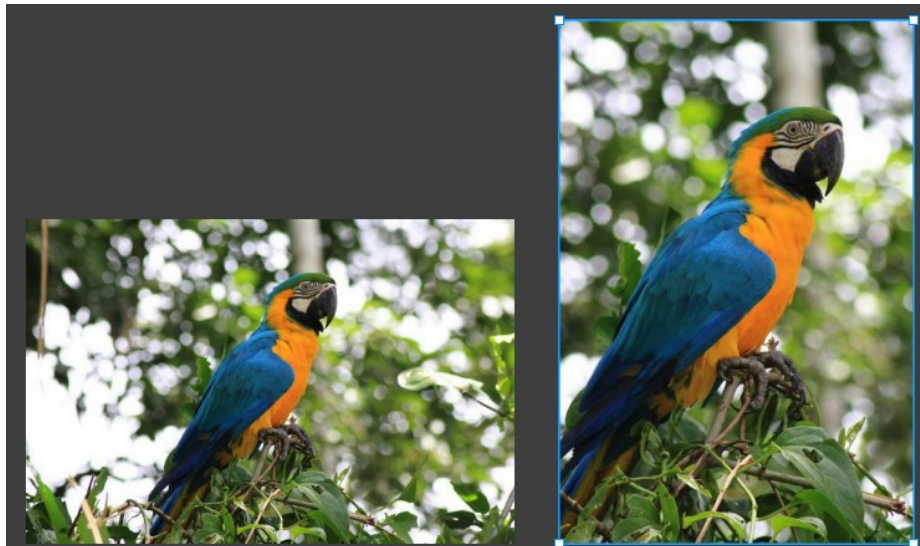


Рис. 144. Застосування режиму Image / Fill під час розтягування у висоту

РЕЖИМ IMAGE / FIT

У режимі Image / Fit растр вписується у фігуру за меншою стороною: він видимий і не обрізається прямокутником.

Якщо прямокутник і растр у ньому мають різні пропорції, з боків з'являються порожнечі (Рис. 145).

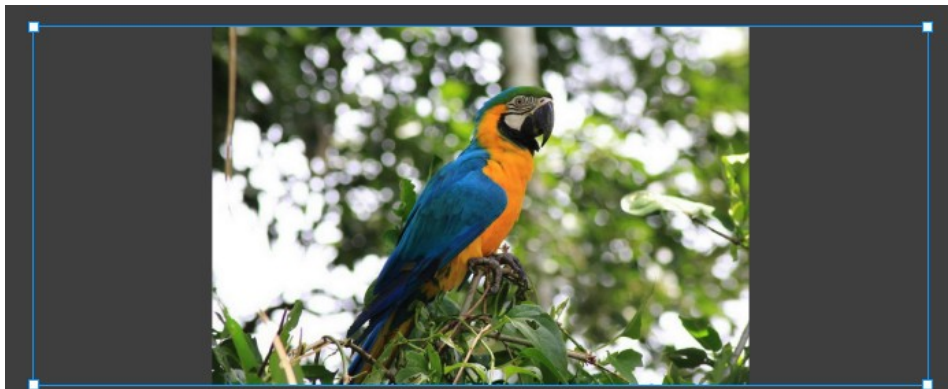


Рис. 145. Застосування режиму Image / Fit

РЕЖИМ IMAGE / CROP

У разі, коли користувач вставляє зображення до Figma, дуже часто виникає необхідність у його обтинанні. Режим **Crop** дає змогу змінювати пропорції векторної рамки, де є растрова заливка. Водночас растр не зміщується. Для використання цього режиму потрібно:

- Перемкнутись у режим **Crop**.

- Схопити за будь-який синій маркер і перетягнути його до центру зображення. Водночас рамка стискається (Рис. 146).

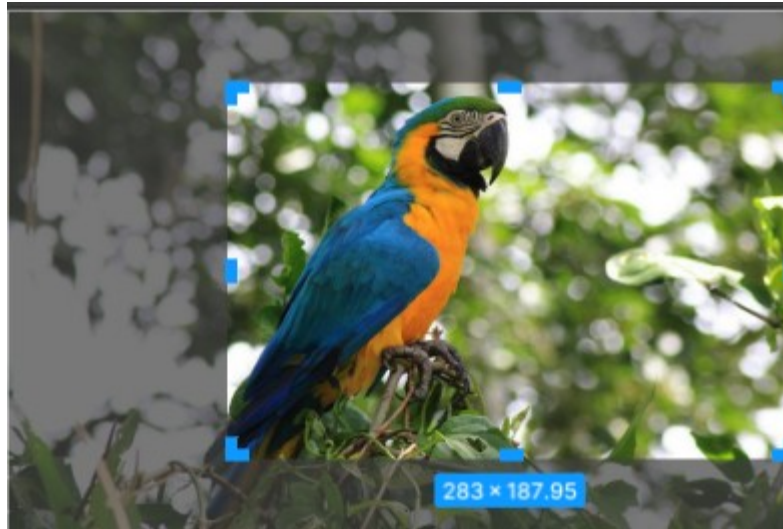


Рис. 146. Застосування режиму Image / Crop

Для **кадрування від центральної точки** під час перетягування маркера, затискаємо **Alt** (Рис. 147).

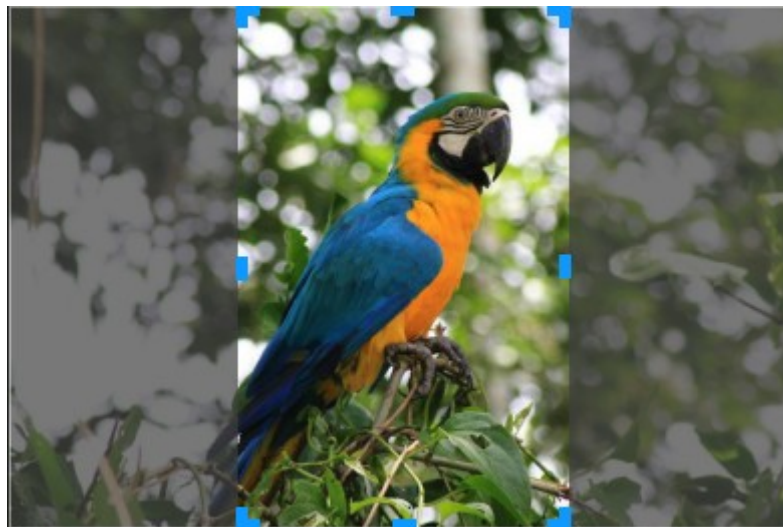


Рис. 147. Кадрування від центральної точки у режимі Image / Crop

Також у цьому режимі можна **пересувати зображення** у рамці. Для цього слід схопити за растр і перетягнути його в потрібний бік (Рис. 148).

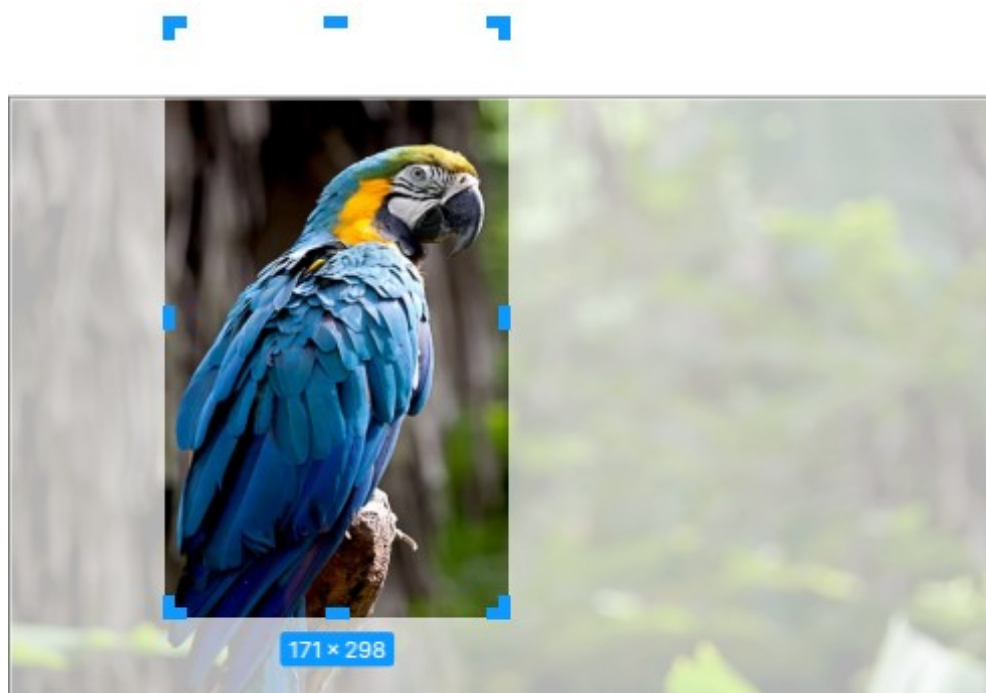


Рис. 148. Пересування зображень у режимі Image / Crop

Рамку **Crop** можна нахилити і таким чином виправляти «завалений» обрій. Для цього потрібно навести вказівник мишки на будь-який із кутових маркерів, доки він не перетвориться на вигнуту стрілку та нахилити рамку (Рис. 149).



Рис. 149. Нахилення рамки Crop у режимі Image / Crop

Також можна спершу нахилити сам растр, утримуючись за його кут, а потім виправити пропорції перетягуванням синіх маркерів, щоб обрізати порожні краї.

—Для того, щоб зафіксувати зміщене або обрізане зображення, потрібно двічі натиснути ліву клавішу мишки у межах рамки або натиснути клавішу **Enter**.

Водночас вихідне зображення не змінюється, а лише маскується. До нього можна повернутися, виділивши шар і вкотре натиснувши клавішу **Enter**. Щоб скинути зсув і кут нахилу **Crop**, можна повернутися у режим **Fill**.

РЕЖИМ IMAGE / TILE

Режим **Tile** використовується для заповнення фігури фрагментами, що повторюються – безшовними патернами. Зазвичай вони намальовані так, що якщо їх розмножити і поставити впритул один до одного, шви між ними будуть непомітними.

Для використання режиму **Tile** потрібно вставити растровий патерн у Figma та переключитися у режимі **Image** з **Fill** на **Tile**. З'явиться темна рамка з блакитними маркерами (Рис. 150).

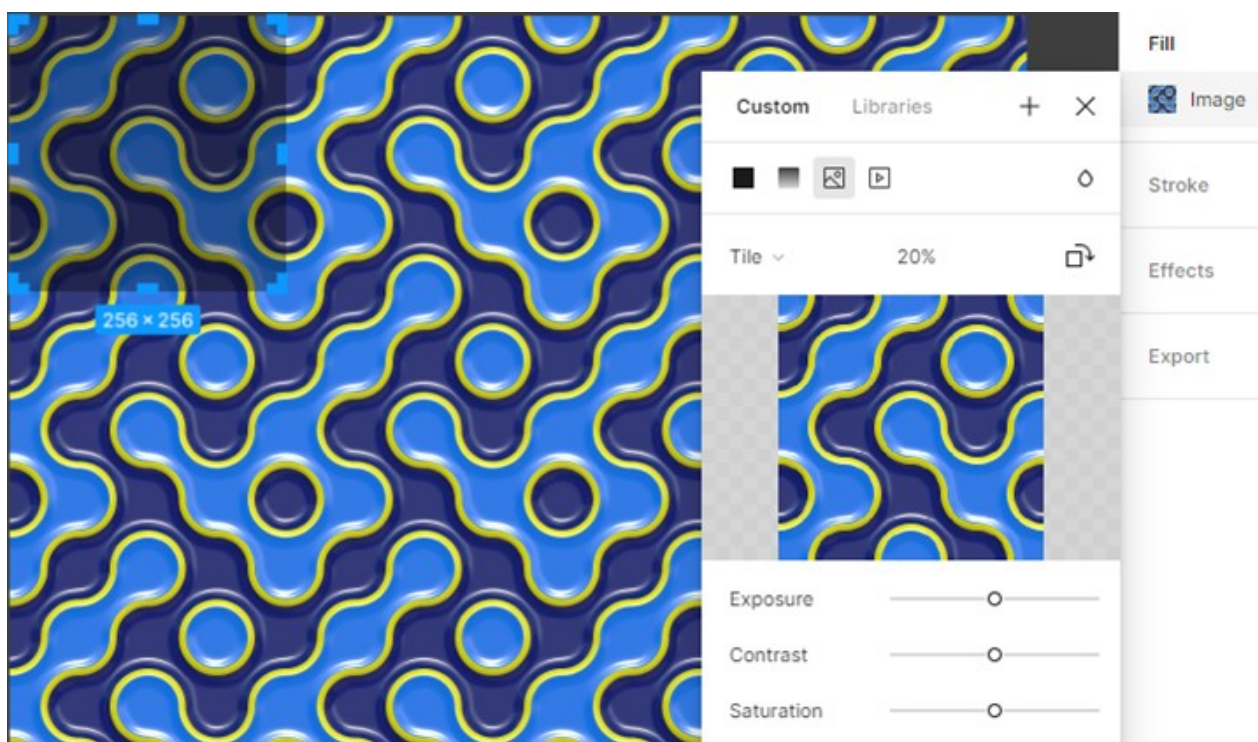


Рис. 150. Перехід у режим Image / Tile

Вона визначає, якого розміру буде сітка, за якою патерн замостить фігуру. Нижній правий маркер дає змогу налаштовувати бажаний розмір плитки.

Незалежно від розміру фігури патерн поширюватиметься на всю доступну площу. Також його розмір можна задавати у відсотках на панелі **Image**.

КОРЕКЦІЯ КОЛЬОРУ

У Figma доступні прості, але потужні фільтри швидкої корекції кольору світлин без додаткового використання зовнішніх редакторів (Рис. 151). Їх цілком достатньо, щоб зробити ілюстрації та аватари у дизайні яскравішими або надати їм потрібний акцент. Усі зміни корекції не змінюють вихідне растрове зображення.

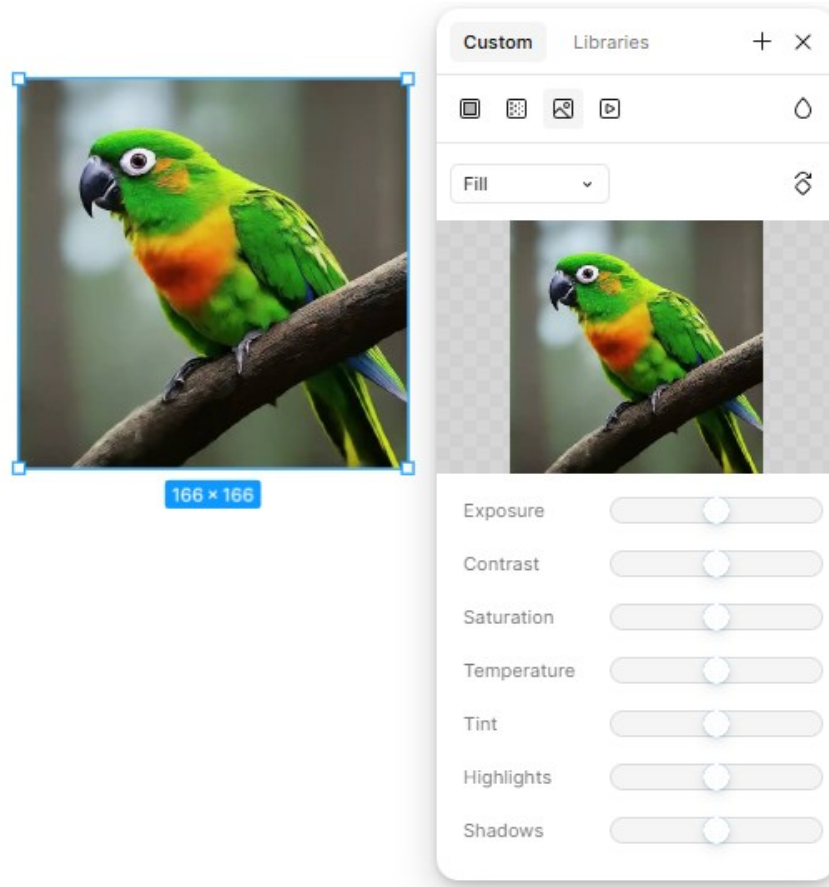


Рис. 151. Фейдери корекції кольору

Коригувати кольори можна на панелі **Fill** у тих шарах, де є растрова заливка. У режимі Image, доступні такі фейдери:

EXPOSURE

Exposure (експозиція) – базове налаштування яскравості світлин. Якщо кадр недотриманий, він темний і має чорні провали. Якщо світло занадто довго падало в об'єктив, кадр занадто світлий і має підсвічені ділянки – білі провали. У фото з пташкою (Рис. 152) експозиція була оптимальною, тому її вартувало б залишити у початковому значенні.

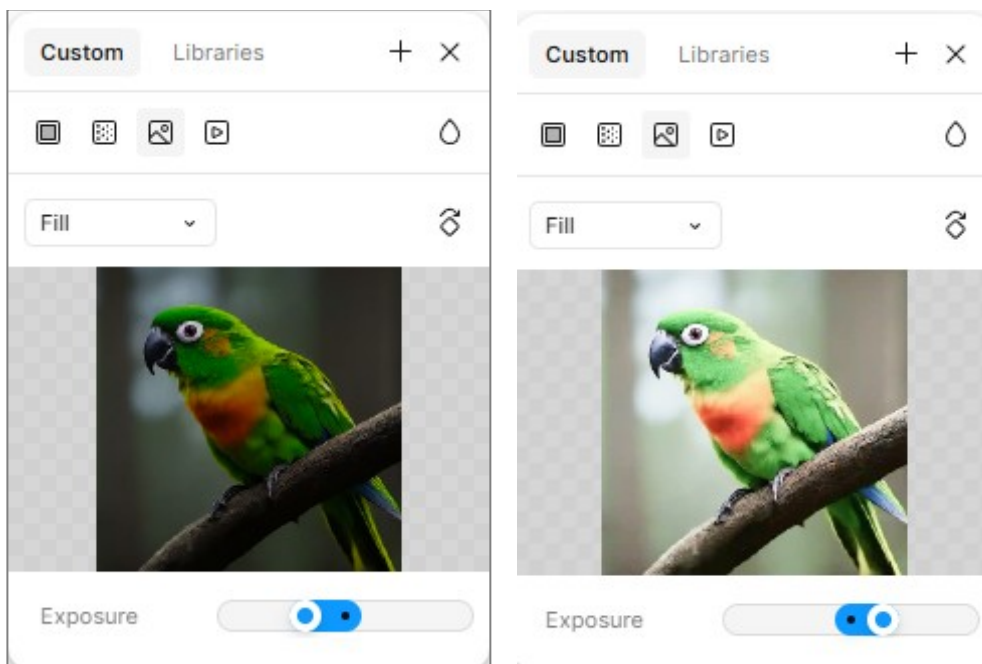


Рис. 152. Застосування експозиції

Якщо немає мети досягти певного художнього ефекту, підсилених чорних тіней і не потрібно підсвічувати окремі ділянки зображення, зміну експозиції слід за можливості уникати.

CONTRAST

Contrast (контраст) – збільшення **контрасту** дає змогу робити темні ділянки темнішими, а світлі світлішими. Зменшення контрастності приглушує між ними різницю. Якщо контрасту не вистачає, то зображення виглядає невиразно. Якщо його занадто багато, переходи кольору занадто різкі. На фото праворуч (Рис. 153) вони видимі як чорні та білі плями.

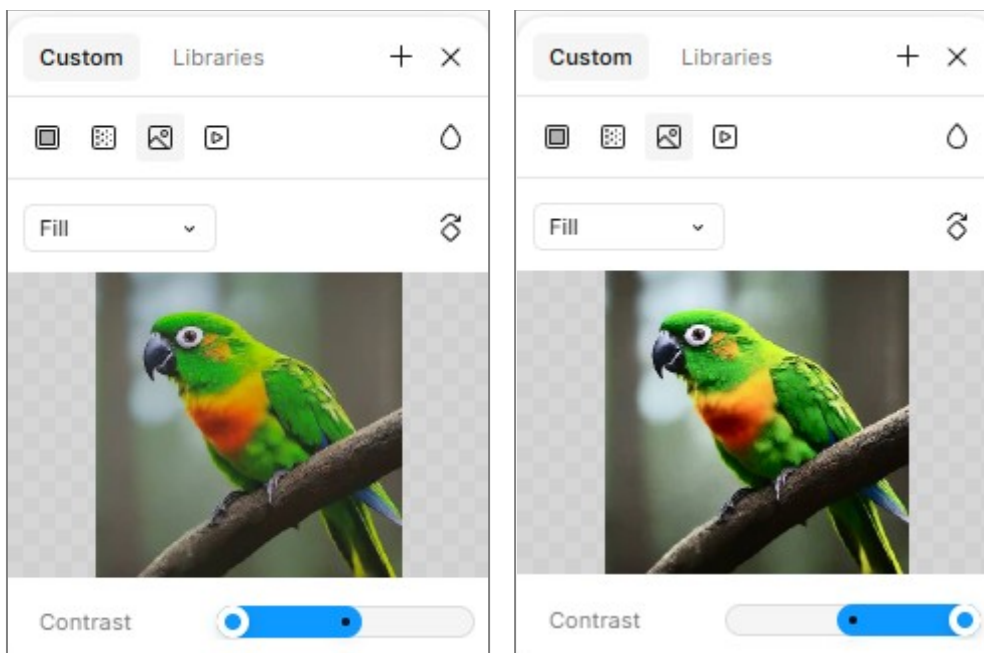


Рис. 153. Застосування зміни контраста

SATURATION

Saturation (насиченість кольору, відтінків) – дає змогу зробити кольори яскравішими (Рис. 154).

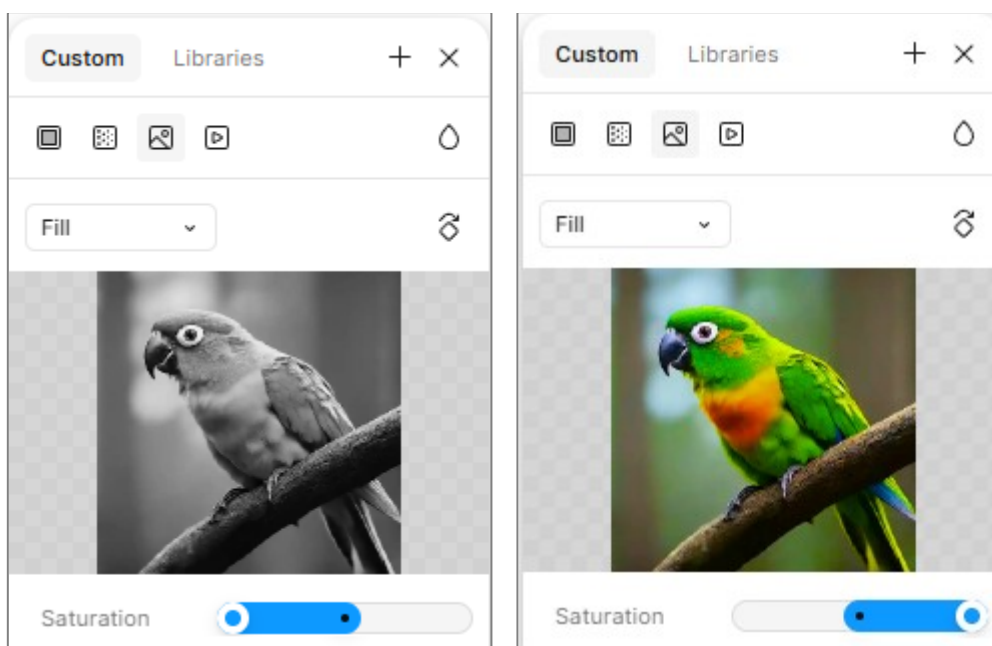


Рис. 154. Приклади зміни насиченості кольорів

Якщо встановити це налаштування на нуль, то у такому разі світлини позбавляться кольору відтінків. Таким чином світлини стануть чорно-білими. Максимальне значення насиченості перетворить кольори на неприродно-неонові.

Щоб чорно-біла світлина виглядала якісно, відвести контраст у нуль недостатньо. На світлині залишиться багато сірого. Компенсувати його можна високим контрастом і глибоким чорним (Рис. 155).

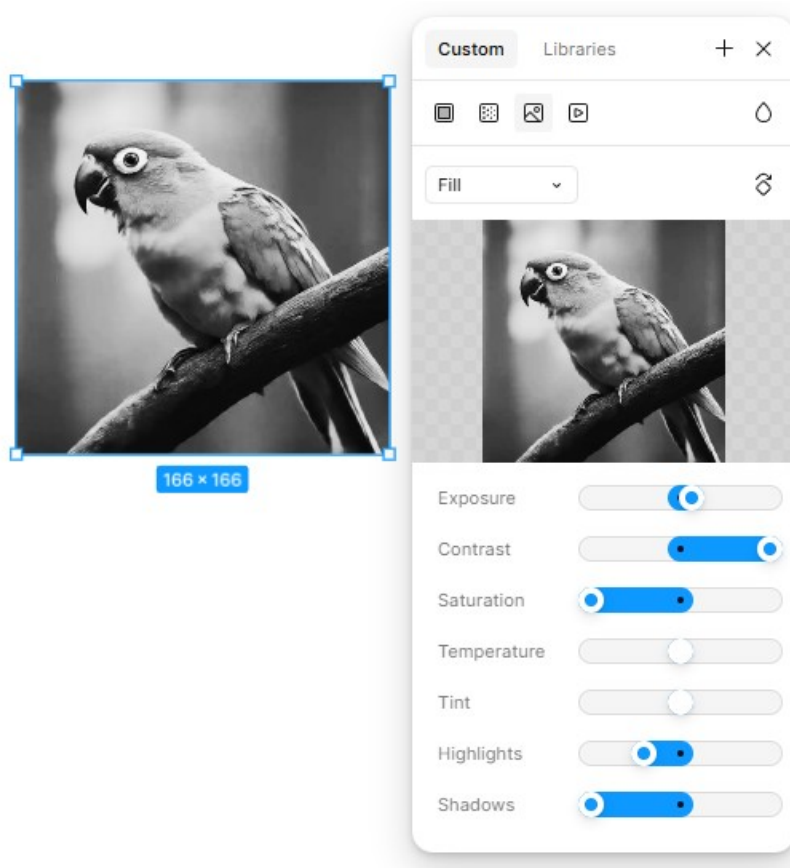


Рис. 155. Компенсування сірого кольору високим контрастом і глибоким чорним

TEMPERATURE

Налаштування **Temperature** (температура світла) дає змогу працювати з балансом білого. Якщо кадр виглядає занадто теплим або холодним, то це налаштування дає змогу відкоригувати його, тобто відновити нейтральне освітлення. Сонячне світло вдень і надвечір має надто теплий відтінок. Для відновлення нейтрального освітлення можна теплий відтінок компенсувати холодним синім. У світлі від ламп люмінесцентного освітлення навпаки немає температури. Такі фото компенсують жовтим (Рис. 156).

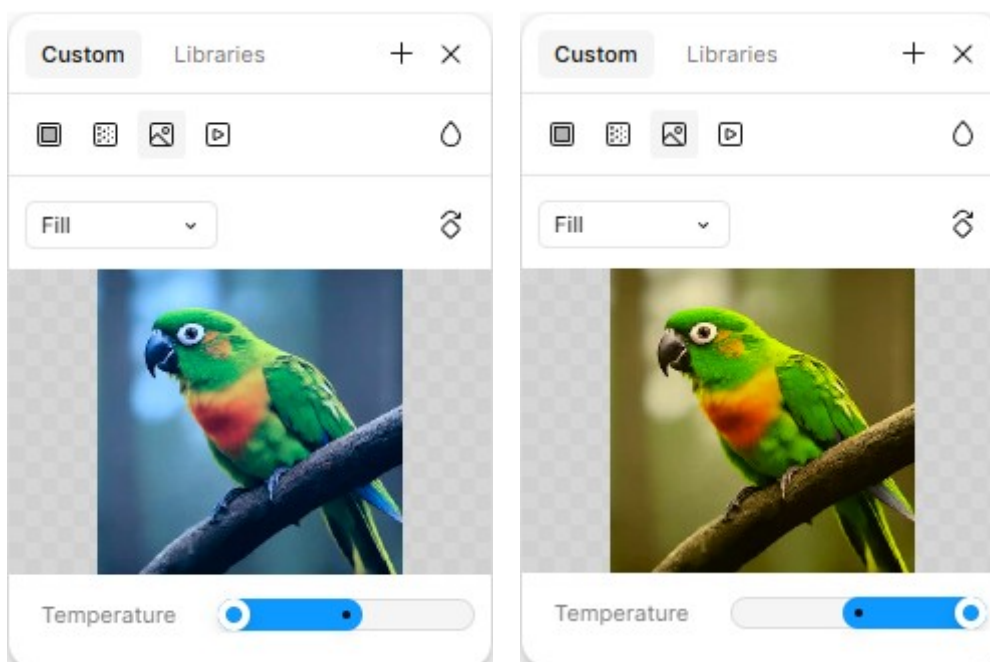


Рис. 156. Приклад компенсування теплого відтінку

TINT

Налаштування **Tint** (відтінок світла) дасть змогу досягти ефекту вицвілого фото. Фотографи використовують її, щоб приглушувати зайве почервоніння обличчя, відводячи відтінок трохи вниз.

Цей інструмент корисний також для компенсації балансу білого у складних умовах зйомки, коли не вистачає нормальної корекції температури. Наприклад, коли на об'єкт зйомки впливають сильно освітлені кольорові стіни (Рис. 157).

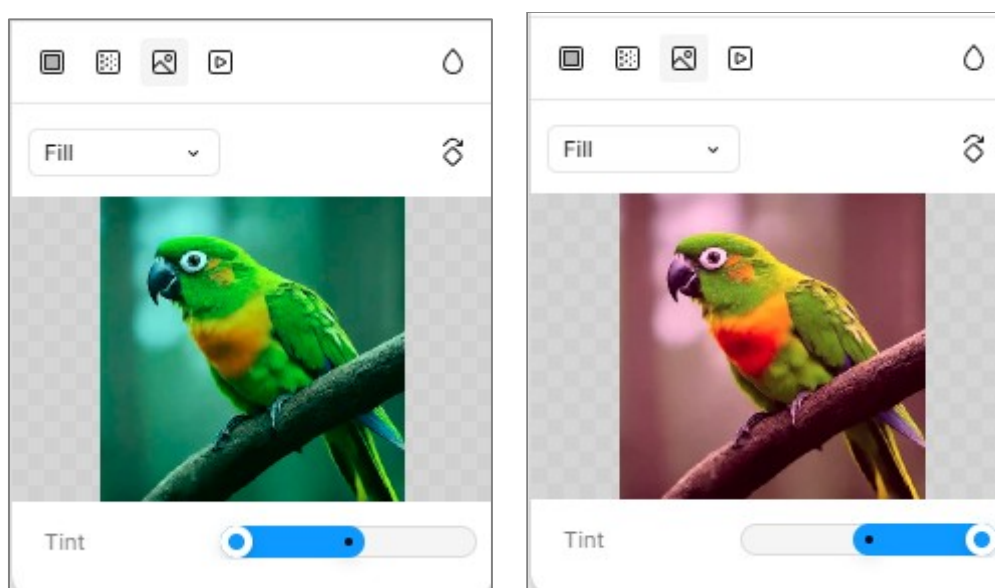


Рис. 157. Приклад використання інструменту Tint

HIGHLIGHTS

Інструмент **Highlights** (підсвічування) дає змогу додатково налаштувати корекції світлих і «пересвічених» ділянок. Його дія схожа на контраст. Різниця між ними у тому, що зміни застосовуються переважно до яскравих ділянок. Інструмент допомагає автоматично виділяти важливі елементи у дизайні (Рис. 158).

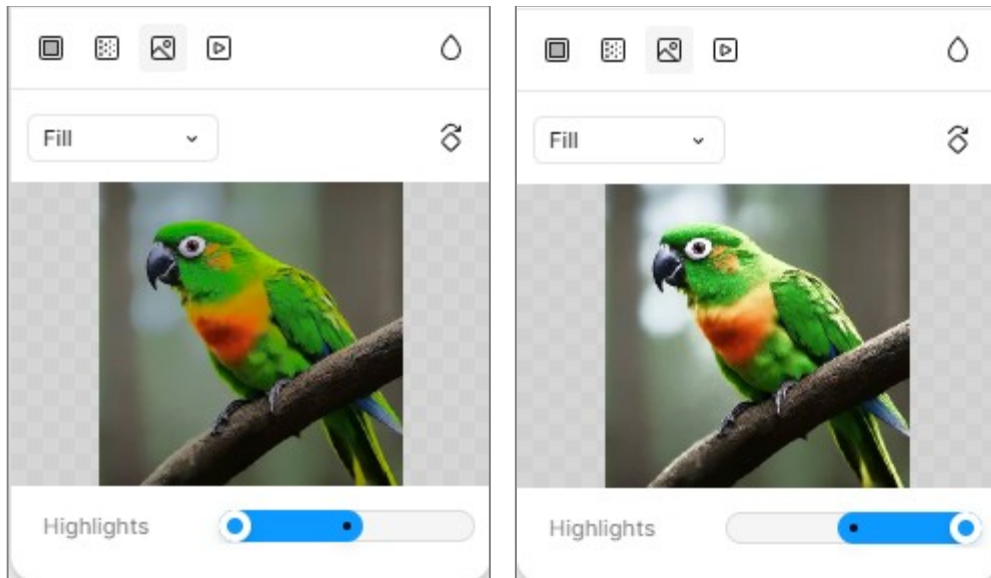


Рис. 158. Приклад використання інструменту Highlights

SHADOWS

Інструмент **Shadows** (Тінь) є аналогом інструменту **Highlights**, але використовується лише для темних ділянок світлин. Він дає змогу додавати тіні до об'єктів, що створює ефект об'ємності та глибини (Рис. 159).

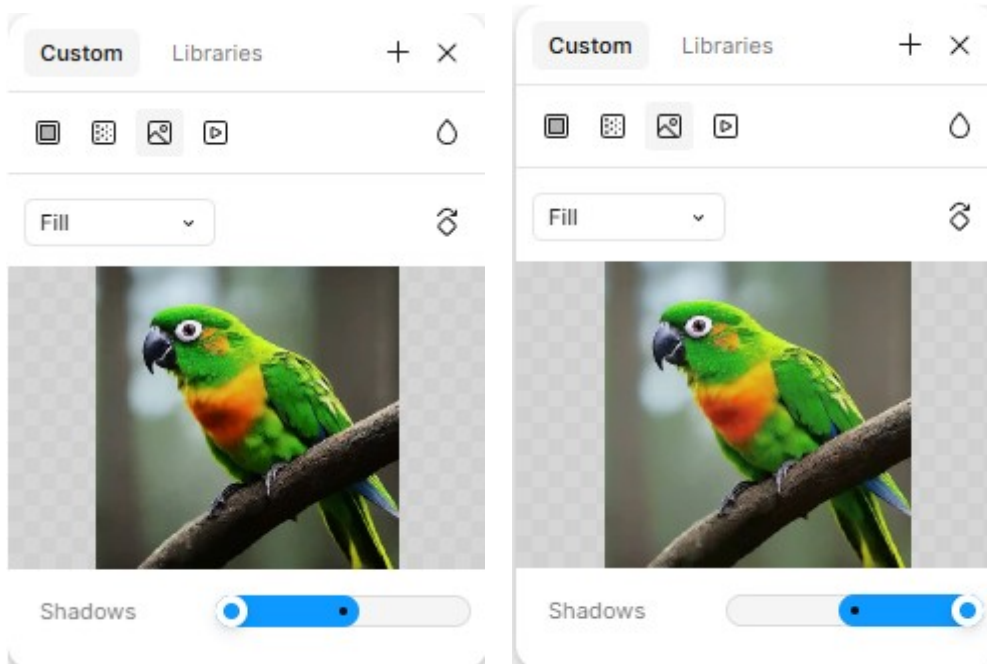


Рис. 159. Приклад використання інструменту Shadows

МЕЖІ

У Figma **Borders** (Межі) – це лінії, що окреслюють контури фігури чи об'єкта. Вони можуть бути використані для підкреслення, створення рамок або для візуального розділення елементів дизайну.

Щоб додати межі до фігури або об'єкта, необхідно у панелі властивостей праворуч знайти секцію **Stroke** та натиснути на кнопку «+». Якщо фігура замкнена, панель властивостей матиме вигляд, як показано на Рис. 160.

Якщо фігура не замкнена, з'являться випадаючі меню **Start point** і **End point**, що задають форму початкової та кінцевої точки контуру (Рис. 161).

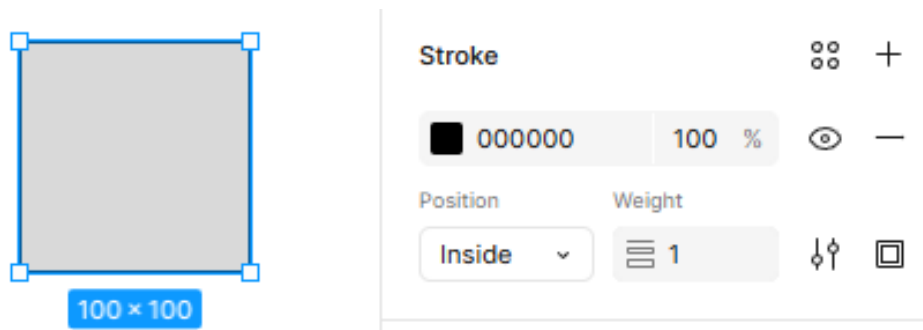


Рис. 160. Вигляд панелі властивостей у випадку замкненої фігури

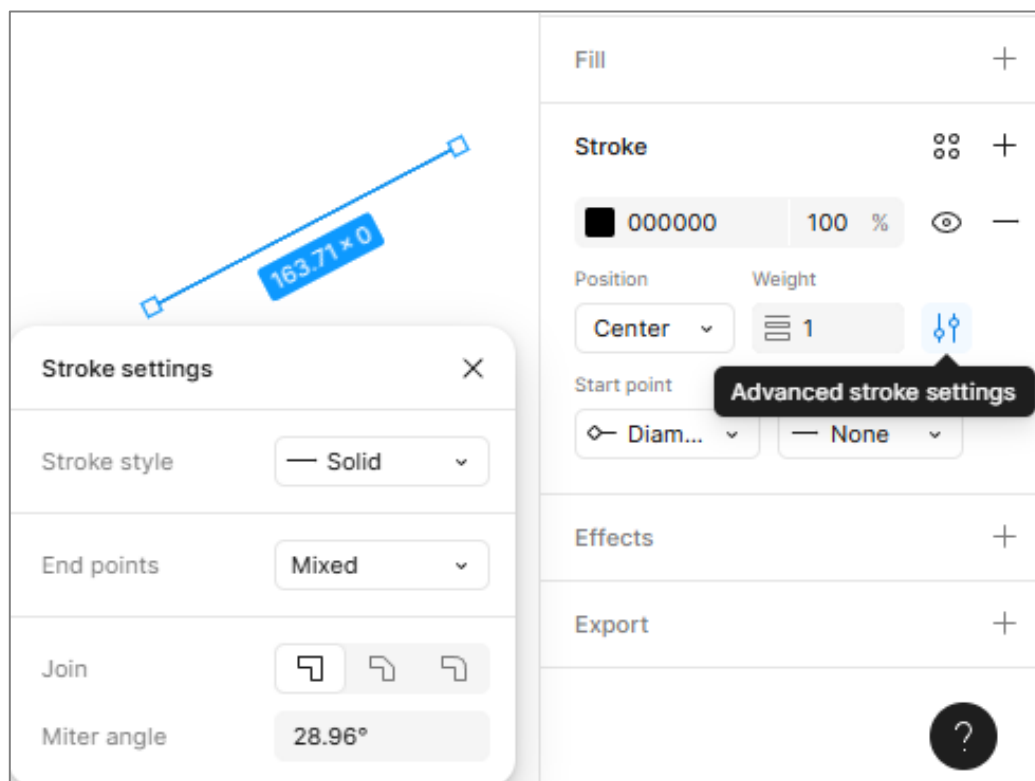


Рис. 161. Вигляд панелі властивостей у випадку розімкненої фігури

Натиснувши на іконку **Advanced stroke settings**, можна отримати доступ до таких налаштувань:

Stroke style – стиль обведення:

- **Solid** – суцільна лінія;
- **Dashed** – пунктирна лінія. Під час вибору пунктирного стилю додаються додаткові параметри: **Dash** – довжина пунктиру, **Gap** – довжина пропуску, **Dash cap** – форма пунктиру.

Join – форма з'єднання точок:

- **Miter** – автоматично скошується, якщо кут нахилу менше заданого полем **Miter angle**;
- **Bevel** – постійно скошена;
- **Round** – заокруглена.

Розглянемо додаткові налаштування обведення примітивів, векторних ліній і текстових шарів. Для демонстрації різних режимів створимо квадрат, **R** та задамо йому обведення червоного кольору завтовшки 10 пікселів.

РЕЖИМИ ОБВЕДЕННЯ

Обведення може бути накладено на фігуру трьома способами:

- **Center** – центр обведення збігається з межами примітива (Рис. 162);

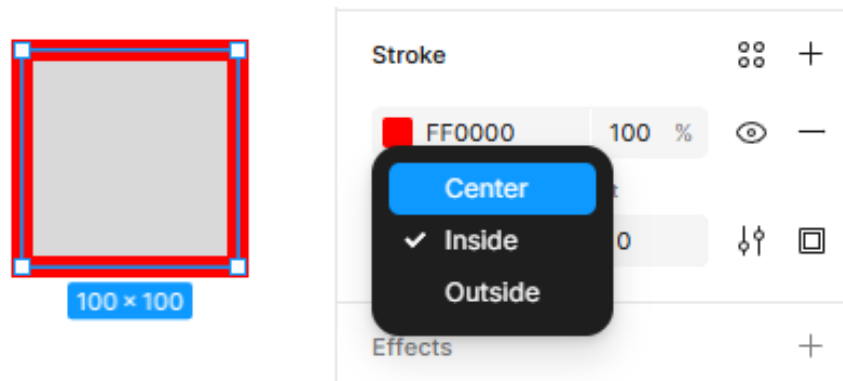


Рис. 162. Приклад застосування обведення Center

- **Inside** – усередині. Обведення не виходить за межі примітива (Рис. 163).

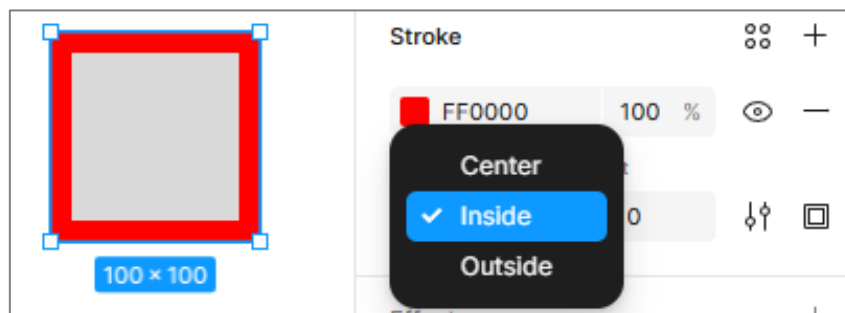


Рис. 163. Приклад застосування обведення Inside

- **Outside** – зовні. Обведення виходить за межі примітива (Рис. 164).

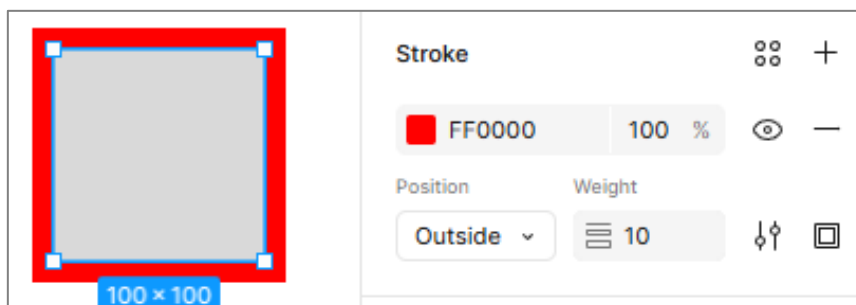


Рис. 164. Приклад застосування обведення Outside

Якщо поставити режим обведення з **Inside** в **Center**, а прозорість зробити рівною 50%, буде явно помітно, де розміщується центр контуру фігури (Рис. 165).

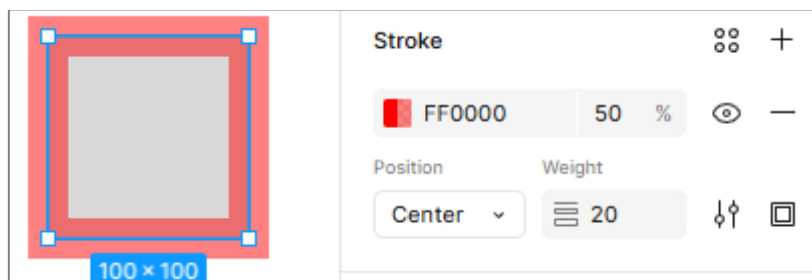


Рис. 165. Визначення центру контуру фігури

ПУНКТИРНЕ ОБВЕДЕННЯ

Для того, щоб створити пунктирне обведення, необхідно:

1. У правій частині розділу **Stroke** натиснути на кнопку **Advanced stroke settings** (два блакитні повзунки) (Рис. 166).
2. У вікні, що відкриється в полі **Stroke style**, вибрати пункт **Dashes**.
3. Задати параметри пунктирної лінії.

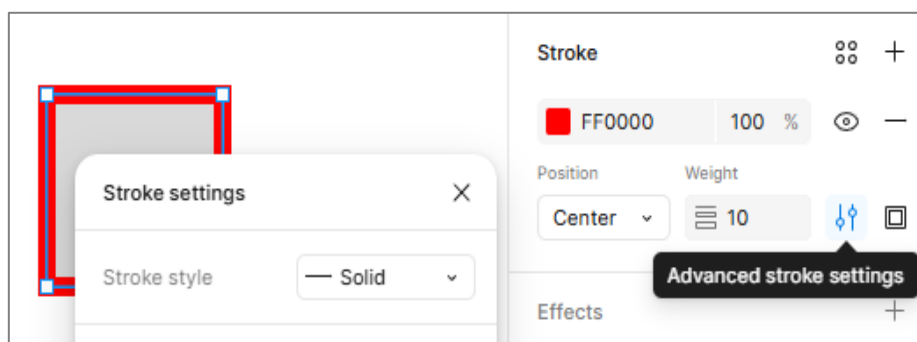


Рис. 166. Створення пунктирного обведення

ДОВЖИНА ПУНКТИРІВ

Figma підлаштовує пунктир так, щоб він був симетричним. Червона лінія обов'язково попадає на кути фігури. Водночас Figma нехтує математично-точними значеннями поля **Dash** – довжина пунктиру в пікселях (користувач задає приблизне значення, а Figma підганяє його так, щоб пунктир виглядав найестетичніше). Значення **Gap** (розрив) – відступ у пікселях між пунктирами також установлюється автоматично (Рис. 167).

Під час установлення значення у полі **Stroke style** на **Custom** задається кастомний стиль межі. Наприклад:

- 10, 5 – штрихи довжиною 10 одиниць з інтервалами 5 одиниць між ними;

– 5, 2, 1, 2 – чергуються довгі та короткі штрихи з різними інтервалами (Рис. 168).

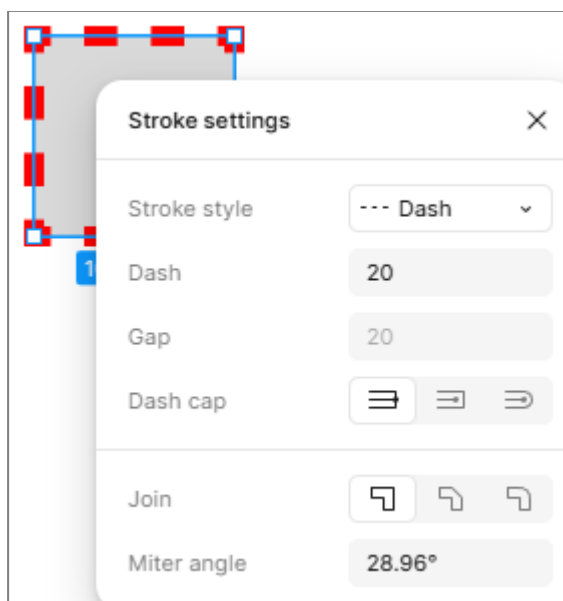


Рис. 167. Застосування обведення пунктирною лінією

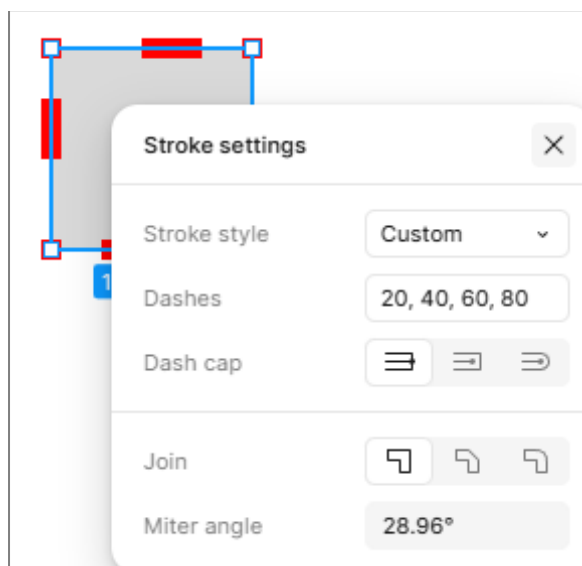


Рис. 168. Застосування обведення пунктирною лінією з різною довжиною штрихів

СТИЛІ ВИГИНІВ ОБВЕДЕННЯ

Обведення може мати три стилі вигинів, які перемикаються у сегменті **Join**:

– **Miter** (Звичайний стик) – прямий кут, за замовчуванням (Рис. 169);



Рис. 169. Застосування стилю обведення Miter

- **Bevel** (Скошений стик) – зрізаний на 45° (Рис. 170);



Рис. 170. Застосування стилю обведення Bevel

- **Round** (Круглий стик) – закруглений (Рис. 171).



Рис. 171. Застосування стилю обведення Round

ТИПИ ЗАКІНЧЕННЯ ЛІНІЇ

Налаштування **Cap** (Наконечник) відповідає за те, як буде оформлено закінчення лінії. Типи закінчень розглянуто у розділі [Примітиви](#). За замовчуванням **Cap** встановлено в **None** і переривається.

До ліній в обведенні можна застосувати тип **Round**, щоб заокруглити пунктир (Рис. 172).



Рис. 172. Застосування типу Round для заокруглення пунктиру

БУЛЕВІ ОПЕРАЦІЇ

У разі необхідності створення складнішої форми, ніж стандартні прямокутники, кола або трикутники, її можна намалювати за допомогою пера. Однак, якщо складну форму можна розбити на декілька базових геометричних фігур, Figma швидше комбінуватиме їх в одну.

Булевими групами у Figma називають принципи взаємодії примітивів один із одним.

Прикладом, в якому перо буде недоречно, може бути малювання пончика: його можна скласти з двох кіл, наклавши біле коло на сіре та колір фону зробити білим (Рис. 173).



Рис. 173. Накладання двох примітивів для створення складної фігури

Однак тут виникає проблема: якщо зміниться колір фону, колір внутрішнього кола залишиться без змін (Рис. 174).



*Рис. 174. У разі зміни кольору фону
колір внутрішнього кола залишається без змін*

Щоб у центрі темно-сірого кола утворився отвір, потрібно відняти від нього форму білого кола. Для цього потрібно об'єднати обидва кола у булеву групу і поставити її в режим віднімання.

Булеве значення може бути у двох режимах: **true** або **false**. Тобто примітив може бути видимим (true) або невидимим (false). Залежно від режиму, примітив відображається певним чином.

Усього є 4 режими булевих груп (Рис. 175):

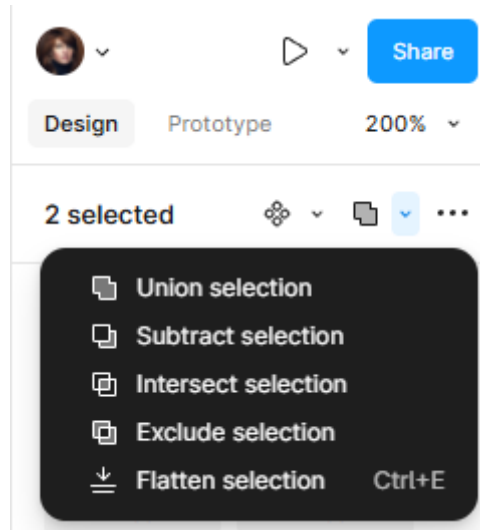


Рис. 175. Режими булевих груп

- **Union** – об'єднання двох фігур в одну. Видимі обидві;
- **Subtract** – віднімання верхньої від нижньої. Видима нижня;
- **Intersect** – видима область перетину двох фігур;
- **Exclude** – видимі обидві фігури, не видима область перетину.

Розгрупувати групу можна комбінацією клавіш **Shift + Ctrl + G**.

SUBTRACT (ВІДНІМАННЯ)

Використаємо наведений приклад із пончиком для демонстрації роботи інструменту **Subtract**. Для створення фігури «пончика» необхідно:

1. Намалювати коло, **O**. За бажанням залити його певним кольором у розділі **Fill**.
2. Продублювати його комбінацією клавіш **Ctrl + D**. Тепер два однакові кола лежать один над одним.
3. Для зменшення верхнього кола відносно центру потрібно затиснути нижній правий маркер і комбінацію клавіш **Shift + Alt** (**Alt** – щоб опорою для масштабування кола була не ліва верхня точка, а геометричний центр; **Shift** – фіксує пропорції кола, не дозволяючи малювати еліпс) і перетягнути маркер до центру кола, щоб задати необхідний розмір (Рис. 176).



Рис. 176. Створення фігури «пончика» (кроки 1–3)

4. Виділити обидва примітиви, послідовно клікнувши мишкою по кожному колу при затисненій клавіші **Shift**.
5. На Панелі властивостей натиснути кнопку **Union selection** (Рис. 177) і виконати команду **Subtract Selection**.

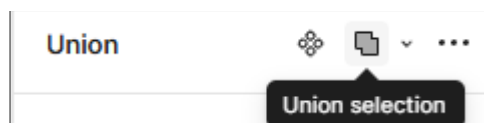


Рис. 177. Кнопка Union selection

У результаті в лівій панелі з'являється булева група з назвою **Subtract**, у якій з'єднані обидва кола. Водночас верхнє коло віднімається від нижнього (Рис. 178).



Рис. 178. Застосування булевої операції Subtract Selection

Під час віднімання нижній шар є базовою фігурою, з якої відбувається віднімання. Якщо дублювати фігуру, що віднімається, і зрушити її в сторону, вона збільшить площу віднімання (Рис. 179).

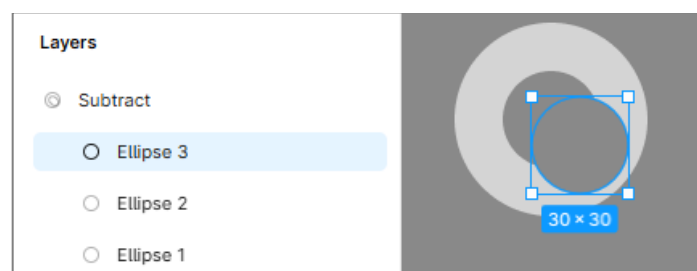


Рис. 179. Збільшення площі віднімання у булевій фігурі

Із Рис. 179 видно, що, якщо у групі стоїть режим **Subtract**, усі верхні фігури всередині цієї групи (Ellipse 2 та Ellipse 3) віднімаються з нижньої (Ellipse 1).

Оскільки шар 3 дубльований із шару 2, він успадковує режим **Subtract**. У правій частині списку відображується його іконка.

UNION (ДОДАВАННЯ)

Як і у попередньому підрозділі **Subtract** продемонструємо використання інструменту **Union** на прикладі. Для цього створимо булеву фігуру – «хмаринку».

Для створення такої фігури необхідно:

1. Намалювати три еліпса, **O**, щоб отримати щось подібне до «хмаринки».
2. Виділити їх усіх з використанням клавіші **Shift** (Рис. 180)

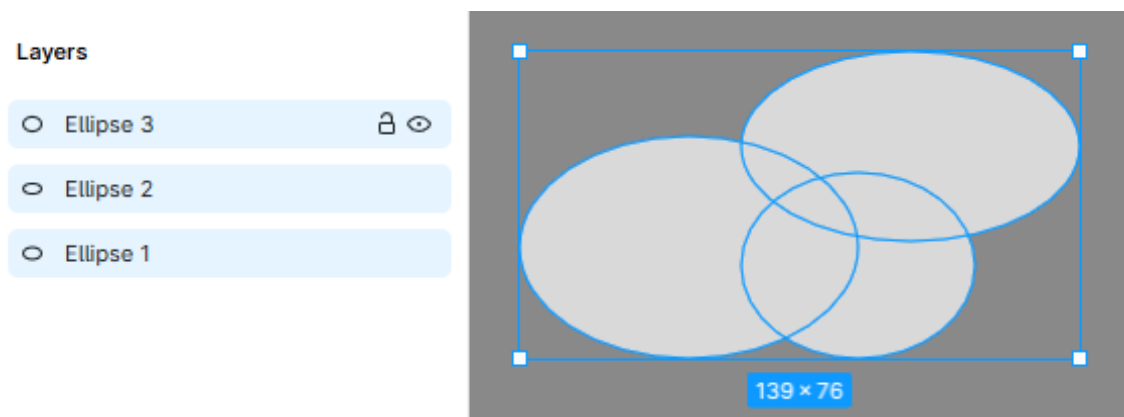


Рис. 180. Виділення кіл для створення булевої фігури

3. Застосувати команду **Union Selection** (Рис. 181).

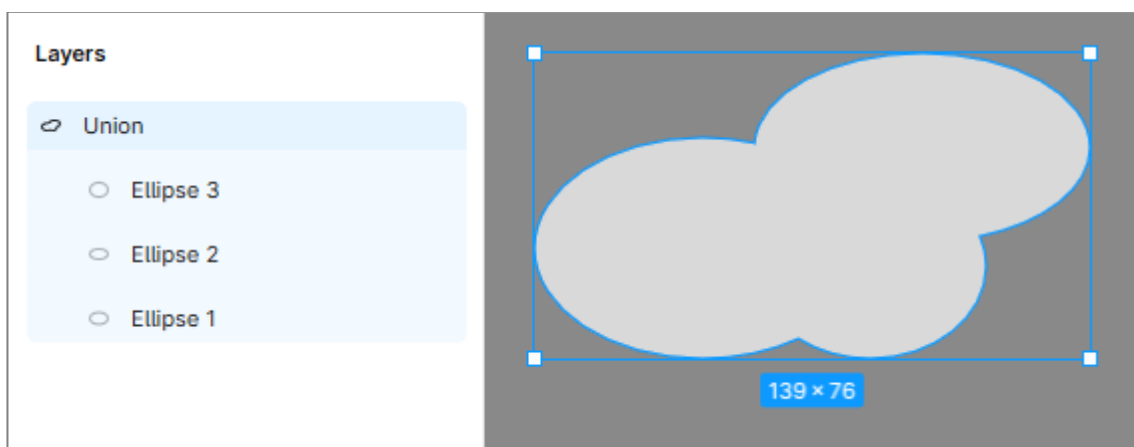


Рис. 181. Застосування булевої операції Union Selection

INTERSECT (ПЕРЕТИН)

Застосування інструменту **Intersect** під час створення булевої фігури зручно продемонструвати на прикладі малювання «ока» за допомогою перетину двох кіл. Для цього:

1. Намалюємо коло, **O**.
2. Продублюємо його, затиснувши клавішу **Alt** та протягнувши його до низу (Рис. 182).

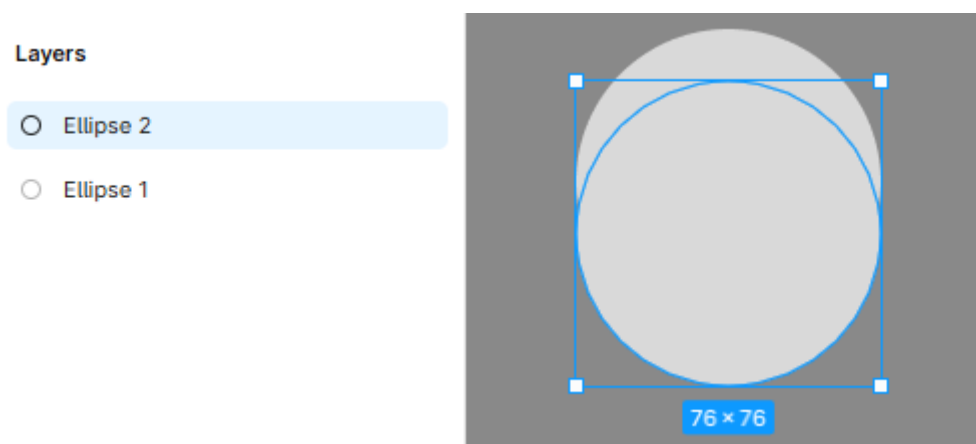


Рис. 182. Створення булевої фігури «око» (крок 1–2)

4. Виділимо обидва кола і застосуємо інструмент **Intersect Selection** (Рис. 183).

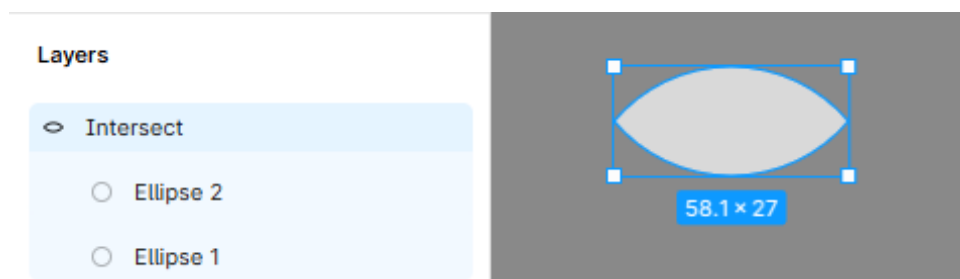


Рис. 183. Застосування булевої операції

Intersect Selection

EXCLUDE (ВИКЛЮЧЕННЯ)

Режим **Exclude** застосовується для приховування спільних фрагментів у фігурах, що перетинаються. Для прикладу намалюємо таку фігуру:

1. Намалюємо коло, **O**.
2. Затиснувши клавішу **Alt**, витягуємо його копію так, щоб кола певним чином перекривались.

3. Виділяємо обидва кола, перемикаємо їх у режим **Exclude Selection** (Рис. 184).

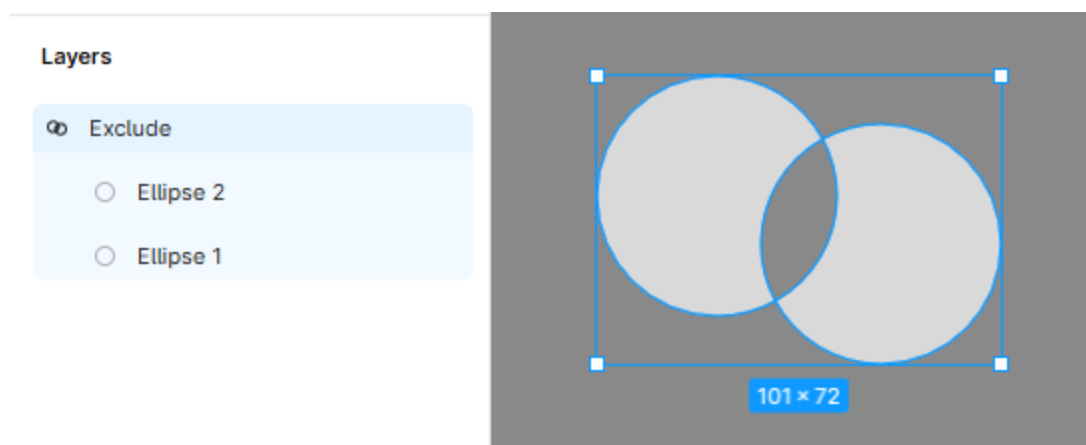


Рис. 184. Застосування булевої операції Exclude Selection

Exclude – інструмент імпровізаторів. Цей режим дає змогу досягати цікавих і непередбачуваних результатів. Якщо додати до початкової фігури з двох кіл ще два і застосувати інструмент **Exclude**, вийде складніший геометричний візерунок (Рис. 185).



*Рис. 185. Приклад застосування булевої операції Exclude Selection
для створення складної фігури*

Намалювати таку геометричну фігуру за кілька хвилин без інструменту **Exclude** було б неможливо.

Булеві режими працюють також і з текстом. Два шари у режимі **Exclude** та з невеликим зміщенням можуть давати, наприклад, ефект, який продемонстровано на Рис. 186.



*Рис. 186. Приклад застосування інструменту **Exclude Selection** для роботи з текстом*

ВКЛАДЕНІ БУЛЕВІ ГРУПИ

Вкладені булеві групи – це групи, що містять інші булеві групи, створюючи водночас складніші форми.

Наприклад, примітиви 1 і 2 об'єднані у булеву групу **Subtract** (Віднімання). Після цього примітив і група **Subtract** об'єднуються у булеву групу **Union** (Додавання). У результаті отримаємо «напівмісяць» у колі (Рис. 187).

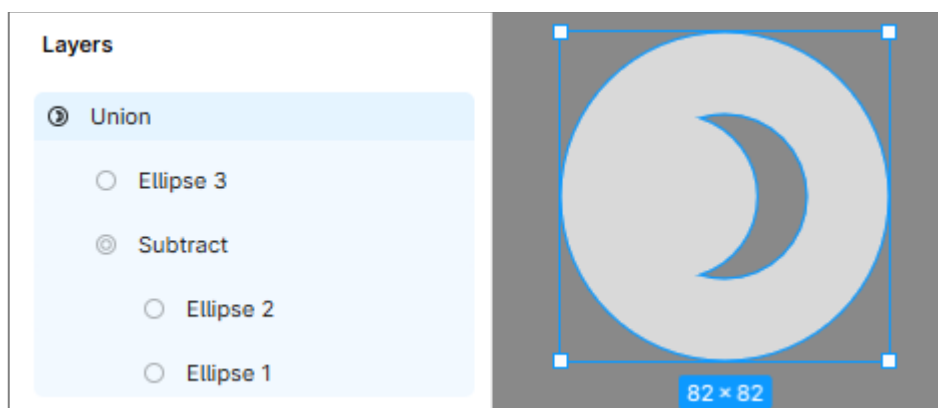


Рис. 187. Приклад вкладки булевої групи

FLATTEN (ОБ'ЄДНАННЯ) СКЛАДНИХ ФІГУР

Фігури, які створюються за допомогою булевих груп, складаються з простих об'єктів. Однак буває необхідність з'єднати багат шарову форму в одношарову, коли не має потреби звертати увагу на структуру фігури, а потрібно працювати з нею як з єдиним цілим. Для цього у Figma використовується функція **Flatten Selection**, **Ctrl + E**.

Для того, щоб зробити об'єднання виконуємо такі операції:

1. Виділяємо булеву групу (звертаємо увагу на трикутник ліворуч, який її розгортає (Рис. 188)).

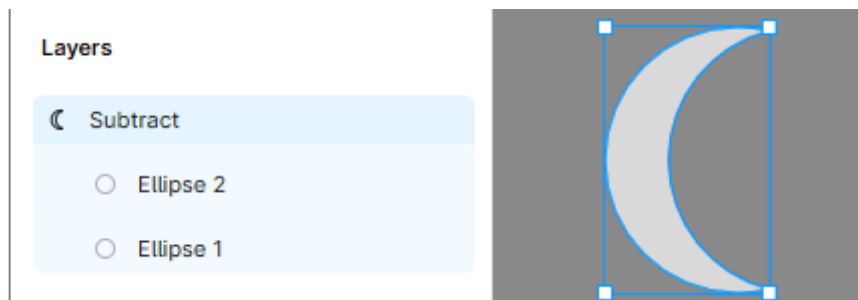


Рис. 188. Виділення булевої групи у процесі об'єднання складних фігур

2. Натискаємо **Flatten**, **Ctrl + E**. Розкриваючий трикутник ліворуч від назви шару зник. Група перетворилася на примітив у формі півмісяця (Рис. 189).

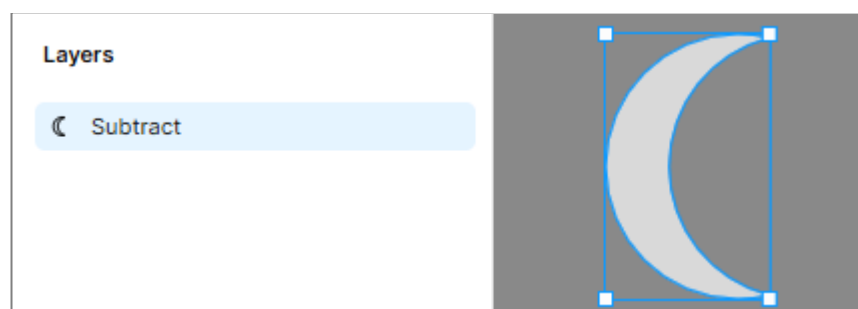


Рис. 189. Застосування інструменту Flatten для об'єднання складних фігур

3. Для виходу з режиму натискаємо на клавішу **Enter**. Тепер його можна змінювати лише у режимі редагування.

Якщо виконувати операцію **Flatten**, бажано робити копію вихідної форми, щоб завжди мати можливість розібрати її на складові, а не редагувати плоску фігуру в кривих.

ЕКСПОРТ У SVG

Для експорту іконки у SVG-файл її необхідно попередньо підготувати. Для цього потрібно:

1. Створити порожній фрейм і вставити у середину об'єкт (їх може бути декілька). Розміри фреймів для іконок зазвичай кратні 8 пікселям: 16×16, 24×24, 32×32 тощо. Якщо іконка не поміщається у створеному фреймі, необхідно її зменшити.

2. Вирівняти іконку по центру фрейму. Переконалися, що вертикальні та горизонтальні відступи рівномірні.
3. Перевести всі межі у криві.
4. Перефарбувати всі елементи у чорний колір.

На цьому процес попередньої підготовки завершено. Залишається виконати експорт.

Для цього потрібно:

1. Виділити фрейм з іконкою.
5. У правій панелі клацнути на плюську, навпроти секції **Export**.
6. У меню, що з'явиться, обираємо формат **SVG**.

РЕЖИМ OUTLINE: ВЕКТОРНІ КОНТУРИ

Режим **View** → **Outline**, **Ctrl + Shift + O** у Figma дозволяє відобразити ієрархію всіх елементів і шарів проєкту у виді списку. В цьому режимі відображаються лише векторні контури шарів без урахування обведення та заливок (Рис. 190). Цей режим корисний для організації, переміщення та редагування елементів, насамперед якщо користувач працює з великими або складними файлами.

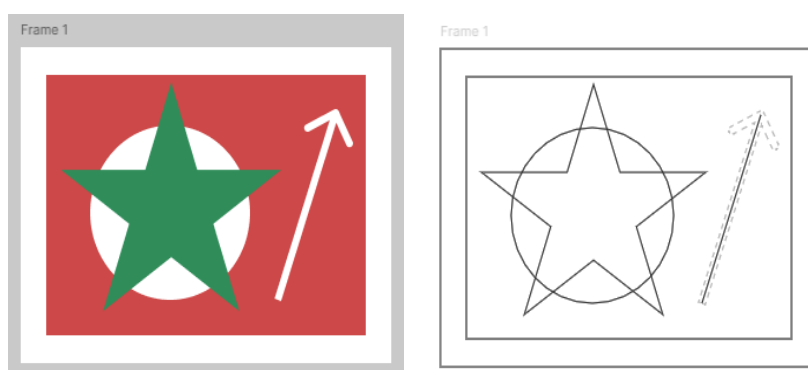


Рис. 190. Приклад відображення ієрархії шарів

Фрейми відображаються за допомогою ліній. У примітивів помітні заокруглення. Наконечники стрілок відображаються пунктиром. У текстових шарах видно примітиви букв, але не видно бокси.

Режим **Outline** також зручно використовувати, коли потрібно знайти невидиму фігуру, яка не має ні обведення, ні заливки, але залишилася її векторна форма.

Для виділення складно досяжних шарів у Figma (Рис. 191) (наприклад, для виділення нижнього квадрата, який перекривається верхнім) потрібно:

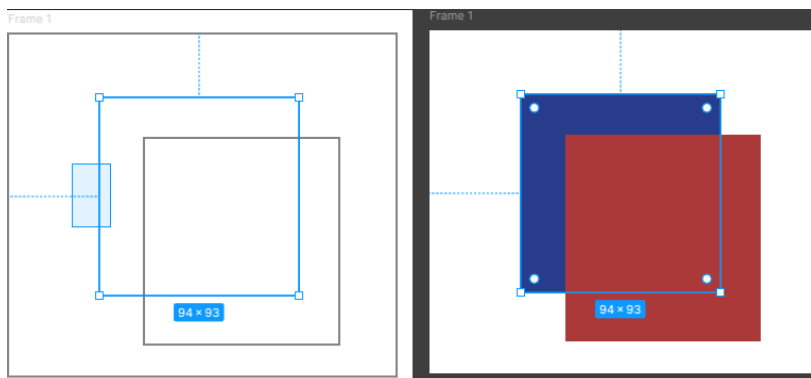


Рис. 191. Виділення складно досяжних шарів у Figma

1. Перемикнутися у режим **Outline**, **Ctrl + Shift + O**.
7. Затиснути клавішу **Alt** і розтягнути прямокутник, який застосовується для виділення, лівою кнопкою мишки так, щоб у нього потрапив контур лише нижньої фігури. Водночас потрібно не зачіпати верхню (див. рис. 185).
8. Вийти з **Outline**, **Ctrl + Shift + O**. Нижній квадрат залишиться виділеним.

МАСКИ

Маски дають змогу приховувати або відображати певні частини об'єктів (зображень, фігур, тексту), використовуючи інший об'єкт як форму. Все, що є всередині цієї форми, буде видно, а решта – приховано. Це один із ключових інструментів для роботи з дизайном у Figma, насамперед для створення складних композицій і графічних ефектів.

КАДРУВАННЯ ФОТО

Оскільки у Figma наявний дуже розвинутий режим кадрування заливки **Crop**, немає сенсу використовувати маски для кадрування аватарів та ілюстрацій. Однак для розуміння принципу їхньої роботи створимо найпростішу маску.

1. Вставимо у робочу область світлину.

Поверх неї розтягнемо коло, **O**. Воно має перекривати те місце у світлині, яке користувач збирається залишити видимим. Для наочності можна знизити прозорість кола до 50% (Рис. 192).

2. Виділимо коло. Поміняємо його місцями із світлиною, **Ctrl + [**. Коло має опинитися знизу.
3. Затискаємо клавішу **Shift** і виділяємо обидва шари.
4. Викликаємо команду **Use as Mask**, **Ctrl + Alt + M** або виконуємо відповідну команду на Панелі інструментів (Рис. 193).

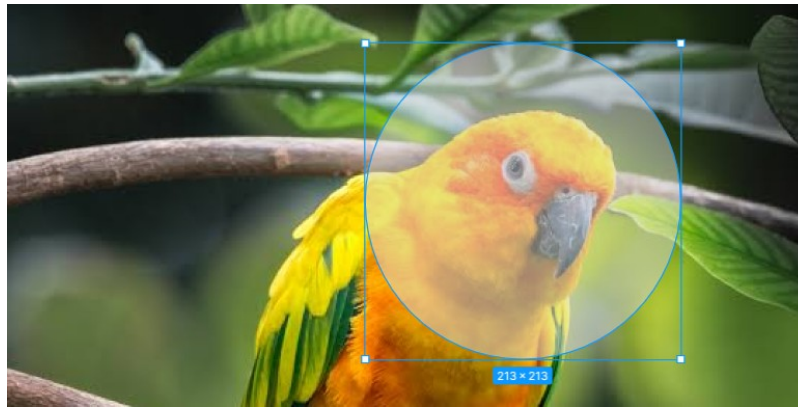


Рис. 192. Встановлення меж кадрювання

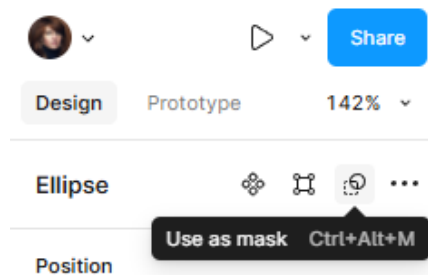


Рис. 193. Застосування команди *Use as Mask* з Панелі інструментів

Так створиться нова група із назвою **Mask Group**. Шар із колом тепер знаходиться у режимі маски (Рис. 194).



Рис. 194. Результат кадрювання світлин у Figma

Доки шар світлини є над маскою, він обтинатиметься за її формою. Оскільки у кола була задана прозорість 50%, світлина теж її успадкує. Тому встановлюємо для кола прозорість на максимум – **100%**.

У Figma є тогл-режим **View** → **Mask Outlines**, **Ctrl + /**. Цей режим показує межі масок зеленим обведенням (Рис. 195).

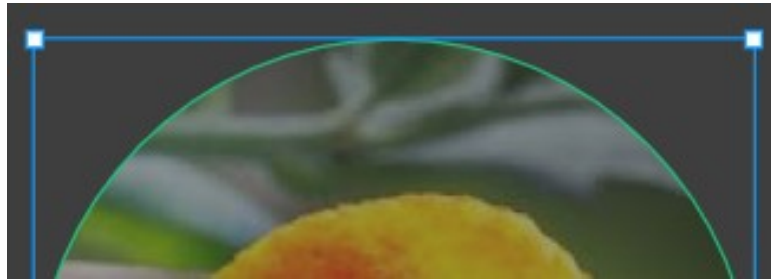


Рис. 195. Застосування тогл-режиму Mask Outlines

АЛЬФА-МАСКА

У Figma **альфа-маска** – це засіб маскування елементів за допомогою рівня прозорості (альфа-каналу) іншого об’єкта. Цей метод дає змогу створювати маски з плавними переходами та градієнтами.

Такий прийом використовується у дизайні, якщо потрібно показати, що праворуч або ліворуч є контент і його можна перетягувати вправо-вліво.

Для того, щоб створити альфа-маску в Figma потрібно:

1. Створити об’єкт, який маскується. Це об’єкт, який користувач хоче замаскувати (наприклад, зображення або фігура).
2. Створити звичайну маску за інструкцією вище.
3. Створити об’єкт маски. Це може бути будь-який об’єкт (наприклад, прямокутник або еліпс) із градієнтом або зміненою прозорістю.

Колір градієнта не важливий, тому що примітив із градієнтом використовуватиметься як маска. У цьому випадку Figma буде використовувати його градієнт для роботи з прозорістю контенту. Для наочності виберемо червоний.

4. Кінцеву контрольну точку градієнта відводимо в нуль.
5. Помітно, як права частина світлини плавно зникає (Рис. 196).

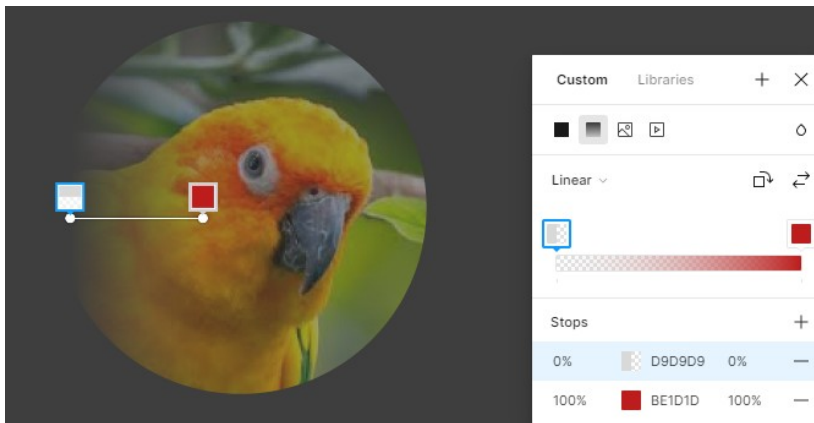


Рис. 196. Застосування альфа-маски

ПЛАГІНИ

Figma плагіни – це додаткові інструменти або розширення, які допомагають поліпшити і пришвидшити роботу в Figma, додаючи нові функціональні можливості або автоматизуючи процеси. Ось кілька популярних плагінів для Figma:

- **Unsplash** – дає змогу додавати безкоштовні зображення з Unsplash безпосередньо у ваші проекти Figma.
- **Iconify** – збірка різноманітних іконок із популярних бібліотек, таких як Material Design Icons та FontAwesome.
- **Blush** – плагін для створення ілюстрацій із колекцій художників та адаптації їх під власні проекти.
- **Stark** – інструмент для тестування контрасту кольорів, що допомагає зробити ваші дизайни доступними для людей із порушеннями зору.
- **Figmotion** – анімаційний плагін, що дає змогу створювати анімації у Figma без потреби експортувати в інші програми.
- **Content Reel** – плагін для швидкого вставлення тексту, зображень або інших елементів для демонстрації в макетах.
- **Map Maker** – дозволяє створювати карти в Figma, додаючи локалізацію для дизайнів із картографічними елементами.
- **Autoflow** – автоматично малює з'єднувальні лінії між елементами, що зручно для створення діаграм або wireframe.

- **FigJam** – хоча це окремий інструмент від Figma, плагін FigJam дозволяє інтегрувати візуальні нотатки й інші елементи у ваші проєкти.
- **Pexels** – ще один плагін для додавання безкоштовних зображень із Pexels.

СТИЛІ У FIGMA

ТИПИ СТИЛІВ: КОЛІР, ТЕКСТ, ЕФЕКТИ, СІТКИ

Стилі у Figma – це основа системного дизайну. Вони дають змогу працювати не з окремими кольорами чи шрифтами, а з цілими наборами налаштувань, які можна повторно використовувати у різних елементах. Завдяки цьому дизайн стає не лише візуально цілісним, а й технічно впорядкованим.

Figma підтримує чотири основні типи стилів: **кольори**, **текст**, **ефекти** та **сітки**. Кожен із них зберігає певний набір параметрів, який можна застосувати до будь-якого об'єкта в кілька кліків.

Кольорові стилі використовують для створення палітр – наприклад, основних кольорів бренду або системних відтінків інтерфейсу. Вони можуть застосовуватись до заливки, обведення, тіней чи іконок. Коли колір оформлений як стиль (наприклад, **Primary / Blue 500** або **Text / Secondary**), його можна швидко змінити у всьому макеті одним натисканням. Це корисно для великих проєктів або командної роботи, де важливо, щоб всі користувалися однаковими кольорами з брендбуку.

Текстові стилі визначають основні параметри типографіки – шрифт, розмір, міжрядковий інтервал, трекінг і вагу. Зазвичай у проєктах створюють систему назв на кшталт:

- Heading / H1 (32px, Bold);
- Body / Regular (16px);
- Caption / Small (13px).

Це допомагає витримувати логіку візуальної ієрархії: великі заголовки привертають увагу, а дрібні тексти виглядають легше. Якщо потрібно оновити типографіку – наприклад, замінити шрифт або збільшити міжрядковий інтервал – достатньо внести зміни лише в один стиль.

Стилі ефектів зберігають налаштування тіней, розмиття або світіння. Це дає змогу створити єдину візуальну глибину для карток, кнопок чи зображень. Наприклад, стиль **Shadow / Soft 8px** може бути застосований до всіх елементів, які мають однакову тінь. Якщо потрібно трохи змінити прозорість або напрямок світла, зміни одразу відобразяться в усьому проєкті. Водночас зберігається послідовність і візуальна чистота інтерфейсу.

Стилі сітки допомагають структурувати макет і підтримувати однакові відступи між блоками. Figma дозволяє створювати сітки кількох типів: **Grid** (квадратна), **Columns** (стовпці) і **Rows** (ряди). У вебдизайні найчастіше застосовують **12-колонну сітку** з відстанями 8 або 16 пікселів. Вона допомагає будувати адаптивні сторінки, де всі елементи чітко вирівняні по вертикалі та горизонталі. Збереження сітки як стилю дає змогу використовувати її на всіх екранах – для desktop, tablet чи mobile – без повторного налаштування.

СТВОРЕННЯ ГЛОБАЛЬНИХ СТИЛІВ

Глобальні стилі – це спільні налаштування, доступні у всіх файлах проєкту або командної бібліотеки. Вони допомагають підтримувати єдиний стиль і спрощують редизайн: змінюючи параметр у бібліотеці, ви автоматично оновлюєте його у всіх пов'язаних елементах.

Щоб створити глобальний стиль потрібно:

1. Створити елемент, наприклад, коло із заливкою або текст та виділити його. Цей елемент буде пробником, для якого підбиратимуться параметри.
2. На панелі **Properties** у правій частині екрана натиснути чотири крапки (**Apply styles and variables**) поруч із потрібною властивістю (Рис. 197).

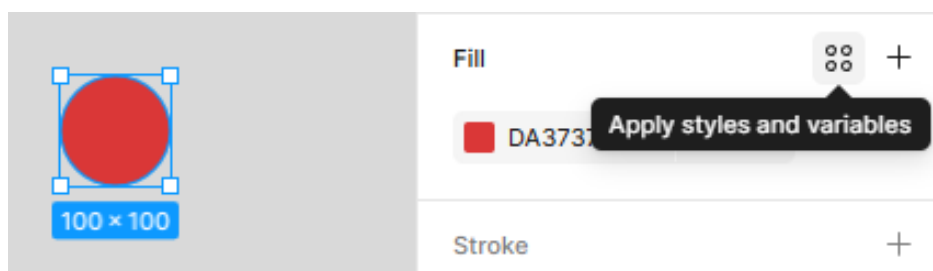


Рис. 197. Кнопка із чотирма крапками відкриває вікно для роботи зі стилями

3. Натиснути кнопку +, **New style** (Рис. 198).

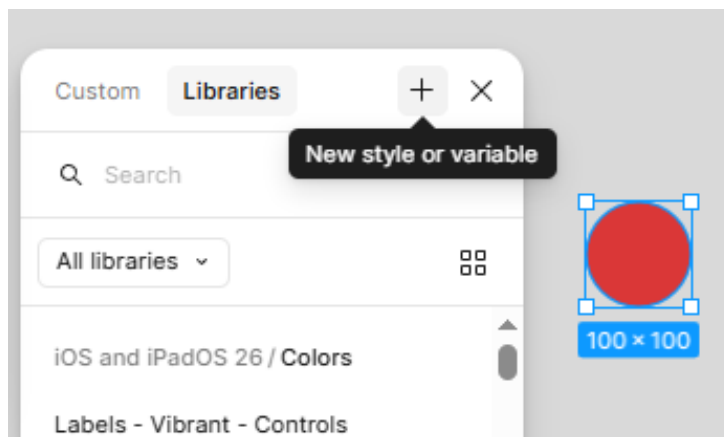


Рис. 198. Створення нового стилю

4. У вікні, що відкриється дати зрозумілу назву стилю, бажано латинцею і без пробілів (замість них можна використовувати дефіс або camelCase). Назви мають бути короткими, логічними та структурованими – це допомагає команді швидко орієнтуватися в бібліотеці, а також щоб надалі розробники могли так називати змінні, наприклад:

- Primary / Blue 500 для кольору;
- Text / H1 / 32 Bold для заголовку;
- Shadow / Soft для ефекту тіні.

Для копіювання стилю з одного об'єкта на інший використовуються команди **Copy Properties**, **Alt + Ctrl + C** та **Paste Properties**, **Alt + Ctrl + V**

ВИКОРИСТАННЯ СТИЛІВ У КОМАНДНІЙ РОБОТІ

У спільних проєктах глобальні стилі виконують роль «мови дизайну». Всі члени команди бачать ті самі кольори, розміри шрифтів і відступи. Це забезпечує узгодженість навіть тоді, коли над макетом одночасно працюють кілька дизайнерів.

Такі глобальні стилі можна зберігати не лише у файлі, а й у спільній бібліотеці (**Team Library**). Це дає змогу підключати їх до різних проєктів: наприклад, одна дизайн-система може одночасно використовуватись у сайті, мобільному додатку та дашборді.

Усі стилі, створені в бібліотеці, позначаються спеціальною іконкою та стають доступними іншим користувачам після публікації (**Publish Styles**). Інші учасники

можуть **підключити її (Enable)** до своїх файлів і використовувати у власних макетах.

Для контролю доступу адміністратор може надати право лише на використання, а не редагування стилів. Це запобігає випадковим змінам і гарантує стабільність системи.

ООНОВЛЕННЯ ТА СИНХРОНІЗАЦІЯ СТИЛІВ

Підтримка актуальності бібліотеки – важливий етап у роботі з великими проєктами. Зміна кольору бренду, новий шрифт або редизайн теми не повинні перетворюватися на хаотичний ручний процес.

Figma автоматично пропонує **оновити стилі (Update Styles)**, якщо у бібліотеці з'явилися зміни. У розділі **Assets** → **Library Updates** можна побачити список усіх оновлень і прийняти їх вибірково. Це дає змогу зберігати контроль: дизайнер сам вирішує, коли застосовувати зміни.

Для системної роботи рекомендують призначити **відповідального за дизайн-систему**. Він слідкує за чистотою бібліотеки, видаляє дублікати, стандартизує назви стилів і координує оновлення між членами команди.

КОМПОНЕНТИ У FIGMA

Компонент у Figma – це шаблон елемента інтерфейсу. Вони дозволяють створювати повторювані елементи, які можна багаторазово використовувати без втрати єдності стилю. Кнопки, картки товарів, поля вводу, меню – усе це зручно оформлюється у виді компонентів. Ідея проста: один раз створюєш – потім змінюєш у будь-який момент, і оновлення застосовується до всіх копій одночасно.

Ці копії називаються **інстансами (instances)**. Вони поведуться як «дзеркала» **головного компонента (Master)** – автоматично оновлюються, коли ви змінюєте базовий шаблон. Проте кожен інстанс може мати власні невеликі зміни (наприклад, інший текст або іконку), не втрачаючи зв'язку з оригіналом. Якщо потрібно відновити всі властивості до базових, використовується команда **Reset all overrides**.

СТВОРЕННЯ КОМПОНЕНТА

Щоб створити компонент потрібно:

1. Створити об'єкт, який буде універсальним. Наприклад, кнопку, картку товару, поле введення або навігаційний пункт. Зазвичай це група кількох елементів: прямокутник із заливкою, текст і, можливо, іконка.
2. Виділити всі частин, які повинні входити до компонента (**Shift** + **клік** або обведення мишкою) (Рис. 199).

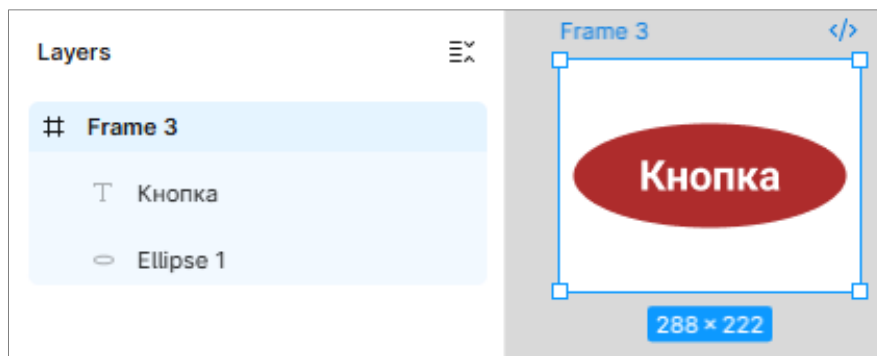


Рис. 199. Звичайне виділення елементів

3. Створіть компонент командою контекстного меню **Create Component**, **Ctrl** + **Alt** + **K**. Навколо елемента з'явиться **фіолетова рамка**, а в панелі шарів – спеціальна іконка ромба , що позначає компонент.
4. Надайте компоненту логічну назву, наприклад Button / Primary або Card / Product. Це допоможе підтримувати порядок у бібліотеці (Рис. 200).

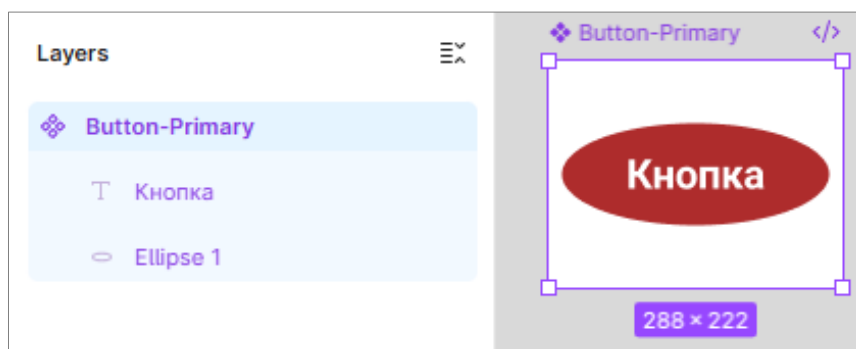


Рис. 200. Компонент Button-Primary

Після створення компонент автоматично з'являється у вкладці **Assets**, де зберігаються всі повторно використовувані елементи вашого файлу (Рис. 201).

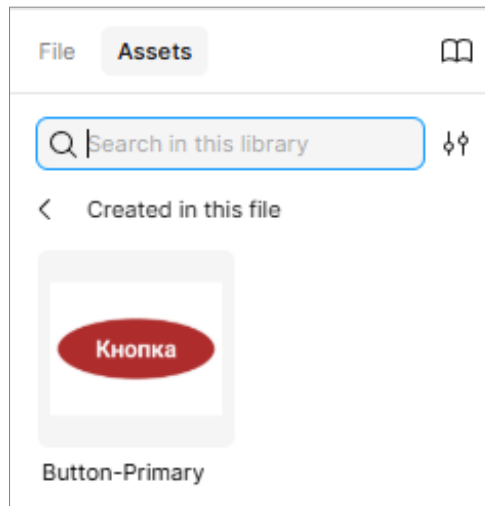


Рис. 201. Відображення компонента в бібліотеці

СТВОРЕННЯ, РЕДАГУВАННЯ І ОНОВЛЕННЯ INSTANCE

Щоб додати копію компонента у дизайн, достатньо просто **перетягнути його з вкладки Assets** у робочу область.

Щоб внести правку лише в конкретній інстансі (наприклад, написати інший текст), просто виберіть її й змініть потрібне поле – зв'язок із головним компонентом залишиться.

Для **повернення інстанс до вихідного вигляду** необхідно скористатися командою контекстного меню **Reset Instance**.

Для повного розірвання зв'язку із головним компонентом, необхідно виконати команду контекстного меню **Detach Instance** – елемент стане звичайною групою і більше не оновлюватиметься автоматично.

КОМПОЗИЦІЯ КОМПОНЕНТІВ

Figma підтримує **композицію компонентів** – це коли всередині одного компонента є інші. Наприклад, кнопка може містити іконку (яка теж є компонентом). Завдяки цьому можна змінювати іконку лише в одному місці, і вона автоматично оновиться у всіх кнопках.

Такий підхід створює «живу структуру» інтерфейсу, де всі частини взаємопов'язані, але водночас легко підтримуються.

Для створення композиції необхідно:

1. Створити базові компоненти. Наприклад, іконку (Icon-Search), кнопку (Button-Primary), текстовий блок або контейнер. Кожен із них потрібно перетворити на **компонент** (**Ctrl + Alt + K**).
2. Вкласти один компонент у інший перетягуванням, наприклад, компонента іконки всередину компонента кнопки. Figma автоматично «впізнає», що це інший компонент, і створить зв'язок між ними.
3. Налаштувати Auto Layout для кнопки, щоб елементи автоматично розташовувалися з правильними відступами між іконкою та текстом. Це дозволить кнопці розширюватися або звужуватися залежно від довжини напису.

Тепер можна використовувати інстанси вкладених компонентів, які містять посилання на головний компонент іконки. При зміні вкладених компонентів усі кнопки, меню чи картки, які його використовують, миттєво оновляться.

СТВОРЕННЯ ВАРІАНТІВ КОМПОНЕНТІВ (VARIANTS)

Figma дає змогу об'єднувати кілька станів одного компонента в єдиний набір варіантів. Це значно спрощує роботу з інтерактивними елементами: дизайнер може перемикає стани за допомогою панелі властивостей, не створюючи окремі копії для кожного випадку. Водночас усі варіанти мають спільне ім'я, але різні властивості.

Наприклад, кнопка може мати кілька станів:

- Default – звичайний вигляд;
- Hover – коли курсор наведено;
- Pressed – натиснута;
- Disabled – неактивна.

Замість чотирьох окремих компонентів, Figma дає змогу об'єднати їх у **один набір Variants**, який виглядає як контейнер із кількома підкомпонентами всередині. Це не просто зручно – це системно. Дизайн наближається до логіки коду, де різні стани елемента керуються через властивості.

Для створення варіантів необхідно:

1. Створити базові стани – намалювати або скопіювати кілька версій одного компонента. Всі вони мають бути однакової структури – з текстом, іконкою чи автолейаутом. Наприклад, створити кнопку:
перша – звичайна (фіолетовий фон, білий текст);
друга – при наведенні (світліший фон);
третя – натиснута (темніший фон);
четверта – неактивна (приглушений колір).
2. Перетворити кожну кнопку на компонент (виділити й натиснути **Ctrl + Alt + K**).
3. Об'єднати у Variants. Для цього слід виділити всі створені компоненти й обрати команду **Combine as Variants** на панелі властивостей). Figma об'єднає їх у єдиний контейнер з іконкою  та тегом **Component Set** (Рис. 202).

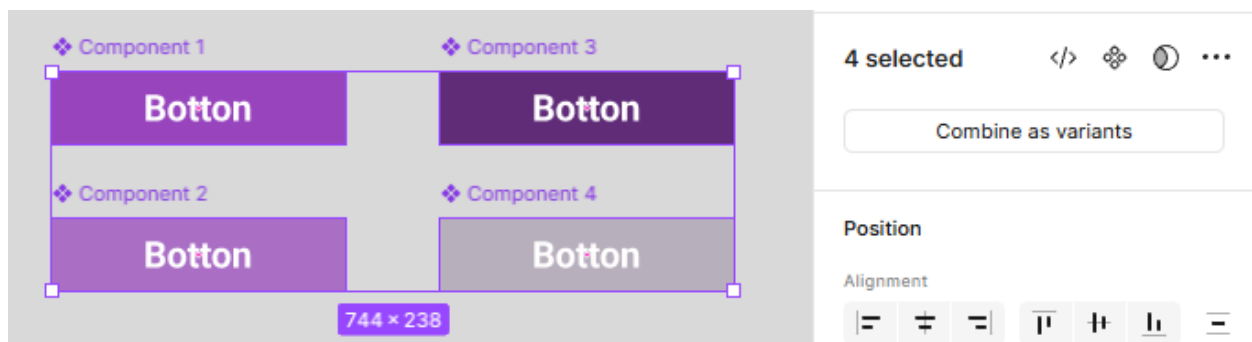


Рис. 202. Об'єднання компонентів у Variants

Тепер у панелі праворуч з'являться властивості, за якими можна перемикаєти стани. За замовчуванням це буде поле Property 1, але можна перейменувати його, наприклад, на State.

4. У правій панелі кожному стану надати значення, наприклад (Рис. 203.

Налаштування властивостей варіантів):

- State = Default;
- State = Hover;
- State = Pressed;
- State = Disabled.

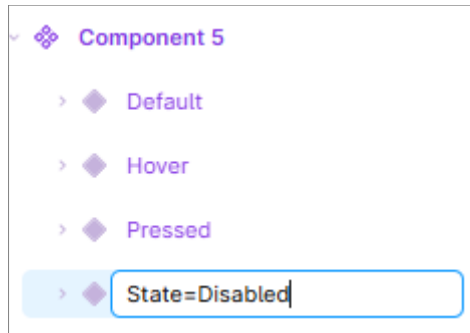


Рис. 203. Налаштування властивостей варіантів

Це дозволить перемикаати стани безпосередньо у вікні властивостей інстанса. Тепер, коли після того, як кнопка буде вставлена в макет, достатньо обрати потрібний стан із випадаючого списку і не потрібно міняти компоненти вручну (Рис. 204).

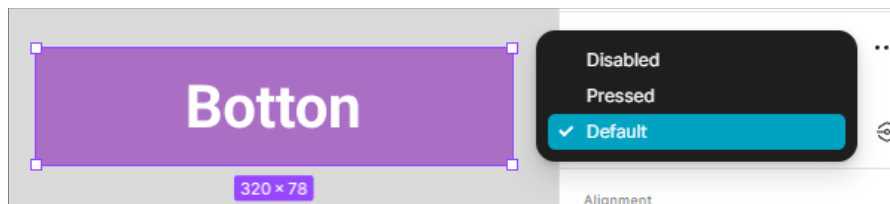


Рис. 204. Зміна стану компонента

Усі варіанти залишаються пов'язаними з головним компонентом. Якщо змінити, наприклад, шрифт або колір тексту в базовому варіанті, зміни автоматично застосуються до всіх станів.

Це надзвичайно зручно при роботі з кнопками, перемикачами, чекбоксами, картками товарів, вкладками або іншими елементами, що мають кілька станів.

Окрім стану State, можна додати ще декілька властивостей, наприклад Size (Small / Medium / Large). Тоді можна комбінувати: State=Hover + Size=Large. Водночас Figma автоматично створює логічну ієрархію назв. Кожен варіант може мати власний Auto Layout, що дає змогу адаптувати компоненти до тексту чи контенту. Також Variants чудово працюють із інтерактивними переходами. Наприклад, у режимі Prototype можна налаштувати: при наведенні (on hover) → перемкнути на State=Hover.

ПУБЛІКАЦІЯ У БІБЛІОТЕКУ

Так само, як і стилі, компоненти можна публікувати у спільній бібліотеці (**Publish Components**). Це дає змогу всім членам команди використовувати єдині елементи у своїх проєктах. Коли головний компонент оновлюється, Figma пропонує іншим користувачам прийняти зміни (**Library Updates**) – так забезпечується синхронність у всіх макетах.

Зазвичай бібліотека створюється у виді окремого файлу. В ньому містяться всі базові елементи: кнопки, поля введення, чекбокси, заголовки, текстові стилі, кольори, ефекти, сітки. Усі вони мають бути оформлені як **компоненти** або **стилі**.

Потім у верхньому правому куті Figma натискають на вкладку **Assets** та обирають іконку  **Library (Бібліотека)** і підтверджують публікацію кнопкою **Publish**.

Після цього бібліотека стане доступною для всіх учасників команди, які мають до неї доступ.

Figma автоматично відстежує зміни у бібліотеці. Якщо головний компонент у бібліотеці оновлюється – наприклад, змінюється колір кнопки або стиль тексту – всі користувачі, які використовують цей компонент у своїх файлах, отримають сповіщення: «Library Updates available». Далі вони можуть переглянути зміни у спливаючому вікні та натиснути **Assent All**, щоб оновити всі компоненти у своєму макеті. Це гарантує, що дизайн у всіх файлах залишається **синхронним і актуальним**, навіть якщо над ним працюють різні люди.

Компоненти з бібліотек позначаються іконкою «ланцюжка»  – це означає, що вони мають зв'язок із оригіналом. Якщо від'єднати компонент (**Detach Instance**), він стане звичайною групою елементів і перестане оновлюватися автоматично.

2.3. СТВОРЕННЯ ІКОНОК, ЛОГОТИПІВ ТА ІЛЮСТРАЦІЙ

Figma – це не лише інструмент для створення макетів інтерфейсів, а й потужний редактор векторної графіки. У ній можна легко створювати **іконки, логотипи й ілюстрації**, які гармонійно поєднуються зі стилем вашого вебсайту

чи застосунку. Завдяки векторній природі всі елементи зберігають чіткість під час масштабування, що важливо для адаптивного дизайну.

Іконки – основа системного інтерфейсу. Вони допомагають користувачеві швидше орієнтуватися, підкріплюють текстові дії візуальними підказками й створюють відчуття цілісності інтерфейсу.

Для створення іконки у Figma необхідно:

1. Створити фрейм потрібного розміру – зазвичай це 24×24 або 32×32 px (для вебу).
2. Використовуючи базові фігури: прямокутник (**R**), коло (**O**), лінії (**L**) та інструмент Pen (**P**), побудувати контури.
3. Поєднати фігури булевими операціями (**Union**, **Subtract**, **Intersect**, **Exclude**), щоб сформувати складні об'єкти.
4. Вирівняти елементи через панель **Align** і встановити точні відступи за допомогою сітки (Grid).
5. Залити іконку одним кольором (зазвичай нейтральним чорним або за дизайн-токеном).
6. Перетворити на компонент (**Ctrl + Alt + K**) і додати в бібліотеку іконок.

Іконки краще малювати зі **Stroke = 2px** без ефектів і градієнтів. Це забезпечує чіткість на різних екранах і під час масштабування.

Логотипи – обличчя бренду. Figma чудово підходить для створення логотипів, насамперед якщо потрібно поєднати текстову частину й графічний символ у єдиному стилі.

Для створення логотипу в Figma потрібно:

1. Вибрати шрифт, який передає характер бренду – сучасний, технологічний, класичний чи рукописний.
2. Створити текстовий шар (**T**) і відрегулювати інтерліньяж, трекінг і міжлітерний інтервал.
3. Перетворити текст у контури – командою **Outline Stroke** або **Convert to Vector**. Це зробить логотип незалежним від шрифтів.

4. Додати графічний символ – просту форму, монохромну і зрозумілу (коло, квадрат, лінії, знак нескінченності тощо).
5. Використати симетрію та пропорції – Figma дає змогу легко копіювати, віддзеркалювати і точно вирівнювати об'єкти.
6. Застосувати фірмові кольори бренду з палітри або токенів.

Логотипи рекомендується створювати у кількох версіях – повний (з текстом) і скорочений (із піктограмою). Це дасть змогу використовувати його як favicon, app icon або водяний знак.

Для презентації логотипу можна створити **mockup** – наприклад, нанести його на візитку, телефон чи сторінку сайту. Це допоможе перевірити, як він виглядає в реальних умовах.

Ілюстрації – характер і настрої інтерфейсу. Вони допомагають зробити дизайн живішим та емоційним, добре поєднуються з мінімалістичними інтерфейсами, якщо не перевантажують композицію.

Для створення ілюстрації у Figma потрібно:

1. Побудувати основні форми за допомогою примітивів і кривих, використовуючи кольори з однієї палітри, щоб уникнути дисонансу.
2. Згрупувати елементи логічно: фон, персонажі, деталі. Це спрощує редагування.
3. Додати відтінки через прозорість або градієнти. У Figma доступні режими **Linear, Radial, Angular, Diamond**.
4. Використати **Pen Tool** для створення плавних контурів (наприклад, волосся, хвиль чи листя).
5. Застосувати **Auto Layout** для ілюстрацій, які мають масштабуватись або складатись із повторюваних частин.
6. Перетворити фрагменти на компоненти, якщо вони повторюються – наприклад, обличчя персонажа чи лист дерева.

Доречно створити власну мінібібліотеку ілюстрацій для продукту – це допоможе зберегти єдиний стиль у всіх екранах.

Усі створені іконки, логотипи чи ілюстрації можна експортувати у формати:

- **SVG** – для веба (зберігає векторність, малу вагу файлу);
- **PNG** – для попереднього перегляду або соціальних мереж;
- **PDF** – для друку чи презентацій.

РОЗДІЛ 3. РАСТРОВА ГРАФІКА У ВЕБДИЗАЙНІ

А. ОСНОВНІ ПОНЯТТЯ РАСТРОВОЇ ГРАФІКИ

Растрова графіка є одним із основних типів комп'ютерної графіки, що широко використовується для створення, редагування та відображення зображень.

Растрова графіка ґрунтується на використанні пікселів. Кожне зображення в такому форматі складається з великої кількості маленьких точок, які називаються пікселями (англ. «pixel»). Пікселі розташовані в сітці, і кожен піксель має свій колір, що сукупно створює зображення.

Роздільна здатність (Resolution) PPI (pixels per inch) – вказує на кількість пікселів, які формують зображення. Роздільна здатність зазвичай вимірюється в пікселях на дюйм або в загальному числі пікселів у горизонтальному і вертикальному напрямках (наприклад, 1920×1080). Що більша роздільна здатність, то більше деталей і чіткіше зображення.

Щільність DPI (dots per inch, крапок на дюйм) – кількість крапок на одиницю площі (зазвичай на дюйм) для друку (наприклад, журнал – 300 DPI, газета – 72 DPI).

Розмір зображення визначається кількістю пікселів по горизонталі та вертикалі (ширина × висота). Це впливає на загальний розмір файлу і якість зображення.

Глибина кольору (Color Depth) визначає кількість кольорів, які можуть бути відображені на зображенні. Що більша глибина кольору, то більше відтінків може бути представлено. Визначається як 2^N , де N – глибина кольору – кількість бітів, які використовуються для зберігання кольору кожного пікселя, наприклад:

- **1 біт (чорно-біле)**: може відображати тільки два кольори – чорний і білий;
- **8 біт**: може відображати до 256 кольорів. Це часто використовувалося для графіки в старих системах або простих іконках;
- **16 біт**: може відображати до 65,536 кольорів. Це часто використовується в деяких графічних програмах та іграх;

- **24 біт:** зазвичай називається True Color, дозволяє відображати до 16,777,216 кольорів. Це стандартний формат для більшості сучасних зображень;
- **32 біт:** включає 24 біти для кольору і 8 біт для альфа-каналу, що відповідає за прозорість. Кожен піксель у зображенні з альфа-каналом має значення прозорості від 0 (повністю прозорий) до 255 (повністю непрозорий).

Шари (Layers) – це різні рівні, на яких можна розмістити об’єкти або зображення. Використовуються для організації елементів графіки, дозволяючи легко редагувати окремі частини зображення без впливу на інші.

ПЕРЕВАГИ ТА НЕДОЛІКИ РАСТРОВОЇ ГРАФІКИ

Переваги:

- **Деталізація і фотореалістичність зображень:** растрові зображення можуть містити велику кількість деталей завдяки пікселям, які забезпечують високу якість відображення текстур і кольорів, а також забезпечують тонке регулювання кольорів.
- **Простота автоматизованого вводу:** багато сучасних камер і сканерів створюють зображення у растровому форматі, що робить ці формати загальнодоступними.
- **Використання різних ефектів:** туман, розмитість тощо
- **Висока швидкість обробки** складних зображень, якщо не потрібне масштабування.

Недоліки:

- **Залежність від роздільної здатності:** растрові зображення мають фіксовану роздільну здатність, і у разі збільшення її розміру можуть втрачати якість, виглядаючи розмитими або пікселізованими.
- **Великий розмір файлів:** висока якість растрових зображень часто призводить до великих розмірів файлів, що може бути проблемою для зберігання та швидкості завантаження.

- **Обмеженість редагування:** зміна окремих частин растрового зображення може бути складною та призводити до втрати якості. виправлення помилок або змінення деталей може бути непростим.

АЛГОРИТМИ СТИСКУ

Растрові зображення зазвичай зберігаються в стислому вигляді. Залежно від типу стиснення може бути можливо або неможливо відновити зображення точно таким, яким воно було до стиснення. Стиск графічних файлів відрізняється від їхньої архівації за допомогою програм архіваторів (rar, zip, arj та ін.) тим, що алгоритм стиску включається у формат графічного файлу.

Стиснення без втрат:

- **RLE (Run-Length Encoding).** Стискає зображення шляхом заміни послідовності повторюваних пікселів на один піксель і кількість його повторень. Наприклад, «AAAAA» стає «A5». Використовується для стиску малюнків типу аплікації, що містять великі області однотонного зафарбування. Наприклад, **BMP** і **PCX**.
- **LZW (Lempel-Ziv-Welch).** Використовує таблиці для зберігання шаблонів зображення, що дає змогу замінювати повторювані частини зображення більш компактним кодом. Використовується для малюнків типу діаграми, здійснюючи пошук повторюваних у малюнку «візерунків». Наприклад, **TIFF** і **GIF**.
- **Huffman Coding.** Використовує частотний аналіз даних для створення оптимальних бітових кодів для кожного символу. Символи, що зустрічаються частіше, мають коротші коди. Наприклад, Huffman Coding може використовувати **JPEG** після дискретного косинусного перетворення (DCT), **PNG** та **TIFF** як один із варіантів стиснення даних.
- **LZ77 (Lempel-Ziv 77).** Це алгоритм, який виконує пошук повторюваних частин у зображенні та замінює їх на посилання на попередні частини. Він ефективно зменшує розмір даних шляхом заміни повторюваних фрагментів на коротші коди. Наприклад, **PNG**.

Стиснення з втратами:

- **JPEG (Joint Photographic Experts Group)**. Найпоширеніший алгоритм стиснення для фотографій. Використовує дискретне косинусне перетворення (DCT) для перетворення зображення у частотну область, де менш важливі частоти можуть бути видалені. Наприклад, **JPEG, JFIF**.

ГРАФІЧНІ ФОРМАТИ

Найпоширенішими типами растрових файлів є:

RAW. Зберігає інформацію, безпосередньо одержувану з матриці цифрового фотоапарата або аналогічного пристрою без застосування до неї будь-яких перетворень. Майже всі напівпрофесійні та професійні цифрові фотоапарати дозволяють зберігати зображення у форматі RAW. Порівняно з іншими типами растрових файлів формат файлу RAW зберігає найбільшу кількість деталей, які можна редагувати, стискати та конвертувати в інші формати надалі.

Розширення файлів для RAW може змінюватися залежно від виробника камери. Наприклад, **.cr2** (Canon RAW Version 2), **.cr3** (Canon RAW Version 3), **.nef** (Nikon Electronic Format), **.nrw** (Nikon RAW), **.arw** (Sony Alpha RAW), **.sr2** (Sony RAW), **.raf** (Fujifilm RAW File), **.dng** (Digital Negative, універсальний формат RAW, розроблений Adobe).

Щодо колірної моделі формат RAW зазвичай зберігає дані у форматі, близькому до того, як їх захоплює сенсор камери. Це може бути: RGB або CMYK.

Втрата інформації відбувається в результаті округлення і виходу кольору пікселя за межі допустимих значень, а не внаслідок стиснення.

Використовується під час знімання в складних умовах (недостатня освітленість, неможливість виставити баланс білого тощо) та для подальшої обробки на комп'ютері (зазвичай в ручному режимі).

BMP (Bitmap). Графічний формат для зберігання растрових зображень у виді двовимірного масиву пікселів із розширеннями файлів: **.bmp** або **.dib** (Device-Independent Bitmap).

BMP файли можуть підтримувати різні глибини кольору, включаючи монохромні (1-біт), 16 кольорів (4-біт), 256 кольорів (8-біт), та 16,7 мільйона кольорів (24-біт). Колір у будь-якому разі можна задати тільки в колірній моделі RGB, але в бітності 16 і 32 можна отримати відтінки сірого з глибиною до 16 і 32-ох біт відповідно.

Більшість BMP файлів зберігаються без стиснення, що забезпечує високу якість зображення, але збільшує розмір файлу. Однак є підтримка стиснення за допомогою RLE для 4-бітних та 8-бітних зображень.

У 32-бітних BMP-файлах підтримується альфа-канал, що дає змогу зберігати інформацію про прозорість.

У цьому форматі можна зберігати тільки одношарові растри.

Через свою простоту та відкритість BMP часто використовують у простих додатках і для зберігання проміжних зображень. Однак через великий розмір файлів у сучасних застосуваннях віддають перевагу форматам з ефективнішим стисненням, таким як PNG або JPEG.

GIF (Graphics Interchange Format). Растровий формат, який витісняється **PNG**, але все ще використовується через підтримку анімації, яка відсутня в чистому PNG, хоча програмне забезпечення дедалі більше підтримує формат **APNG**. Має розширення файлів .gif.

Використовує палітру з 256 кольорів із 24-бітного діапазону моделі RGB.

Використовує алгоритм стиснення LZW (Lempel-Ziv-Welch), який дає змогу зберігати зображення без втрат якості.

Один із кольорів у палітрі може бути оголошений повністю прозорим, але напівпрозорість пікселів (технологія альфа-каналу) не підтримується.

Дає змогу створювати анімовані зображення за допомогою кількох шарів або кадрів. Кожен кадр зображення зберігається як окремий шар і ці кадри відтворюються послідовно, повторюючись по колу, створюючи ефект анімації.

Цей формат популярний в інтернеті, насамперед для створення мемів, анімованих зображень і простих графічних елементів через свою малу вагу та сумісність із більшістю браузерів і платформ.

PNG (Portable Network Graphics). Цей формат був створений як покращена альтернатива формату GIF, а також подекуди для заміни складнішого формату TIFF. Файли мають розширення .png, .apng і можуть зберігати метадані, такі як інформація про авторські права або коментарі.

PNG підтримує колірні глибини до 48 біт на піксель для кольорових зображень і до 16 біт на піксель для відтінків сірого. Використовує колірну модель RGB.

Використовує стиснення даних без втрат, це означає, що зображення не втрачає якість під час зберігання та завантаження, але файли можуть бути доволі великими порівняно з форматами зі стисненням зі втратами, такими як JPEG. Основний метод стиснення PNG ґрунтується на алгоритмі Deflate, який є комбінацією двох технік: LZ77 та Huffman Coding.

Підтримує альфа-канали, що дозволяє використовувати прозорість у зображеннях.

Не підтримує шари та анімацію (для цього є формат .apng Animated PNG).

Цей формат широко використовується у вебдизайні, графічному дизайні та для збереження зображень високої якості.

TIFF (Tag Image File Format). Це формат для зберігання растрових зображень який широко використовується для високоякісних зображень, таких як фотографії, медична візуалізація та скановані документи, завдяки своїй гнучкості та можливості зберігати зображення без втрат якості.

Файли мають розширення .tiff, .tif і можуть зберігати метадані разом із зображенням, такі як інформація про авторство, дата створення, і налаштування камери.

Формат підтримує різні колірні моделі, включаючи RGB, CMYK, Grayscale різні колірної глибини, від 1-бітного монохромного до 32-бітного кольору.

Може використовувати різні методи стиснення без втрат (LZW, PackBits, Huffman Coding), з втратами (JPEG, якість зображення може знижуватися, але розмір файлу може бути значно зменшений) або не стискати зображення взагалі. Формат TIFF накладає обмеження на розмір файлу до 4 GB.

Формат підтримує альфа-канал та може містити кілька зображень в одному файлі, що дає змогу зберігати графічні зображення у виді шарів або кадрів. Анімація не підтримується.

JPEG (Joint Photographic Experts Group). Формат, що часто використовується як формат даних у цифрових камерах. У мережі «Інтернет» формат JPEG застосовується для відображення напівтонових ілюстрацій та графічної інформації з плавним переходом тонів. Формат добре підходить для фотографій і складних зображень, але може бути менш ефективним для зображень із великими рівними кольоровими ділянками або текстом, де формати без втрат, як PNG, можуть бути кращими. Розширення файлів: .jpg, .jpeg.

JPEG підтримує колірні моделі RGB та YCbCr і стиснення з втратою **JPEG**. Під час детального розгляду сильно стисненого зображення помітно розмиття меж і характерний **муар** поблизу них.

Формат не підтримує альфа-канал, шари й анімацію.

PSD (PhotoShop Document). Власний формат програми Adobe Photoshop, один із найбільш потужних за можливостями зберігання растрової графічної інформації, що зберігається без стиснення і використовується для професійного редагування зображень. Розширення файлу .psd.

Підтримуються 48-розрядне кодування кольору та різні колірні моделі: RGB, CMYK, Grayscale, Lab Color.

Наявна багатошаровість, що дає змогу редагувати окремі елементи зображення незалежно один від одного та редагувати текстові елементи без потреби їх конвертації в растрові зображення.

Формат зберігає додаткові канали, такі як **альфа-канали** для прозорості, **канали кольору**, що містять інформацію про яскравість конкретного кольорового компонента в зображенні, спеціальні канали для створення спеціальних ефектів (тіні, світіння, розмиття...) або масок (для управління видимістю частин зображення).

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

Платне:

- **Adobe Photoshop** – один із надпотужних редакторів для роботи з растровою графікою. Має величезну кількість інструментів і можливостей для редагування зображень.
- **Corel PaintShop Pro** – інший популярний редактор, який пропонує різноманітні інструменти для оброблення фотографій і створення графіки.
- **Affinity Photo** – професійний редактор із великою кількістю функцій за зниженою ціною порівняно з Adobe Photoshop.

Безплатне:

- **GIMP** – безплатний і відкритий редактор, який пропонує чимало функцій, схожих на Photoshop.
- **Krita** – безплатний редактор, орієнтований на цифрових художників. Можливість створення плоскої анімації.
- **Paint.NET** – безплатний растровий графічний редактор для Microsoft Windows, створений на основі .NET Framework. В деяких випадках можна розглядати як полегшену безкоштовну альтернативу Adobe Photoshop.

В. ОСНОВНІ ІНСТРУМЕНТИ РАСТРОВИХ РЕДАКТОРІВ

Растрова графіка є одним із основних інструментів для створення та редагування графічних зображень у сучасних комп'ютерних програмах.

Розглянемо головні концепції растрової графіки, зокрема основні функції та можливості графічних редакторів на прикладі такого інструменту, як Paint.NET, що допоможе краще зрозуміти, як ефективно використовувати растрову графіку для досягнення ваших дизайнерських цілей.

Графічний редактор Paint.NET є одним із популярних інструментів для роботи з растровою графікою. Ця програма поєднує простий у використанні інтерфейс і потужні функції, що дають змогу і новачкам, і досвідченим користувачам виконувати широкий спектр завдань: від базового редагування до створення складних зображень.

У цьому розділі ми розглянемо:

Основні можливості Paint.NET для роботи з растровою графікою.

Інструменти й техніки редагування зображень.

Методи поліпшення якості, корекції кольорів і застосування ефектів.

Вивчаючи цей матеріал, ви отримаєте базові навички роботи з растровою графікою та зрозумієте, як використовувати Paint.NET для реалізації творчих ідей.

Paint.NET – це безплатний графічний редактор, який підтримує роботу з шарами, необмежену історію, спеціальні ефекти та великий набір інших корисних інструментів для редагування зображень.

Завантажити Paint.NET можна з офіційного сайту getpaint.net або з сайту <https://paintnet.org.ua/>.

ВСТУП ДО PAINT.NET

ІНТЕРФЕЙС КОРИСТУВАЧА PAINT.NET

Інтерфейс користувача Paint.NET розділений на десять основних областей (Рис. 205):

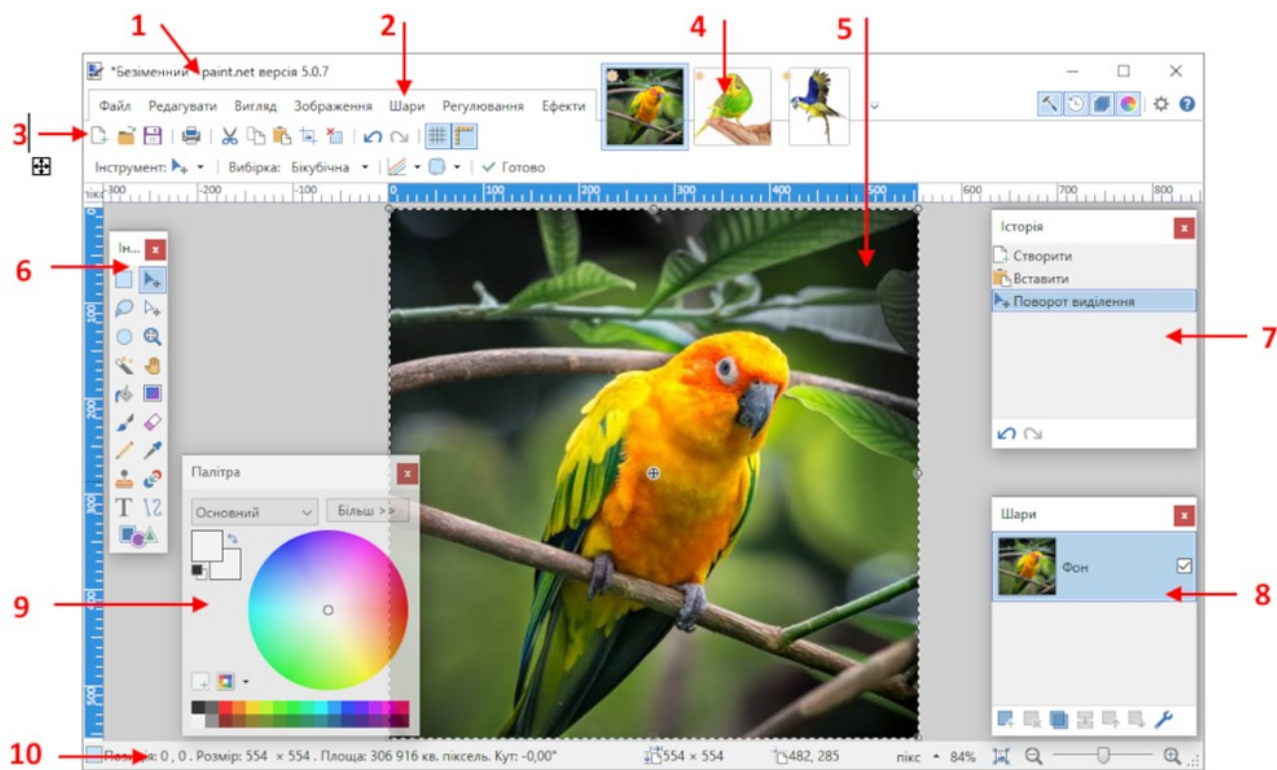


Рис. 205. Інтерфейс користувача Paint.NET

1. НАЗВА ВІКНА

У цьому рядку вказується назва файлу, з яким ви працюєте у конкретний момент, і його формат. Також тут відображається поточна версія Paint.NET.

2. ПАНЕЛЬ МЕНЮ

Панель меню надає доступ до основних інструментів і функцій для роботи із зображеннями. На цій панелі розміщено сім пунктів меню, чотири іконки для керування плаваючими вікнами (**Вікно інструментів**, **Журнал**, **Вікно шарів** та **Вікно палітри**), іконка **Параметри** та іконок **Довідка** (Рис. 206).

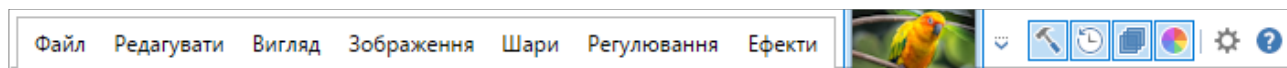


Рис. 206. Панель меню Paint.NET

3. ПАНЕЛЬ ІНСТРУМЕНТІВ

На панелі інструментів (Рис. 207) розміщені іконки для швидкого доступу до команд, які використовуються найчастіше. Панель складається з двох рядів: верхній

(розміщені постійні іконки) і нижній (відображаються параметри для активного інструменту).

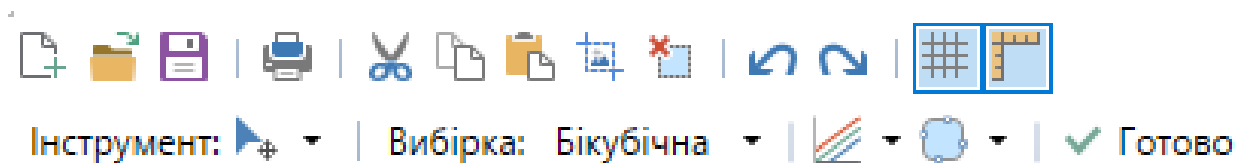


Рис. 207. Панель інструментів Paint.NET

Детальніше буде розглянута у розділі [Основні інструменти та функції](#).

4. СПИСОК ЗОБРАЖЕНЬ

Список зображень (Рис. 208) відображається у верхній частині вікна Paint.NET, поряд із Панеллю меню. Мініатюра відображається для кожного відкритого зображення та дозволяє швидко перемикається між ними. Синє підсвічування показує, яке зображення редагується зараз.

Щоб перейти на потрібне зображення, необхідно клікнути його мініатюру. Якщо для показу всіх відкритих зображень на панелі не вистачає місця, по краях з'являються стрілки, що дозволяють зміщувати іконки попереднього перегляду вліво та вправо. Також можна гортати мініатюри, використовуючи коліщатко мишки, навівши вказівник на список зображень.

У правій частині списку зображень є стрілка, спрямована вниз. Під час її натискання відкривається список відкритих зображень (рис. 208).



Рис. 208. Список відкритих зображень

Закрити зображення можна, натиснувши хрестик у верхньому правому куті мініатюри активного зображення (Рис. 209) або комбінацією клавіш **Ctrl + F4**.

Якщо зображення містить незбережені зміни, то у лівому верхньому куті його мініатюри відображається помаранчева зірочка.



Рис. 209. Інструменти керування мініатюрою

Під час спроби закрити зображення, яке містить незбережені зміни, з'явиться запит на підтвердження дії з опціями: **Зберегти**, **Не зберігати** або **Скасувати**.

Мініатюри можна перевпорядкувати за допомогою перетягування.

При натисканні правою кнопкою миші на будь-який ескіз зображення у списку зображень відкривається контекстне меню, яке містить команди: **Копіювати шлях**, **Відкрити папку з файлом**, **Зберегти**, **Зберегти як ...** та **Закрити**.

Контекстне меню для поточного зображення можна відкрити також за допомогою клавіш **Alt** + **-** (символ мінус).

5. РОБОЧА ОБЛАСТЬ

У цій області відбуваються всі дії зі створення та редагування зображень.

6. ВІКНО ІНСТРУМЕНТІВ

Вікно **Інструменти** містить набір основних інструментів, необхідних для редагування зображень (Рис. 210). Активний інструмент виділяється у вікні **Інструменти** синім підсвічуванням. При наведенні вказівника на піктограму інструмента з'явиться підказка з його назвою та гарячою клавішею.

Вікно інструментів можна приховати або відобразити, натиснувши клавішу **F5** або клацнувши по іконці цього вікна на панелі меню.

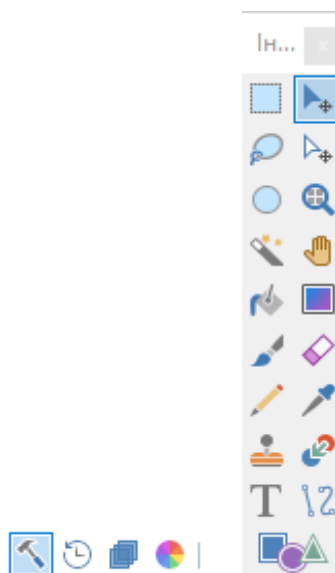


Рис. 210. Іконки вікон на панелі меню та вікно Інструменти

Вікно інструментів можна перемістити, клацнувши та перетягнувши його за рядок заголовка.

Вікно можна повернути у вихідне положення, натиснувши **Ctrl+Shift+F5** або утримуючи **Ctrl+Shift** та клацнувши відповідну іконку на **Панелі меню**.

7. ВІКНО ІСТОРИЇ

У вікні історії (Рис. 211) відображаються всі дії, які виконуються над зображенням. Дозволяє легко скасувати декілька операцій або повторити їх знову. Закриття Paint.Net, або закриття зображення видаляє історію редагування.

Щоб переглянути зміни у зображенні, використовуйте стрілки **Скасувати / Повернути** або клацніть відповідний запис у вікні **Історія**. Зображення буде відновлено на вибрану точку.

Скасування будь-якої дії буде відзначено внизу списку сірим фоном.

Після виконання нової операції всі скасовані дії, які можна було повернути, будуть видалені зі списку.

Якщо нова дія виконується під час перегляду попередніх дій з історією, нова дія замінює та стирає всі інші дії. В такому разі відновити елементи історії неможливо.

Вікно **Історія** можна приховати або відобразити, натиснувши клавішу **F6** або клацнувши його піктограму на панелі меню.

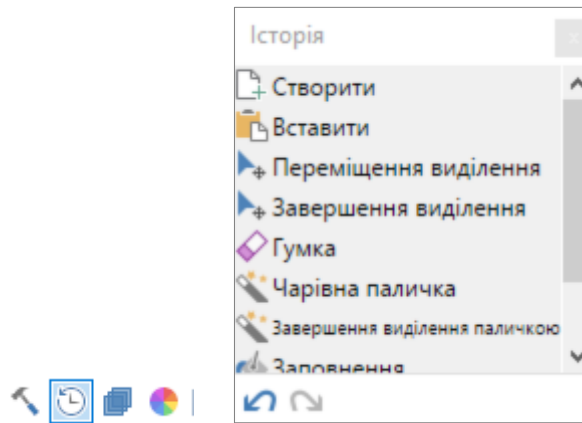


Рис. 211. Іконки вікон на панелі меню та вікно Історія

Вікно можна повернути у вихідне положення, натиснувши **Ctrl+Shift+F6** або натиснувши **Ctrl+Shift** та клацнувши піктограму **Історія**.

Перемістити вікно **Історії** можна, утримуючи його за рядок заголовка.

8. ВІКНО ШАРИ

Вікно **Шари** (Рис. 212) – це основна область для додавання, видалення, доступу та керування шарами зображення. Кожне зображення містить щонайменше один шар.

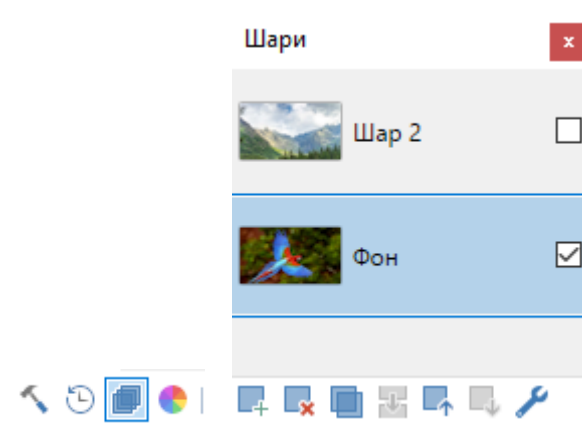


Рис. 212. Іконки вікон на панелі меню та вікно Історія

Більш детально це вікно буде розглянуто у розділі **Шари**.

9. ВІКНО ПАЛІТРА

Вікно **Палітра** (Рис. 213) використовується для встановлення кольорів і керування ними.

Приховати або відобразити палітру кольорів можна, натиснувши клавішу **F8** або клацнувши її піктограму на панелі меню.

Вікно можна повернути у вихідне положення, натиснувши **Ctrl+Shift+F8** або натиснувши **Ctrl+Shift** і клацнувши піктограму **Палітра**.

Перемістити вікно можна, утримуючи його за рядок заголовка.

Розглянемо функціонал вікна **Палітра**.

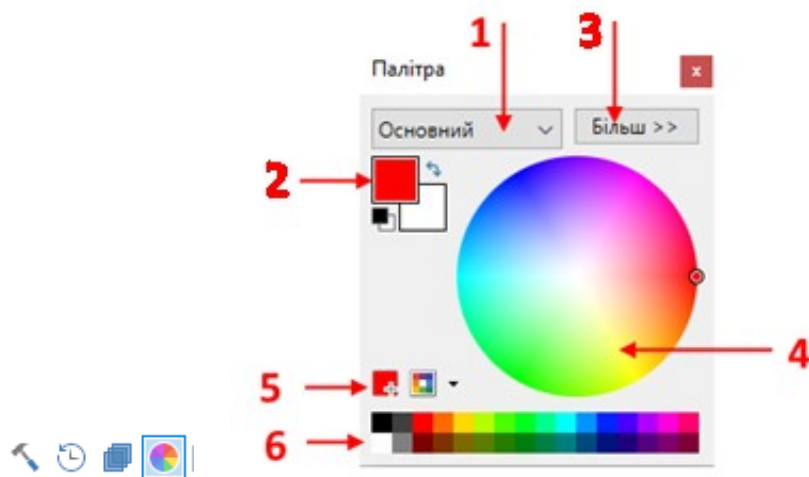


Рис. 213. Іконки вікон на панелі меню та вікно **Палітра**

1. Перемикач кольору Основний / Додатковий. Список, що розкривається, пропонує активацію панелі Основного (верхній) або Додаткового (нижній) кольорів на слоті палітри (2). Обраний у слоті колір буде використано для наступних операцій редагування.

2. Вибір поточного кольору. Квадрати, що перекриваються, у верхньому лівому куті вікна **Палітра** показують вибрані в цей момент кольори **Основного** (квадрат на передньому плані) та **Додаткового** (квадрат на задньому плані) кольорів.

Натискання на двосторонню стрілку між цими квадратами міняє місцями кольори в слотах Основного та Додаткового кольорів.

Маленький чорно-білий значок у зоні квадратів установлює кольори за замовчанням Основний – чорним та Додатковий – білим.

3. Кнопка Більше >>. Натисніть цю кнопку, щоб розгорнути вікно **Палітра** для відображення детальної інформації про колір, що міститься в активному слоті (Основного або Додаткового кольору) (Рис. 214). У відкритому вікні буде додатково видно повзунки **RGB** (Red – червоний, Green – зелений та Blue – блакитний),

HSV (Hue – відтінок, Saturation – насиченість та Value – значення), Альфа-прозорість, поле **Hex** для введення шістнадцяткового значення кольору активного кольорового слота та палітра на 96 кольорів.

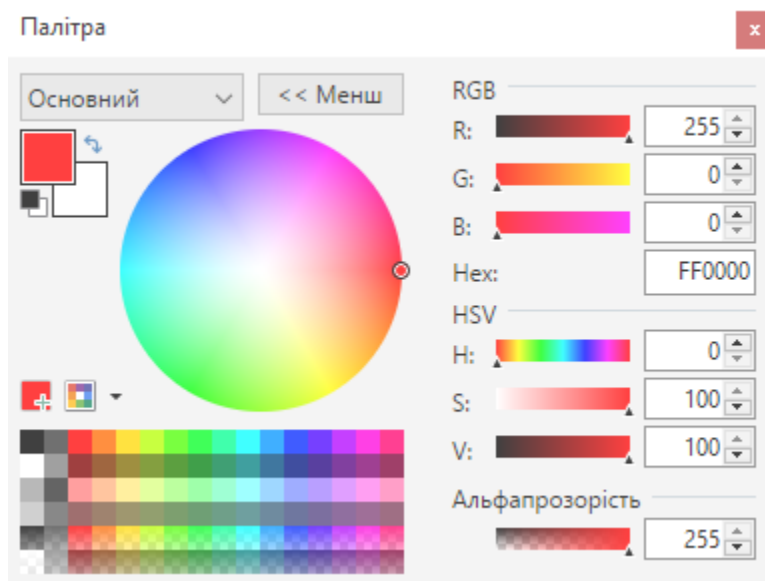


*Рис. 214. Розгорнуте вікно **Палітра***

У розкритому вікні **Палітра**, кнопка **Більше** змінює назву на **Менше**. Натискання цієї кнопки повертає вікно Палітри до розміру за замовчуванням.

4. Колірне коло. Колірне коло дозволяє вибрати колір для поточного активного кольорового слота. Якщо активний Основний колір, клацання на колірному колі встановить цей колір на слот Основного кольору і навпаки.

Якщо навести вказівник миші на довільний колір на колі та затиснути клавішу **Ctrl** вибір кольору обмежиться колом того ж радіусу, що й поточний вибраний колір. Це те саме, що й зміна значення відтінку **H** у розділі HSV розширеного вікна кольору (Рис. 215).

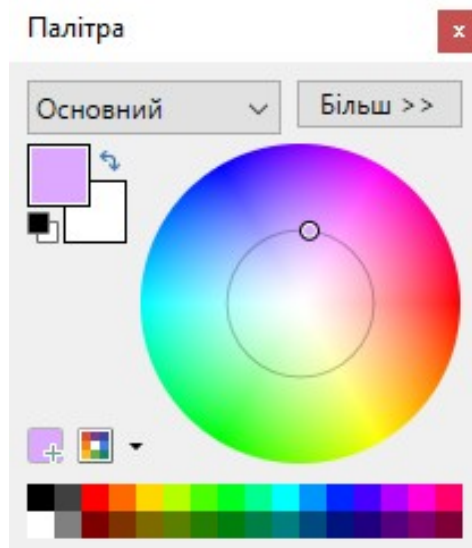


Рис. 215. Зміна відтінку на колірному колі

Якщо навести вказівник миші на довільний колір на колірному колі та затиснути клавішу **Alt** вибір кольору обмежиться одним напрямком по радіусу. Кутове положення радіуса буде згенеровано поточним кольором. Це дорівнює зміні значення насиченості **S** у розділі HSV (Рис. 216).

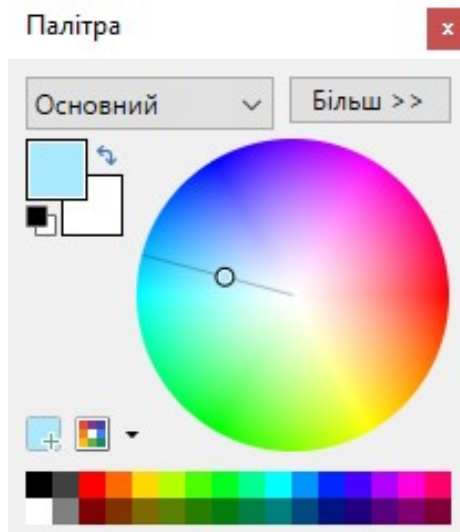


Рис. 216. Зміна насиченості на колірному колі

Утримування **Shift** з курсором над будь-яким кольором на колі обмежує вибір кольору деякими променями на колірному колі. Поточний колір може не знаходитися на жодному промені. Це те саме, що й зміна насиченості **S** та/або відтінку **H** одночасно у розділі HSV (Рис. 217).

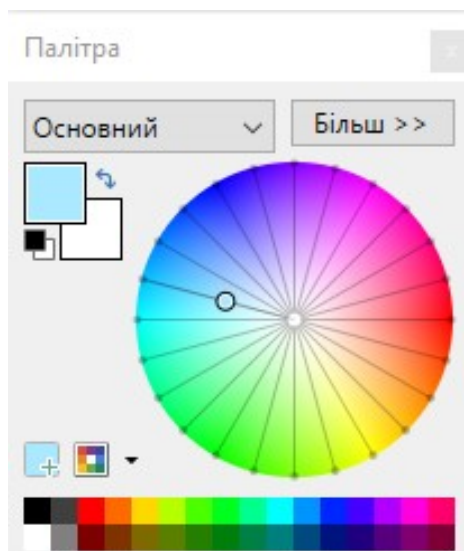


Рис. 217. Одночасна зміна насиченості та відтінку на колірному колі

5. Елементи керування палітрою. Вище панелі палітри знаходяться дві іконки, що дозволяють керувати кольором: **Додати колір в палітру** (ліворуч) та **Керування палітрами** (праворуч).

Перша з них додає поточний колір до палітри: виберіть колір, клацніть по цій іконці, а потім по комірці палітри, щоб помістити туди поточний колір.

Друга іконка відкриває меню, яке надає різні команди для керування палітрою загалом:

Зберегти поточну палітру як... – зберігає поточну палітру з іменем файлу користувача. Після збереження файлу він буде доступний для вибору у верхній частині списку.

Відкрити теку з палітрами – відкриває у провіднику Windows папку, де зберігаються файли палітри. За замовчуванням це:

C:\Users\[UserName]\Documents\paint.net User Files\Palettes. Якщо папка палітр не існує, її можна створити, клацнувши на пункт меню, а потім відкривши її.

Типова палітра – відкривається панель Paint.NET за замовчуванням.

6. Палітра. Цей елемент керування використовується для швидкого вибору одного з кольорів. Клацніть **лівою** клавiшею миші на кольорі, щоб завантажити його в слот **Основного кольору**. Клацніть **правою** клавiшею миші по кольору, щоб завантажити його у слот **Додаткового кольору**.

Якщо панель не розкрита, на панелі відображаються лише 32 поточні кольори. При відкритій палітрі відображається вся палітра з 96 відтінків.

10. РЯДОК СТАНУ

Ця область розділена на кілька секцій. Перша зліва показує назви вибраних інструментів. Наступна секція відображає або процес обробки зображення (якщо воно відбувається зараз), або інформацію про розмір зображення, масштаб, положення курсора (розташування курсора відраховується від верхнього лівого кута зображення, що знаходиться в позиції (0,0)) та одиниці вимірювання (пікселі, дюйми, сантиметри). Також за допомогою повзунка масштабування можна швидко змінити масштаб зображення

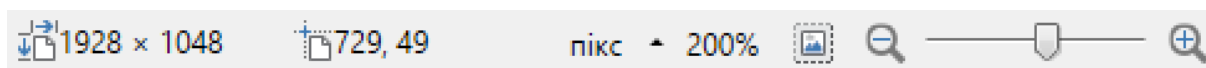


Рис. 218. Рядок стану

НАЛАШТУВАННЯ РОБОЧОГО ПРОСТОРУ

Налаштувати робочий простір у Paint.NET можна у діалоговому вікні **Параметри**, що відкривається, клацнувши на піктограму шестерні (Рис. 219) у верхньому правому кутку вікна програми.

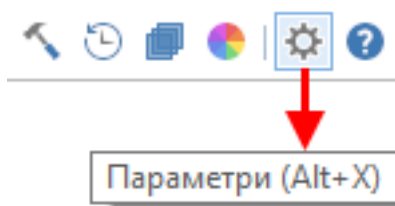


Рис. 219. Іконка Параметри

У вікні **Параметри** доступні такі вкладки:

Інтерфейс. Ця вкладка містить налаштування зовнішнього вигляду програми та її поведінку (Рис. 220). А саме можна вибрати між світлою та темною **Кольоровою схемою** інтерфейсу, вибрати мову, на якій відображатиметься інтерфейс Paint.NET, змінити налаштування прокрутки.

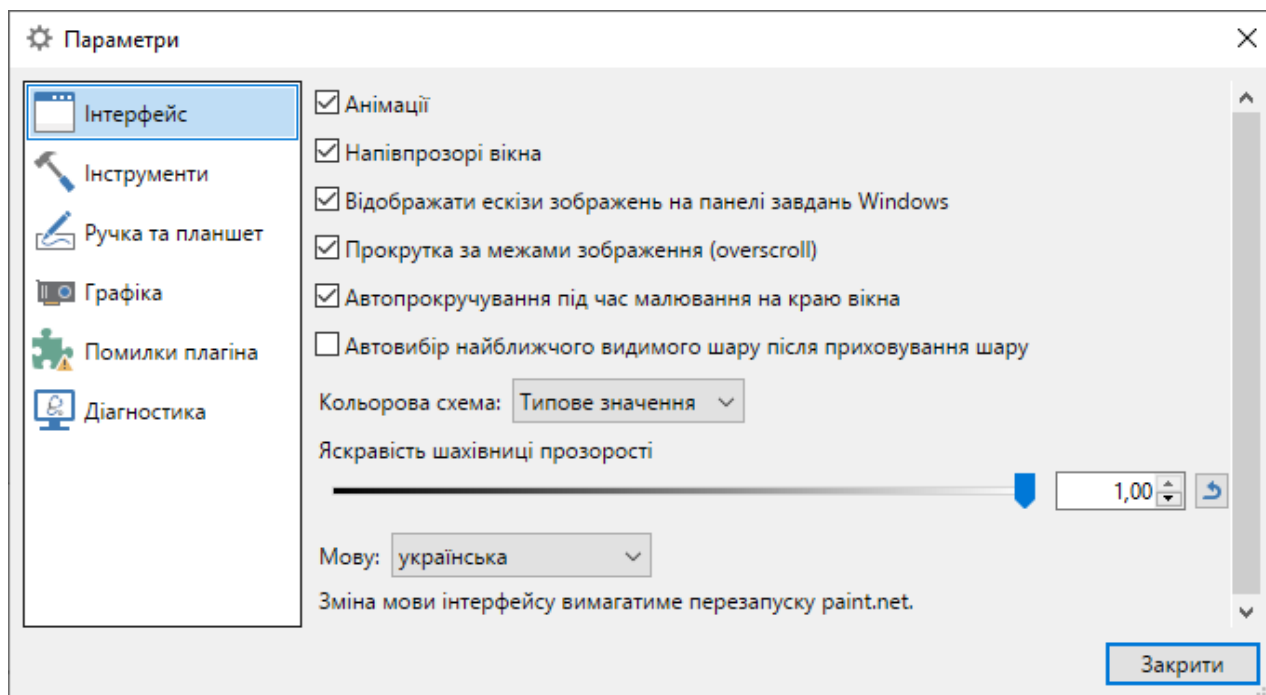


Рис. 220. Вікно Параметри. Вкладка інтерфейс

Інструменти. Ця вкладка містить дії та налаштування за замовчуванням для деяких інструментів: виділення, тексту, градієнту, чарівної палички тощо.

Ручка та планшет. Вкладка дозволяє користувачам налаштовувати параметри для малювання за допомогою графічних планшетів або стилусів: чутливість до натиснення, товщина лінії / інструмента залежно від тиску, реагування на нахил стилуса. Також наявна опція, яка допоможе калібрувати ваш стилус для точного використання.

Графіка. Вкладка дозволяє користувачам налаштувати графічні параметри програми для поліпшення продуктивності або якості відображення: увімкнення або вимкнення використання GPU (графічного процесора) для обробки зображень, налаштування способу згладжування малюнка при зміні масштабу або при переміщенні зображення тощо.

Помилки плагіну. Вкладка містить інформацію про плагіни, що підключаються, які не вдалося завантажити або заблоковані для завантаження через відомі проблеми сумісності.

Діагностика. Вкладка містить інформацію про систему та запущену версію Paint.NET.

НАЛАШТУВАННЯ РОБОЧОЇ ОБЛАСТІ

За налаштування зовнішнього вигляду робочої області Paint.NET відповідають команди, що розташовані у меню **Вигляд** (Рис. 221). Вони змінюють спосіб представлення зображення у вікні редагування, перемикають видимість піксельної сітки, лінійок і прапорців для встановлення одиниць виміру.

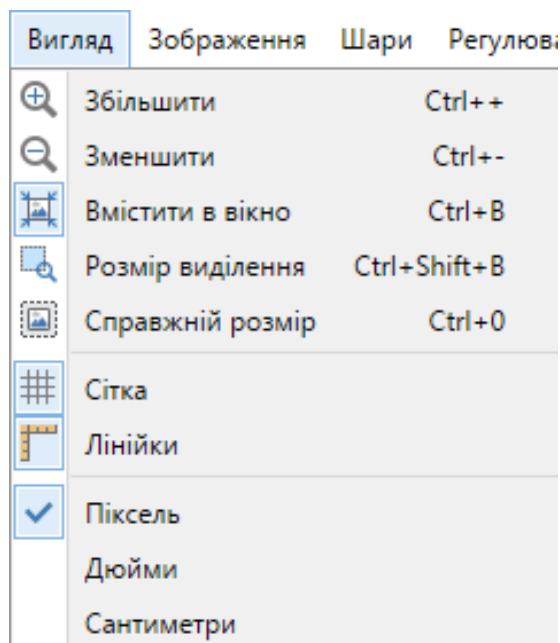


Рис. 221. Меню Вигляд

Збільшити, Ctrl + + – збільшення масштабу відображення зображення на екрані з фіксованим кроком (наприклад, 100%, 150%, 200%...).

Зменшити, Ctrl + - – зменшення масштабу відображення зображення на екрані з фіксованим кроком (наприклад, 100%, 67%, 50%, 33%, 25%...).

За розміром вікна, Ctrl + B – зміна масштабу відображення зображення до меж вікна редагування або до повного розміру.

Якщо зображення більше, ніж вікно редагування, воно буде зменшено, щоб поміститися у вікні редагування. Якщо зображення менше розміру вікна редагування, воно відображатиметься у повному розмірі.

За виділенням, Ctrl + Shift + B – збільшення масштабу активного виділення до розміру вікна редагування.

Справжній розмір, Ctrl + O – встановлення масштабу зображення рівним 100%.

Сітка – накладання на поточне зображення сітки, кожна комірка якої має розмір, що дорівнює одному пікселю (Рис. 222). Піксельна сітка особливо корисна при точному попівксельному редагуванні. Піксельна сітка не видима за масштабу менше за 200%.

Цю команду меню можна використовувати й для відключення сітки. Панель інструментів також має піктограму, яка виконує такі самі дії.

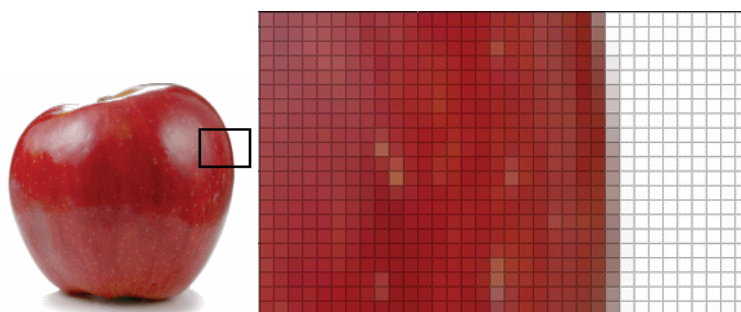


Рис. 222. Піксельна сітка

Лінійки – відображення або приховування лінійок уздовж верхнього та лівого країв полотна зображення (Рис. 223). Панель інструментів має відповідний значок.

Лінійки можуть допомогти під час вирівнювання елементів зображення, виконання вимірювань або коригування зображення до певного розміру. На лінійках завжди встановлюються поточні одиниці виміру.

Коли виділена область активна, прямокутник, що її обмежує, буде виділено на лінійках синім кольором, що спрощує вимір. Положення курсора також відображається на лінійках чорними рисками, що полегшує точне застосування інструментів.

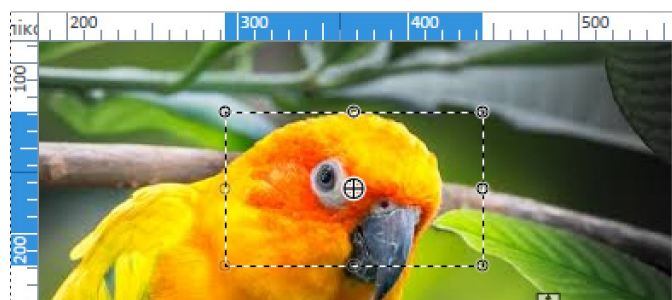


Рис. 223. Горизонтальна та вертикальна лінійки

Одиниці виміру: пікселі, дюйми та сантиметри – цей набір із трьох прапорців визначає, які одиниці виміру використовує Paint.NET.

Одиниця виміру за замовчуванням – пікселі.

ДОВІДКА

Доступ до меню **Довідка** (Рис. 224) можна отримати, клацнувши по піктограмі зі знаком питання на панелі Меню. Paint.NET надає швидкий доступ до довідкової документації, існує можливість надіслати побажання або повідомлення про помилку авторам програми, а також отримати доступ до різних онлайн-ресурсів і переглянути вікно **Про програму**.

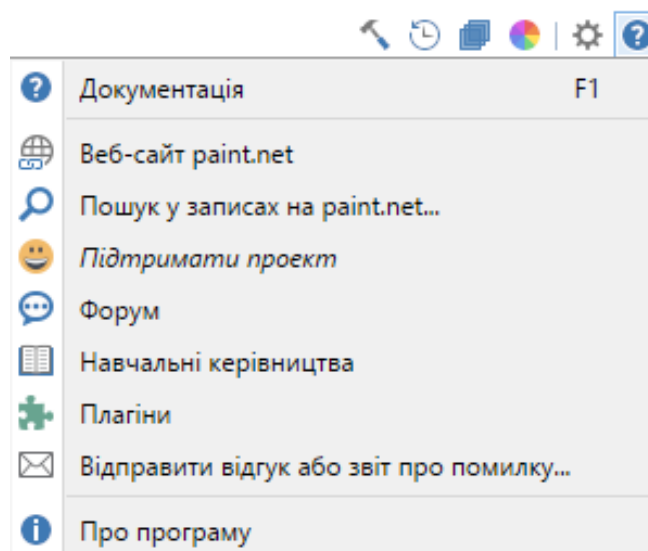


Рис. 224. Довідка Paint.NET

РОБОТА З ФАЙЛОМ ЗОБРАЖЕННЯ

Усі команди, призначені для роботи з файлом зображення, а саме: створення, відкриття, збереження файлів, друк зображень, доступ до списку файлів, з якими працювали нещодавно, зібрані у меню **Файл** (Рис. 225).

Ці команди схожі до аналогічних команд в інших програмах для редагування зображень і документів. Розглянемо їх детальніше.

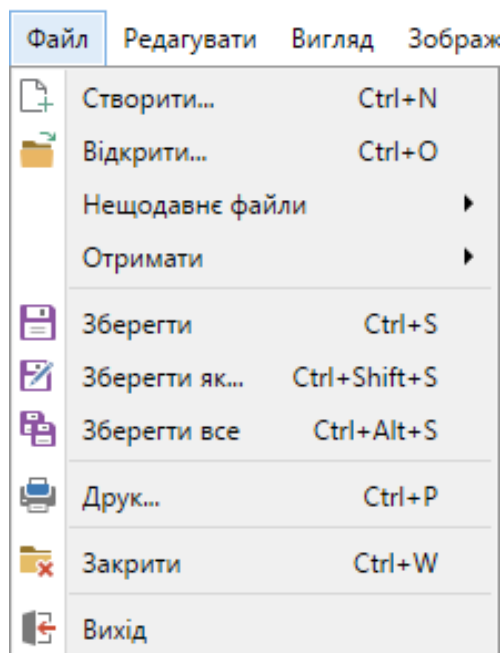


Рис. 225. Меню Файл

Створити, Ctrl + N – створення нового порожнього одношарового зображення. Єдиний шар буде заповнений білим кольором. Розмір полотна за замовчуванням становить 800×600 пікселів або матиме розмір зображення, що міститься в буфері обміну.

Після виконання цієї команди, відкриється вікно (Рис. 226), у якому ви зможете вручну змінити параметри зображення (ширину, висоту, роздільну здатність).

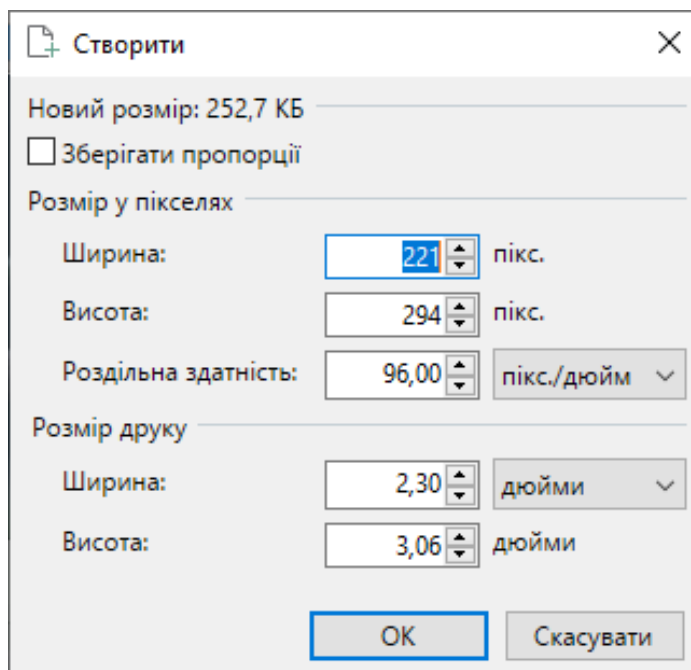


Рис. 226. Вікно Створити

Також у цьому вікні можна встановити прапорець **Зберігати пропорції**, щоб при зміні однієї сторони (ширини або висоти) інша сторона змінюватиметься автоматично, зберігаючи оригінальне співвідношення сторін.

У верхній частині вікна відображається розмір файлу зображення. Але зазвичай після збереження на жорсткому диску зображення займатиме менше місця. Це пов'язано із методом стиснення, що буде застосовано залежно від вибору формату.

Відкрити, Ctrl + O – відкриття наявного зображення для редагування. Відкрите зображення буде додано до списку зображень і відображено у вікні редагування.

Нещодавні файли – швидкий доступ до останніх десяти зображень, відкритих в Paint.NET. Кожне зображення супроводжується мініатюрою, що допомагає його ідентифікувати. Також тут зазначається формат зображення.

Команда **Очистити цей список** видаляє всі записи зі списку, що може бути корисним, якщо папка, де містяться зображення, була змінена або перейменована.

Якщо у цьому списку курсор навести на будь-яке із зображень, шлях до файлу відобразиться у виді підказки.

Отримати. Це підменю містить лише одну команду – **Зі сканера або камери**, яка дозволяє відкрити зображення, отримані зі сканера або камери, підключених до комп'ютера.

Зберегти, Ctrl + S – збереження зображення у файлі з поточним ім'ям. Для різних форматів зображень доступні різні параметри збереження, які можна встановити самостійно.

Якщо зображення раніше не зберігалось і якщо формат файлу, що зберігається, вимагає налаштування, то з'явиться вікно збереження файлу (Рис. 227). Якщо зображення ще не має імені (за промовчанням **Безіменний**), то потрібно дати йому ім'я і вказати формат зображення.

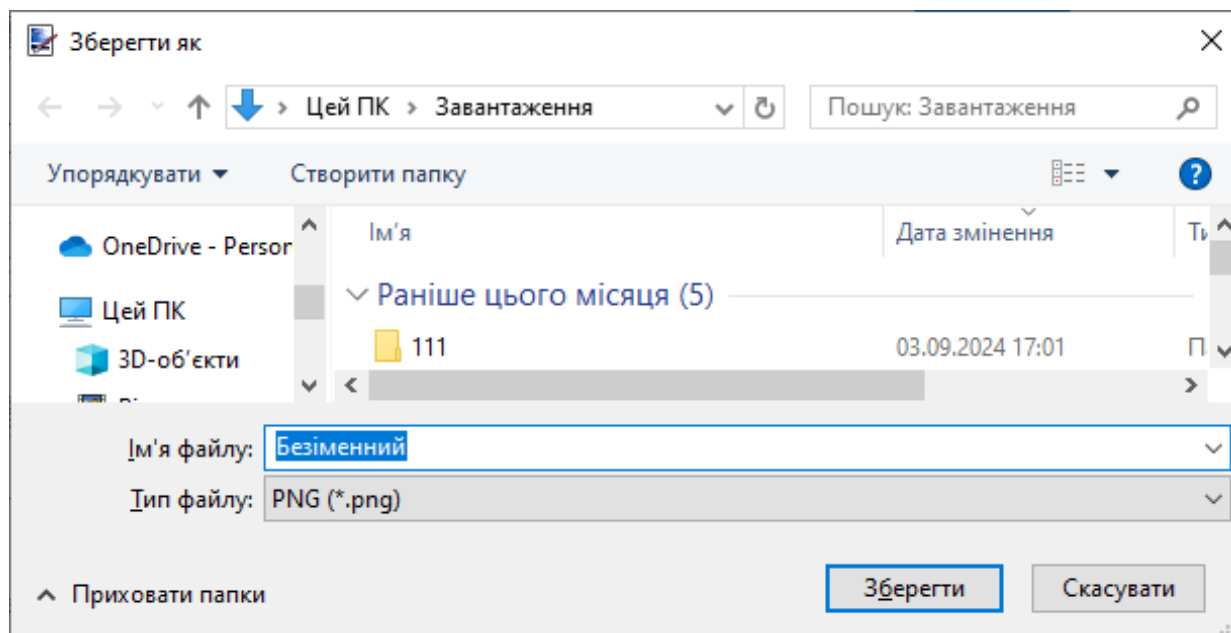


Рис. 227. Вікно Зберегти як

При збереженні зображень у форматі JPEG можна налаштувати якість зображення. Для зображень GIF представлені параметри прозорості та збудження. Зображення TGA можна налаштувати на збереження у 24- або 32-бітному форматі, зі стиском RLE або без нього. Для зображень у форматі PNG можна настроїти глибину кольору, рівень розмивання, поріг прозорості тощо. Водночас слід зазначити, що деякі параметри стають доступними лише у разі вибору певної глибини кольору.

Також у вікні вибору параметрів збереження зображення можна переглянути те, як воно виглядатиме в іншій програмі після збереження. Над сектором попереднього перегляду вказано розмір файлу.

Зберегти як, Ctrl + Shift + S – збереження із новим ім'ям, форматом, розташуванням або налаштуваннями. Це дуже корисно при створенні копії відредагованого зображення зі збереженням оригіналу.

Зберегти все, Ctrl + Alt + S – збереження всіх зображень, які було змінено з моменту відкриття. Наявне ім'я файлу, шлях, формат і налаштування будуть перезаписані. Якщо зміни роблять чинний формат непридатним (наприклад, додавання шару до JPG), з'явиться діалогове вікно конфігурації або попередження.

Друк, Ctrl + P – друк зображення із указаними параметрами.

Закрити, Ctrl + W. Команда закриває активне зображення. Якщо в зображенні були проведені зміни, і вони не були збережені, то з'явиться вікно, в якому потрібно буде вибрати бажану дію (зберегти, закрити без збереження або скасувати закриття).

Вихід. Команда для виходу з Paint.NET. Вас попросять зберегти зображення, якщо ви ще цього не зробили. Незалежно від кількості незбережених зображень, запит на підтвердження буде лише один.

ШАРИ

Для формування зображення Paint.NET використовує шари. Шари можна уявити, як стопку прозорих слайдів або аркушів. Кожен слайд містить окремий елемент, який є частиною загального зображення. Непрозорі ділянки на шарі, що лежить вище, закриватимуть ділянки, що містяться в шарах нижче.

Нові зображення завжди створюються з одним шаром, на ім'я **Фон**. Усі зображення мають щонайменше один шар.

Ярлики шарів, з яких складається зображення, можна побачити у **вікні Шари**, що відображається або приховується натисненням клавіші **F7** або кнопки **Шари** на панелі меню (Рис. 228) і переміщується за допомогою рядка заголовка.



Рис. 228. Іконка Шари на панелі Меню

У зображення завжди є лише один активний шар. Його можна визначити за синім виділенням. Операції малювання та редагування зображення можна здійснювати лише з активним шаром. Ніякий інший шар не буде змінений. Щоб зробити активним інший шар, достатньо клацнути на його ім'я у вікні **Шари** (Рис. 229).

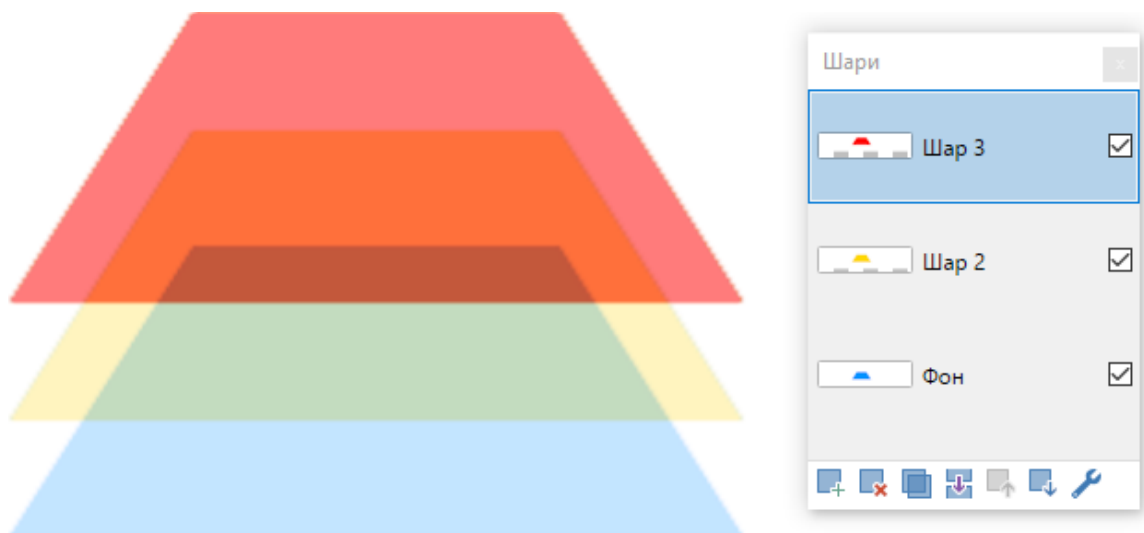


Рис. 229. Вікно керування шарами

У Paint.NET операції над шарами є важливим інструментом для редагування зображень. Шари дозволяють працювати з різними елементами зображення незалежно одне від одного, надаючи більше гнучкості при створенні та редагуванні.

Операції над шарами можна здійснювати за допомогою декількох способів: за допомогою команд меню **Шари** (Рис. 230–231), кнопками вікна **Шари** та комбінаціями клавіш.

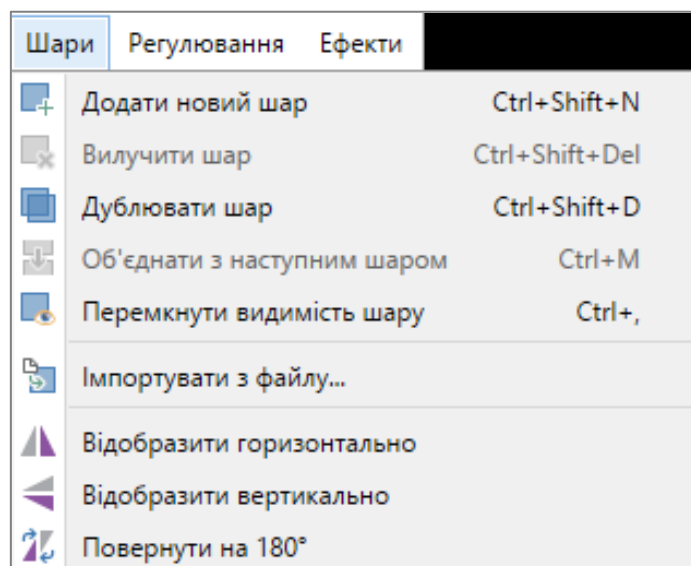


Рис. 230. Меню Шари

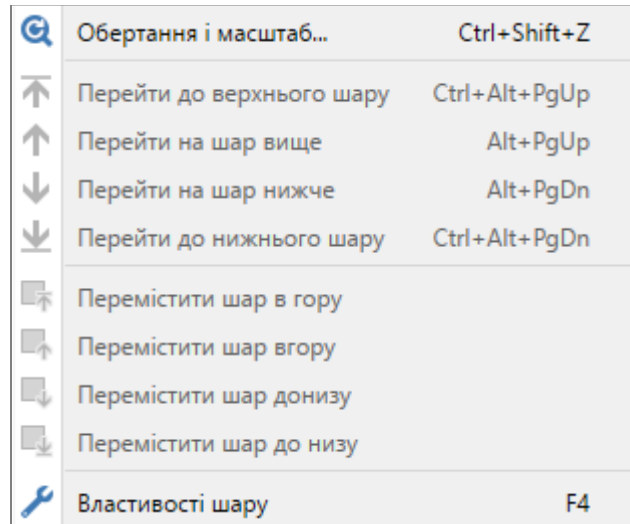


Рис. 231. Меню Шари (продовження)

Необхідно пам'ятати, що операції над шарами впливають лише на активний шар, вибраний у вікні **Шари**.

ДОДАВАННЯ ШАРУ

Додати новий, повністю прозорий шар до зображення безпосередньо над активним шаром можна за допомогою команди меню **Шари – Додати новий шар**, **Ctrl + Shift + N** або кнопкою вікна **Шари – Додати новий** (Рис. 232).

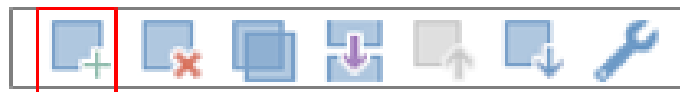


Рис. 232. Кнопка вікна Шари – Додати новий

Новий шар матиме ім'я, яке відповідатиме позиції шару, наприклад **Шар 2**. Ім'я шару можна змінити самостійно.

ВИЛУЧЕННЯ ШАРУ

Для видалення активного шару із зображення використовують команду меню **Шари – Вилучити шар**, **Ctrl + Shift + Del** або кнопку вікна **Шари – Вилучити шар** (Рис. 233).



Рис. 233. Кнопка вікна Шари – Вилучити шар

Перед видаленням вас попросять підтвердити команду.

Оскільки всі зображення повинні мати хоча б один шар, останній шар, що залишиться, видалити не можна.

ДУБЛЮВАННЯ ШАРУ

Щоб створити точну копію активного шару користуються командою меню **Шари – Дублювати шар**, **Ctrl + Shift + D** або кнопкою вікна **Шари – Дублювати шар** (Рис. 234).



Рис. 234. Кнопка вікна *Шари – Дублювати шар*

Будуть продубльовані всі властивості шару, що належать до пікселів. Новий шар матиме таке ж ім'я, що і вихідний, та знаходитись нижче за нього.

ОБ'ЄДНАННЯ ШАРІВ

Об'єднання активного шару із шаром, розташованим нижче, здійснюється командою меню **Об'єднати з наступним шаром**, **Ctrl + M** або відповідною кнопкою вікна **Шари** (Рис. 235).



Рис. 235. Кнопка вікна *Шари – Об'єднати з наступним шаром*

ПЕРЕХОДИ ПО ШАРАХ

Переходити по шарах можна за допомогою команд меню:

- **Перейти до верхнього шару**, **Ctrl + Alt + PgUp** – робить активним верхній шар в списку у вікні Шари.
- **Перейти на шар вище**, **Alt + PgUp** – робить активним шар безпосередньо над поточним.
- **Перейти на шар нижче**, **Alt + PgDn** – робить активним шар безпосередньо під поточним.
- **Перейти до нижнього шару**, **Ctrl + Alt + PgDn** – робить активним нижній шар в списку у вікні Шари.

Або, клацаючи мишкою по назві потрібного шару, у вікні **Шари**.

ПЕРЕМІЩЕННЯ ШАРІВ

Як зазначалося вище, порядок, у якому шари відображаються у вікні шарів, дуже важливий. Шари, розташовані вище в стеку, будуть приховувати (затуляти) шари, що розташовані нижче (якщо тільки у вищого рівня немає прозорих областей, тоді нижній шар буде видно). Розглянемо способи переміщення шарів.

Переміщення шарів за допомогою меню **Шари**:

- **Перемістити шар в гору** – переміщує активний шар в верхню позицію у вікні шарів.
- **Перемістити шар вгору** – переміщує активний шар на одну позицію вгору у списку в вікні **Шари**.
- **Перемістити шар донизу** – переміщує активний шар на одну позицію нижче в списку у вікні **Шари**.
- **Перемістити шар до низу** – переміщує активний шар у нижню позицію в списку у вікні **Шари**.

Переміщення шарів за допомогою кнопок вікна **Шари** (Рис. 236):

- **Перемістити шар догори**. З кожним клацанням переміщає активний шар на один рівень вище у списку шарів.
- **Перемістити шар донизу**. З кожним клацанням переміщає активний шар на один рівень нижче у списку шарів.



Рис. 236. Переміщення шарів за допомогою кнопок вікна **Шари**

- Швидко перемістити шар у верхню або нижню позицію стопки шарів можна за допомогою клавіші **Ctrl**. Утримуючи **Ctrl**, клацніть значок **Перемістити шар догори** (або **Перемістити шар донизу**), щоб відправити активний шар у верхню (або нижню) частину списку.

- Змінювати порядок шарів можна також за допомогою перетягування на нове місце у списку шарів.

ВИДИМІСТЬ ШАРІВ

Шари можна вмикати та вимикати. Поруч із кожним шаром у вікні **Шари** є прапорець, який контролює видимість шару. Цей прапорець також доступний у діалоговому вікні **Властивості шару**.

Також керувати видимістю шару можна за допомогою команди **Перемкнути видимість шару, Ctrl + ,** з меню **Шари**.

Для того, щоб невидимий шар залишився частиною зображення, його потрібно зберігати у форматі *.pdn.

ІМПОРТ У ВИГЛЯДІ НОВОГО ШАРУ

У Paint.NET можна імпортувати зображення для редагування, які будуть завантажені як один або кілька нових шарів. Для цього відкрийте меню **Шари** та виконайте команду **Імпорт із файлу....** Відкриється стандартне вікно відкриття файлу Windows.

ВІДОБРАЖЕННЯ, ПОВОРІТ І МАСШТАБ ШАРІВ

Команда **Обертання і масштаб, Ctrl + Shift + F**, що входить у меню **Шари**, дає змогу вільно повертати зображення по трьох осях: X, Y, Z, нахилити чи масштабувати зображення на активному шарі.

У вікні цієї команди (Рис. 237) є такі області, в яких можна регулювати параметри зображення:

Нахил і поворот. За допомогою цих повзунків користувач може повертати зображення навколо різних осей:

- X-вісь – обертання навколо горизонтальної осі.
- Y-вісь – обертання навколо вертикальної осі.
- Z-вісь – обертання об'єкта на площині.

Зліва від повзунків знаходиться кулька «глобус», що дозволяє повертати зображення навколо осей одночасно.



Рис. 237. Вікно Обертання і масштаб

Навігація. Ці повзунки дозволяють зсувати вміст шару вліво / вправо (по осі X) або вгору / вниз (по осі Y) в межах та за межі області, обмеженої розмірами полотна.

Масштаб. Повзунок дає змогу «наближати» й «віддаляти» зображення так, щоб сфокусувати вашу увагу на потрібній області. Переміщення повзунка праворуч збільшує об'єкт, а ліворуч – зменшує. Також можна вручну ввести значення масштабу в полі поруч із повзунком для точнішого налаштування.

Якість. Цей повзунок дає змогу вибрати рівень інтерполяції (якості) для трансформації зображення. Є кілька опцій для вибору, таких як:

- **Низька якість** – швидше виконання операції, але менш точне згладжування зображення.
- **Середня якість** – компроміс між швидкістю та якістю.
- **Висока якість** – найбільш точне перетворення з кращим згладжуванням, але може зайняти більше часу.

Залежно від ваших потреб, ви можете регулювати цей повзунок для досягнення балансу між швидкістю виконання та кінцевою якістю зображення.

Режим плитки. Зображення повторюється по всіх напрямках у режимі простого повтору або дзеркального.

Вибірка. Під час обертання та масштабування важливо враховувати налаштування **інтерполяції** (вибірки), яке впливає на кінцеву якість зображення після трансформацій:

- **Метод сусідніх пікселів** – найшвидший метод, але з грубими краями та низькою якістю при масштабуванні. Підходить для зображень із низькою деталізацією або піксельною графікою.
- **Білінійна** – більш згладжене, підходить для звичайних фотографій. Підвищує якість, зменшуючи «шум» при зміні розмірів або обертанні.
- **Бікубічна** – найкраща якість для фотографій. Додає гладкості, передусім на великих зображеннях при масштабуванні.

Окрім команди **Обертання і масштаб**, у меню **Шари** наявні ще три команди для швидкого редагування шару:

Відобразити горизонтально / вертикально – відображення активного шару відносно уявної вертикальної / горизонтальної осі.

Повернути на 180° – поворот активного шару на 180 градусів.

Обертання і масштаб, Ctrl + Shift + Z – обертання, панорамування, збільшення або зменшення активного шару.

НАЛАШТУВАННЯ ШАРУ

Властивості шару, F4 – встановлення режиму змішування поточного шару із шаром, який розташований під ним у списку вікна **Шари** (Рис. 238). Також у діалоговому вікні, що відкриється, можна задати ім'я шару, його видимість, режим змішування та рівень непрозорості.



*Рис. 238. Встановлення режиму змішування поточного шару із попереднім у списку вікна **Шари***

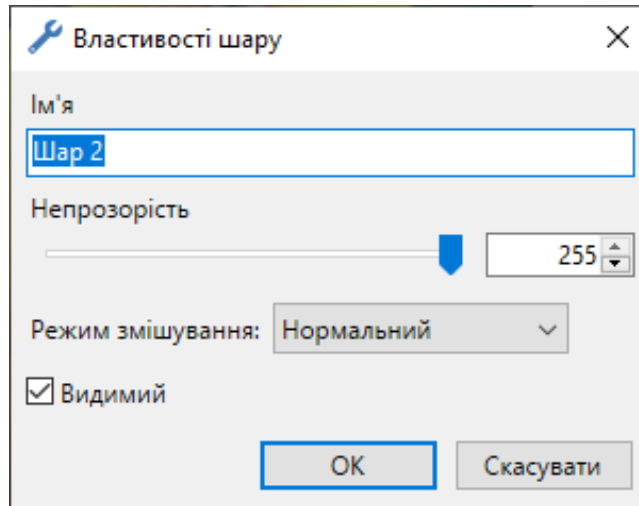


*Рис. 239. Встановлення режиму змішування поточного шару
із попереднім Прозорість шарів*

Кожен шар Paint.Net складається з пікселів, які збережені у форматі RGBA. Частина аббревіатури «RGB» позначає кольори (червоний, зелений і синій). Частина «А» (Альфа) позначає змінну для зберігання інформації про прозорість пікселя. Альфа може набувати значення від 0 (повністю прозорий) до 255 (повністю непрозорий).

Якщо піксель прозорий, то замість його кольору на полотні буде показаний колір пікселя шару, що розташований нижче.

Оскільки, прозорі пікселі не можуть відображатися на моніторі, щоб якось вказати на прозорість шару, редактор використовує тло, що нагадує зображення шахової дошки.

Якщо ви бачите такий фон, це означає, що частина зображення прозора. Зображення шахівниці не є частиною зображення.

Якщо ви зберігаєте зображення, то при його перегляді або використанні в іншій програмі ніякої «шахової дошки» там не буде (якщо інша програма також не використовує «шахівницю» для позначення прозорості).

Аналогічно до пікселів, шари також мають значення непрозорості (Рис. 240).

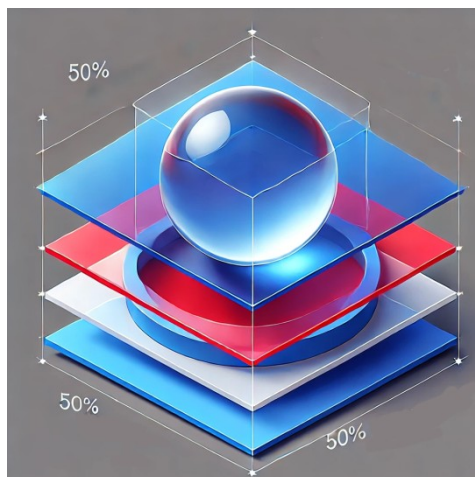


Рис. 240. Демонстрація прозорості шарів

РЕЖИМИ ЗМІШУВАННЯ

Режим змішування визначає, як шар поєднується з нижніми шарами під час складання зображення.

Як і прозорість, режими змішування застосовуються до кожного пікселя у шарі.

Щоб змінити режим змішування шару, виділіть його у вікні шарів і натисніть

F4. Відкриється діалогове вікно властивостей шару. Новий режим змішування можна вибрати зі списку, що розкривається (Рис. 241).

Усього доступно 14 режимів змішування. Не всі назви режимів змішування шарів інтуїтивно зрозумілі, тому в будь-якому разі рекомендується експериментувати. Кожен режим змішування, описаний нижче, застосований до двох шарів із рівнем непрозорості, встановленим на позначці 255.

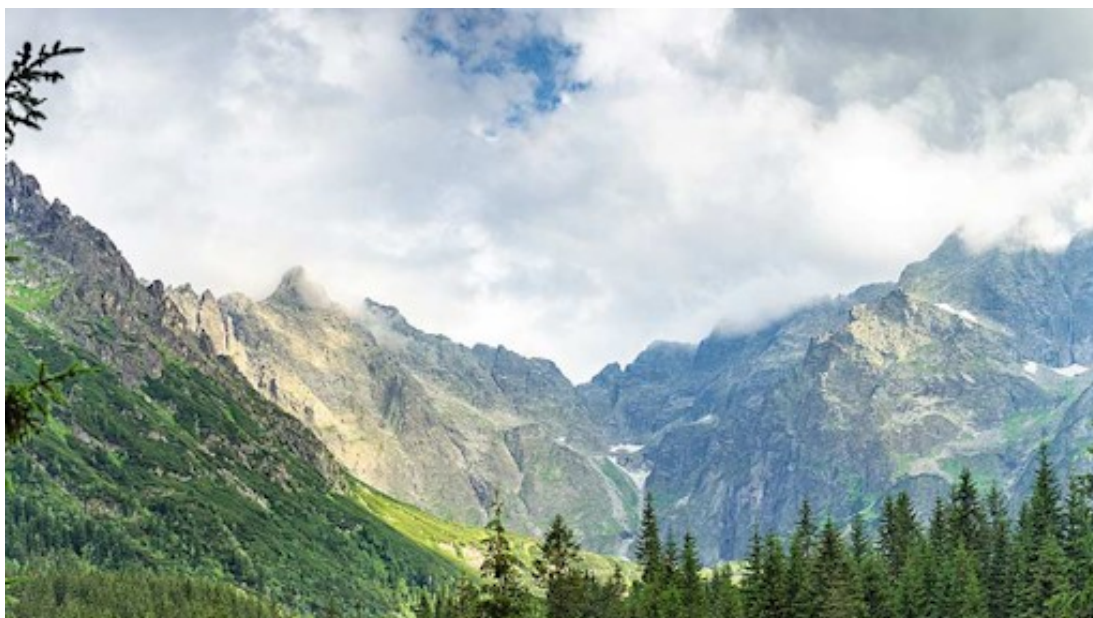


Рис. 243. Вихідне зображення 2

Нормальний. Стандартний режим, що використовується за замовчуванням. Кожен піксель у шарі поєднується з композицією залежно від значення прозорості. Якщо верхній шар повністю непрозорий, він закриває собою нижній шар. При зменшенні прозорості верхнього шару починає проступати нижній шар (Рис. 244).

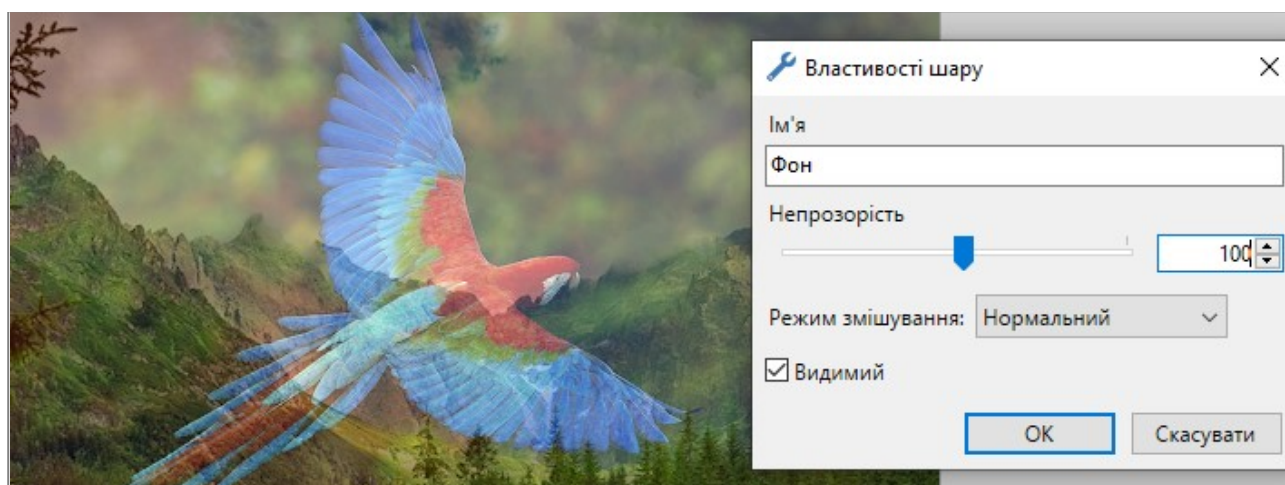


Рис. 244. Режим змішування Нормальний

Множення. Цей режим використовується для затемнення зображення, накладаючи колір одного шару на інший. У результаті зображення стає темнішим. При множенні білого з іншим кольором він не змінюється (Рис. 245).

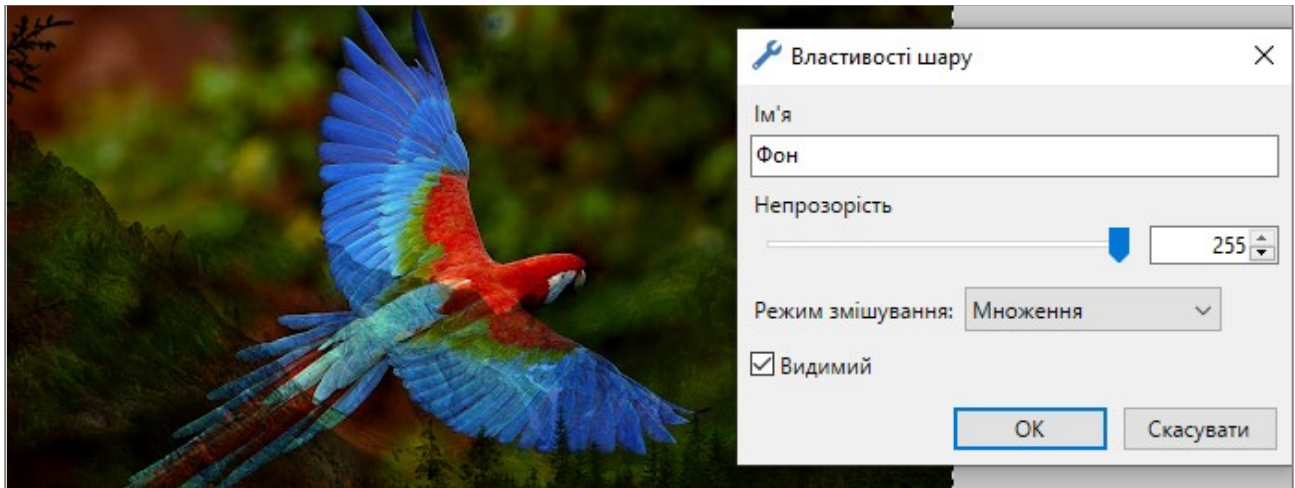


Рис. 245. Режим змішування Множення

Додавання. Інтенсивність кольору пікселів обох шарів складаються разом. Композиція завжди яскравіша, за винятком повністю чорних пікселів зображень (Рис. 246).

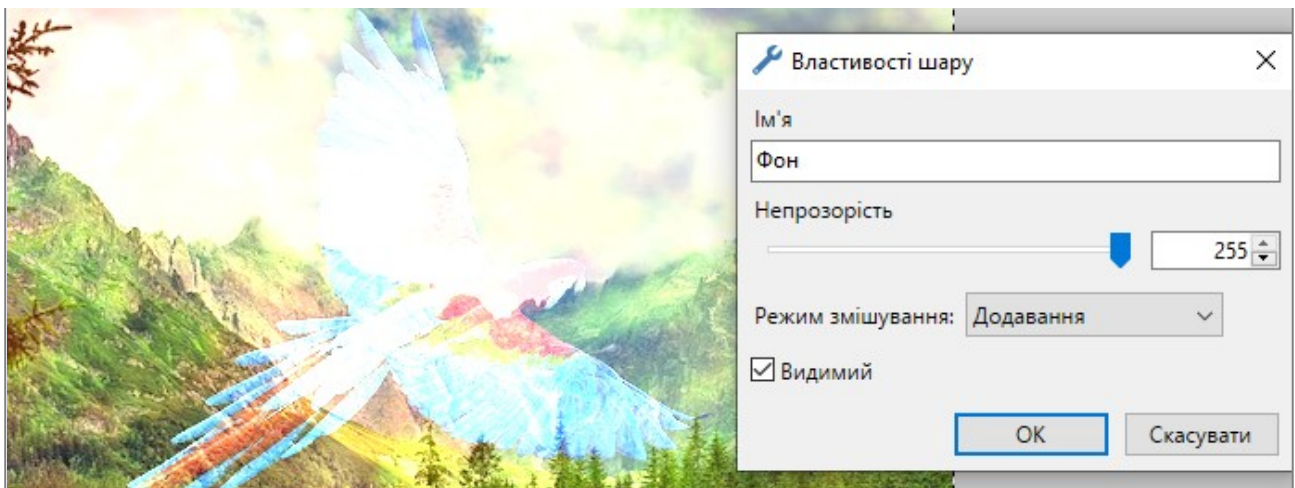


Рис. 246. Режим змішування Додавання

Затемнення основи. Цей режим змішування дає змогу робити темні пікселі темнішими, а світліші пікселі змішуються з іншими світлими пікселями, щоб залишатися яскравими (Рис. 247).

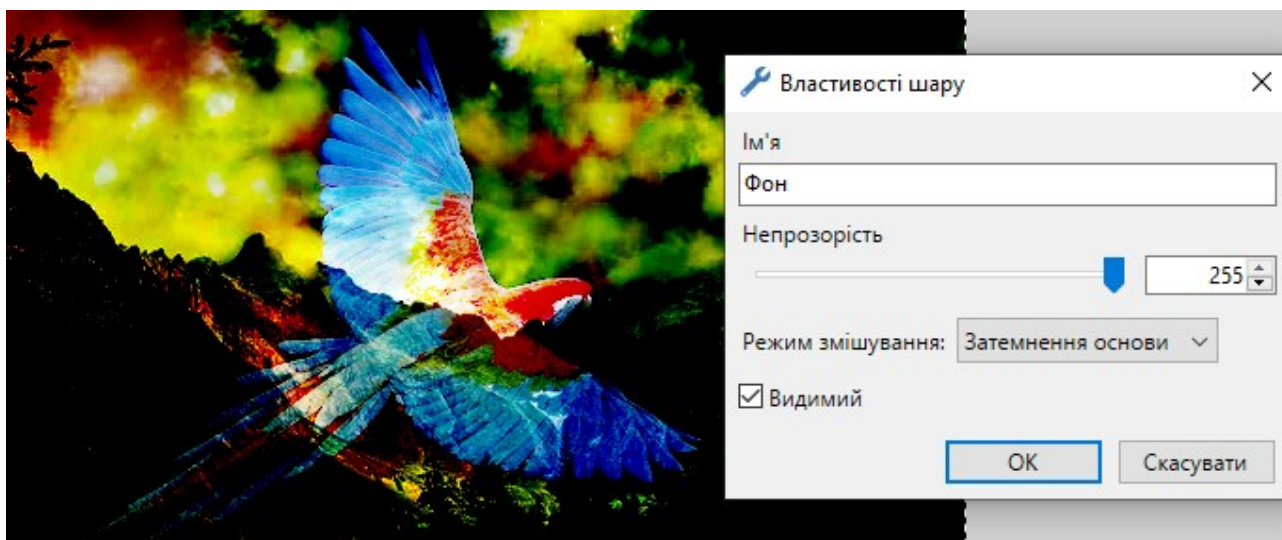


Рис. 247. Режим змішування Затемнення основи

Освітлення основи. Режим протилежний до попереднього – нижнє зображення «вигоряє» під впливом верхнього. У разі використання цього режиму світлі ділянки верхнього шару посилюють яскравість нижнього шару. Темні ділянки не мають жодного впливу. Тобто найбільші зміни відбуваються у бік білого кольору (Рис. 248).

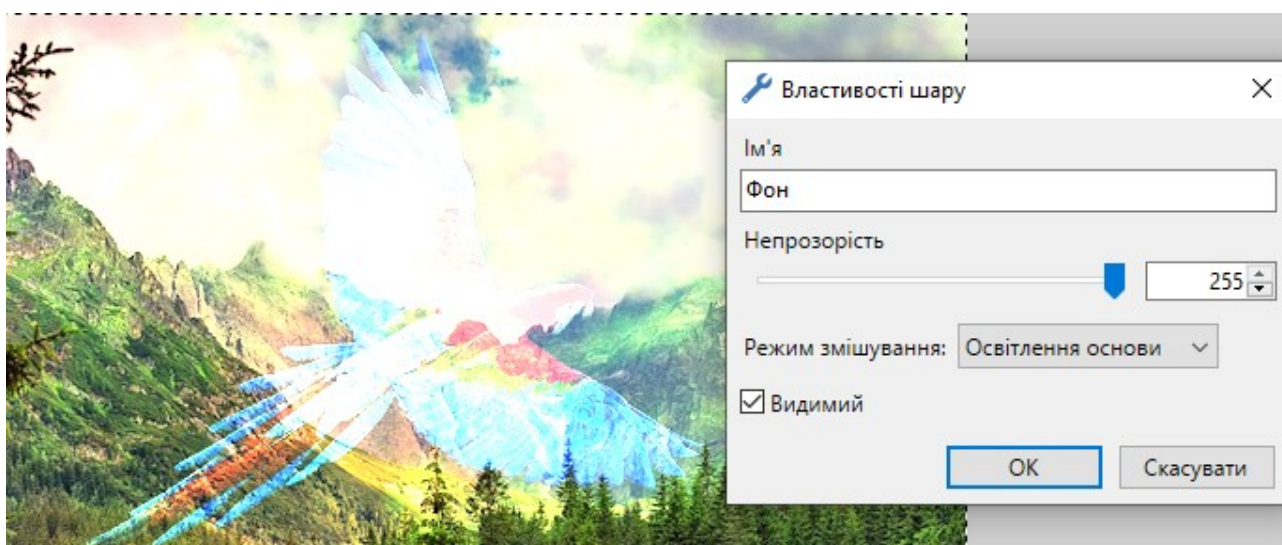


Рис. 248. Режим змішування Освітлення основи

Відбиття. Цей режим змішування можна використовувати для надання блиску світлим об'єктам або ділянкам (Рис. 249).

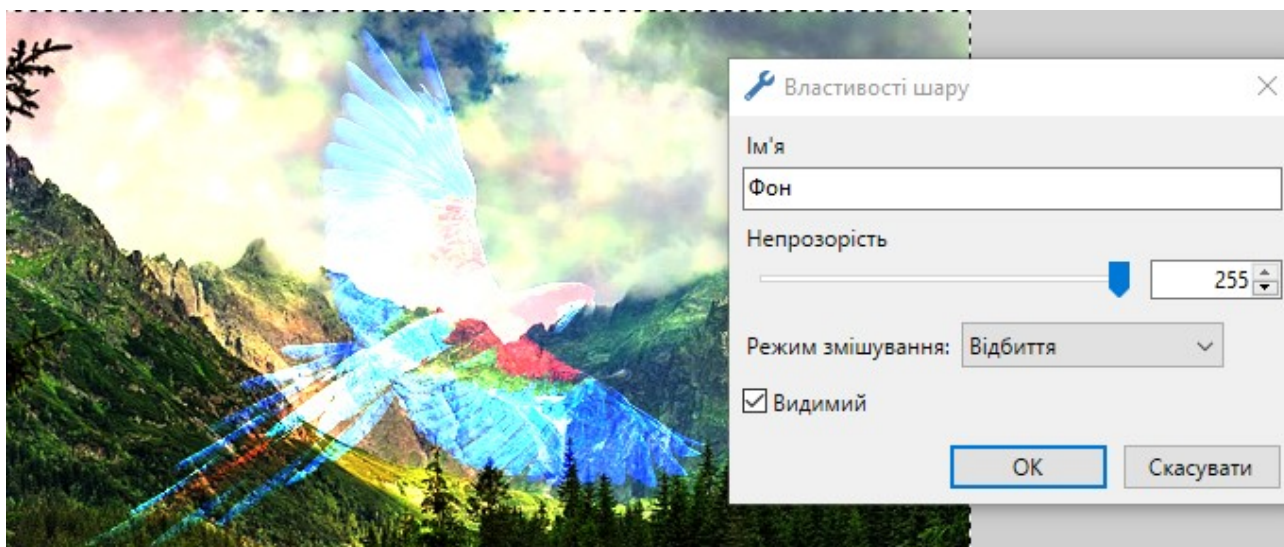


Рис. 249. Режим змішування Відбиття

Свічення. Режим створює ефект світіння, роблячи пікселі яскравішими та з'являється ілюзія світла, яке виходить з певних ділянок зображення. Цей ефект може бути особливо корисним для підсвічування тексту, створення атмосферних фонів або акцентування на певних елементах зображення (Рис. 250).

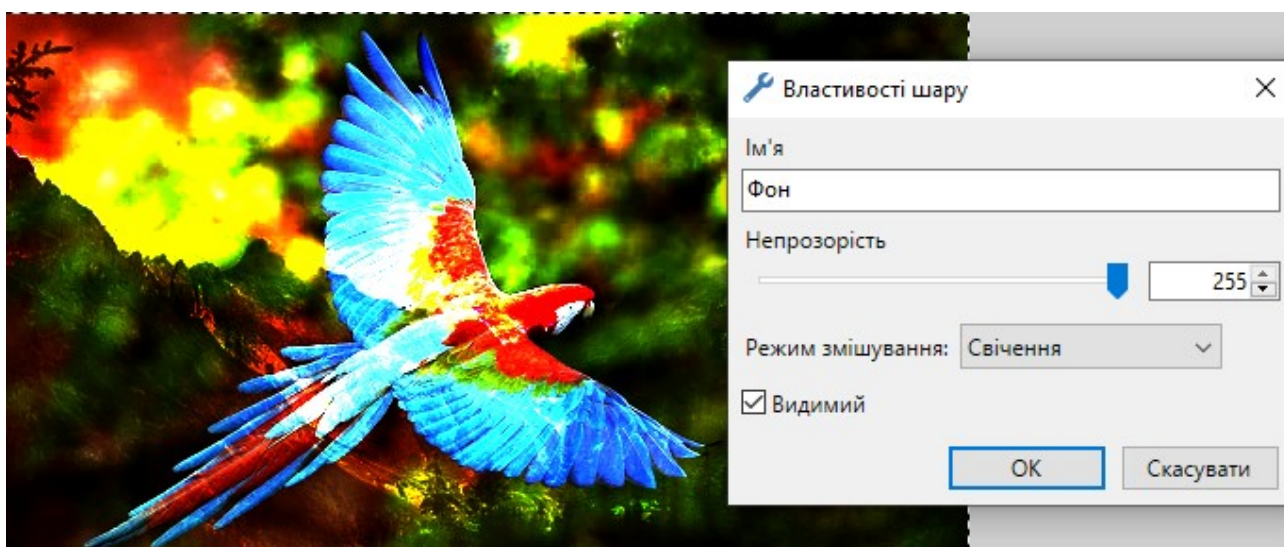


Рис. 250. Режим змішування Свічення

Різниця. Віднімає значення пікселів одного шару від іншого, що створює ефект інверсії кольорів. Якщо піксель на верхньому шарі білий, піксель на нижньому шарі інвертується. Якщо піксель на верхньому шарі чорний, піксель на нижньому шарі не змінюється. Якщо піксель на верхньому шарі збігається з нижнім, результат – чорний піксель. Тобто кольори, що збігаються, будуть чорними. Фрагменти, що не збігаються, будуть кольоровими (Рис. 251).

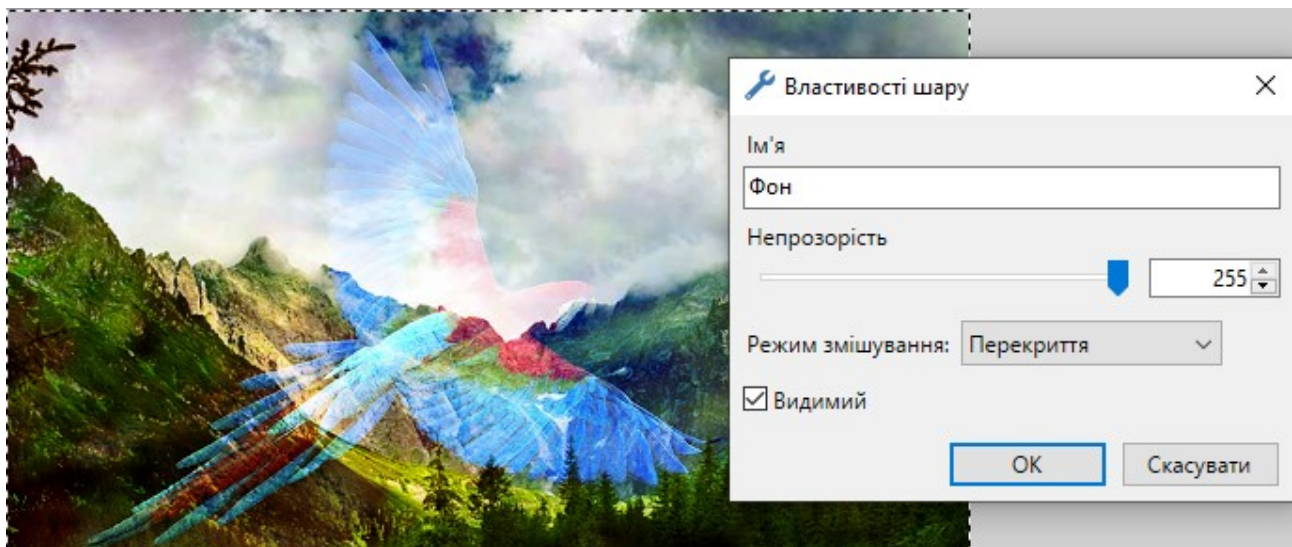


Рис. 251. Режим змішування Різниця

Заперечення. Цей режим використовується для створення ефектів, основаних на відмінностях між двома шарами. Коли ви застосовуєте цей режим, то кольори верхнього шару віднімаються від кольорів нижнього шару. Може бути корисним для створення артефактів, а також для корекції кольору в деяких випадках (Рис. 252).

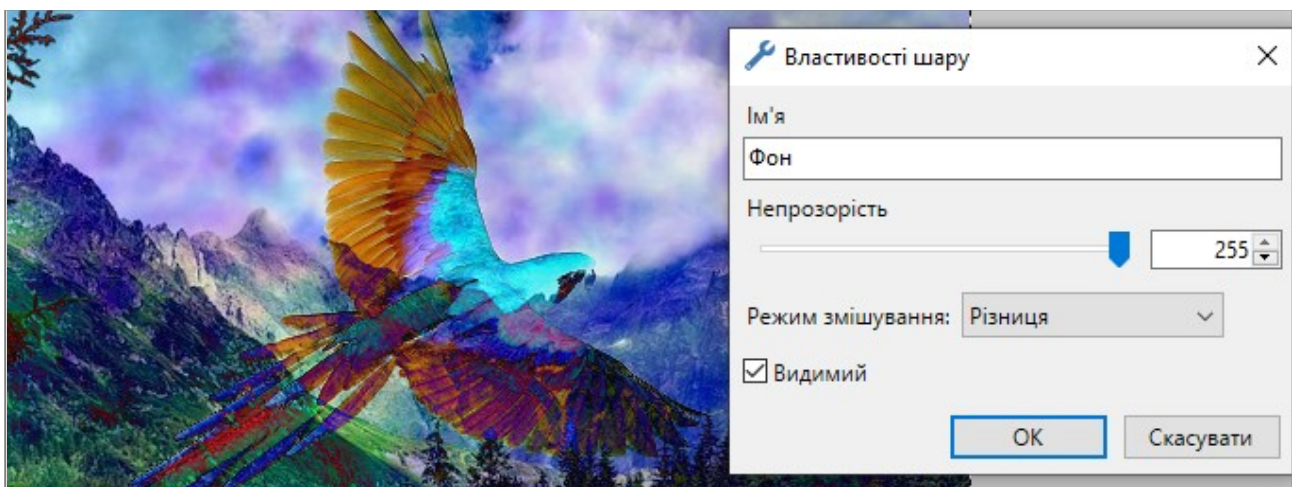


Рис. 252. Режим змішування Заперечення

Перекриття. Залежно від інтенсивності кольору пікселів шару застосовується режим **М'яке світло** (для темних кольорів) або **Затемнення** (для світлих). Цей режим надає зображенню насиченості та контрастності (Рис. 253).

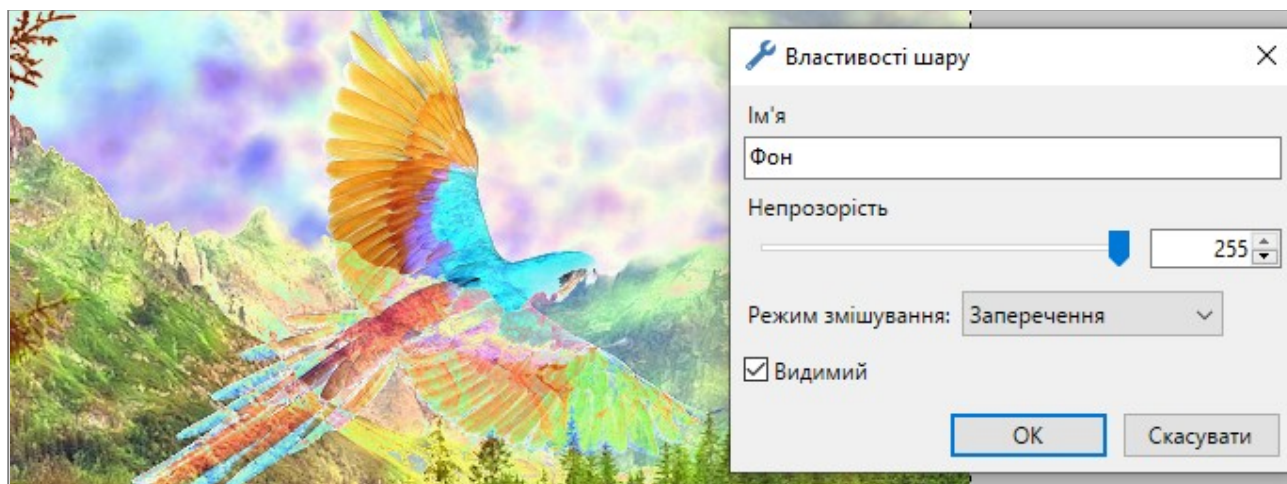


Рис. 253. Режим змішування Перекриття

Заміна світлим. Два шари поєднуються так, щоб світліші пікселі з верхнього шару замінювали темніші пікселі з нижнього шару. Це корисно для освітлення зображення або додавання ефектів освітлення (Рис. 254).

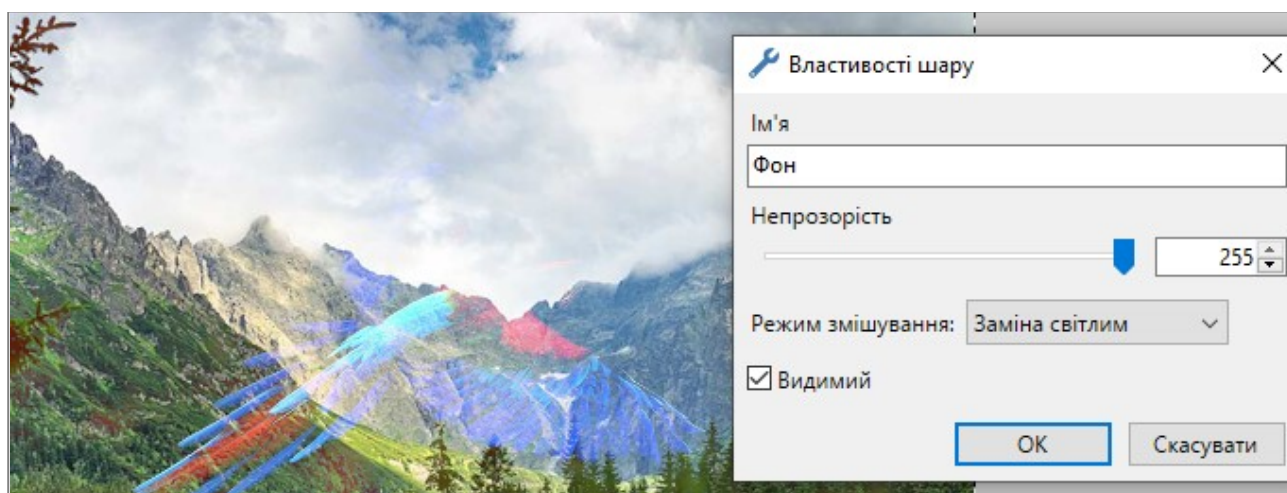


Рис. 254. Режим змішування Заміна світлим

Заміна темним. У цьому режимі пікселі шару поміщаються в фінальне зображення тільки в тому випадку, якщо вони темніші за відповідні пікселі іншого шару (Рис. 255).

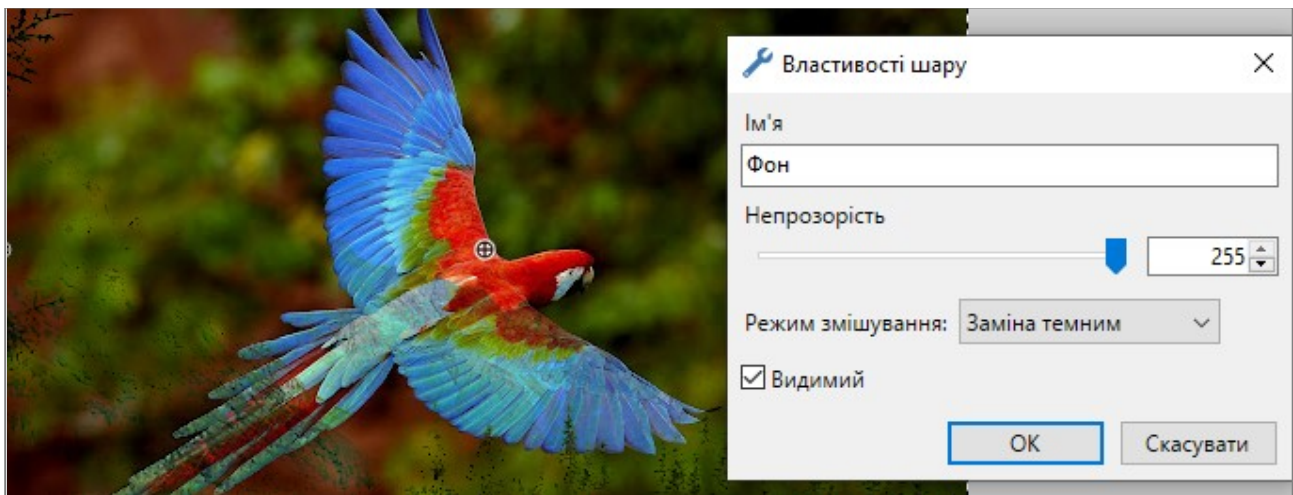


Рис. 255. Режим змішування Заміна темним

Екранне освітлення. Протилежний режиму **Множення** у тому сенсі, що пікселі верхнього шару освітлюють пікселі нижнього шару. В результаті картинка посвітлішає, начебто ми проєктували її за допомогою двох діaproекторів. Використовується для освітлення зображення, зберігаючи водночас кольори (Рис. 256).

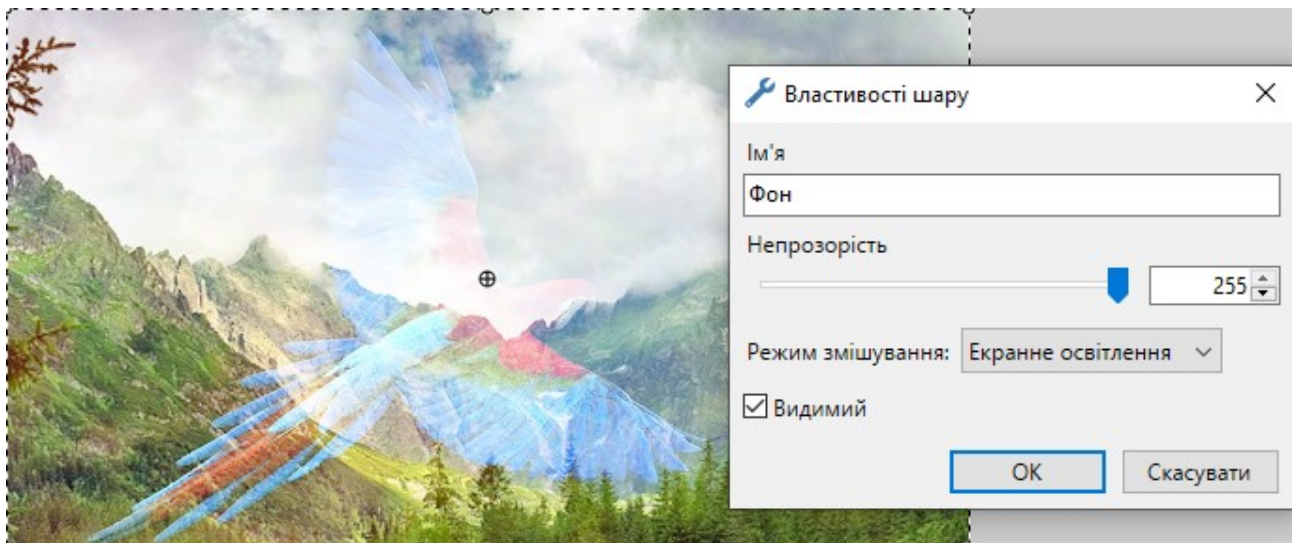


Рис. 256. Режим змішування Екранне освітлення

Xor (Виняток). Цей режим використовують зазвичай для аналізу зображення, а не для його обробки. Пікселі, які збігаються за кольором, стають прозорими, а інші – отримують новий колір, у результаті чого створюється унікальний ефект (рис. 257).

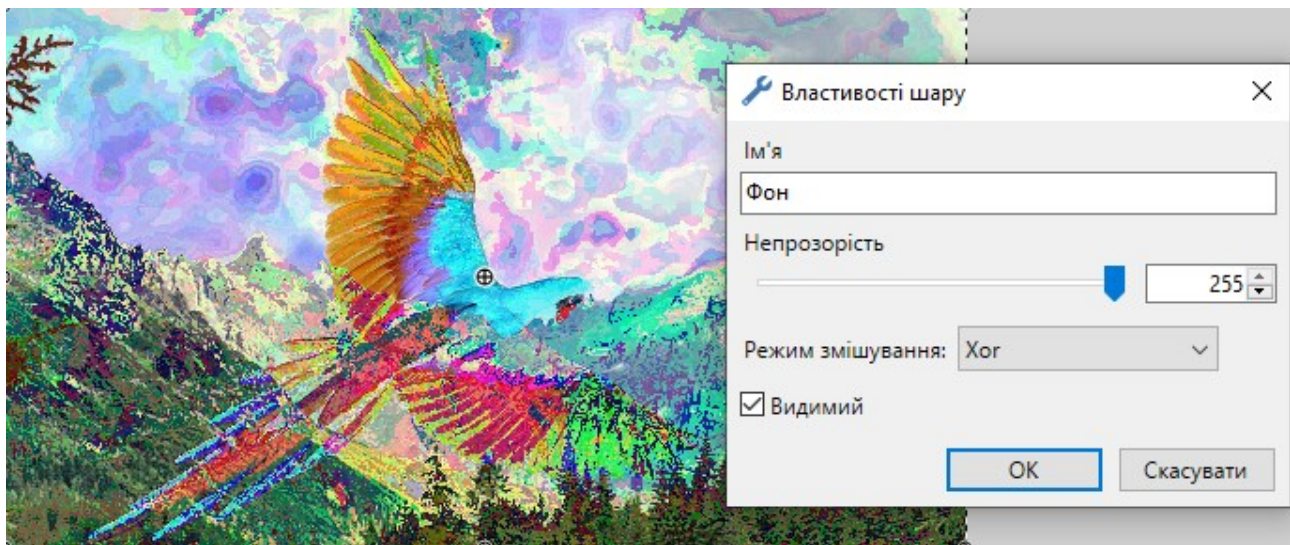


Рис. 257. Режим змішування Хор

Перезаписати. У цьому режимі будь-який новий намальований піксель повністю замінює будь-який піксель, який уже був у цьому місці активного шару. Відмінності можна побачити нижче, де жирну лінію було намальовано двічі, кожного разу зі значенням Alpha 128 (приблизно 50% прозорості) (Рис. 258).



Рис. 258. Режим змішування Перезаписати

Лінія зверху була намальована під час використання режиму **Нормальний**. Лінія видима там, де її було проведено.

Лінія знизу намальована за допомогою режиму **Перезаписати**. Напівпрозорі пікселі повністю замінюють наявні пікселі, що означає, що птаха більше не видно там, де намальована лінія. Цей режим може бути особливо корисним з інструментом **Олівець** під час редагування зображень на рівні пікселів.

ОСНОВНІ ІНСТРУМЕНТИ ТА ФУНКЦІЇ

ПАНЕЛЬ ІНСТРУМЕНТІВ

Панель інструментів (Рис. 259) містить кнопки для доступу до багатьох найуживаніших команд для редагування зображень.

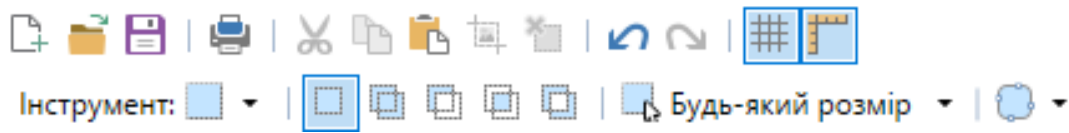


Рис. 259. Панель інструментів Paint.Net

Найуживаніші команди для роботи із зображенням зібрані у верхній частині панелі.

Зліва направо: **Створити, Відкрити, Зберегти, Друк, Вирізати, Копіювати, Вставити, Обрізати, Зняти виділення, Скасувати та Повторити**. Ці дії також можна знайти в меню **Файл і Редагувати**.

Наступні дві кнопки на Панелі інструментів **Відобразити сітку при збільшенні** та **Лінійки** перемикають видимість піксельної сітки та лінійки. Дія кнопок аналогічні до дії відповідних команд меню **Вигляд** і були розглянуті у розділі [Налаштування робочої області](#).

Другий рядок Панелі інструментів починається зі списку, в якому можна вибрати інструменти Paint.Net. Цей елемент керування повністю дублює плаваюче вікно **Інструменти** (Рис. 260).

Кнопки, що розташовані після списку **Інструмент**, призначені для встановлення налаштувань активного інструменту. Детально вони описані у розділі [Інструменти](#).

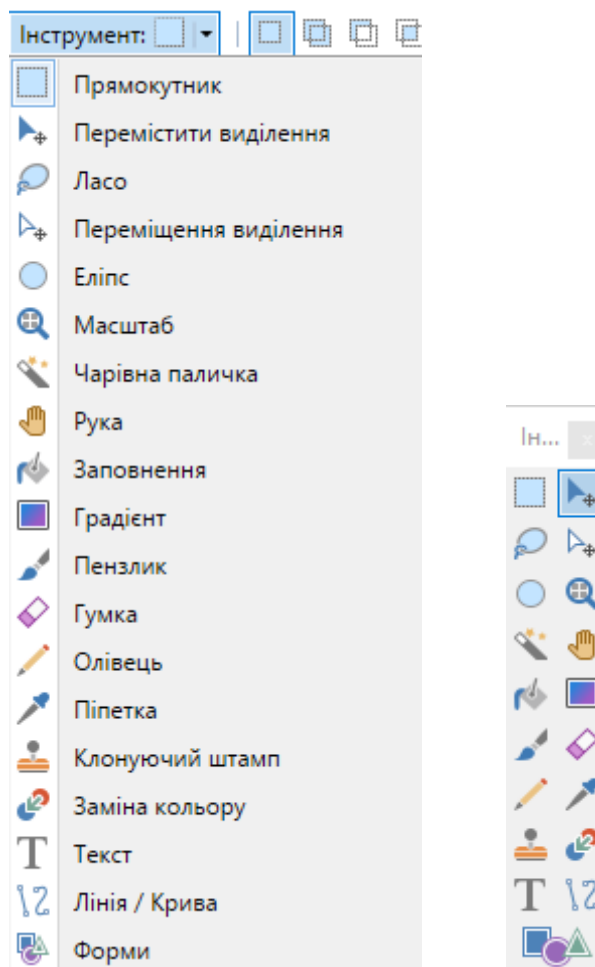


Рис. 260. Список інструментів Paint.Net

Одна або всі кнопки останнього блоку Панелі інструментів, використовується у багатьох інструментах Paint.NET для налаштування того, як відбуватиметься попиксельна обробка того чи іншого процесу (Рис. 261).

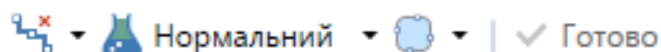


Рис. 261. Блок по піксельної обробки процесів

Згладжування – це функція допомагає зробити краї об’єктів більш плавними й менш пікселізованими (Рис. 262).

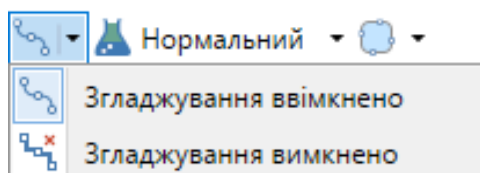


Рис. 262. Вибір параметрів згладжування

Якщо згладжування ввімкнено, краї траєкторії інструмента будуть плавнішими (перший малюнок). Якщо вимкнено, краї траєкторії інструмента

будуть прив'язані до найближчого пікселя. Це призведе до появи нерівностей чи сходів.



Рис. 263. Увімкнутий і вимкнутий режим згладжування ліній

Режими змішування – цей елемент керує тим, як кольори накладаються один на одного при малюванні або редагуванні зображень (Рис. 263). Режими змішування інструментів подібні до режимів змішування шарів і розглянуті у розділі Налаштування шару → [Режими змішування](#).

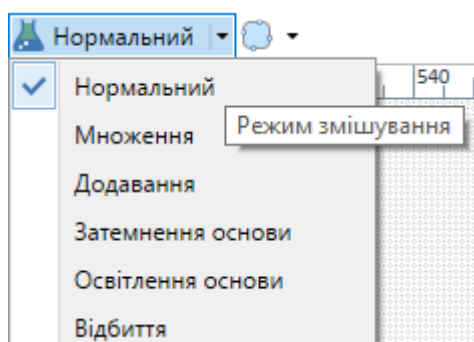


Рис. 263. Режими змішування

Вибір якості – визначає, чи активний вибір повинен бути згладженим (піксельним) або без згладжування (Рис. 264).

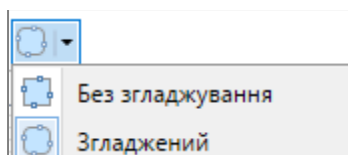


Рис. 264. Вибір якості зображення

Якщо вибрано якість піксельного виділення **Без згладжування**, межа виділення буде прив'язана до найближчого пікселя, що призведе до появи нерівностей або сходів. Якщо вибрано якість виділення **Згладжений**, межа виділення буде більш гладкою.

Кнопка **Готово** – деякі інструмент та параметри редагування залишаються активними, доки не буде вибрано елемент керування **Готово**, що фіксує форму, текст, інструмент чи операцію на полотні та виходить із режиму редагування.

Усі інструменти редактора Paint.NET можна згрупувати у 7 категорій, що подані нижче.

ІНСТРУМЕНТИ ВИДІЛЕННЯ

У графічному редакторі Paint.NET є чотири інструменти виділення: **Прямокутник**, **Ласо**, **Еліпс** і **Чарівна паличка**, що призначені для виділення області зображення для подальшої модифікації, видалення, копіювання або кадрування (Рис. 265).

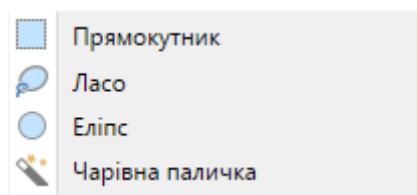


Рис. 265. Інструменти виділення

Розглянемо можливості та особливості кожного з них окремо.

ПРЯМОКУТНИК

Прямокутник, **S** використовується для визначення прямокутного чи квадратного виділення (Рис. 266).

Виділена область на зображенні обмежена тонкою пунктирною лінією та підсвічується синім кольором. Після перемикання на інший інструмент це підсвічування зникає.

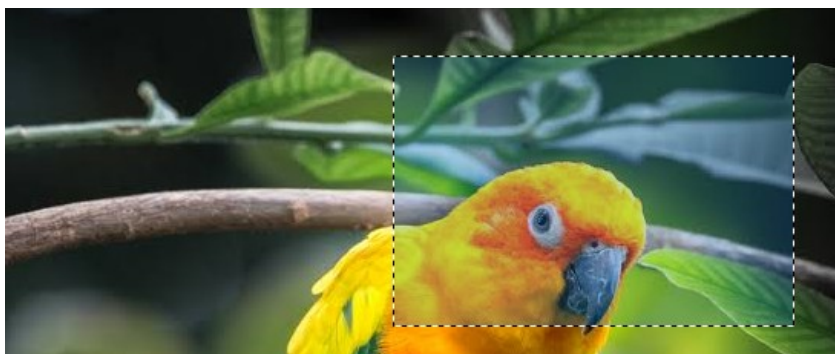


Рис. 266. Прямокутне виділення

Якщо на зображенні є одна або декілька виділених областей, всі операції малювання (а також ефекти й регулювання) будуть впливати тільки на ці виділені області.

Створення прямокутного виділення

Щоб створити прямокутне виділення, натисніть на ліву клавішу мишки та, не відпускаючи її, перетягніть вказівник так, щоб створити виділення.

Квадратне виділення створюється при натиснутій клавіші **Shift**.

Видалити поточне виділення можна або командою **Скасувати виділення** меню **Редагувати**, **Ctrl+D**, або **Esc**.

Режими вибору прямокутної області

Є три режими (Рис. 267), які визначають розмір прямокутного виділення при його витягуванні:

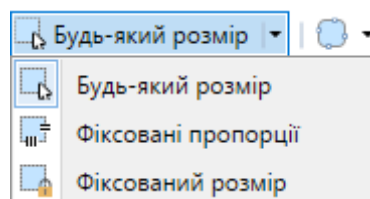


Рис. 267. Вибір режиму прямокутного виділення

Будь-який розмір. Висота та ширина прямокутного виділення не обмежені.

Фіксовані пропорції. Висота і ширина прямокутного виділення фіксуються у зазначеному співвідношенні при створенні виділення (Рис. 268).

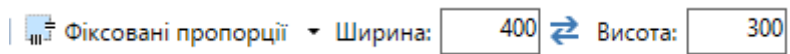


Рис. 268. Встановлення фіксованих пропорцій виділення

Щоб змінити співвідношення, введіть нові значення у текстові поля. Можна використовувати цілі та десяткові числа.

Фіксований розмір. Висота і ширина прямокутного виділення фіксуються до вказаних розмірів. Перетягування виділення призводить до переміщення місця виділення (Рис. 269).

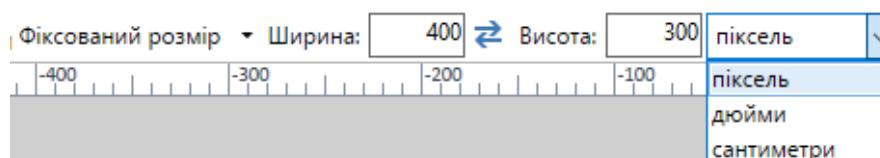


Рис. 269. Встановлення фіксованих розмірів виділення

Щоб змінити висоту та ширину, клацніть на текстове поле та введіть нове число. Можна використовувати цілі та десяткові числа. Одиниці вимірювання

можна змінити зі списку, що розкривається. Підтримуються пікселі, дюйми та сантиметри.

ЛАСО

Ласо, **S**, **S** використовується для створення виділення довільної форми (Рис. 270). Край виділеної області слідуватиме за курсором миші, коли він переміщається по полотну. Фігура автоматично закрийється прямою лінією від поточного розташування курсора до початкової точки.



Рис. 270. Виділення інструментом Ласо

ЕЛІПС

Еліпс, **S**, **S**, **S** створює еліптичні або круглі виділення (Рис. 271). Натисніть, утримуйте та перетягніть, щоб створити виділення.

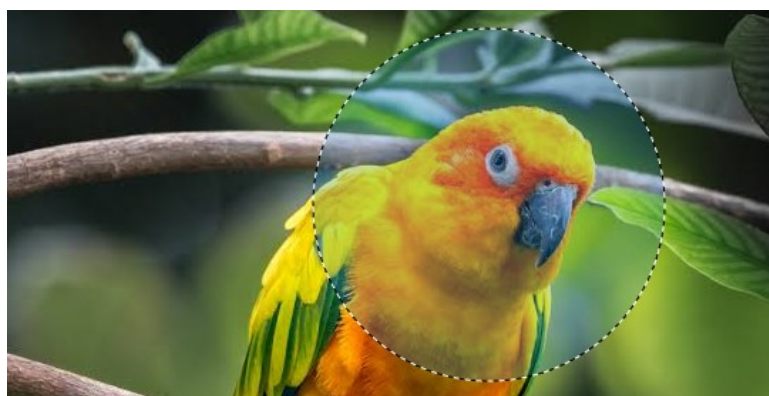


Рис. 271. Виділення у формі еліпса

Кнопка **Shift** змушує виділення залишатися круглим при зміні розміру. Якщо використовується клавіша **Shift**, діаметр кола вибору визначається початковою точкою клацання та поточним положенням вказівника.

ЧАРІВНА ПАЛИЧКА

Інструмент **Чарівна паличка**, **S, S, S, S** дає змогу вибрати ділянки зображення зі схожим кольором аналогічно до того, як інші інструменти виділення дозволяють вибрати ділянки заданої форми. Однак, на відміну від інших інструментів виділення, **Чарівна паличка** бере до уваги такі фактори, як фактичний колір зображення та поточне значення чутливості. Зазвичай все, що потрібно для вибору потрібної області зображення, це просто клацнути по ній.

Чарівну паличку можна розглядати як аналог інструмента **Заливання**, за винятком, що замість того, щоб заповнити область кольором, вона вибирає саму область.

РЕЖИМИ ЗАПОВНЕННЯ

Чарівна паличка використовує два режими заповнення (Рис. 272): **Безперервний** і **Глобальний**.

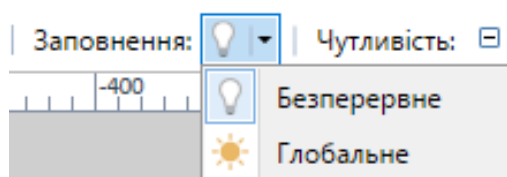


Рис. 272. Вибір режиму заповнення інструменту Чарівна паличка

Безперервний – цей режим виділяє лише ті пікселі, які мають схожу колірну гаму і є поруч один з одним. Тобто, якщо є розриви або відмінності в кольорах між пікселями, то вони не будуть виділені. Результати виділення області можуть відрізнятися залежно від того, який відсоток буде встановлено як параметр **Чутливість**.

Глобальний – у цьому режимі чарівна паличка вибирає всі пікселі одного кольору по всьому зображенню, незалежно від їхніх сусідів. Тобто, якщо ви клацнете на піксель, програма вибере всі пікселі цього кольору на всьому зображенні. Цей режим підходить, коли потрібно змінити колір об'єкта, що розкиданий по різних частинах зображення. Результат буде поширюватися на всі пікселі в шарі, що задовольняють налаштування **Чутливість**.

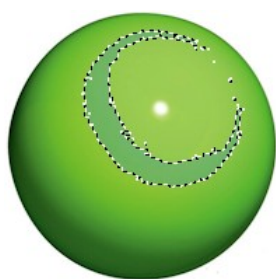
ЧУТЛИВІСТЬ

Чутливість визначає, наскільки схожими за кольором мають бути пікселі, щоб їх можна було виділити.

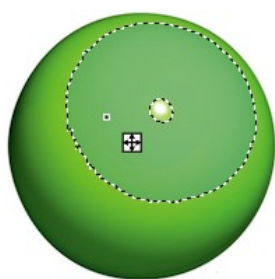
Низька чутливість (близько 0%) означає, що будуть обрані лише ті пікселі, які мають майже ідентичний колір до натиснутого пікселя.

Висока чутливість (до 100%) збільшує діапазон вибору, включаючи ширший спектр відтінків і кольорів, схожих на початковий.

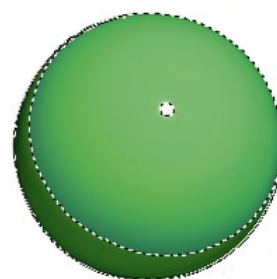
Ефект такої настройки показаний нижче (Рис. 273):



Чутливість 14%



Чутливість 23%



Чутливість 50%

Рис. 273. Вибір чутливості виділення інструментом Чарівна паличка

РЕЖИМИ ВИБІРКИ

Інструмент **Чарівна паличка** має два режими вибірки: **Зображення** та **Шар**, які стосуються того, як вибираються кольори для інструментів малювання, таких як пензель чи ластик (Рис. 274).

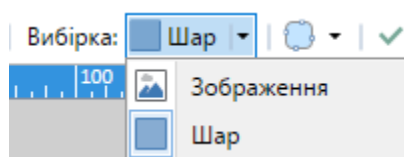


Рис. 274. Вибір режиму вибірки інструменту Чарівна паличка

Зображення – чарівна паличка вибирає колір з урахуванням усіх шарів. Це означає, що якщо в точці, на яку ви натискаєте, є кілька шарів із різними кольорами, Paint.NET врахує підсумковий результат – тобто змішаний колір усіх видимих шарів.

Шар – чарівна паличка вибирає колір лише з поточного активного шару. Якщо на цьому шарі нічого немає (наприклад, прозорий піксель), Paint.NET не буде враховувати інші шари під ним.

Ці режими використовуються також інструментами **Заливання** та **Піпетка**.

ПЕРЕТЯГУВАННЯ ТОЧКИ КЛІКА

Точку кліка інструмента **Чарівна паличка** можна коригувати, поки вибір активний. Це замінює необхідність скасувати та повторно застосувати виділення, якщо воно не охоплює потрібну область (Рис. 275).

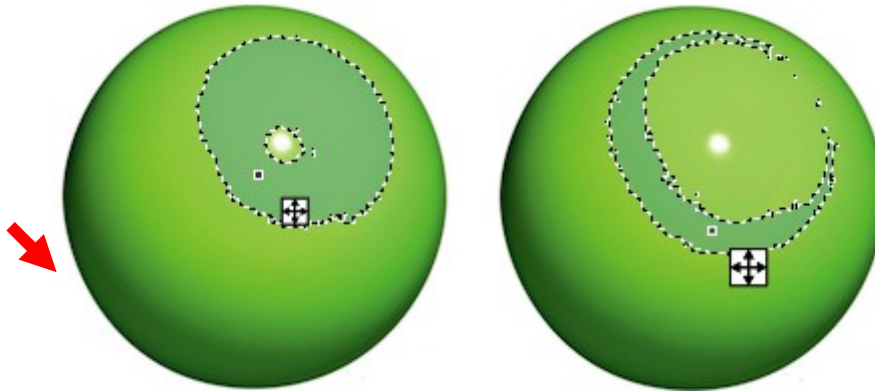


Рис. 275. Перетягування точки кліка

Маркер, що керує, виглядає як білий квадрат із чотирма стрілками всередині нього. Натисніть і перетягніть значок, щоб перемістити точку кліка. Виділення буде проінтерпретовано так, ніби користувач скасував виділення та повторно застосував його, використовуючи нову вихідну точку.

РЕЖИМИ ВИДІЛЕННЯ

Є п'ять режимів вибору: **Замінити**, **Додавання** (об'єднання), **Віднімання**, **Перетин** та **Інверсія** (Хог – виключення). Вони визначають, як взаємодіють наступні вибірки з наявною вибіркою. Для спрощення доступу до кожного режиму є гарячі кнопки (Рис. 276).

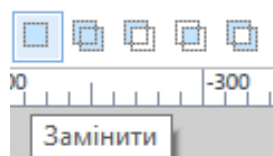


Рис. 276. Гарячі кнопки вибору Режиму виділення

Замінити. Це стандартний режим виділення, який скидає поточне виділення при створенні нового. Тобто нове виділення замінить будь-яке попереднє.

Додавання. Дає змогу додавати нові області до поточного виділення (Рис. 277). Це дозволяє вибрати одразу кілька областей. Для додавання можна використовувати будь-який інший інструмент виділення.

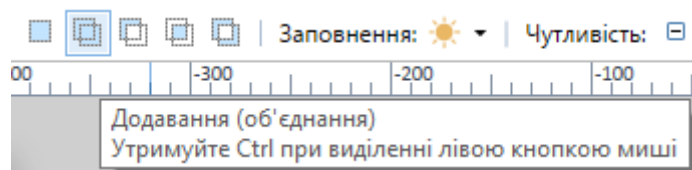


Рис. 277. Вибір режиму виділення Додавання

Нижче показаний вибір, зроблений шляхом виділення кола за допомогою інструмента **Еліпс**, із подальшим створенням квадратного виділення за допомогою інструмента **Прямокутник** (Рис. 278).

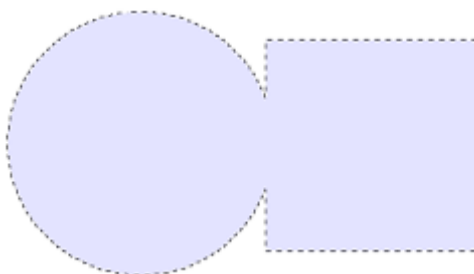


Рис. 278. Приклад виділення в режимі додавання

Додавати до наявного виділення нове можна також при натиснутій клавіші **Ctrl**.

Віднімання. У цьому режимі будь-які нові області видаляються з раніше створеного виділення (Рис. 279). Цей режим дає змогу змінювати або скорочувати вибір, який неможливо зробити, роблячи виділення за один раз. Для подальшого видалення можна вибирати будь-який інструмент виділення.

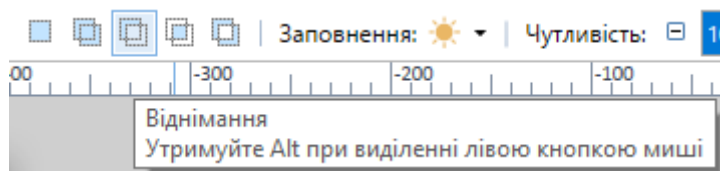


Рис. 279. Вибір режиму виділення Віднімання

Наприклад, виділяючи коло за допомогою інструмента **Еліпс**, режим віднімання дозволяє легко видалити центр, створивши в ньому інше коло (Рис. 280).

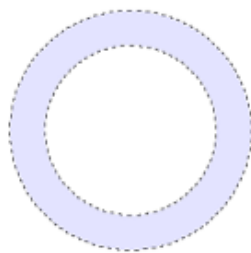


Рис. 280. Приклад виділення в режимі віднімання

Видалити фрагменти з наявного виділення можна також при натиснутій клавіші **Alt**.

Перетин множин. У цьому режимі зберігаються лише виділення, що перетинаються. Області, які не є спільними для вихідної та нової областей, відкидаються (Рис. 281).

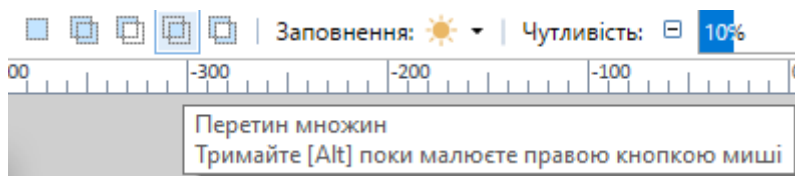


Рис. 281. Вибір режиму виділення Перетин множин

Наприклад, виділення у чверть кола (Рис. 282) було створено за допомогою створення великого круглого виділення інструментом **Еліпс**, а потім меншого квадратного виділення з активним режимом перетину.

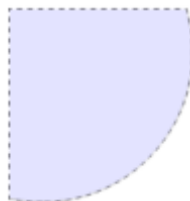


Рис. 282. Приклад виділення в режимі Перетину множин

Перетин можна також створити затисненням клавіші **Alt** під час створення нового виділення **Правою** клавішею миші.

Інверсія (Xor). Цей режим діє як режим **Додавання**, за винятком того, що області, що перекриваються, видаляються із загального виділення (Рис. 283).

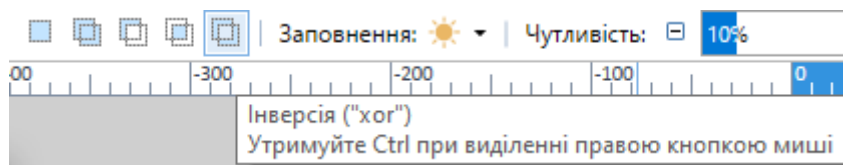


Рис. 283. Вибір режиму виділення Інверсія

Наприклад, комбінація інструментів **Еліпс** і **Прямокутник** матиме вигляд як на рисунку (Рис. 284).

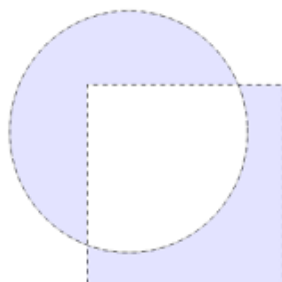


Рис. 284. Приклад виділення в режимі Інверсії

Увімкнути режим **Інверсія** можна також затисненням клавіші **Ctrl** під час створення нового вибору **Правою** клавішею миші.

ВИБІР ЯКОСТІ

Є дві опції для вибору якості виділення: **Згладжений** і **Без згладжування** (Рис. 285).

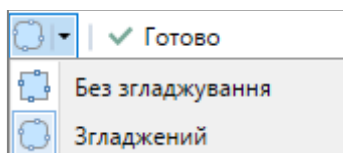


Рис. 285. Вибір якості виділення

Ці параметри задають поведінку краю виділення: чи відсікається край виділення по межі кожного пікселя чи край буде згладжений (Рис. 286).

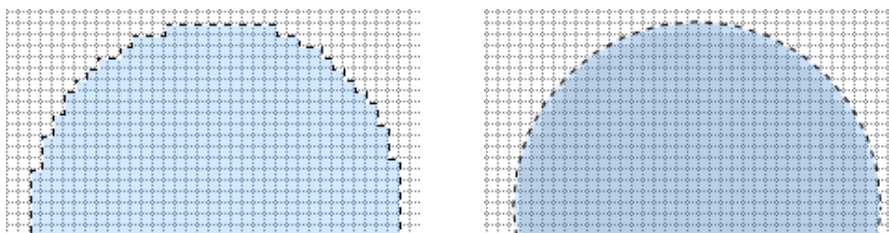


Рис. 286. Демонстрація виділення в режимах Без згладжування та Згладжений

Краї фігури зліва нерівні. Фігура праворуч згладжена.

ІНСТРУМЕНТИ ДЛЯ ПЕРЕГЛЯДУ

Для зміни поточного вигляду зображення використовуються два інструменти: **Масштаб** і **Рука** (Рис. 287). Ці інструменти змінюють спосіб перегляду зображення, не змінюючи його.

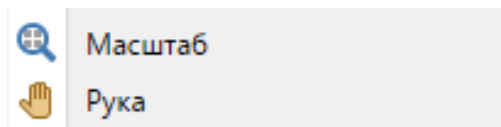


Рис. 287. Інструменти для перегляду

МАСШТАБ

Масштаб використовується для збільшення або зменшення вигляду зображення.

Щоб збільшити полотно, використовується ліва клавіша миші. Права клавіша миші зменшує масштаб. Індивідуальні рівні масштабування заздалегідь визначені та не підлягають налаштуванню.

Колесо миші, яке використовується разом із клавішею **Ctrl**, дублює дію інструменту.

Обведення прямокутної області при активному інструменті **Масштаб**, збільшить вказану область до розміру вікна редагування.

РУКА

Інструмент **Рука** дозволяє прокручувати зображення у вікні редагування. Це корисно під час перегляду зображення, розмір якого більший, аніж вікно редагування (при збільшенні).

Функціонал цього інструменту також доступний, якщо активний будь-який інший інструмент. Прокручайте зображення по вертикалі у будь-який час за допомогою колеса миші. Утримуйте **Shift**, щоб змінити напрямок прокручування на горизонтальний.

Щоб вимкнути надмірне прокручування за межі зображення потрібно натиснути кнопку **Параметри** та на вкладці **Інтерфейс** діалогового вікна **Параметри** зняти прапорець **Прокрутка за межами зображення**.

Також можна панорамувати зображення за допомогою клавіш зі стрілками **↑ ← ↓ →**, утримуючи **пробіл**. Водночас відстань перетягування обернено пропорційно масштабу. Зображення, що мають більший масштаб, прокручуються на меншу кількість пікселів при кожному натисканні кнопок. І навпаки.

Якщо одночасно утримувати клавішу **Ctrl** та клавішу **пробілу**, величина панорування при натисканні клавіші зі стрілкою збільшується в десять разів.

Центрування можна досягти, використовуючи комбінацію клавіш **Ctrl + B** двічі. Перша комбінація масштабує зображення вікна, друга комбінація відновлює попередній рівень масштабування. Цього ж ефекту можна досягти за допомогою меню **Вигляд → Вмістити в вікно**.

ІНСТРУМЕНТИ ПЕРЕМІЩЕННЯ

У Paint.NET є два інструменти переміщення: **Перемістити виділене** та **Переміщення виділення** (Рис. 288).

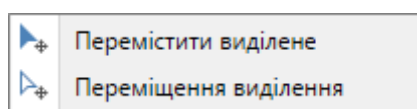


Рис. 288. Інструменти переміщення

ІНСТРУМЕНТ ПЕРЕМІСТИТИ ВИДІЛЕНЕ

Перемістити виділене, M – цей інструмент дозволяє переміщувати, повертати і масштабувати виділену область разом із пікселями зображеннями, які потрапили до цієї області (Рис. 289). Якщо цей інструмент активний, блакитне підсвічування області виділення (для кращого відображення кольору) вимикається.



Рис. 289. Дія інструменту Перемістити виділене

Натисніть і перетягніть, щоб перемістити пікселі в інше місце. Як видно на зображенні, наведеному вище, після переміщення пікселів замість них залишається прозора область. Це позначено сіро-білим малюнком шахівниці.

Для переміщення копії виділення потрібно утримувати клавішу **Ctrl**.

ІНСТРУМЕНТ ПЕРЕМІЩЕННЯ ВИДІЛЕННЯ

Переміщення виділення, M, – цей інструмент дає змогу переміщувати, повертати та масштабувати виділену область, не змінюючи положення пікселів зображення, що потрапили до цієї області (Рис. 290). Це корисно для уточнення положення та розміру області вибору. Якщо цей інструмент активний, області виділення підсвічується блакитним кольором.



Рис. 290. Дія інструменту Переміщення виділення

ЗМІНА РОЗМІРУ

При перемиканні на один з інструментів переміщення виділення, на кутах і сторонах області виділення з'являються маркери, які використовуються для керування перетвореннями форми та розміру виділення.

Для зміни розміру наведіть курсор миші на будь-який із маркерів межі виділення і він змінить вигляд на «руку». Натисніть на ліву клавішу мишки та перетягніть виділення / виділене (Рис. 291).

Щоб зберегти вихідні пропорції сторін, можна натиснути на клавішу **Shift**.

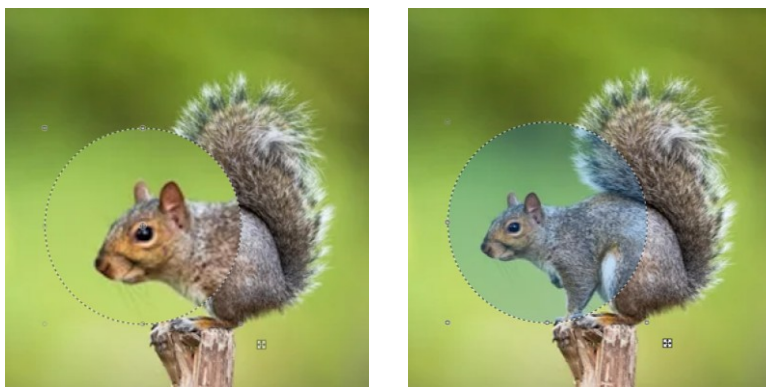


Рис. 291. а) зміна розміру обраної області; б) зміна розміру області виділення

ОБЕРТАННЯ

Якщо курсор розташований неподалік маркера, він змінюється на подвійну вигнуту стрілку. Ця двонаправлена (Рис. 292) стрілка курсора використовується для вказівки на можливості обертання. Клацніть лівою кнопкою миші та перетягніть курсор, щоб повернути виділення навколо центру обертання.

Права кнопка миші завжди викликає поворот незалежно від розташування курсора.

Центр обертання, виділення або шару відображається під час активації інструменту переміщення. Він виглядає як коло з хрестиком та визначає точку повороту виділення при повороті.



Рис. 292. Обертання виділеної області

Центр обертання може бути переміщений за межі виділення або навіть за межі полотна.

Під час обертання кут повороту відображається у рядку стану. Здійснювати поворот із кроком 15 градусів можна, утримуючи клавішу **Shift** під час обертання.

Кожне переміщення, зміна розміру та поворот, що здійснюється над областю виділення, додається у вікно **Історія**, тому завжди можна простежити зміни.

Повертати можна і виділену область, і область виділення (Рис. 293).

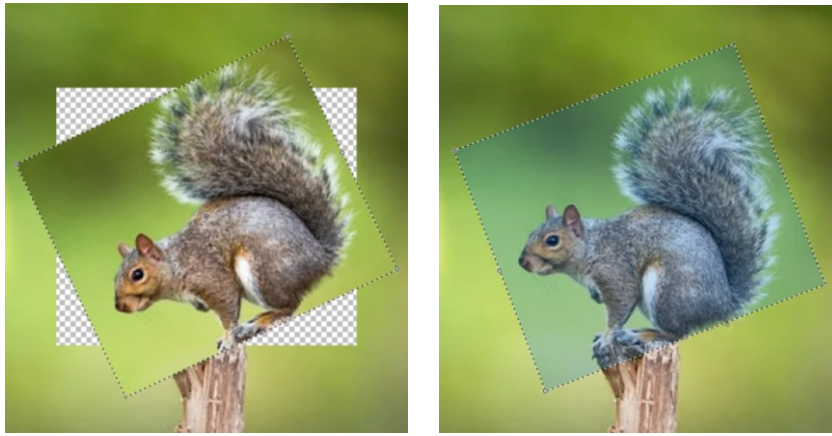


Рис. 293. а) поворот виділеної області; б) поворот області виділення

ІНСТРУМЕНТИ ЗАЛИВАННЯ

У Paint.NET є два інструменти заливання: **Заповнення** та **Гرادієнт** (Рис. 294).

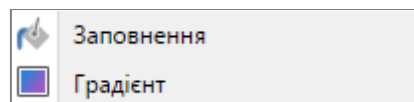


Рис. 294. Вибір інструмента заливання

ЗАПОВНЕННЯ

Інструмент **Заповнення**, **F** можна використовувати для заповнення окремої області вказаним кольором. Для цього потрібно розмістити курсор в області, яку потрібно зафарбувати, а потім клацнути лівою клавішею миші, щоб залити її **Основним кольором** або правою кнопкою, щоб залити **Додатковим**.

Область зображення для заливки визначається як область зі схожим кольором навколо курсору. Поріг чутливості кольору регулюється повзунком **Чутливість** на Панелі інструментів. Також тут можна встановити стиль, який використовуватиметься при заливці, та багато інших параметрів.

РЕЖИМИ ЗАПОВНЕННЯ

Є два режими заповнення: **Глобальне** та **Безперервне**. Ці режими визначають спосіб заливки та вибираються на **Панелі Інструментів** (Рис. 295).

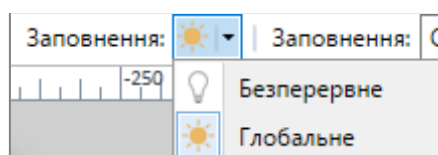


Рис. 295. Вибір режиму заповнення

Безперервний – заливка поширюється лише на ті пікселі, які мають схожу колірну гаму і є поруч один із одним.

Глобальний – заливка поширюється на всі пікселі одного кольору по всьому зображенню. Цей режим заливки можна ввімкнути, натиснувши клавішу **Shift**.

Результат заповнення залежатиме від налаштування параметра **Чутливість**.

ЧУТЛИВІСТЬ

Процес налаштування чутливості інструмента **Заповнення** подібний до описаного вище процесу налаштування чутливості інструмента **Чарівна паличка**.

Зліва вихідне зображення. На другому зображенні показано заповнення зеленого кольору помаранчевим із чутливістю 36%. Третє зображення показує чутливість 42%. І зображення праворуч показує, що область заповнення ще збільшилася у розмірі, тому що чутливість було збільшено до 60% (Рис. 296).

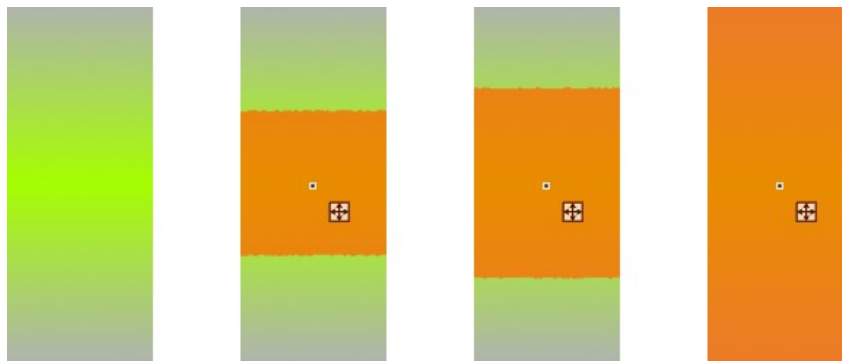


Рис. 296. Приклади заповнення кольором з різною чутливістю

Чутливість можна коригувати навіть доки заливка залишається не завершеною. Також можна переміщувати точку кліка.

ЗМІНА КОЛЬОРУ АКТИВНОЇ ЗАЛИВКИ

Якщо заливка активна, зміна кольору у вікні **Палітра** негайно перефарбовує заповнену область.

Також у Paint.NET є можливість коригувати точку кліка (виглядає як білий квадрат із чотирма стрілками всередині нього) інструменту **Заповнення**, поки заливка ще активна. Це замінює необхідність скасовувати та повторно застосовувати операцію заповнення до іншої області (Рис. 297).

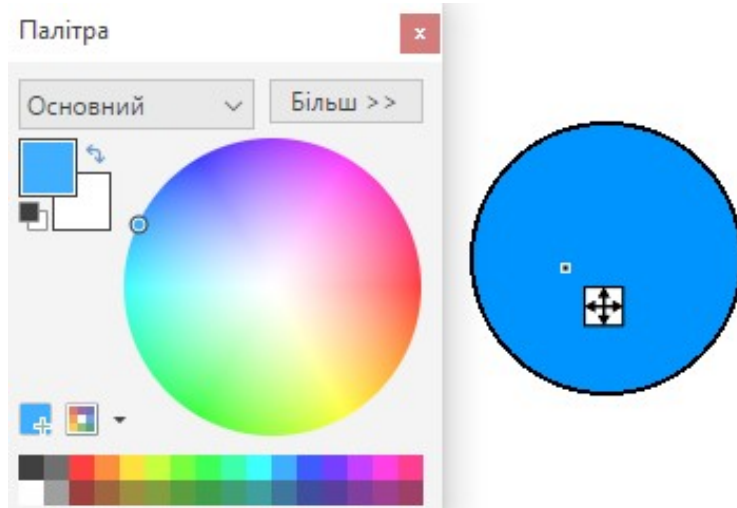


Рис. 297. Коригування точки кліка інструменту Заповнення

На зображенні нижче показано, що початкова операція заливки забарвлює ліве коло в синій колір. Цю заливку легко нанести на інше коло, перетягнувши точку кліку на область, яку потрібно заповнити. На правому зображенні показана переміщена точка клацання та отримана в результаті змінена заливка (Рис. 298).

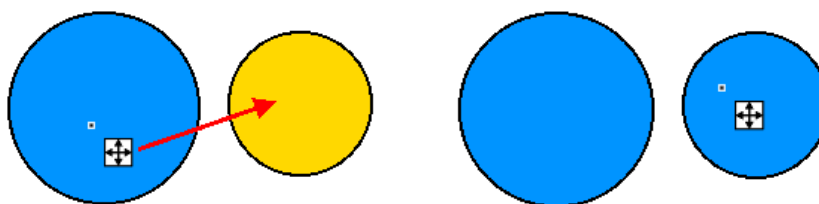


Рис. 298. Приклад зафарбовування області перенесенням точки кліка

СТИЛІ ЗАПОВНЕННЯ

Інструмент **Заповнення** дає змогу заливати області візерунком, а не суцільним кольором (Рис. 299). Заповнення за допомогою стилю заливки використовуватимуть **Основний** і **Додатковий** кольори (Рис. 300).

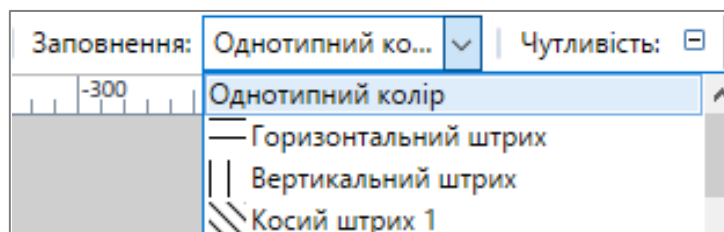


Рис. 299. Вибір стилю заповнення

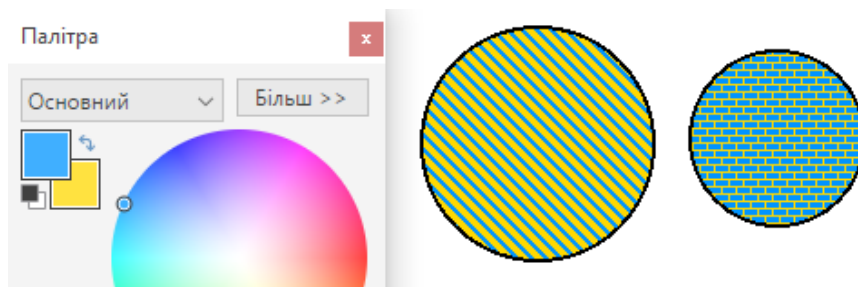


Рис. 300. Приклади заповнення основним і додатковим кольорами

РЕЖИМИ ВИБІРКИ

Інструмент **Заповнення** має два режими вибірки: **Зображення** та **Шар**, що визначають, які пікселі будуть відбиратися або тестуватись, щоб визначити ступінь заповнення (Рис. 301).

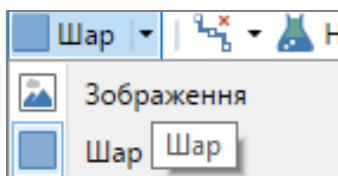


Рис. 301. Вибір режиму Вибірки

Зображення – інструмент аналізує всі шари, щоб знайти схожі кольори для заливки. Це дає змогу заливати пікселі, навіть якщо їхній колір визначається інформацією з інших шарів.

Шар – інструмент заливки працює лише на активному шарі. Тобто він заповнюватиме лише ті області, кольори яких відповідають вибраному кольору на поточному шарі.

Ці режими спільно використовуються інструментами **Заповнення**, **Чарівна паличка** та **Піпетка**.

РЕЖИМИ ЗМІШУВАННЯ

У Paint.NET інструмент заповнення використовує також режими змішування, які впливають на те, як нові кольори взаємодіють з уже наявними на шарі. Ці режими дають змогу створювати різноманітні ефекти під час редагування зображень (Рис. 302).

Більш детально режими змішування розглядаються у розділі Налаштування шару → [Режими змішування](#).

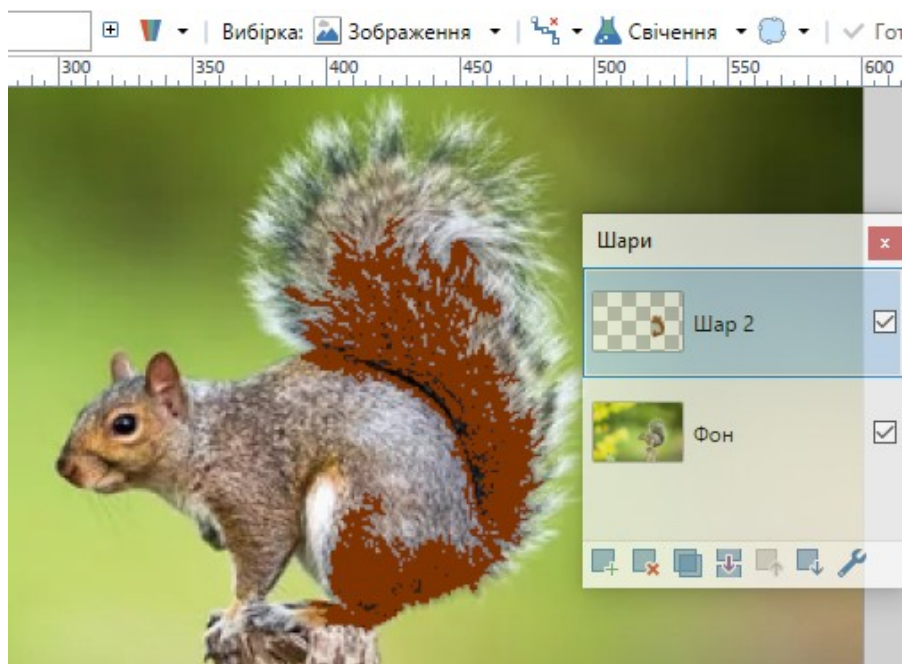


Рис. 302. Приклад застосування режиму змішування

ГРАДІЄНТ

Щоб намалювати градієнт, виберіть інструмент **Градiєнт**, **G**, натисніть на полотні та протягніть вказівником миші. Градієнт буде намальований як перехід між **Основним** і **Додатковим** кольорами під час руху миші.

Інструмент **Градiєнт** дає змогу малювати градієнти семи типів, які можна вибирати на Панелі інструментів (Рис. 303).



Рис. 303. Вибір типу градієнта на Панелі інструментів

Доки інструмент **Градiєнт** активний, градієнт можна відрегулювати, перетягнувши **керівні маркери** (точки початкового та кінцевого кліку). Якщо клацнути правою клавішею миші по керівному маркеру, зміняться ролі **Основного** та **Додаткового** кольорів, змінивши напрям градієнта.

Нижче наведено приклади впливу кожного із 7 типів градієнта на зображення (Рис. 304–309). Зліва показано зображення після обробки градієнтом із двома непрозорими кольорами (значення альфа 255). Праворуч показано зображення після обробки градієнтом із прозорим Додатковим кольором (значення альфа 0).

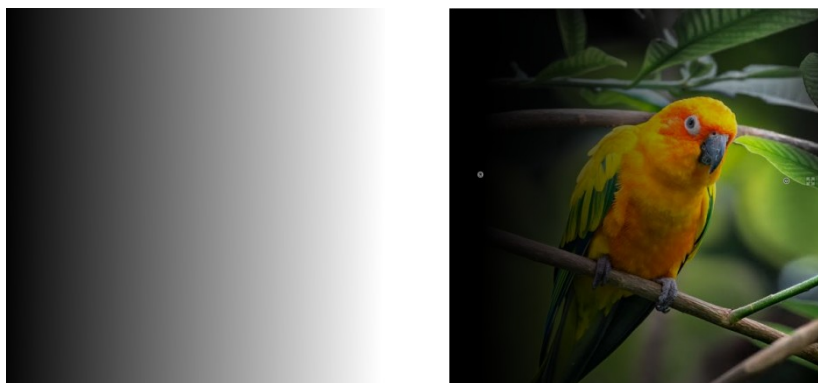


Рис. 304. Лінійний градієнт у кольоровому та прозорому режимах

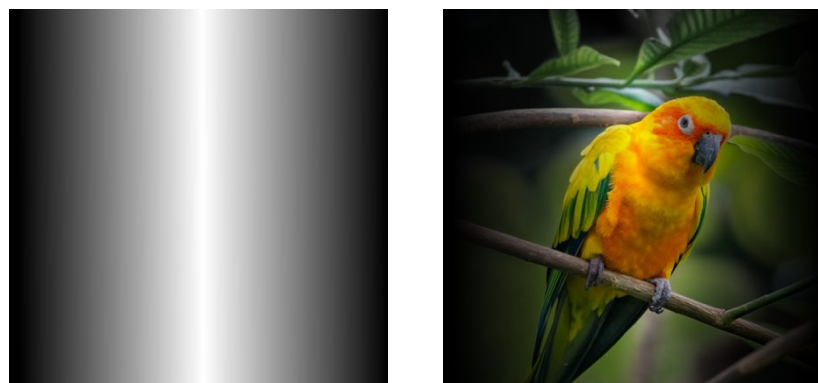


Рис. 305. Лінійний (відбитий) градієнт у кольоровому та прозорому режимах

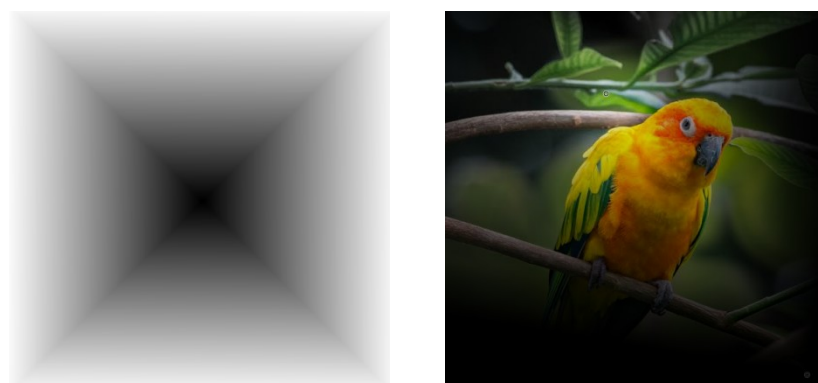


Рис. 306. Ромбоподібний градієнт у кольоровому та прозорому режимах

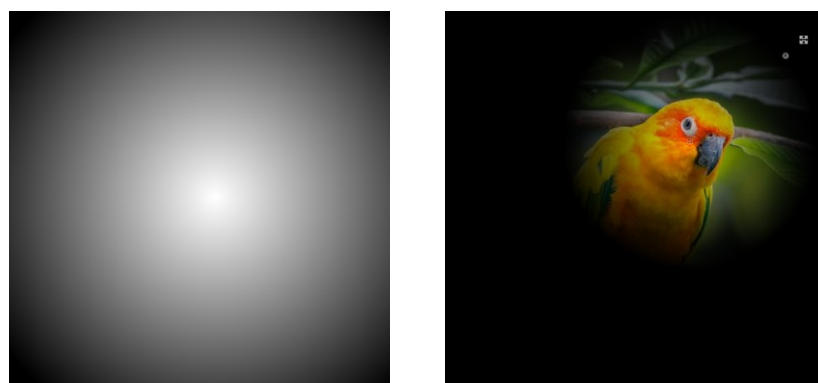


Рис. 307. Круговий градієнт у кольоровому та прозорому режимах

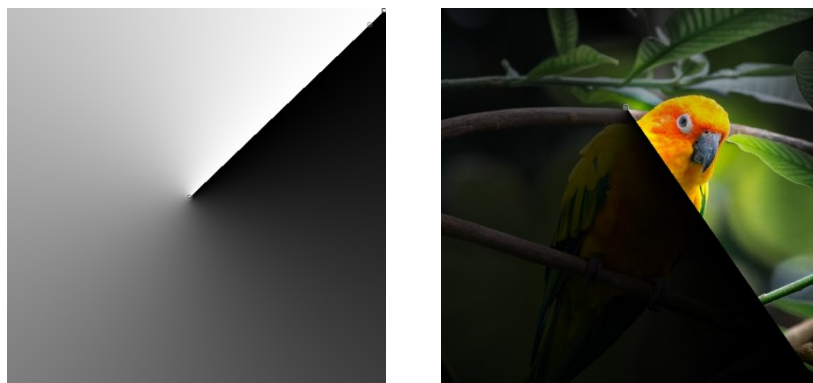


Рис. 308. Конічний градієнт у кольоровому та прозорому режимах

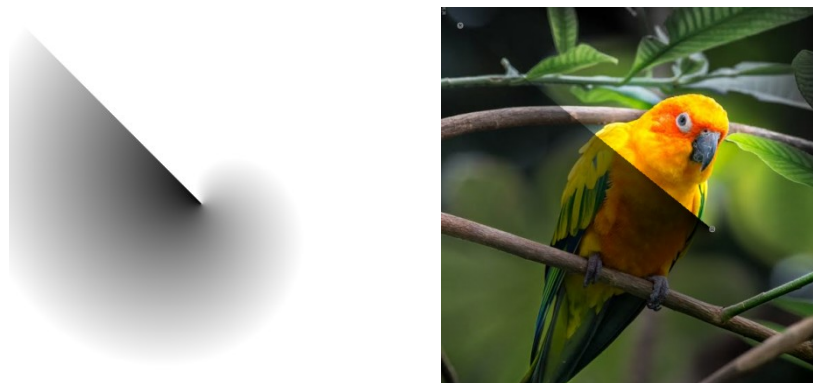


Рис. 309. Спіраль за годинниковою стрілкою (Спіраль проти годинникової стрілки аналогічна) у кольоровому та прозорому режимах

Режими градієнта

Градієнт може бути в двох режимах: Режим прозорості та Колірний режим (Рис. 310).

Режим градієнта за замовчуванням – це **Колірний режим**, який впливає на всі колірні канали та альфа-канал.

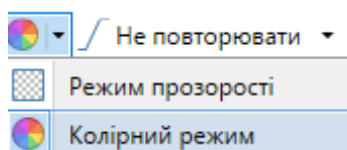


Рис. 310. Режими градієнта

Режим прозорості використовується для створення поступового «зникнення» частини зображення. Це може бути корисним для змішування двох зображень разом (Рис. 311).

Цей режим градієнта впливає лише на альфа-канал активного шару. Градієнт зникатиме від альфа-значення Основного кольору до інверсії альфа-значення

Додаткового кольору. Отже, можливим є перехід між двома станами часткової прозорості.

Щоб зробити згасання або змішування, помістіть два зображення на окремі шари, а потім скористайтесь прозорим градієнтом будь-якої форми на верхньому шарі.



Рис. 311. Приклад плавного переходу одного зображення в інше

Режими повтору

Режими повтору визначають дію градієнта поза межами форми. Користувачеві доступно 3 режими: **Не повторювати**, **Повторювати** й **Відбивати** (Рис. 312, 313).

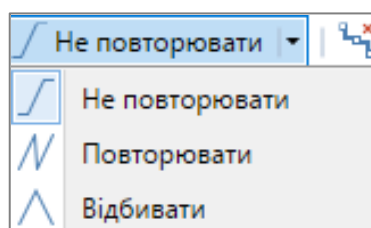


Рис. 312. Режими повтору

Не повторювати – градієнт не виходить за межі обмежень, що накладаються керівними маркерами.

Повторювати – градієнт між керівними маркерами повторюється у фрагментах такої ж ширини, як і відстань між керівними маркерами. Фрагменти відокремлені різкою гранню закінчення одного градієнта та початком наступного.

Відбивати – градієнт дзеркально відбитий уздовж країв. Це призводить до безшовного патерну, що повторюється.



Рис. 313. Приклади застосування режимів повтору

РЕЖИМИ ЗМІШУВАННЯ

Градієнти можна створювати, використовуючи опції режимів змішування, вибрані на Панелі інструментів із меню, що випадає.

Градієнт буде застосований відповідно до інших параметрів (Колір / Прозорість, Режим повторення тощо), а потім проінтерпретується так, якби пікселі були на своєму шарі з встановленим режимом змішування шарів.

Більш детально режими змішування розглядаються у розділі Налаштування шару → [Режими змішування](#).

ІНСТРУМЕНТИ ДЛЯ МАЛЮВАННЯ

У Paint.NET наявні такі інструменти для малювання, як **Пензлик**, **Гумка** та **Олівець** (Рис. 314).

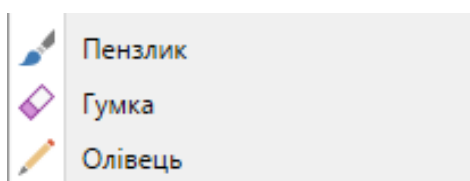


Рис. 314. Інструменти для малювання

ПЕНЗЛИК

Пензлик, **B** – один з основних інструментів малювання, який імітує ручний пензлик. Він використовується для створення смуг суцільного кольору або кольору з візерунком.

Щоб малювати за допомогою цього інструмента, натисніть і тягніть покажчик миші по полотну. При використанні лівої клавіші миші застосовуватиметься

Основний колір. При використанні правої клавіші миші буде застосовуватися **Додатковий колір.**

У будь-якому разі інструмент **Пензлик** успадкує непрозорість вихідного кольору, який встановлено у розширеному вікні **Палітра.**

Розмір, Жорсткість, Інтервал, Заповнення та Режим змішування інструменту можна налаштувати з Панелі інструментів після активації пензлика.

У деяких версіях Paint.NET ви можете вибрати форму кінчика пензлика, що може вплинути на вигляд нанесених мазків.

Стили заповнення

Якщо на Панелі інструментів вибрано стиль заповнення для створення візерунка, використовуватимуться і **Основний**, і **Додатковий кольори.** Якщо використовується права кнопка миші (замість лівої), ролі Основного та Додаткового кольорів міняються місцями. При використанні стилю **Однотипний колір** використовується лише основний колір.

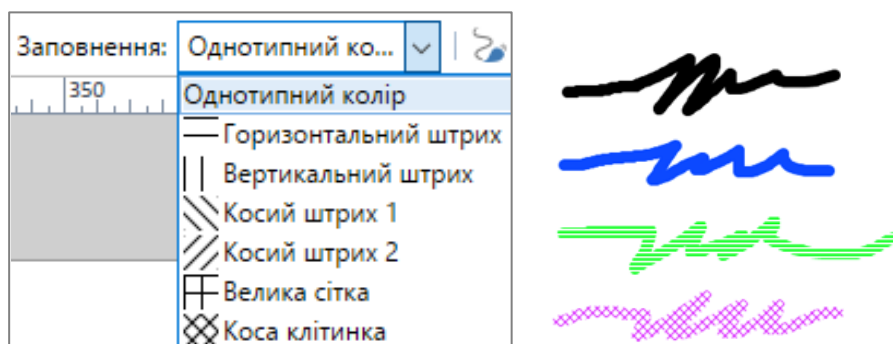


Рис. 315. Вибір стилю заповнення

Жорсткість

Параметр **Жорсткість** визначає, наскільки жорстким чи м'яким буде край пофарбованої смуги. Низькі величини жорсткості надають пензлику м'який край, тоді як високі величини роблять його більш чіткішим. Жорсткість можна вважати силою згладжування, що використовується для пом'якшення країв. Із цієї причини жорсткість ігнорується, якщо вимкнено згладжування.

На наведеному нижче прикладі зверху малювання було здійснене пензликом із низьким значенням Жорсткості (10%), а знизу з високим (90%) (Рис. 316).



Рис. 316. Приклад малювання пензликом з різною жорсткістю

Інтервал

Інтервал визначає, як часто пензлик залишає слід на полотні під час малювання. Що менше значення інтервалу, то більш безперервним буде слід, а що більше значення, то більш переривчастим (Рис. 317). Це може бути корисно для створення текстур або специфічних ефектів.

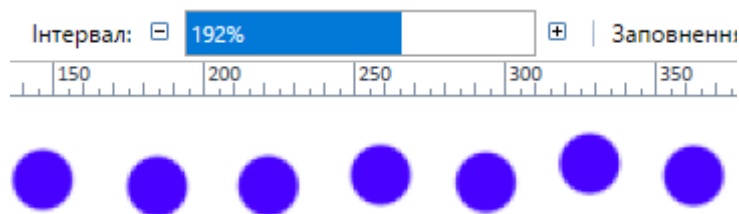


Рис. 317. Встановлення інтервалу слідів пензлика

ГУМКА

Інструмент **Гумка**, **E** використовується для видалення частин активного шару або виділення. Це може бути корисним при роботі з багатошаровими зображеннями, оскільки видалені частини верхнього шару дозволяють бачити нижні шари.

Стирання може призвести до відображення шахівниці. Вона не є частиною зображення і не відобразатиметься поза редагуванням Paint.NET.

Гумка має параметри **Розміру**, **Жорсткості** та **Інтервалу** на Панелі інструментів, аналогічні до відповідних налаштувань **Пензлика**.

Гумка успадковує значення альфа-каналу **Основного кольору** при використанні лівої кнопки миші та значення альфа-каналу **Додаткового кольору** при використанні правої кнопки миші. Це дозволяє робити м'які стирання.

Нижче наведені приклади трьох налаштувань альфа-каналу: у лівій частині значення альфа-каналу встановлено на 60; у середній частині на 125; у правій частині на 255 (максимум) (Рис. 318).

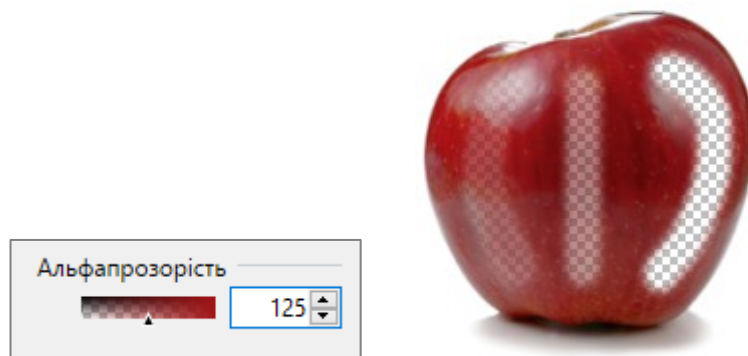


Рис. 318. Приклад налаштування альфа-каналу

ОЛІВЕЦЬ

Інструмент **Олівець**, **P** використовується для попиксельного редагування активного шару. Перетягування інструмента **Олівець** створює лінію завширшки один піксель.

За допомогою лівої клавіші мишки **Олівець** малює піксель **Основним кольором**. За допомогою правої клавіші мишки **Олівець** малює піксель **Додатковим кольором**.

У будь-якому разі інструмент **Олівець** успадкує непрозорість вихідного кольору, який встановлено у розширеному вікні **Палітра**.

Під час використання олівця корисно включити **Піксельну сітку**. Піксельна сітка доступна з коефіцієнтом масштабування вище 200% і показує контур окремих пікселів, що полегшує попиксельне редагування (Рис. 319).

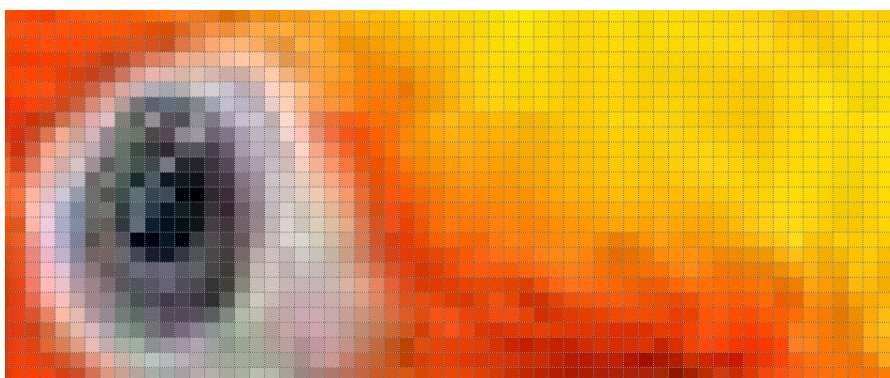


Рис. 319. Вигляд зображення із включеною піксельною сіткою

Інструмент можна застосувати, використовуючи **Режим змішування**, вибраний на Панелі інструментів. Більш детально режими змішування розглядаються у розділі Налаштування шару → [Режими змішування](#).

ФОТОІНСТРУМЕНТИ

До інструментів Paint.NET, що полегшують роботу з кольорами та ретушуванням зображень, належать: **Піпетка**, **Клонуючий штамп** і **Заміна кольору** (Рис. 320)

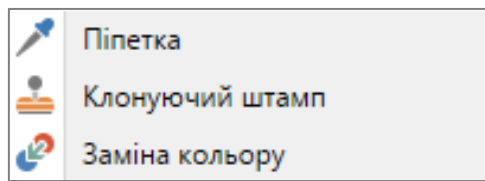


Рис. 320. Фотоінструменти

ПІПЕТКА

Інструмент **Піпетка**, **К** використовується для перенесення кольору з активного шару до слотів **Основного** або **Додаткового** кольору вікна **Палітра**.

Щоб використовувати інструмент, активуйте його та клацніть у будь-якій точці на активному шарі. Колір пікселя під курсором буде встановлений на **Основний колір** (ліва клавіша миші) або **Додатковий колір** (права клавіша миші) з одночасним зазначенням розташування цього кольору на колірному колі у вікні **Палітра** (Рис. 321).

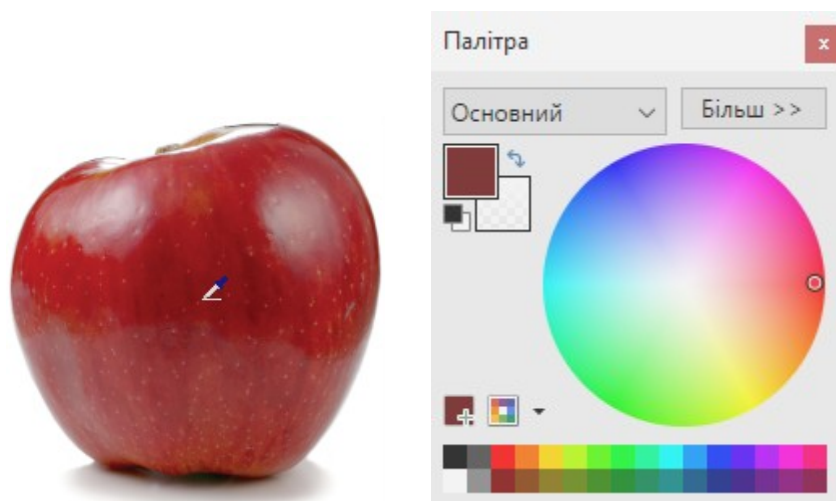


Рис. 321. Вибір кольору інструментом Піпетка

Інструмент **Піпетка** найпростіше використовувати при збільшенні зображення з увімкненою підтримкою **Піксельної сітки**.

Режими Вибірки

Піпетка має два режими вибірки: **Зображення** та **Шар** (Рис. 322). Вони визначають, які пікселі будуть обрані під час вибору кольору.

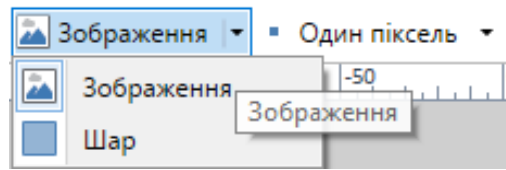


Рис. 322. Режими вибірки інструменту Піпетка

Зображення – програма сприймає шари як одне об'єднане зображення, і вибирає колір на основі кінцевого результату всіх шарів разом.

Шар – якщо зображення складається з кількох шарів, піпетка ігноруватиме всі інші шари та вибиратиме колір тільки з того шару, який ви обрали для редагування.

Розмір Вибірки

Для інструменту **Піпетка** можна використовувати один із шести фіксованих розмірів вибірки (Рис. 323). Це квадрати розміром один піксель, 3×3 , 5×5 , 11×11 , 31×31 , 51×51 . Розмір визначає область, з якої піпетка вибирає колір.

У разі, якщо розмір вибірки перекриває певний діапазон кольорів, то вибрані кольори знаходитимуться усереднені й буде повернено один усереднений колір.

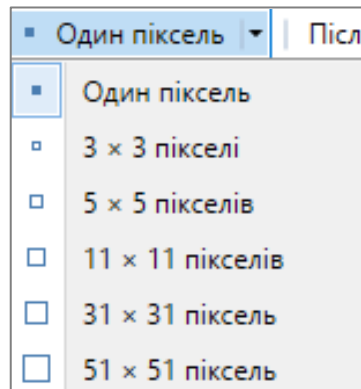


Рис. 323. Фіксовані розміри вибірки

Вибрати поведінку **Піпетки** після натискання можна у списку **Після натискання** (Рис. 324):

- **Не перемикати інструмент** – залишає інструмент **Піпетка** активним після вибору кольору.

- **Перемкнутися на попередній** – після вибору кольору буде активований інструмент, що використовувався останнім.
- **Перемкнутися на Олівець** – після вибору кольору буде активований інструмент **Олівець**.

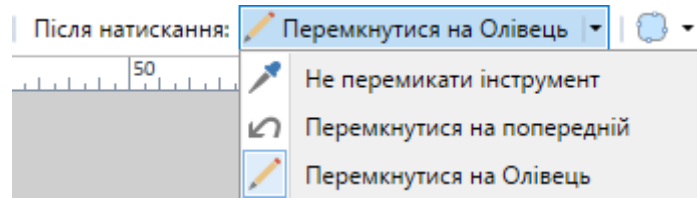


Рис. 324. Вибір поведінки піпетки після натискання

КЛОНУЮЧИЙ ШТАМП

Клонуючий штамп, **L** корисний при копіюванні групи пікселів між різними шарами або в тому самому шарі (Рис. 325).

Для використання цього інструменту виберіть його у вікні інструментів, а потім укажіть джерело копіювання – область, яку потрібно клонувати. Для цього натисніть клавішу **Ctrl** і клацніть у потрібній області мишкою. У будь-який момент можна змінити джерело копіювання, здійснивши аналогічну операцію.

Далі перейдіть до цільової області (область, куди необхідно скопіювати пікселі). Натисніть ліву кнопку миші, щоб почати клонування. Інструмент буде копіювати вміст з області джерела та накладати його на нове місце.



Рис. 325. Приклад використання інструменту Клонуючий штамп

Клонуючий штамп має налаштування **Розміру**, **Жорсткості**, **Інтервалу** та **Згладжування**, які розміщені на **Панелі інструментів** і працюють аналогічно до відповідних налаштувань **Пензлика**.

Ви можете змінювати розмір пензлика, щоб налаштувати, як багато пікселів захоплюється при клонуванні.

Зменшення значення **Жорсткості** призводить до пом'якшення краю клонованого зображення. Низьке значення Жорсткості може використовуватися для надання клонованій області менш чіткого вигляду.

Ви також можете використовувати різні режими змішування для отримання незвичних результатів (Рис. 326).



Рис. 326. На малюнку апельсин був клонований, щоб мати вигляд частини яблука

Якщо непрозорість зменшена, **Клонуючий штамп** буде копіювати частково прозору копію джерела. На наступному зображенні непрозорість основного кольору у вікні **Палітра** зменшена до 150. В результаті клонування за допомогою цього параметра прозорість скопійованих пікселів зменшується порівняно з вихідним кольором (Рис. 327).

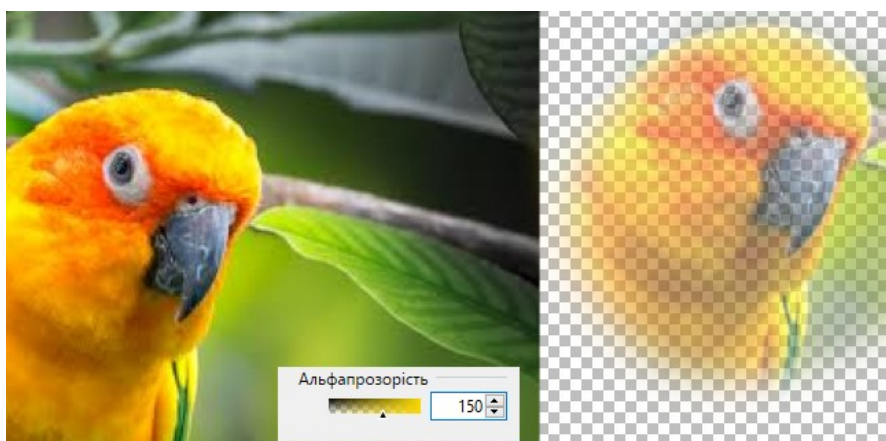


Рис. 327. Приклад клонування із збільшенням прозорості

ЗАМІНА КОЛЬОРУ

Інструмент **Заміна кольору**, **R** використовується для заміни одного кольору іншим. Із його допомогою можна швидко та просто змінити колір будь-якого об'єкта.

При використанні лівої кнопки миші основний колір замінюватиме колір, вибраний як додатковий колір. При використанні правої кнопки миші навпаки.

Режими вибірки

Інструмент **Заміна кольору** має два різні режими вибірки: **Вибірка: один раз** і **Вибірка: додатковий колір** (Рис. 328). Це суттєво змінює спосіб дії інструмента.



Рис. 328. Режими вибірки інструменту Зміна кольору

Вибірка: один раз. Використовуючи ліву кнопку миші, перефарбовуємо в колір, який попереднім кліком вибрали як основний колір. Використовуючи праву кнопку миші, перефарбовуємо в колір, який попереднім кліком вибрали як додатковий колір.

На зображення нижче **Основний колір** – синій, **Додатковий колір** – світло-зелений. Замальовуємо будь-який колір на зображенні в ці два кольори, використовуючи ліву і праву клавіші мишки (Рис. 329).

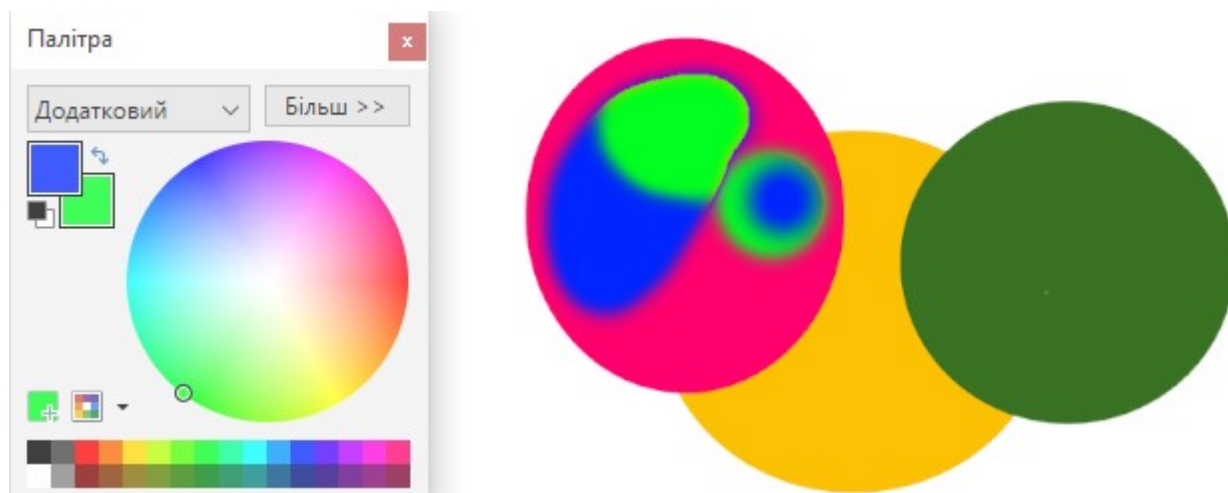


Рис. 329. Перефарбовування в режимі Вибірка: в один раз

Вибірка: додатковий колір. Використовуючи цей режим із лівою кнопкою миші, основний колір стає кольором заміни. Замінюються пікселі в межах

налаштування чутливості додаткового кольору. Використання правої кнопки миші змінює ролі кольорів порівняно з лівою кнопкою миші.

На зображенні нижче: **Основний колір** – синій, **Додатковий колір** – жовтий. Замальовуємо тільки жовтий (залежно від встановленої чутливості) колір синім (Рис. 330).

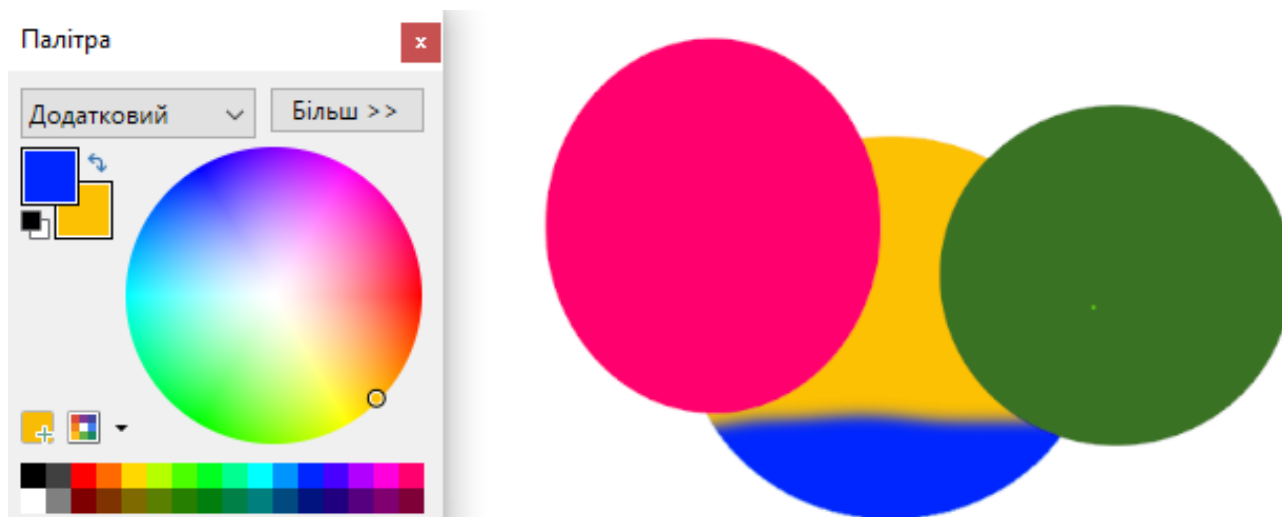


*Рис. 330. Перефарбовування в режимі Вибірка:
додатковий колір*

ЧУТЛИВІСТЬ, РОЗМІР ПЕНЗЛЯ ТА ЖОРСТКІСТЬ

Користувач може налаштувати точність заміни кольору за допомогою налаштування потрібного значення **Чутливості**. Якщо значення **Чутливість** дорівнює 0%, то заміна кольору здійснюватиметься тільки для точного збігу кольорів. Якщо ж встановити значення **Чутливість** – 100%, то заміні підлягатимуть усі кольори (у цьому разі дія інструменту відповідатиме інструменту Пензель).

Ширина пензля, що перефарбовує, задається параметром **Розмір пензля** на Панелі інструментів. Кінчик пензля завжди буде круглої форми.

Величина **Жорсткості** на Панелі інструментів визначає, наскільки жорсткими або м'якими є краї пензля, що перефарбовує. Високий параметр дає більш різкий край, тоді як низький параметр дає більш м'який край. Зверніть увагу, що параметр **Жорсткість** ігнорується, якщо вимкнено згладжування (Рис. 331).

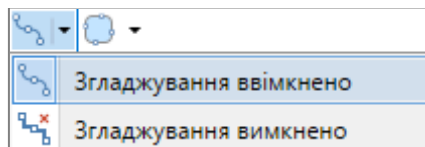


*Рис. 331. Параметр Жорсткість ігнорується,
якщо вимкнено згладжування*

Нижче на зображенні можна побачити застосування інструмента **Заміна кольору**. Зверніть увагу, що було враховано тонкі зміни відтінку (Рис. 332).



Рис. 332. Приклад застосування інструмента Зміна кольору

ІНСТРУМЕНТИ ДЛЯ ТЕКСТУ ТА ФІГУР

До інструментів для тексту та фігур належать: **Текст**, **Лінія / Крива** та **Форми** (Рис. 333).

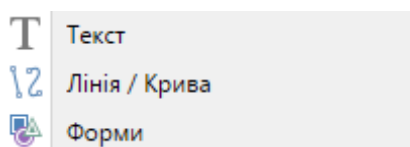


Рис. 333. Інструменти для тексту і фігур

ФОРМИ

Інструмент **Форми**, **O** дозволяє додавати різні геометричні фігури на полотно. Інструмент має 29 визначених фігур: 8 основних, 8 багатокутників або зірок, 4 стрілки, 4 форми виноски та 5 символів (Рис. 334).

Щоб створити фігуру, клацніть на вибраний тип фігури у відповідному списку **Панелі інструментів** і починайте створювати фігуру на полотні. Фігура буде створена в режимі редагування.

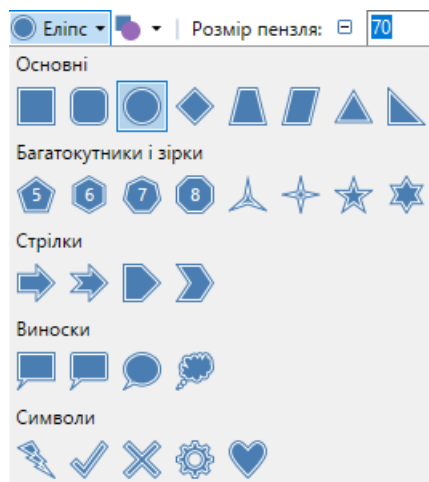


Рис. 334. Список фігур

У режимі Редагування форма не є фіксованою і може бути змінена у розмірі, орієнтації, заливці та кольорі поряд із параметрами згладжування та режимом змішування. Всі ці параметри доступні на Панелі інструментів за активного інструмента **Форми**.

Під час створення фігури утримуйте клавішу **Shift**, щоб зберегти вихідне співвідношення висоти та ширини.

Фіксація фігури на полотні

Зафіксувати створену фігуру на активному шарі та вийти з режиму редагування можна, використовуючи будь-який із цих методів:

- Натисніть клавішу **Enter**.
- Клацніть за межами прямокутника, який обмежує фігуру.
- Натисніть кнопку **Готово** на Панелі інструментів.
- Почніть малювати нову фігуру.

Переміщення фігури

Перед тим, як фігура буде зафіксована на полотні, її можна перемістити у будь-яке місце на полотні.



Щоб змінити положення фігури, клацніть на значок, що пульсує із чотирма стрілками у квадраті, використовуючи ліву кнопку миші.

Або помістіть вказівник усередині фігури (він перетвориться на чотиристоронню стрілку) та натисніть і перетягніть його лівою кнопкою миші.

Клавіші зі стрілками $\uparrow \leftarrow \downarrow \rightarrow$ також можна використовувати для переміщення фігури. Одне натискання клавіші переміщує об'єкт на один піксель у напрямку стрілки. Одночасне натискання клавіші **Ctrl** переміщає об'єкт на 10 пікселів.

Обертання фігури

Перед тим, як фігура буде зафіксована на полотні, її також можна обертати.

Центр обертання виглядає як коло з хрестом усередині й по замовчанню розташоване в центрі фігури. Натисніть і перетягніть центр обертання у потрібне місце (його можна перемістити за межі фігури або навіть за межі полотна) (Рис. 335).

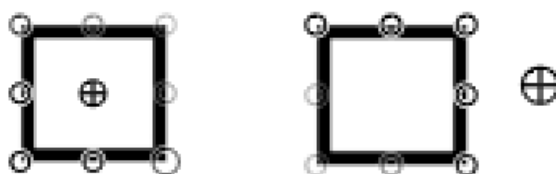


Рис. 335. Переміщення центру обертання

Фігура повертається навколо центру обертання, використовуючи **праву** клавішу миші (Рис. 336).

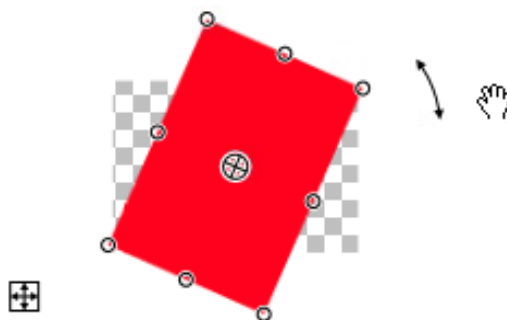


Рис. 336. Поворот фігури навколо центру обертання

Альтернативою є розміщення курсора одразу за межами фігури, коли він стане двонаправленою зігнутою стрілкою. Тоді фігуру можна повернути навколо її центру, клацнувши та перетягнувши за допомогою **лівої** клавіші миші.

При утриманні натиснутою клавіші **Shift** можливий покроковий поворот із дискретністю 15 градусів.

Для повороту фігури можна використовувати клавіші зі стрілками $\uparrow \leftarrow \downarrow \rightarrow$, утримуючи водночас **праву** клавішу миші.

Режими малювання та заливання Форми

Форми можна малювати у трьох різних режимах: **Тільки контур**, **Тільки заливка**, **Контур та заливка** (Рис. 337; Рис. 338).

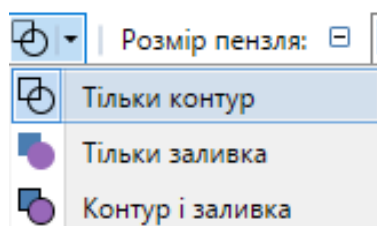


Рис. 337. Режими малювання та заливання Форми

У перших двох варіантах застосовується Основний колір при використанні лівої клавіші миші або Додатковий колір при використанні правої клавіші миші. У режимі **Контур та заливка** заливка буде додатковим кольором, а контур – основним кольором під час використання лівої клавіші миші. Права клавіша миші змінює колір.

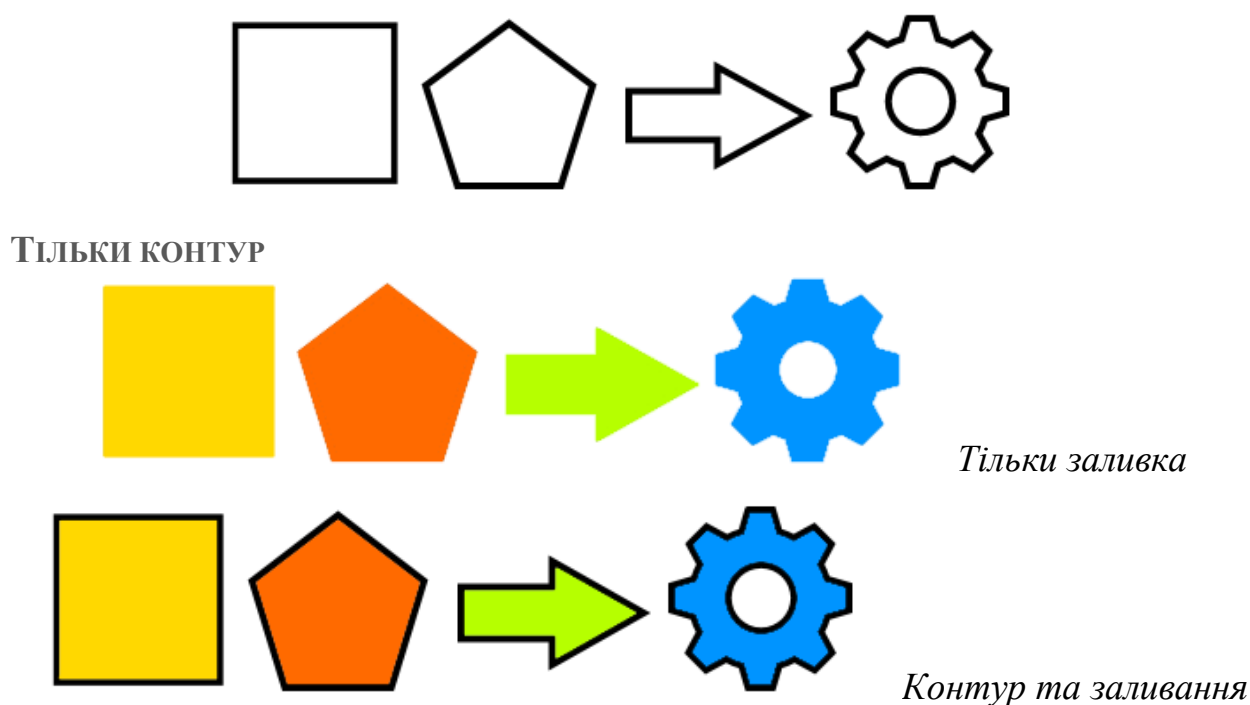


Рис. 338. Приклад малювання фігур

Розмір та стиль пензля

Вигляд межі фігури залежать від параметрів пензля (**Розмір пензля** та **Стиль**) на Панелі інструментів (Рис. 339).

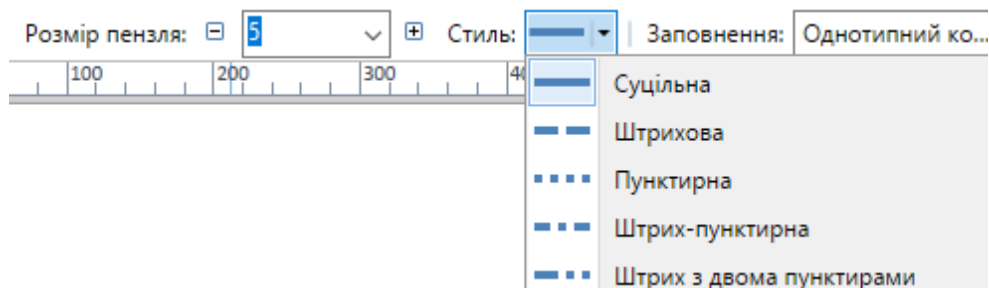


Рис. 339. Вибір стилю та розміру пензля

Скруглений прямокутник має додаткове налаштування на Панелі інструментів: **Радіус заокруглення** (Рис. 340).

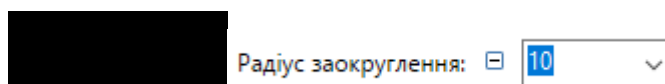


Рис. 340. Встановлення розміру радіуса заокруглення

Стилі Заливки

Фігури можуть мати один із типів заливки, що вибираються зі списку **Заповнення** на Панелі інструментів (Рис. 341).

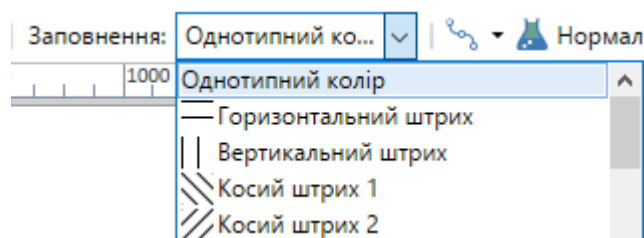


Рис. 341. Вибір стилю заливки форми

Ця функція дає змогу заповнювати фігури одним із численних шаблонів, а не суцільним кольором. Заливка за допомогою вибраного стилю використовуватиме Основний і Додатковий кольори.

Згладжування

Інструментом **Форми** має два режими згладжування:

Згладжування ввімкнено – лінії виглядають гладкішими, жирнішими та м'якшими.

Згладжування вимкнено – лінії відображаються у виді суцільних пікселів і виглядають більш нерівними і тонкими.

Режими змішування

До інструмента **Форма** можуть бути застосовані **Режими змішування**, аналогічно до режимів змішування шарів.

Більш детально режими змішування розглядаються у розділі Налаштування шару → [Режими змішування](#).

ЛІНІЯ / КРИВА

Інструмент **Лінія / Крива**, **O** використовується для малювання ліній і кривих. Ці дві можливості групуються в один інструмент, оскільки лінія є частинним випадком кривої. Тобто фактично цей інструмент завжди малює криві, просто лінія не має кривизни.

Створення Лінії / Кривої

Щоб намалювати лінію потрібно клацнути на точці початку лінії та перетягнути покажчик миші до бажаної кінцевої точки. Звільнення миші завершує етап малювання.

При використанні лівої кнопки миші лінія відображається основним кольором. Використання правої кнопки миші відображає лінію у Додатковому кольорі.

Після закінчення етапу малювання на лінії з'являються чотири керівні маркери, а поруч із кінцевою точкою лінії з'являється маркер із чотиристоронніми стрілками для переміщення (Рис. 342). Значки всіх маркерів пульсують, що забезпечує кращу видимість.



Рис. 342. Керівні маркери

Якщо потрібна пряма лінія, зафіксуйте її одним зі способів:

- Натисніть на клавішу **Enter**.
- Натисніть на клавішу кнопку **Готово** на Панелі інструментів.
- Клацніть за межами поточного прямокутника, що обмежує лінію / криву.
- Почніть малювати нову лінію / криву.

Якщо потрібна крива, будуйте її, переміщуючи керівні маркери у нове місце.

Утримання клавіші **Shift** під час створення прямої лінії обмежує поворот лінії на кути, кратні 15 градусів (0 °, 15 °, 30 °, 45 ° ...).

Переміщення Лінії / Кривої

У режимі редагування створену лінію чи криву можна вільно переміщати й повертати. Для цього натисніть маркер із чотирма стрілками, що пульсують у квадраті, та перемістіть створену лінію, використовуючи ліву кнопку миші.

Також для переміщення можна використовувати клавіші зі стрілками **↑ ← ↓ →**. Одне натискання на клавішу зі стрілкою переміщує об'єкт на один піксель у напрямку стрілки. Одночасне натискання клавіші **Ctrl** переміщає об'єкт на 10 пікселів.

Обертання Лінії / Кривої

Також у режимі редагування можна лінію або криву повернути навколо свого геометричного центру за допомогою правої клавіші миші. Водночас вказівник миші набуде вигляду двонаправленої стрілки (Рис. 343).

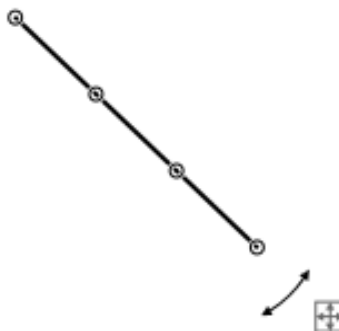


Рис. 343. Обертання Лінії / Кривої

При утриманні натиснутою клавіші **Shift** можливий покроковий поворот із дискретністю 15 градусів.

Також поворот лінії / кривої можна здійснити клавішами зі стрілками **↑ ← ↓ →** при натиснутій **правій** клавіші миші.

Типи Кривих

На Панелі інструментів є три режими для вибору типу кривої: **Прямий**, **Сплайн** і **Безьє** (Рис. 344).

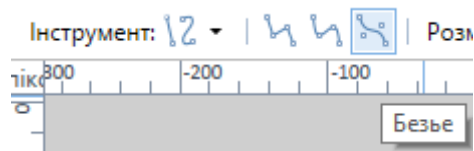


Рис. 344. Кнопки вибору типу кривої на панелі інструментів

При виборі типу **Прямий** креслиться ламана лінія.

При виборі кривої **Сплайн** лінія буде вигнута з використанням кубічної сплайнової інтерполяції. Крива завжди проходитиме через керівні маркери, навіть якщо вони будуть переміщені за межі полотна (Рис. 345).



Рис. 345. Сплайн

При виборі кривої **Безьє** лінія проходитиме від першого (вихідного) до останнього керівного маркера. Другий і третій керівні маркери змінюють криву вздовж її шляху, але необов'язково на ній лежать (Рис. 346).

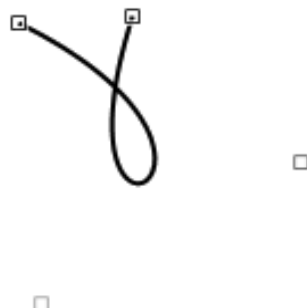


Рис. 346. Крива Безьє

При перемиканні між типами кривих на Панелі інструментів, крива буде переінтерпретована на основі обраного типу кривої та поточного розташування керівних маркерів.

Ширина, Стиль і Заливка

Ширину лінії або кривої можна вибрати зі списку **Розмір пензля** на Панелі інструментів.

Лінія та крива можуть мати різні стилі початку, самої лінії та кінця цієї лінії.

Стиль початку та кінця лінії / кривої може бути: **Звичайна**, **Стрілка**, **Наконечник** або **Закруглена** (Рис. 347).

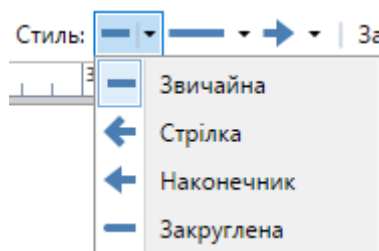


Рис. 347. Вибір стилю початку кривої

Сама лінія може бути: **Суцільна**, **Штрихова**, **Пунктирна**, **Штрихпунктирна**, **Штрих з двома пунктирами** (Рис. 348).

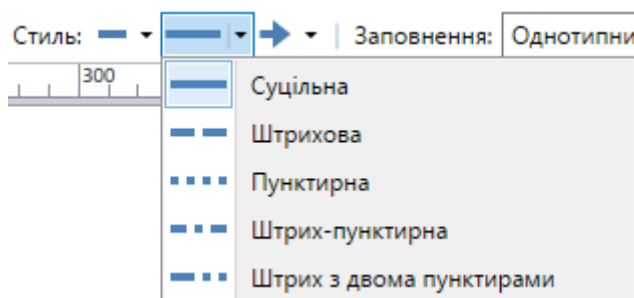


Рис. 348. Вибір стилю лінії кривої

Використовуючи розмір пензля та стилі ліній, можна малювати такі лінії (Рис. 349):

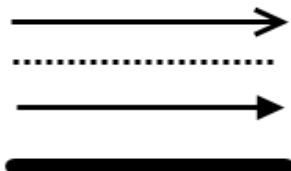


Рис. 349. Приклади ліній із різними розмірами пензля і стилями накреслення

Лінії та криві можуть відображатися одним із багатьох типів заливки, що вибираються зі списку **Заповнення** на Панелі інструментів. Заливка за допомогою вибраного стилю використовуватиме Основний і Додатковий кольори (Рис. 350).

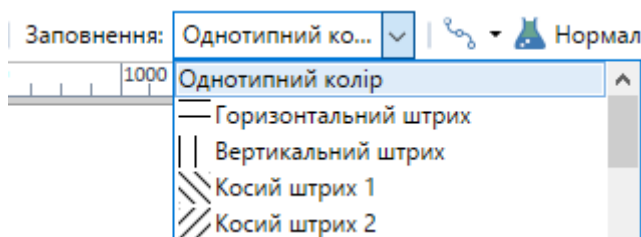


Рис. 350. Вибір стилю заливки кривої

Згладжування

Інструмент **Лінія / Крива** має, як і всі інші форми, два режими згладжування: **Згладжування увімкнено** (лінії виглядають гладкішими, жирнішими та м'якшими) та **Згладжування вимкнено** (лінії відображаються у виді суцільних пікселів і виглядають більш нерівними та тонкими).

Режими змішування

Лінії та криві можна наносити за допомогою Режиму змішування, вибраного на Панелі інструментів.

Більш детально режими змішування розглядаються у розділі Налаштування шару → [Режими змішування](#).

ТЕКСТ

Інструмент **Текст**, **T** дає змогу набирати текст на поточному шарі за допомогою Основного кольору, який можна змінити у вікні **Палітра**.

Завжди розміщуйте текст на прозорому шарі. Якщо потрібно буде відредагувати текст, це можна зробити значно простіше.

Елементи керування на Панелі інструментів можуть використовуватися для вибору шрифту, зміни розміру шрифту, форматування тексту, режиму відтворення тексту, вирівнювання, згладжування та режиму змішування.

Під час набору тексту зверніть увагу на наявність пульсуючої чотиринаправленої стрілки нижче та праворуч від текстового курсору, що миготить. Це керуючий маркер. Він використовується для переміщення тексту доки він активний.

Утримання клавіші **Ctrl** прибирає видимість курсору та керуючого маркера. Після звільнення клавіші **Ctrl** від утримання курсор і маркер знову видно. Це корисно, якщо треба подивитися, як виглядатиме текст без цих елементів.

Після закінчення набору тексту натисніть клавішу **Esc** або кнопку **Готово** на Панелі інструментів для фіксації тексту на шарі.

Коли текст фіксується, він більше не існує як набір символів, а сприймається тільки як пікселі. Це означає, що текст не може бути повторно відредагований інструментом **Текст** після фіксації на шарі.

Шрифт і розмір тексту

Перші два елементи керування на Панелі інструментів визначають шрифт, який буде використовуватися для візуалізації тексту та розмір символів. Потрібний розмір можна і вибрати зі списку, і ввести у вікно вибору розміру. Розмір підтримує тільки цілочисельні значення (Рис. 351).

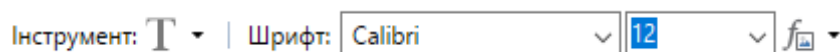


Рис. 351. Встановлення шрифту та розміру тексту

Показник розміру шрифту

Текст за промовчанням відображається за допомогою точок для вимірювання розміру шрифту в двох режимах (Рис. 352):

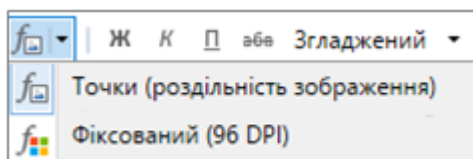


Рис. 352. Режими вимірювання розміру тексту

Точковий (роздільність зображення) – це одиниця виміру довжини, яка використовується в друкованих продуктах. Одна точка дорівнює 1/72 дюйма. Текст, що відображається в точках, буде масштабований відповідно до роздільної здатності зображення (точок на дюйм).

Фіксований (96 пікселів на дюйм) – регулює розмір шрифту для відповідності роздільній здатності Windows. За замовчуванням це 96 пікселів на дюйм, де один піксель дорівнює одній точці на екрані комп'ютера. Текст, що відображається з використанням пікселів, буде відображатися з базовою висотою, такою, як і вказаний розмір у пікселях.

Форматування тексту

Є чотири основні варіанти форматування тексту: **Жирний**, **Курсив**, **Підкреслений** і **Закреслений** (Рис. 353). Ці параметри можна вмикати або вимикати

за допомогою піктограм на Панелі інструментів, коли інструмент **Текст** ще активний. Опції форматування можна поєднувати, вибираючи одразу кілька опцій.



Рис. 353. Вибір формату тексту

Режими відтворення тексту

Режим відтворення можна вибрати на Панелі інструментів під час активного інструмента **Текст**. Є три режими відтворення (рендерингу) тексту: **Згладжений**, **Новий різкий** і **Класичний різкий**, кожен з яких впливає на те, як текст виглядає під час його рендерингу (Рис. 354).

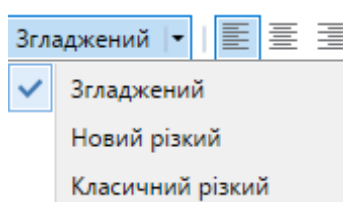


Рис. 354. Режими відтворення тексту

Згладжений – у цьому режимі використовується згладжування країв шрифту для створення м'якого вигляду. Він забезпечує більш плавні переходи між пікселями, що корисно для відображення тексту на екрані, коли важлива чіткість і плавність форм.

Новий різкий – це сучасний режим відтворення, який використовує технологію рендерингу без згладжування. Шрифти виглядають чіткішими, але менш плавними. Це ідеальний варіант, коли потрібен різкий і чіткий текст.

Класичний різкий – цей режим ґрунтується на старих технологіях рендерингу тексту, де також немає згладжування, але використовується інша методика порівняно з Новим різким. У результаті текст виглядає схожим до старих версій Paint.net або інших програм, які використовували класичний алгоритм відтворення тексту.

Вирівнювання тексту

Тип вирівнювання тексту задається за допомогою піктограм на Панелі інструментів під час активного інструменту **Текст**.



Рис. 355. Режими вирівнювання тексту

Вирівнювання тексту – це напрямок, у якому текст, що друкується, графіка відходить від початкової позиції курсора. Є три варіанти вирівнювання:

- **Вирівняти по лівому краю.** Текст розташовуватиметься праворуч від початкової позиції курсору.
- **Вирівняти по центру.** Текст розташовуватиметься рівномірно праворуч і ліворуч від початкової позиції курсора.
- **Вирівняти по правому краю.** Текст буде ліворуч від початкової позиції курсору.

Згладжування

Інструментом **Текст** має два режими згладжування: **Згладжування ввімкнено** (лінії виглядають гладкішими, жирнішими і м'якшими) та **Згладжування вимкнено** (лінії відображаються у виді суцільних пікселів і виглядають більш нерівними та тонкими).

Режими змішування

До інструмента **Текст** можуть бути застосовані **Режими змішування**, аналогічно до режимів змішування шарів.

Більш детально режими змішування розглядаються у розділі Налаштування шару → [Режими змішування](#).

МЕНЮ РЕДАГУВАТИ

Меню **Редагувати** містить команди для переходу вперед і назад по історії редагування зображень, команди керування буфером обміну, маніпулювання обраною областю та виділенням (Рис. 356).

Редагувати	Вигляд	Зображення	Шари	Регул
	Скасувати			Ctrl+Z
	Повторити			Ctrl+Y
	Вирізати			Ctrl+X
	Копіювати			Ctrl+C
	Копіювати об'єднані			Ctrl+Shift+C
	Вставити			Ctrl+V
	Вставити в новий шар			Ctrl+Shift+V
	Вставити в нове зображення			Ctrl+Alt+V
	Копіювати виділення			Ctrl+Alt+Shift+C
	Вставити виділення			▶
	Очистити виділення			Del
	Заповнити виділення			Backspace
	Інвертувати виділення			Ctrl+I
	Вибрати все			Ctrl+A
	Скасувати виділення			Ctrl+D

Рис. 356. Команди меню Редагувати

Скасувати / Повернути, Ctrl + Z / Ctrl + Y – скасовує / повертає останні дії, які застосовувались до активного зображення. Те саме можна зробити і у вікні історії.

Вирізати, Ctrl + X – видалення вибраної частини зображення активного шару. Ви можете вибрати частину зображення за допомогою інструментів вибору: **Прямокутник**, **Ласо**, **Еліпс** або **Чарівна паличка**. Після виконання команди **Вирізати** вибрана область шару переміститься в буфер обміну, а на її місце вставляться «прозорі» пікселі. Лінія, що відмічає поточне виділення, також зникне.

Копіювати, Ctrl + C – копіювання вибраної області у буфер обміну. Виділення залишається на активному шарі. Поточний вибір також залишається активним.

Копіювати об'єднані, Ctrl + Shift + C – копіювання всіх шарів в буфер обміну без необхідності використовувати команду **Об'єднати шари**.

Вставити, Ctrl + V – вставлення зображення з буфера обміну в активний шар. Водночас вставлені пікселі автоматично виділяються та вмикається інструмент **Переміщення виділення, M**. Це дає змогу переміщати, масштабувати або повертати вставлене зображення у бажане місце та з потрібною орієнтацією.

Якщо зображення, що вставляється, занадто широке і/або занадто високе для поточного розміру полотна, то з'явиться вікно, в якому потрібно буде вибрати бажану дію: **Збільшити полотно**, **Зберегти розмір полотна** або **Скасувати вставку**. При виборі пункту **Збільшити полотно** його розмір збільшиться до розміру фрагмента у буфері обміну.

Вставити в новий шар, Ctrl + Shift + V – створення нового шару для розміщення зображення з буфера обміну. Перевага перед попередньою командою полягає у тому, що вихідне зображення не руйнується. Новий шар створюється безпосередньо над активним шаром і стає активним.

Вставити у нове зображення, Ctrl + Alt + V – створення нового зображення для розміщення зображення з буфера обміну. Нове зображення має такі самі розміри, як і зображення з буфера обміну.

Копіювати виділення, Ctrl + Alt + Shift + C – копіює поточну геометрію виділення у буфер обміну. Це корисно, якщо ви хочете зберегти форму виділення, щоб застосувати її до іншого шару, зображення або зберегти для подальшого використання.

Вставити виділення – вставляє геометрію виділення з буфера обміну як активне виділення на активний шар.

Режими комбінування виділення підтримуються через підменю **Вставити виділення**, тому, наприклад, геометрія виділення з буфера обміну може бути додана до наявного виділення. Детально булеві операції були розглянуті у підрозділі [Булеві операції](#) розділу **Векторна графіка в графічних редакторах** (Рис. 357).

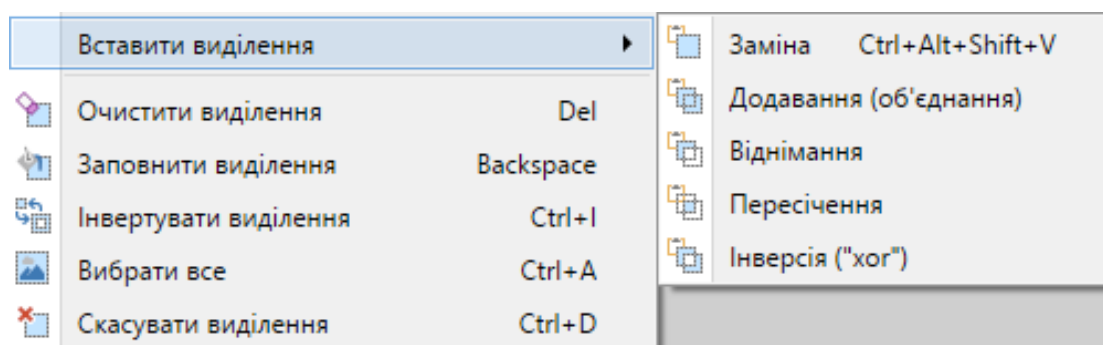


Рис. 357. Режими вставлення виділеного

Очистити виділення, Del – видалення виділеної області на активному шарі. Всі пікселі, розташовані всередині цього виділення, заміщаються прозорими. Ця дія схожа на команду **Вирізати**, за винятком того, що пікселі, що видаляються, не копіюються в буфер обміну.

Заповнити виділення, Backspace – заповнення поточним основним кольором виділеної області.

Інвертувати виділення, Ctrl + A – всі області, які були виділені, будуть не виділені й навпаки.

Вибрати все, Ctrl + Shift + V – виділення всієї робочої області зображення.

Скасувати виділення, Ctrl + D. Після застосування цієї команди із зображення будуть зняті всі виділення.

Коли в Paint.NET виділено деяку область, операції редагування обрізаються по краю цього виділення.

Наприклад, інструмент вибору еліпса використовувався для вибору круглої області у середині зображення. Коли використали інструмент градієнта, градієнт не вийшов за межі активного виділення (Рис. 358).

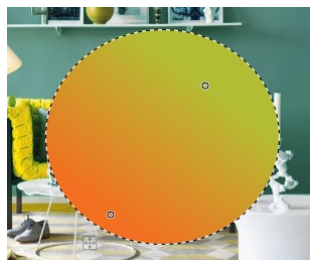


Рис. 358. Приклад обмеження операцій редагування виділеною областю
Ефекти та коригування відображатимуться також тільки у вибраній області.

РЕДАГУВАННЯ ЗОБРАЖЕНЬ

ОБРІЗКА, ЗМІНА РОЗМІРУ, ВИРІВНЮВАННЯ ЗОБРАЖЕНЬ

Меню **Зображення** містить команди, які змінюють розмір зображення або полотна, забезпечують обертання, а також дозволяють обрізати виділення або об'єднати всі шари зображення (Рис. 359).

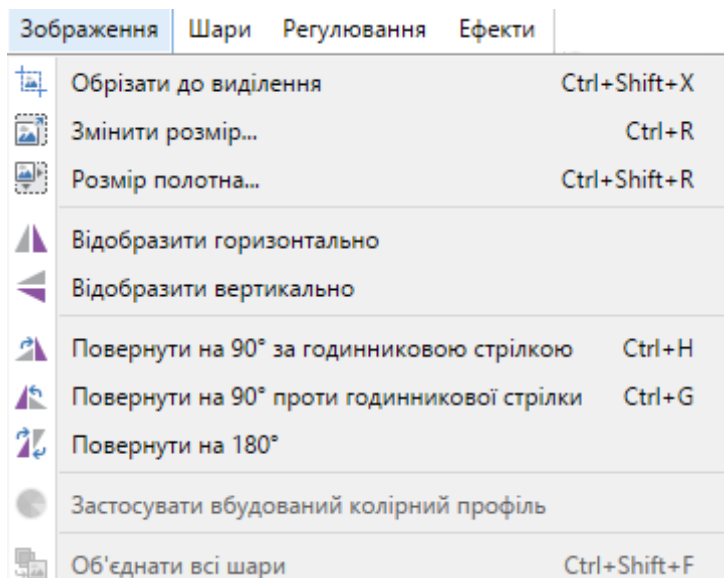


Рис. 359. Меню Зображення

Це меню відрізняється від меню **Шари** тим, що команди меню **Зображення** застосовуються до всіх шарів одночасно, а команди меню **Шари** впливають лише на поточний шар.

Обрізати з виділення, Ctrl + Shift + X – від зображення залишиться тільки та його частина, що була виділена.

Якщо область виділення була прямокутною або квадратною, то нове зображення буде в межах прямокутника або квадрата.

Якщо ж виділення було іншої форми, то як межі зображення буде використаний прямокутник, що включає область виділення. Пікселі, які не потрапили до цієї області, стануть прозорими (малюнок «шахової дошки») (Рис. 360).



Рис. 360. Прозорі області позначаються «шахівницею»

Змінити розмір, Ctrl + R – зміна розміру зображення до вказаного (Рис. 361).

Доступно два режими зміни розміру: **У відсотках** та **Абсолютний розмір**. Вони пропонують можливість зміни розміру зображення у відсотках або із зазначенням нової висоти та/або ширини.

Якщо опцію **Зберігати пропорції** ввімкнено, то встановлення нової ширини призведе до пропорційної зміни висоти та навпаки. Якщо ж ця опція відключена, то при зміні розміру зображення буде спотворене.

Розмір друку розраховується на основі роздільної здатності (DPI). При виборі одиниці вимірювання (дюйми, сантиметри) розмір пікселя буде ґрунтуватися на роздільній здатності принтера, яка за замовчуванням дорівнює 96 пікселів на дюйм (або 37.8 пікселів на сантиметр). Наприклад, якщо ви вказали, що зображення буде шириною 1 дюйм, то при роздільній здатності 96 пікселів на дюйм розмір зображення буде змінено так, що його ширина становитиме 96 пікселів.

Також ви можете змінити розмір зображення під час друку через зміну його роздільної здатності. Однак, якщо ви змінюєте роздільну здатність зображення, розмір пікселя на екрані залишається таким же, а кількість їх змінюється.

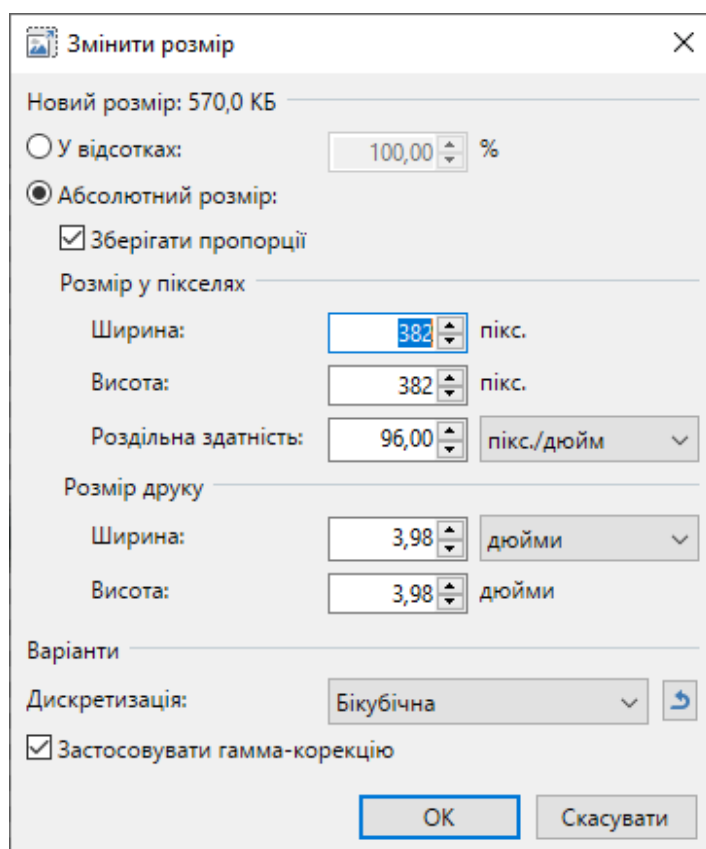


Рис. 361. Вікно Змінити розмір

Дискретизація – це процес перерахунку пікселів, у якому колір нових пікселів ґрунтується на кольорі старих. У Paint.NET можна вибрати кілька алгоритмів дискретизації зображень. Ці алгоритми допомагають керувати тим, як змінюється якість зображення при зміні його розміру (Рис. 362).

Деякі з цих алгоритмів дають кращий видимий результат, але потребують тривалого часу на обробку. Для більшості завдань цілком підійде встановлений за замовчуванням спосіб **Бікубічна**. Інші способи дискретизації (Білінійна, Метод сусідніх пікселів, Метод Фанта тощо) застосовуються лише за наявності специфічних вимог і розуміння принципу їхньої роботи.

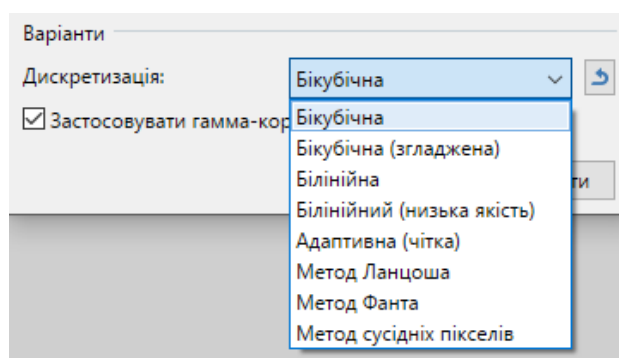


Рис. 362. Алгоритми дискретизації

Розмір полотна, Ctrl + Shift + R – збільшення або зменшення розміру полотна без зміни розміру зображення.

Вікно команди схоже на вікно **Змінити розмір**, за винятком того, що додано вибір області «закріплення» зображення (По центру, Вгорі, Вгорі справа тощо) (Рис. 363).

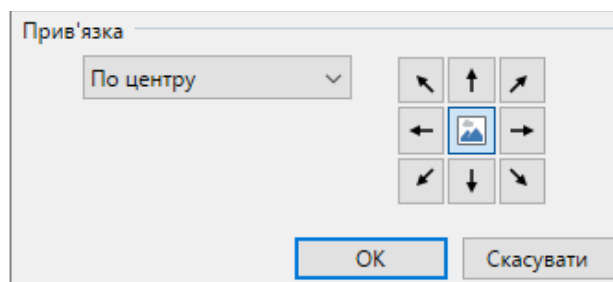


Рис. 363. Вибір прив'язки рисунка при зміні розміру полотна

Якщо полотно збільшено, нова область буде заповнена пікселями того ж кольору, що й вторинний колір у вікні **Палітра**.

Якщо полотно зменшено, краї наявного зображення поза нової області полотна будуть обрізані (Рис. 364).

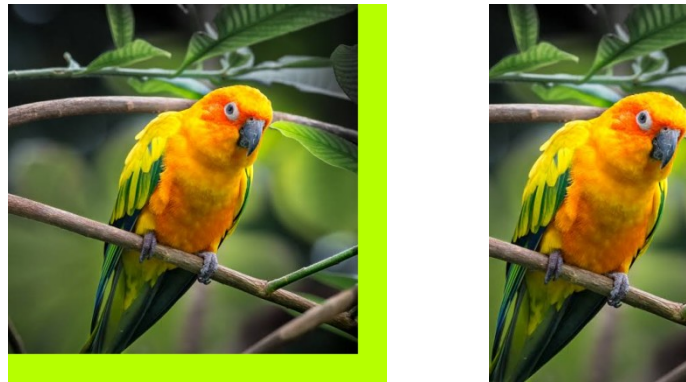


Рис. 364. Зміна розміру полотна

Відобразити горизонтально / Відобразити вертикально – відображення зображення відносно горизонтальної або вертикальної уявної осі (Рис. 365).

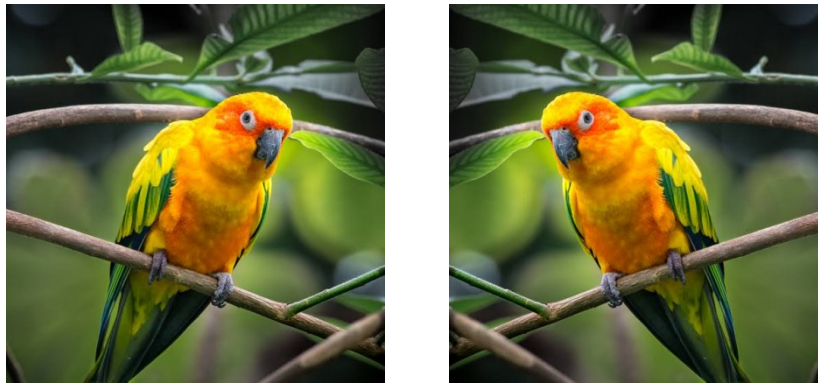


Рис. 365. Відображення зображення

Повернути за годинниковою стрілкою на 90°, Ctrl + H, Повернути проти годинникової стрілки на 90°, Ctrl + G, Повернути на 180° – ці команди дозволяють повернути зображення на 90° за годинниковою стрілкою, на 90° проти годинникової стрілки або на 180°.

Об'єднати всі шари, Ctrl + Shift + F – об'єднання всіх шарів зображення в один. Активна лише тоді, коли зображення містить більше одного шару.

КОРЕКЦІЯ КОЛЬОРУ

Для корекції кольору зображення різними способами призначене меню **Регулювання**, яке містить інструменти налаштування поточного активного шару (Рис. 366).

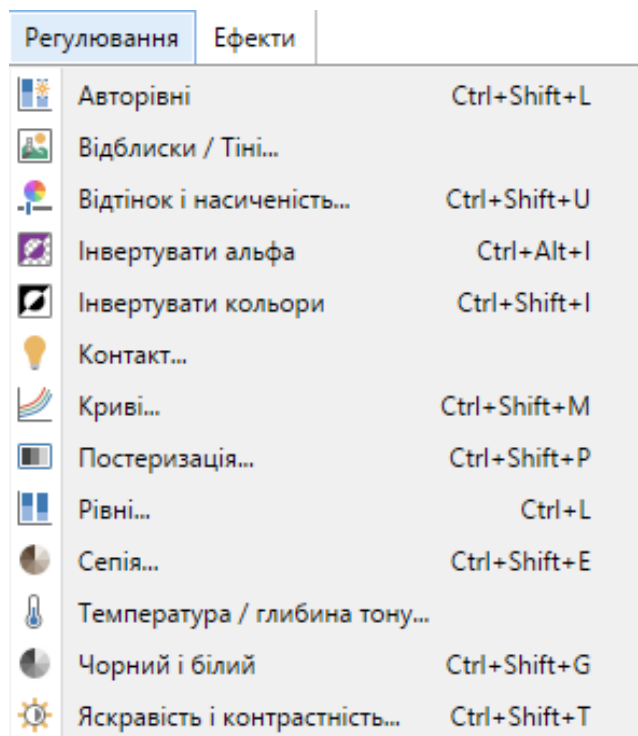


Рис. 366. Меню Регулювання

АВТОРІВНІ

Команду **Авторівні**, **Ctrl + Shift + L** використовують для автоматичного підстроювання кольору зображення. Ця функція автоматично підлаштує рівні яскравості та контрасту для покращення якості зображення, усуваючи пересвітлення чи недостатню освітленість (Рис. 367).



Рис. 367. Дія команди Авторівні

Після використання цієї команди, яка не має опцій конфігурації, наше зображення буде виглядати так.

ВІДБЛИСКИ / ТІНІ

Регулює відблиски та тіні зображення (Рис. 368).

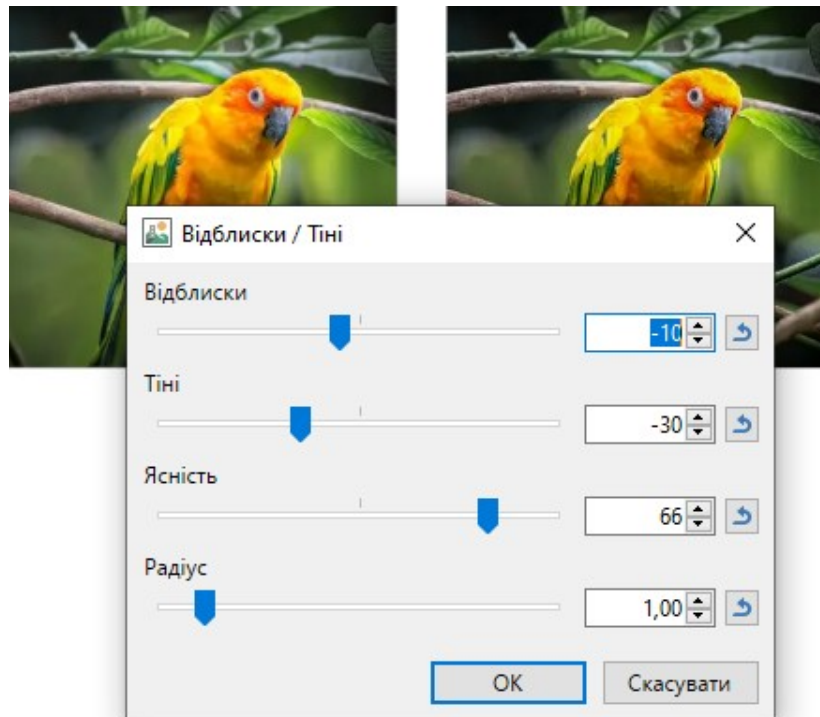


Рис. 368. Налаштування Відблисків і Тіней

ВІДТІНОК І НАСИЧЕНІСТЬ

Команда **Відтінок і насиченість**, **Ctrl + Shift + U** використовується для зміни кольору та насиченості кольору зображення.

Відтінок (Hue) – зсув кольору по колірній стрічці. Змінюючи це значення, ви змінюєте основний колір об'єктів на зображенні.

Якщо уявити, що весь колірний діапазон розташований по колу, цей елемент керування дає змогу циклічно переміщати поточну палітру по колу. Різниця між кольорами панелі зберігатиметься (Рис. 369).

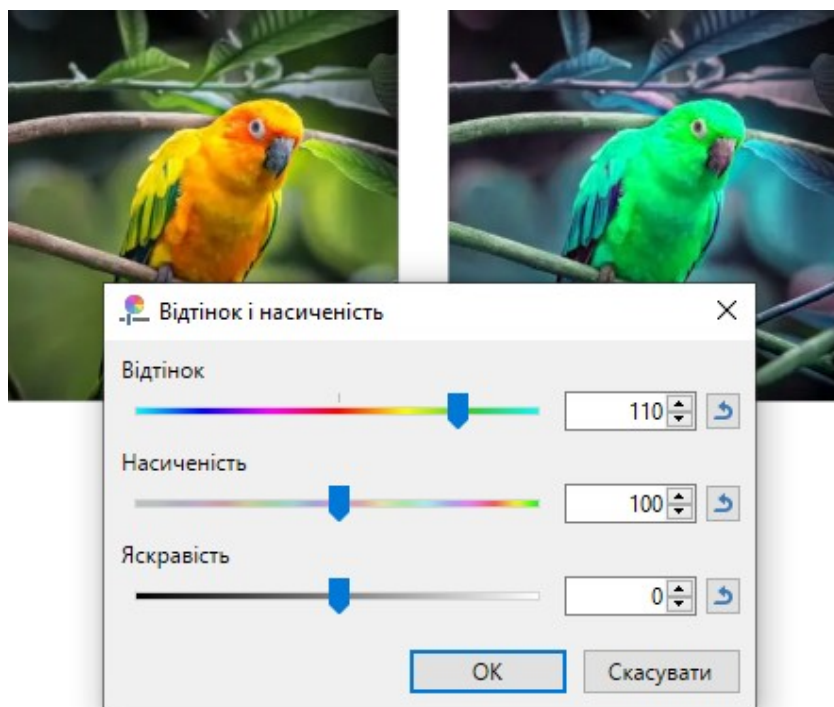


Рис. 369. Налаштування відтінку

Насиченість (Saturation) – контролює інтенсивність кольору. Зменшення цього параметра робить кольори блідими, а збільшення – яскравими (Рис. 370).

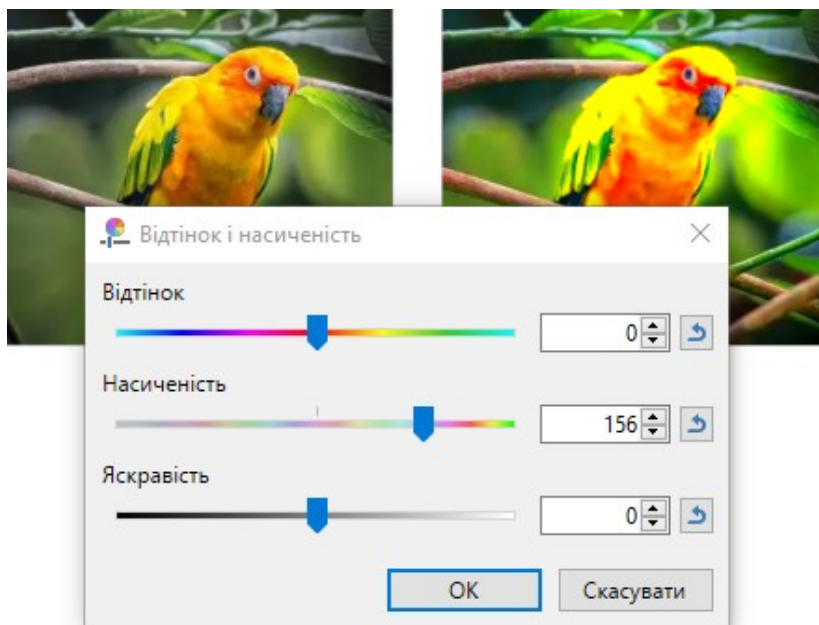


Рис. 370. Налаштування насиченості

Яскравість (Lightness) – додає або забирає світло з кольорів (Рис. 371).

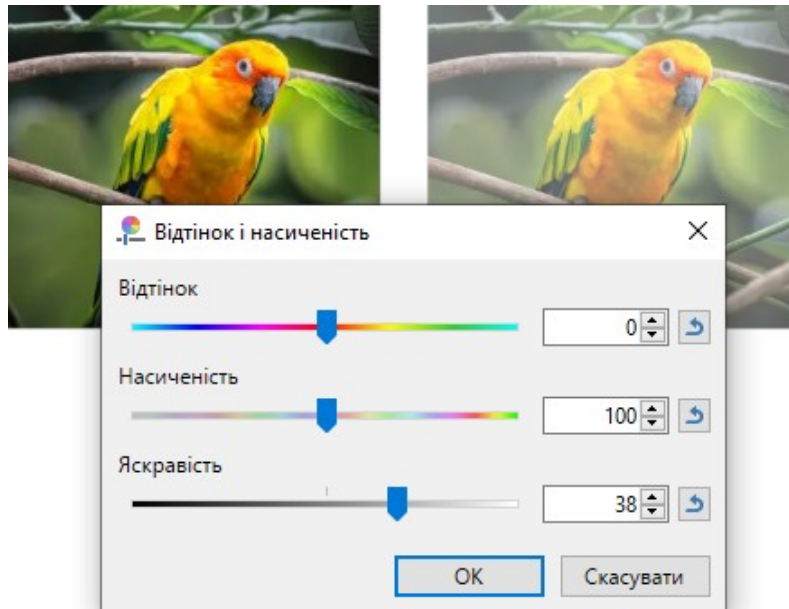


Рис. 371. Налаштування яскравості

Початкове значення елемента управління **Яскравість** дорівнює нулю. Збільшення значення збільшує яскравість і робить зображення яскравішим. Зменшення значення зменшує яскравість, роблячи кольори більш темними.

ІНВЕРТУВАТИ АЛЬФА

Команда **Інвертувати альфа**, **Ctrl + Alt + I** використовується для обробки прозорості (альфа-каналу) зображення. Ця функція змінює значення альфа-каналу на протилежне: всі прозорі області стають непрозорими, а непрозорі – прозорими. Вона працює лише з прозорістю, не впливаючи на інші кольорові компоненти (Рис. 372).

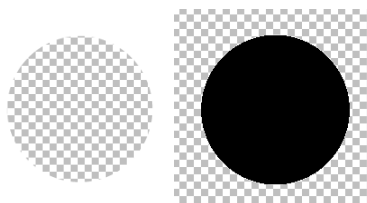


Рис. 372. Дія команди Інвертувати альфа

ІНВЕРТУВАТИ КОЛЬОРИ

Ефект команди **Інвертувати кольори**, **Ctrl + Shift + I** дуже схожий на отримання негативу фотографії. Повторне застосування повертає вихідне

зображення. **Інвертувати кольори** не має параметрів, що настраюються, і не відображає діалогове вікно (Рис. 373).



Рис. 373. Дія команди Інвертувати кольори

КОНТАКТ (ЕКСПОЗИЦІЯ)

Це налаштування збільшує або зменшує експозицію зображення. У фотографії **експозиція** – це кількість світла, яка потрапляє на плівку або матрицю камери. На отриманому зображенні видиме освітлення буде збільшено (що зробить зображення світлішим) або зменшено (що зробить його темнішим) (Рис. 374).

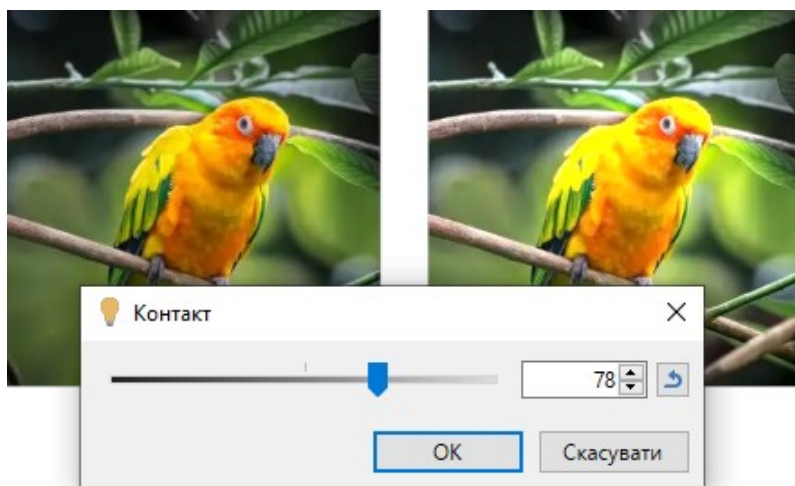


Рис. 374. Налаштування експозиції

КРИВІ...

Пункт меню **Криві**, **Ctrl + Shift + M** можна використовувати для налаштування кольорів та яскравості зображення з дуже високим ступенем контролю над цим процесом.

У деяких випадках цей інструмент забезпечує такі можливості налаштування, яких не можна досягти, за допомогою інших інструментів, наприклад **Рівні...** або **Яскравість і контрастність...**

Одразу після відкриття вікна діалогу команди **Криві...** бачимо графік, на якому вісь X представляє вхідні значення яскравості, а вісь Y – нові значення яскравості.

Переміщення кривої яскравості у верхню половину графіка (вище за діагональну лінію), призведе до того, що зображення стає яскравішим. І навпаки – переміщення контрольної точки кривої нижче діагональної лінії (у нижню половину графіка) призведе до потемніння зображення (Рис. 375).

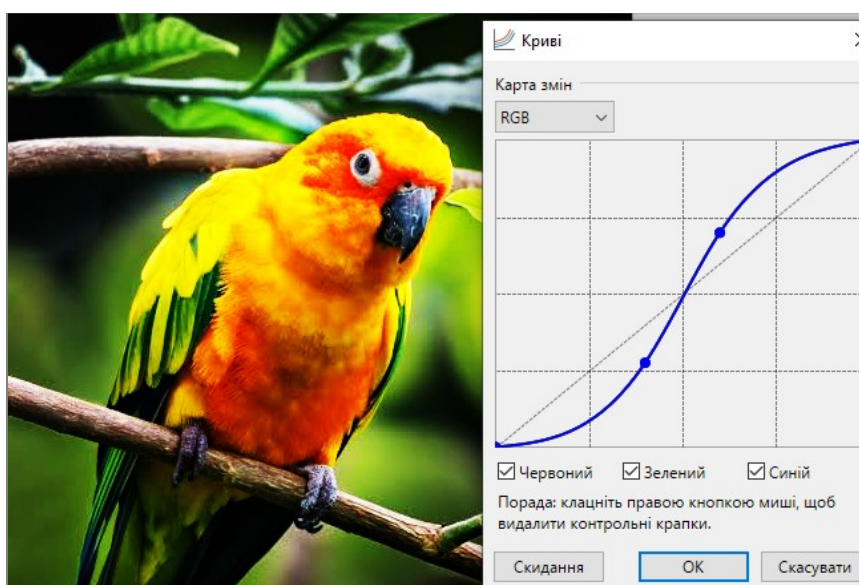


Рис. 375. Налаштування яскравості зображення командою *Криві*

Контрольні точки використовуються для налагодження лінії графіка.

Для додавання контрольної точки потрібно просто клацнути лівою кнопкою миші в будь-якому місці графіка. Видалення контрольної точки здійснюється клацанням по ній правою кнопкою миші.

Для переміщення контрольної точки просто перетягніть її на бажане місце. Зауважте, що немає способу отримати дві точки на одній вертикалі (точно одна над іншою), позаяк це означатиме, що одну і ту ж інтенсивність потрібно одночасно зменшити і збільшити, що неможливо.

Нижче наведено приклади використання команди **Криві...** для налаштування кольору зображення. На першому прикладі показано освітлення зображення за допомогою кривих (Рис. 376), а на другому затемнення зображення (Рис. 377).

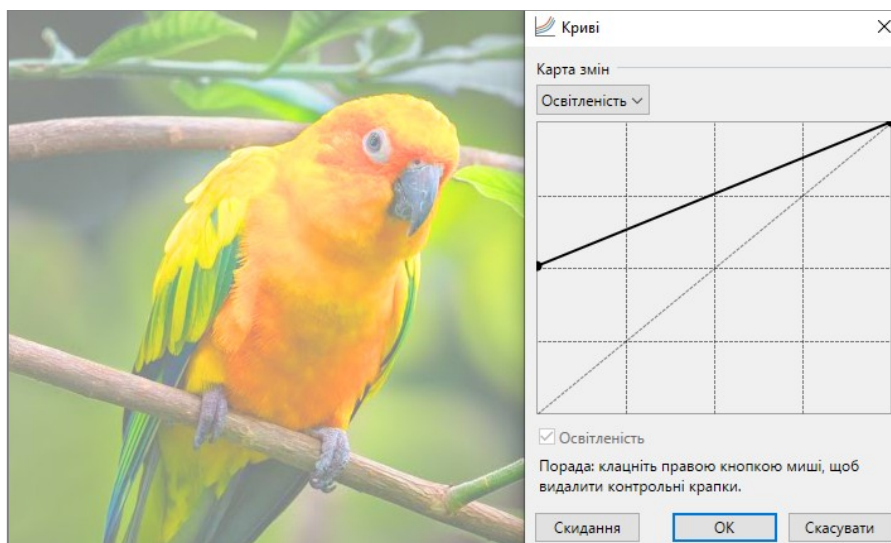


Рис. 376. Налаштування кольору зображення (освітлення) командою *Криві*

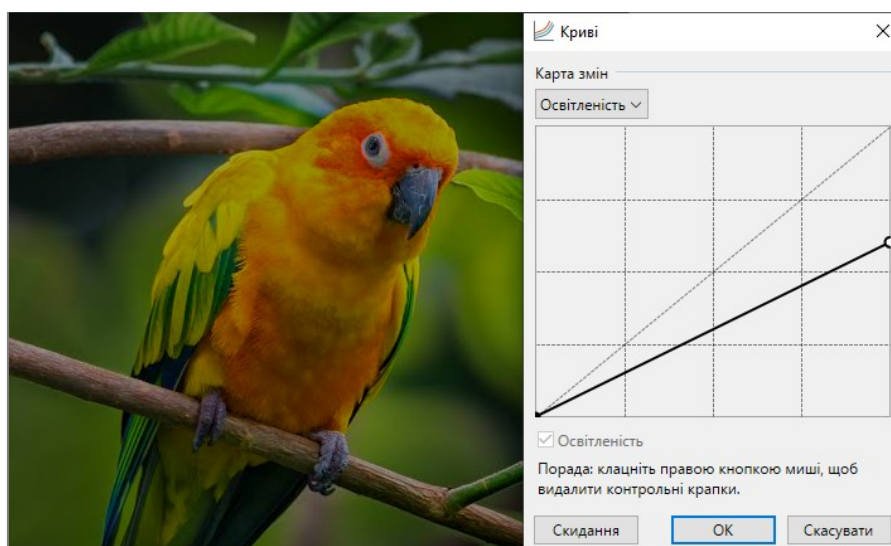


Рис. 377. Налаштування кольору зображення (затемнення) командою *Криві*

Корекція RGB-каналів. За замовчуванням команда **Криві...** використовується для налаштування освітлення зображення. Як альтернативу команда може використовувати окремі канали кольорів Red (червоний), Green (зелений), і Blue (блакитний). Для цього виберіть у списку (вгорі вікна діалогу команди) опцію **RGB** (Рис. 378).

У цьому режимі нові контрольні точки впливатимуть на колірні канали, вибрані за допомогою перемикачів (чекбоксів), які розташовані під графіком.

Для роботи з одним або парю каналів просто змініть стан перемикачів, вибравши потрібні канали.

Нижченаведене зображення було відкориговано за допомогою команди **Криві...**, з метою збільшення контрастності та насиченості кольору.

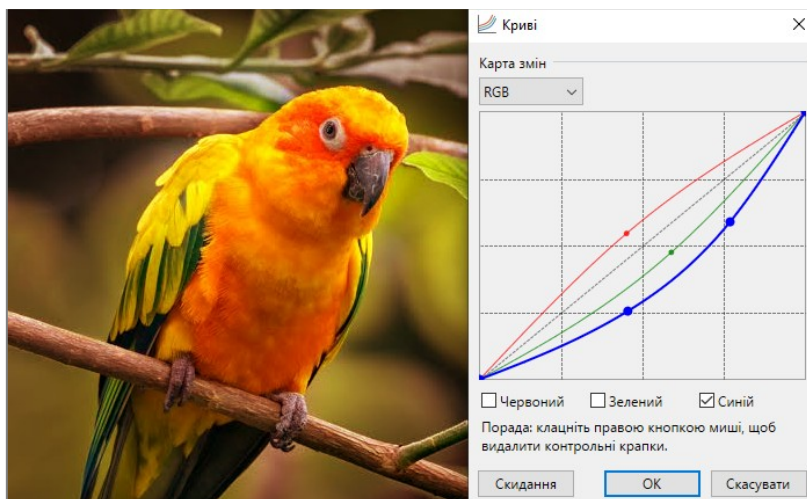


Рис. 378. Корекція RGB-каналів командою *Криві*

За допомогою команди **Криві...** кольори зображення можна навіть повністю інвертувати. Такий результат неможливо отримати за допомогою інших команд регулювання, таких як **Рівні...** та **Відтінок та насиченість...** (Рис. 379).

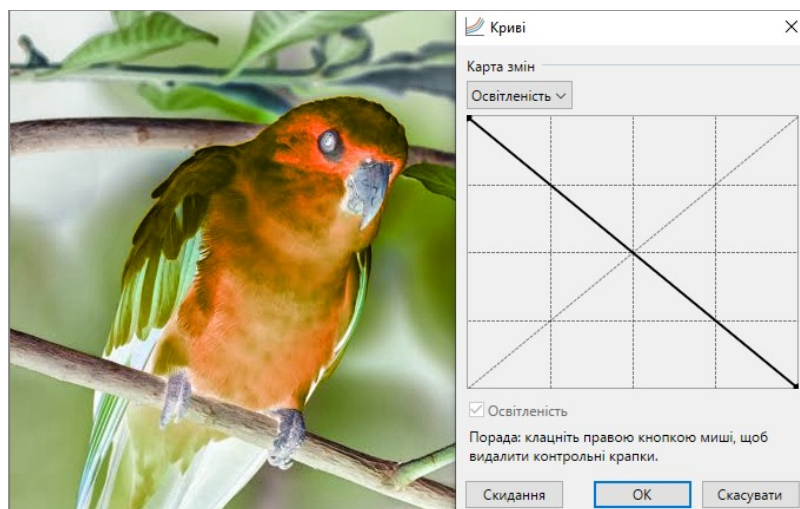


Рис. 379. Інвертування кольорів командою *Криві*

ПОСТЕРИЗАЦІЯ...

Команда **Постеризація**, **Ctrl + Shift + P** зменшує кількість кольорів, які може використовувати кожен піксель. Це дає змогу «постеризувати» зображення, надавши йому стиль «ретро» (Рис. 380).

Зазвичай кожен із колірних каналів RGB (червоний, зелений і синій) має 256 можливих значень (від 0 до 255). Ця установка обмежує діапазон від 2 до 64 можливих значень.



Рис. 380. Постеризація зображення

Колірними каналами RGB можна керувати незалежно один від одного, якщо знято прапорець **Залежні кольори**. Якщо ж цей прапорець встановлено, елементи керування змушені використовувати те саме значення (Рис. 381).

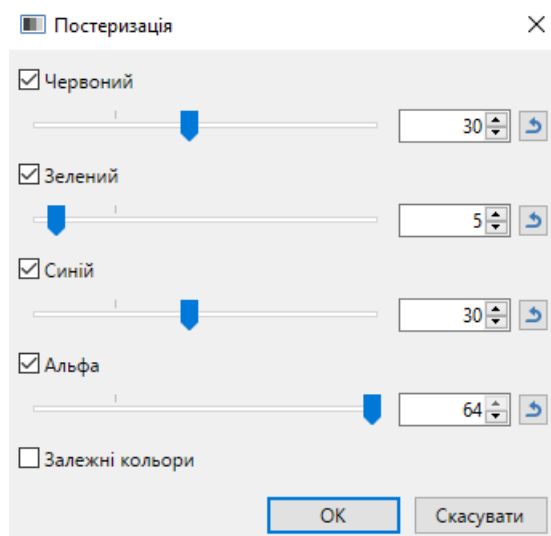


Рис. 381. Вікно налаштування параметрів постеризації

РІВНІ...

Ця операція використовується для регулювання діапазону (або експозиції) кольору та гами зображення, включаючи поканальну гама-корекцію. Більшість регулювань, що виконуються за допомогою команди **Рівні**, здійснюється шляхом зміни вихідних значень білих, сірих і чорних пікселів.

Рівень білого визначає значення найсвітлішої точки на зображенні, тоді як рівень чорного – значення найтемнішого зображення. Рівень сірого становить середнє значення кольорів усього зображення. Кожну з цих точок кольору зображення можна налаштувати незалежно за допомогою рівнів.

Діалогове вікно налаштування рівнів (Рис. 382) розділене навпіл, ліва частина є кольірним діапазоном у вихідному зображенні (Вхід). Права частина є зображенням після внесення коригувань (Вихід).

З лівого боку знаходяться два зразки кольору, білий (значення 255) і чорний (значення 0). Вони вказують на вхідні білі та чорні пікселі.

Праворуч відображаються три кольорні зразки: по одному для вихідних білих і чорних пікселів, а інший для вихідної точки сірого.

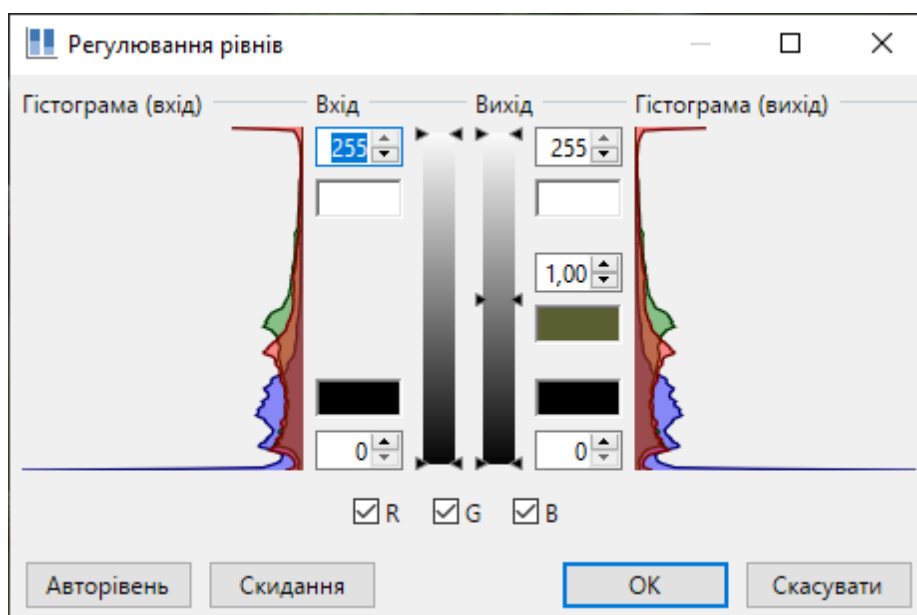


Рис. 382. Вікно Регулювання рівнів

За умовчанням вихідна гістограма повністю еквівалентна вхідній. Однак при кожній зміні налаштувань у цьому вікні, вихідна гістограма буде змінюватися, щоб відобразити зміни, що відбудуться з вихідним зображенням.

Ці налаштування можна змінювати за допомогою введення числових значень, повзунка або подвійним клацанням лівою клавiшею мишки по зразку кольору (Рис. 383).

Біла точка з лівого боку (Вхід) – це елемент керування, який при налаштуванні стає білою точкою з правого боку (Вихід). Коли регулюється точка білого, яскравіші

кольори зображення будуть затемнені, щоб відповідати вихідній білій точці. Так само налаштовуються сіра та чорна точки.

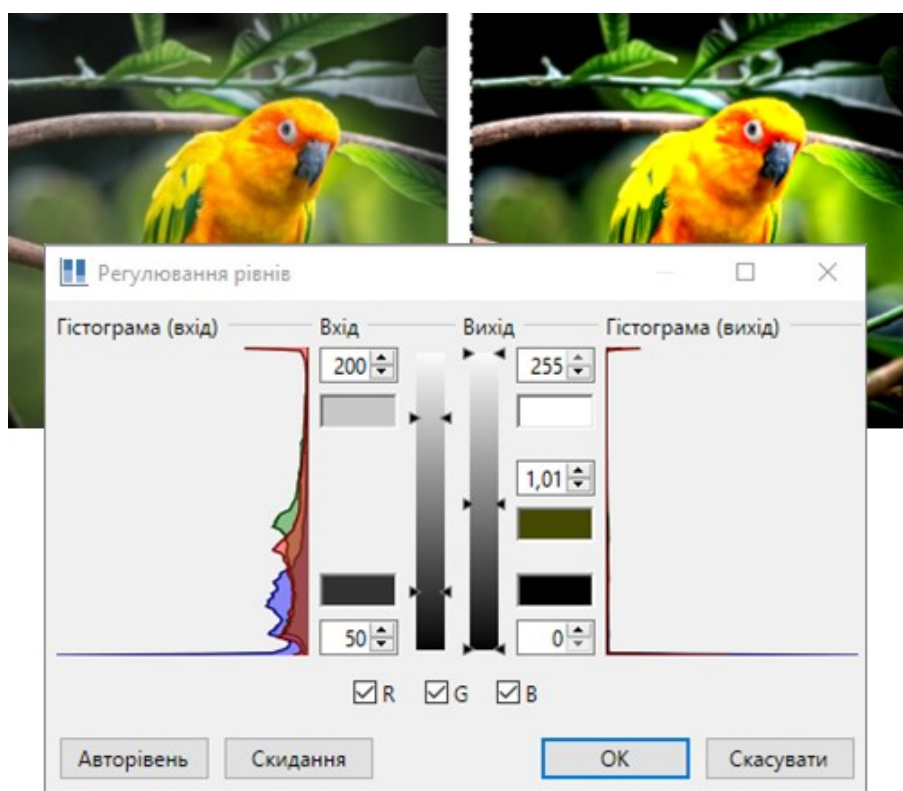


Рис. 383. Зміна налаштувань у вікні Регулювання рівнів

Якщо розтягнути вихідну гістограму в тонкий шар, це призводить до підвищення контрасту (Рис. 384).

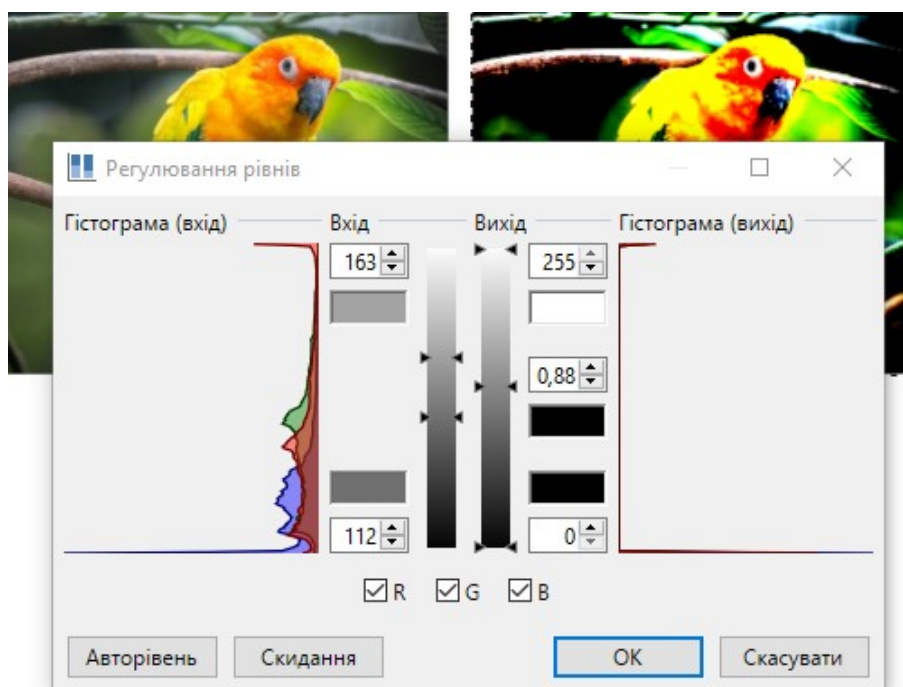


Рис. 384. Підвищення контрасту зображення за допомогою регулювання рівнів

Якщо зтиснути гістограму в один пік, зображення буде мати дуже маленький контраст (Рис. 385).

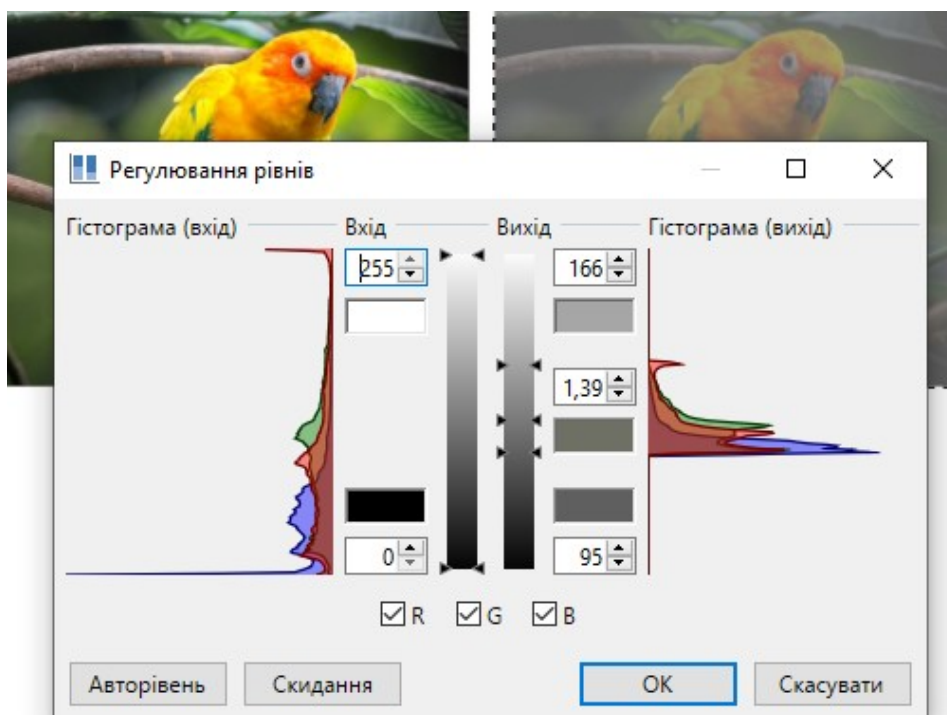


*Рис. 385. Зниження контрасту зображення
за допомогою регулювання рівнів*

Можливі будь-які налаштування між цими двома крайнощами.

Якщо вихідна сіра точка налаштована так, щоб вона була трохи яскравішою (перемістіть середній повзунок вгору або збільште числове значення), це збільшить яскравість загального кольору зображення. Білі та чорні кольори зображення не змінюються (Рис. 386).

СЕПІЯ

Це коригування перетворює зображення на чорно-біле, а потім додає тон «сепія». Це можна використовувати для надання зображенням вигляду старої фотографії (Рис. 388).

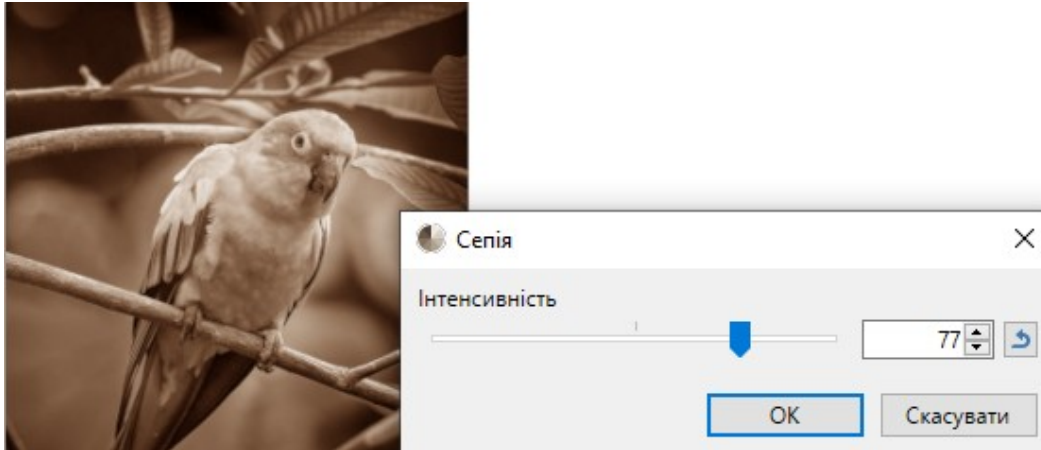


Рис. 388. Ефект Сепія

ЗРОБИТИ ЧОРНО-БІЛИМ

Це коригування видаляє всі кольори, роблячи шар чорно-білим (Рис. 389).



Рис. 389. Чорно-біле зображення

ЯСКРАВІСТЬ І КОНТРАСТНІСТЬ...

Цю команду можна використовувати для регулювання яскравості та контрастності зображення (Рис. 390).

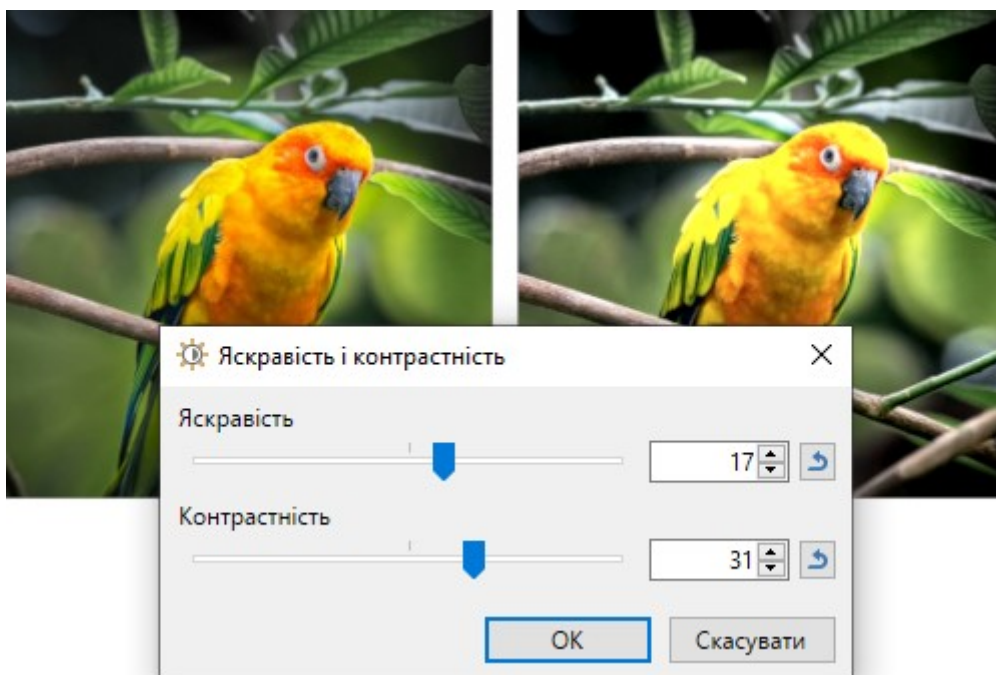


Рис. 390. Регулювання яскравості та контрастності зображення

РЕТУШУВАННЯ ТА ФІЛЬТРИ

Команди меню **Ефекти** використовуються для зміни поточного шару або виділення і містить дев'ять підменю: **Колір**, **Об'єкт**, **Обробка**, **Розмиття**, **Спотворення**, **Стилізування**, **Фотографія**, **Художні** та **Шум** (Рис. 391).

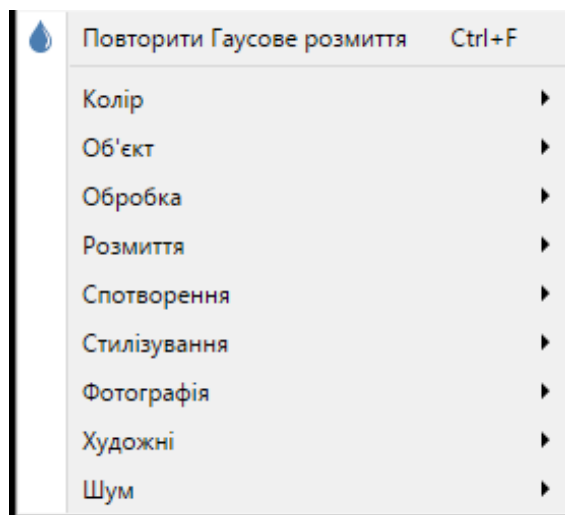


Рис. 391. Команди меню Ефекти

Команда **Повторити**, **Ctrl + F** дає змогу швидко повторити останній ефект, який використовувався. Водночас будуть використані такі ж параметри ефекту, що і останній раз. Ця команда активна лише після застосування хоча б одного ефекту.

КОЛІР

Ефект квантування в Paint.NET дає змогу зменшити кількість кольорів на зображенні, що робить його вигляд спрощеним або стилізованим, схожим на плакат чи ретро-графіку. Це подібно до обмеження кількості кольорів (або «квантування»), яке часто використовується в старих комп'ютерних графіках і мультимедіа.

Кількість кольорів – головний параметр, який визначає, скільки кольорів залишиться на зображенні після квантування. Менша кількість кольорів зробить зображення більш спрощеним і стилізованим (Рис. 392).

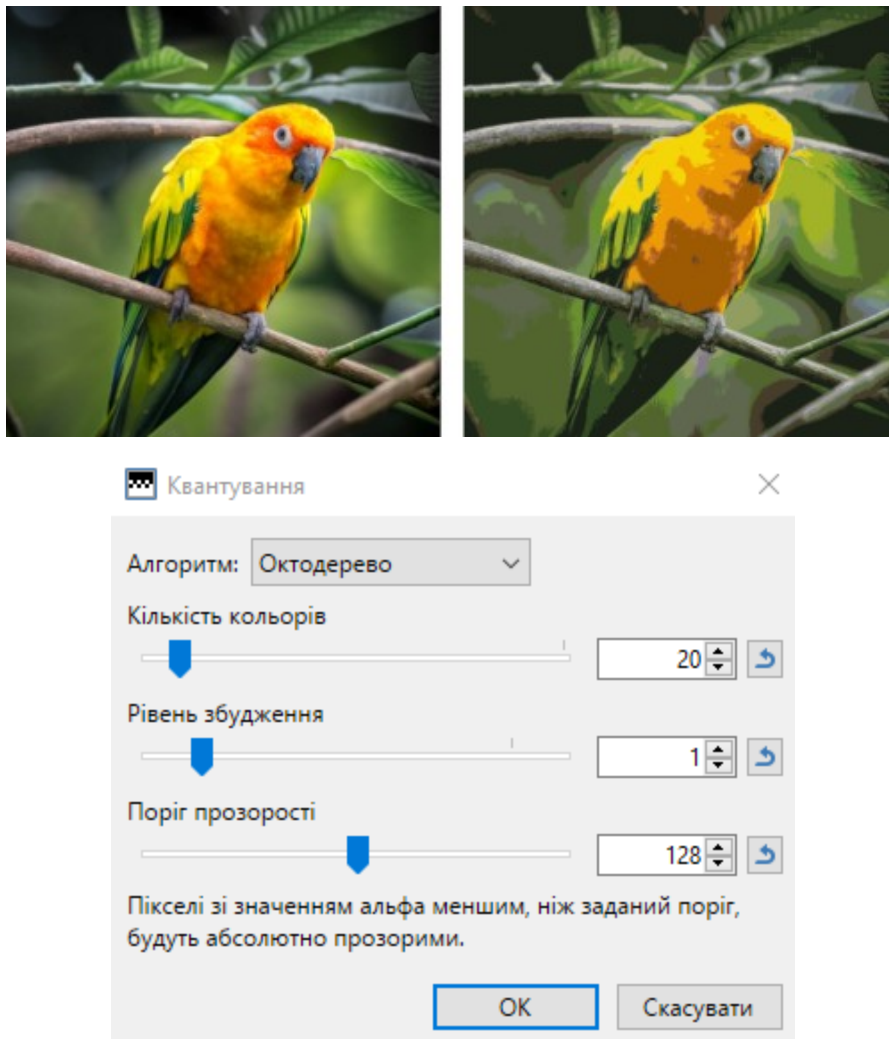


Рис. 392. Налаштування кількості кольорів зображення

ОБ'ЄКТ

У Paint.NET до зображення можна додати **ефект тіні** з вказаними параметрами (Рис. 393).

Радіус – визначає, наскільки розмитою буде тінь. Що більший радіус, то більше розмиття, що робить тінь м'якішою та менш різкою. Високий радіус підходить для великих тіней, які мають виглядати більш натуральними.

Відстань – цей параметр контролює відстань між об'єктом і тінню. Що більше значення, то далі тінь розміщується від об'єкта.

Кут – кут визначає напрямок тіні відносно об'єкта. Змінюючи кут, можна налаштувати, з якого боку падає тінь (наприклад, зліва, справа, знизу чи зверху).

Прозорість – цей параметр керує прозорістю тіні. Якщо встановити меншу прозорість, тінь буде менш насиченою і менш помітною. Вища прозорість робить тінь темнішою і більш виразною.

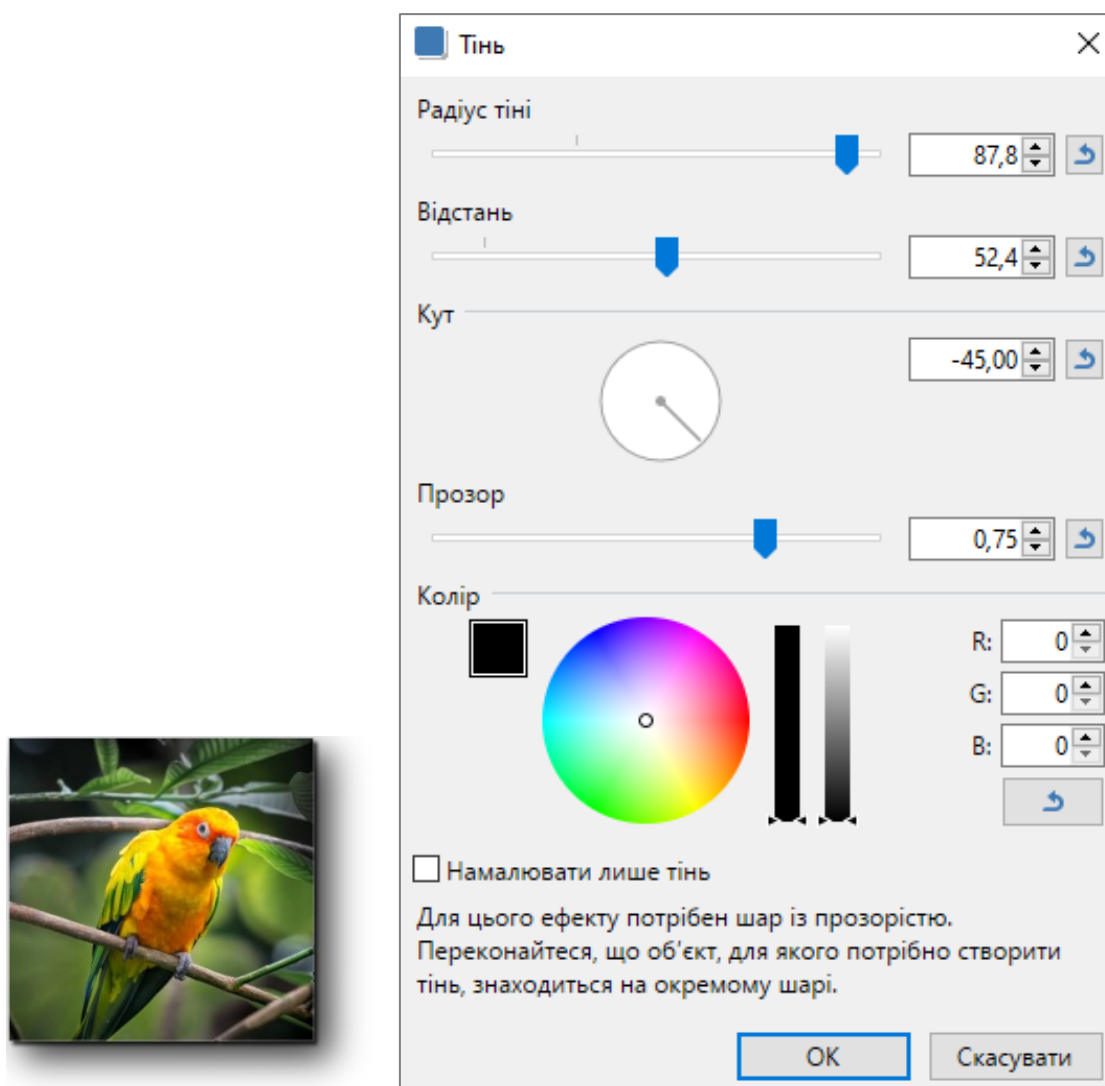


Рис. 393. Налаштування тіні зображення

ОБРОБКА

Це підменю містить ефекти, що створюють візуалізації, які зазвичай замінюють те, що вже було на вибраному шарі (Рис. 394).

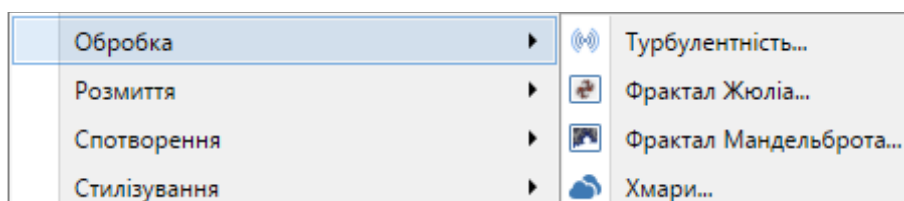


Рис. 394. Команди підменю Обробка

ТУРБУЛЕНТНІСТЬ

Цей ефект пов'язаний із шумом і мало чим відрізняється від кольорової версії хмар. Колір ефекту **Турбулентність** встановлена і не може бути налаштована в діалоговому вікні ефекту (Рис. 395).

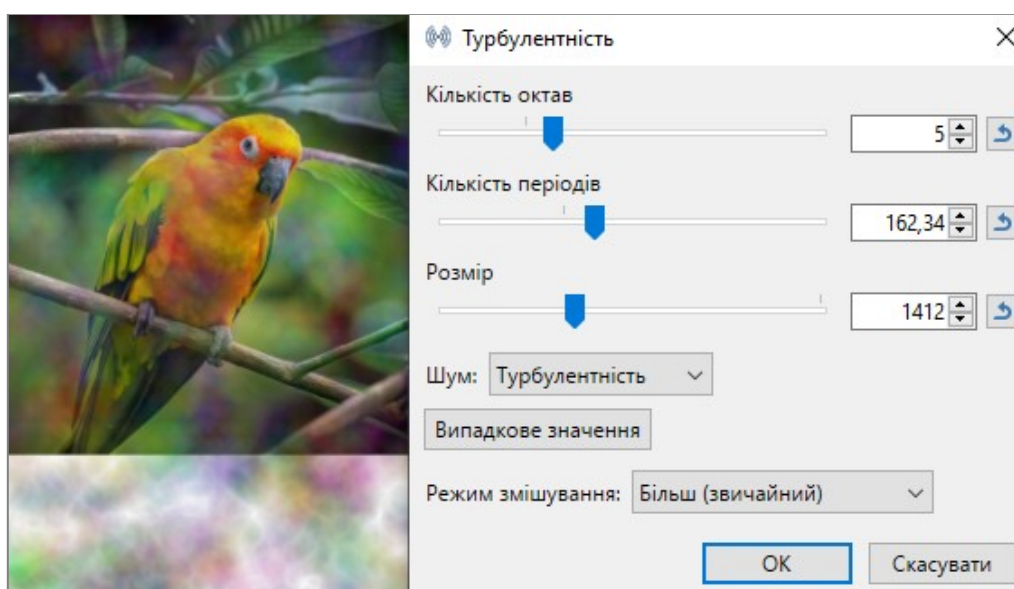


Рис. 395. Налаштування ефекту Турбулентності

Кількість октав визначають кількість плям. Низькі налаштування створюють менше плям, а вищі – збільшують їх.

Кількість періодів – це масштабний коефіцієнт і використовується для ефективного збільшення або зменшення масштабу.

Розмір регулює розмір візерунка. Що вище значення, то більший візерунок.

Параметр **Шум** має два варіанти: **Турбулентність** і **Фрактальна сума**. Фрактальна сума дає більш яскраві та барвисті ефекти.

Режим змішування визначає, як змішуватимуться хмари з вихідним вмістом шару або виділення. Для отримання додаткової інформації див. розділ **Режими змішування**.

ФРАКТАЛ ЖЮЛІА...

Цей ефект генерує на шарі фрактал Жюліа.

У діалоговому вікні ефекту можна налаштувати насиченість кольору, масштаб фракталу, кут обертання фракталу, якість деталей і режими змішування (Рис. 396). Кольори цього фракталу не налаштовуються.

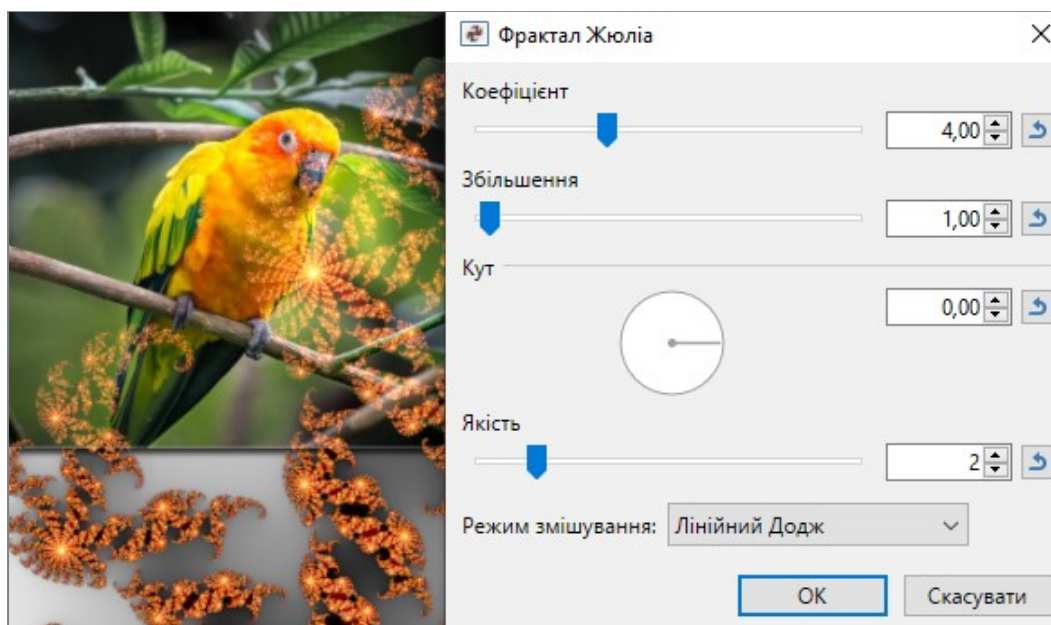


Рис. 396. Налаштування фракталу **Жюліа**

Коефіцієнт визначає насиченість кольору. Низькі налаштування створюють монохромний фрактал, тоді як високі налаштування збільшують яскравість кольору.

Збільшення – збільшує чи зменшує масштаб фракталу.

Кут – обертає фрактал навколо центру виділення або шару.

Якість – визначає кількість деталей. Нижчі налаштування надають фракталу нерівний вигляд, а вищі налаштування забезпечують більш плавні лінії.

Режим змішування визначає, як змішуватиметься фрактал із вихідним вмістом шару або виділення. Для отримання додаткової інформації див. розділ **Режими змішування**.

ФРАКТАЛ МАНДЕЛЬБРОТА

Цей ефект генерує на шарі фрактал Мандельброта.

У діалоговому вікні ефекту можна налаштувати ті ж параметри, що і для фракталу Жюліа (Рис. 397). Кольори фракталу Мандельброта не налаштовуються, за винятком опції **Інвертувати кольори**.

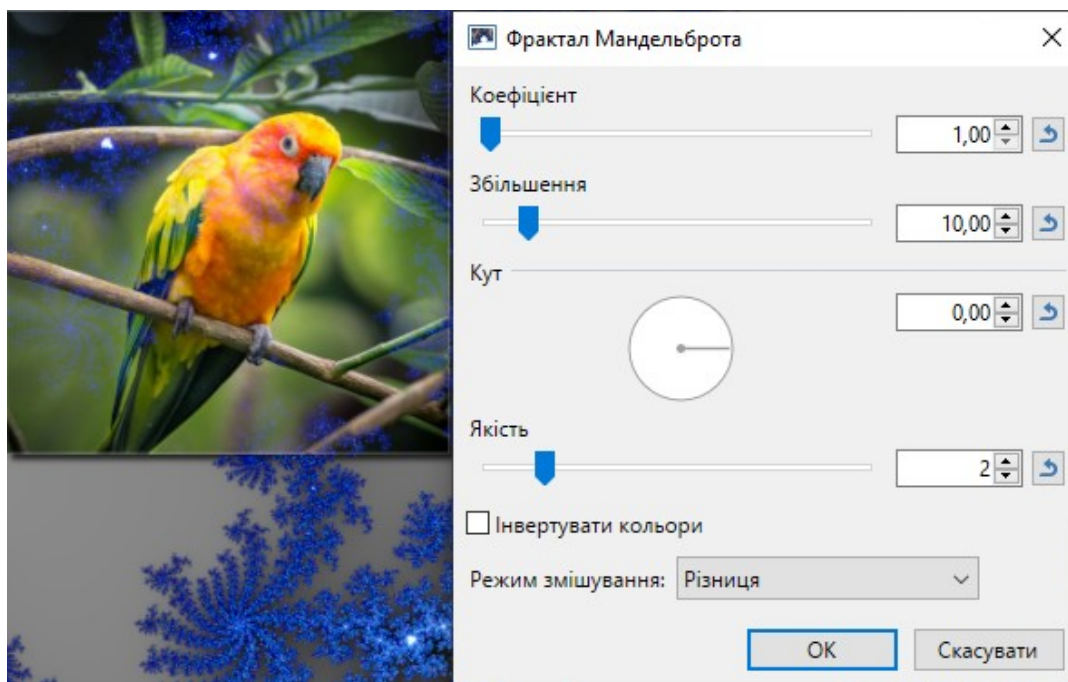


Рис. 397. Налаштування фракталу Мандельброта

Коефіцієнт визначає насиченість кольору. Низькі налаштування створюють монохромний фрактал, а високі налаштування підвищують яскравість кольору.

Збільшення – масштабує фрактал.

Кут – обертає фрактал навколо центру виділення або шару.

Якість визначає кількість деталей. Нижчі налаштування надають фракталу нерівний вигляд, а вищі – плавніший.

Встановлення прапорця **Інвертувати кольори** має той самий ефект, що й команда **Регулювання** → **Інвертувати кольори**. Кольори будуть замінені на протилежні.

Режим змішування визначає, як змішуватиметься фрактал із вихідним вмістом шару або виділення. Для отримання додаткової інформації див. розділ **Режими змішування**.

ХМАРИ...

Створює випадковий «хмарний» ефект для шару або виділення за допомогою Основного та Додаткового кольорів (Рис. 398).

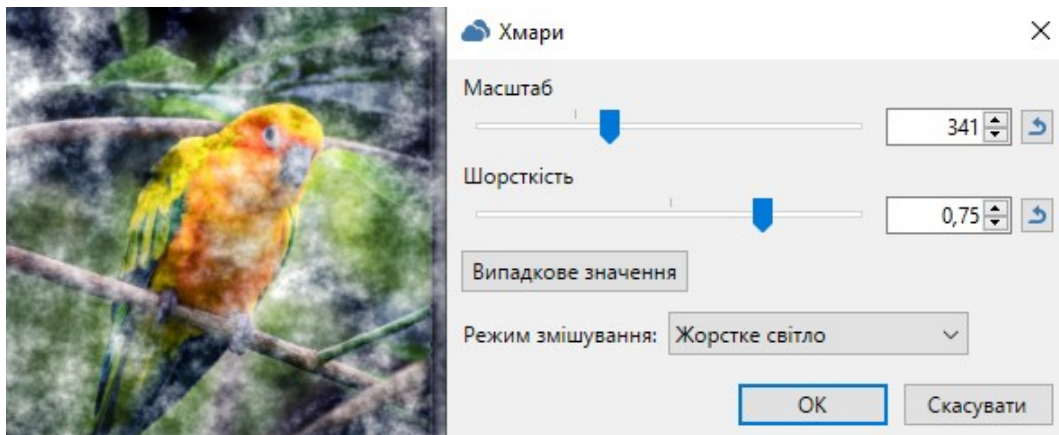


Рис. 398. Налаштування ефекту «Хмари»

Масштаб – установлює розмір хмаринок.

Шорсткість – це показник того, наскільки грубими будуть хмари. Низькі значення створюють ефект розмиття, а високі значення створюють чимало дрібних деталей.

Режим змішування визначає, як змішуватимуться хмари з вихідним вмістом шару або виділення. Для отримання додаткової інформації див. розділ Режими змішування.

Випадкове значення рандомізує функцію хмари, щоб створити новий ефект.

РОЗМИТТЯ

Ця група ефектів призначена для розмиття зображення різними способами (Рис. 399).

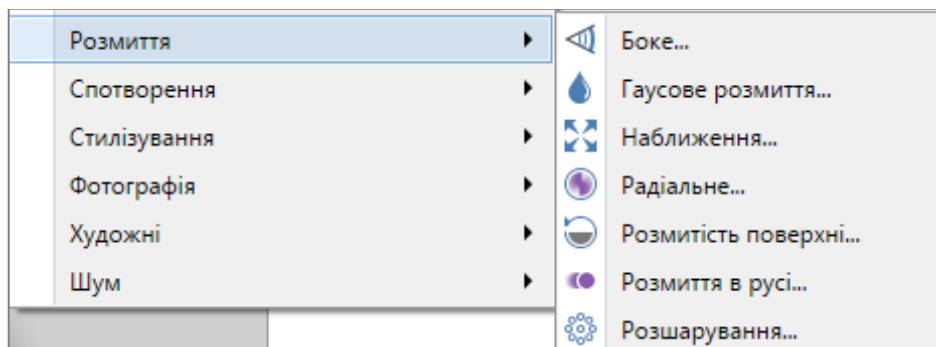


Рис. 399. Команди підменю Розмиття

БОКЕ...

Цей ефект схожий на **Гаусове розмиття**, але дає більш розпливчастий або «розфокусований» вигляд зображення (Рис. 400). Із погляду математики він застосовує більш рівномірні вагові коефіцієнти до пікселів у межах необхідного радіусу.

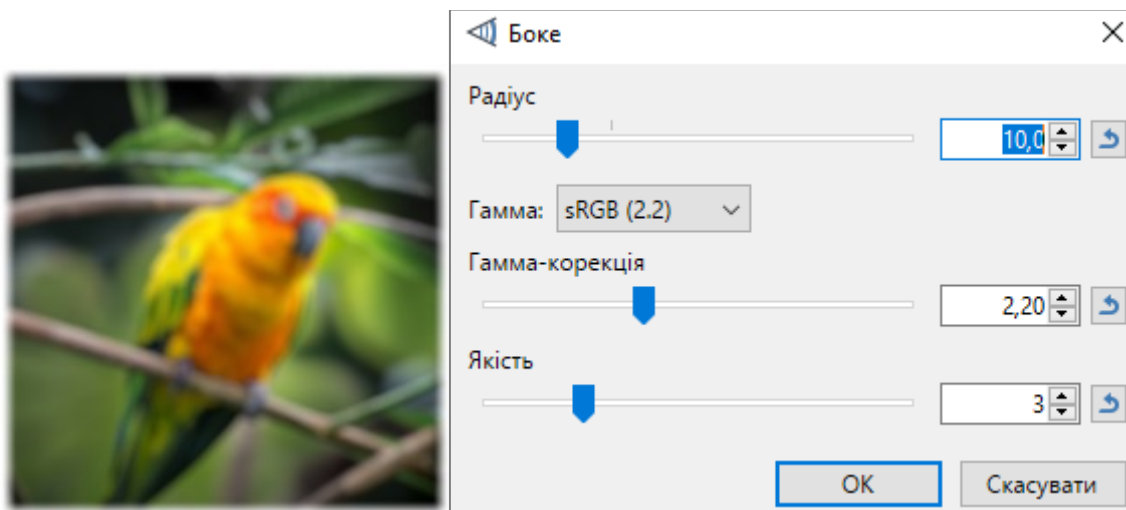


Рис. 400. Налаштування ефекту Боке

Радіус – контролює силу ефекту розмиття.

ГАУСОВЕ РОЗМИТТЯ...

Цей ефект часто використовується для пом'якшення зображень або створення м'якого фону. Розмиття по Гаусу має тільки один параметр, що настраюється: **Радіус**, який показує, наскільки широко та сильно застосовується ефект (Рис. 401).

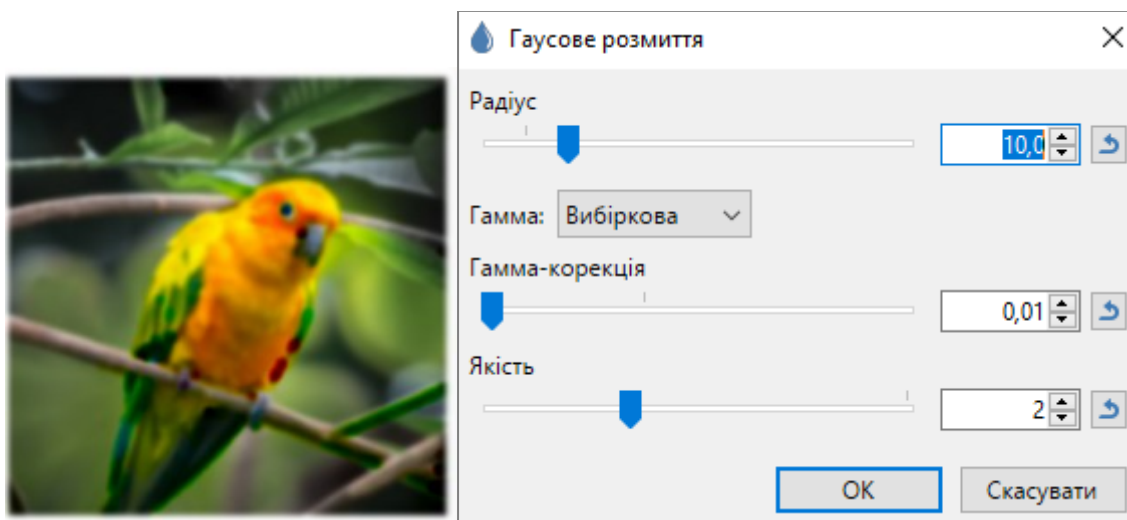


Рис. 401. Налаштування ефекту Гаусове розмиття

НАБЛИЖЕННЯ...

Цей ефект дає ілюзію руху або розмиття у трьох вимірах, що виходять із зображення (Рис. 402).

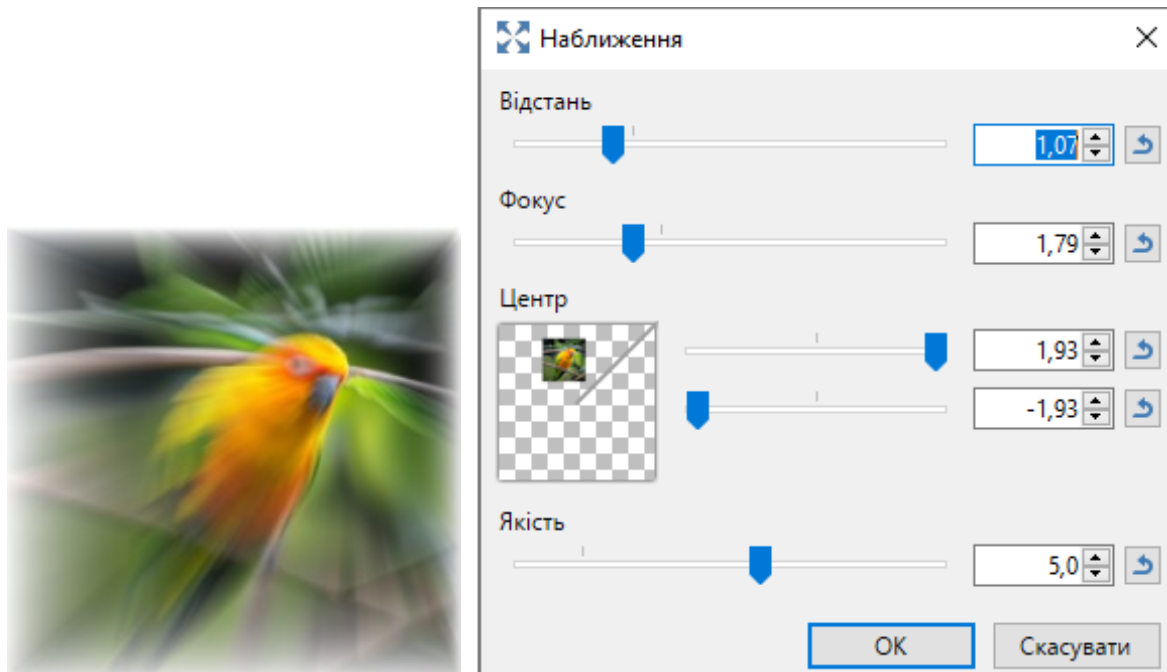


Рис. 402. Налаштування ефекту Наближення

Відстань – цей параметр контролює силу збільшення, тобто наскільки сильно буде видно ефект наближення.

Центр – точка центру розмиття. Її можна перемістити, використовуючи подвійний повзунок або перетягнувши хрестик у попередньому перегляді ескізів, щоб визначити, в якому місці буде центр розмиття.

Якість – параметр, що дає змогу збільшити якість розмиття.

РАДІАЛЬНЕ...

Цей ефект нагадує **Розмиття в русі**... окрім того, що рух об'єкта здійснюється не за прямою лінією, а навколо певного центру (Рис. 403).

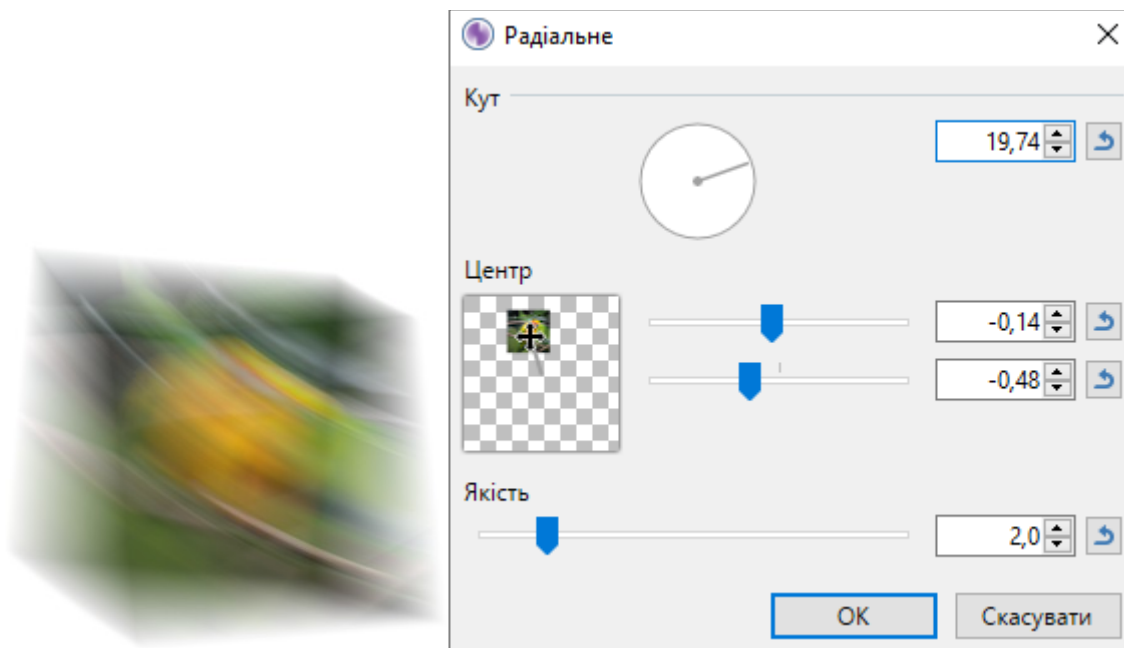


Рис. 403. Налаштування ефекту Радіальне

Кут визначає, наскільки ефект зсуває вихідні пікселі.

Центр – точка центру розмиття. Центр можна переміщати, використовуючи подвійний повзунок або перетягнувши хрестик у вікні попереднього перегляду ескізів.

Параметр **Якість** визначає якість зображення після застосування ефекту.

РОЗМИТІСТЬ ПОВЕРХНІ...

Цей ефект є корисним для розмиття м'яких деталей або шуму в зображенні, зберігаючи межі деталей і контраст. У результаті виходить спрощена версія оригіналу (Рис. 404).



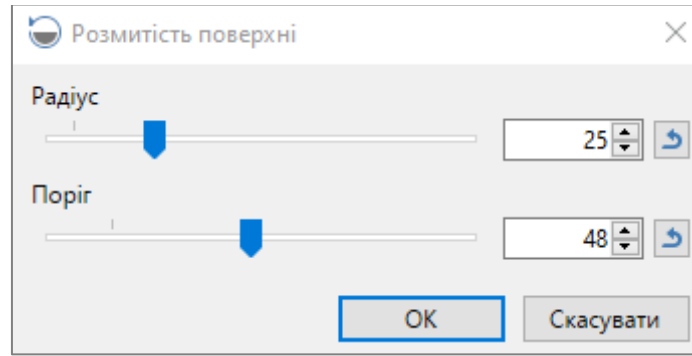


Рис. 404. Налаштування ефекту Розмитість поверхні...

Значення **Радіуса** визначає величину області, що розмивається. Збільшення радіусу призводить до більшого розмиття.

Поріг – указує, наскільки пікселі повинні відрізнятися за кольором від сусідніх, щоб їх можна було розмити. Вищі значення дають змогу зберегти більше деталей, оскільки ефект застосовується лише до значно відмінних пікселів.

РОЗМИТТЯ В РУСІ...

Цей ефект корисний для надання зображенню ілюзії прямолінійного руху, ніби фотографія була зроблена з великою витримкою (Рис. 405).

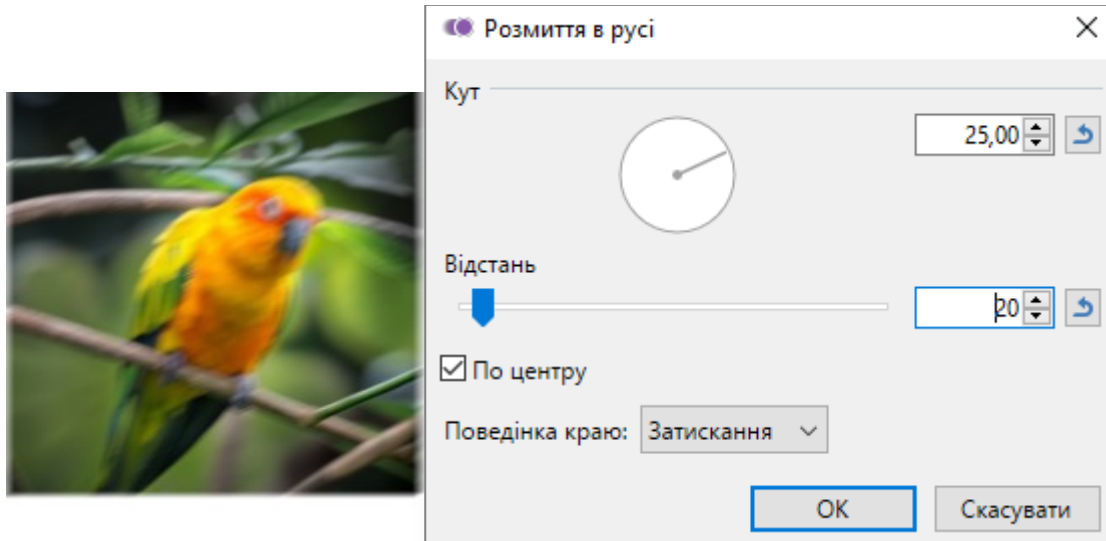


Рис. 405. Налаштування ефекту Розмиття в русі

Кут видимого руху та **Відстань** є мірами того, наскільки ефект зсуває вихідні пікселі.

РОЗШАРУВАННЯ...

Копії або фрагменти зображення відображаються із зазначеним зсувом і поворотом. Цей ефект можна використовувати для деякого розфокусування зображення або надання йому «п'яного» або «напівсвідомого» вигляду (Рис. 406).

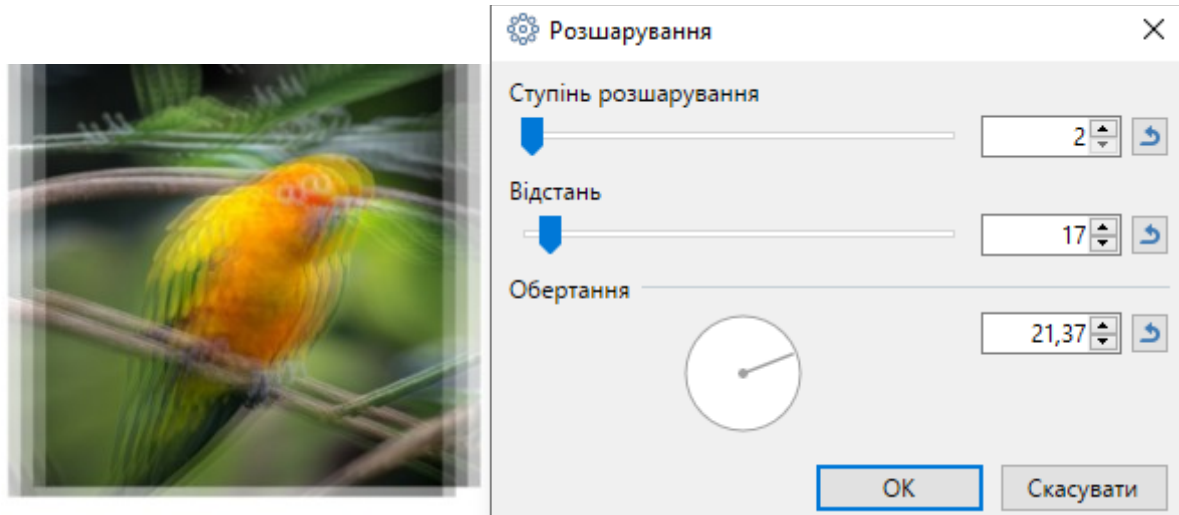


Рис. 406. Налаштування ефекту Розшарування

Указана кількість копій (**Ступінь розшарування**) перемальовується з **Відстанню** та **Обертанням** у вихідне положення, як зазначено у діалоговому вікні.

СПОТВОРЕННЯ

Група ефектів призначена для спотворення або скручування зображення часто до невпізнання (Рис. 407).

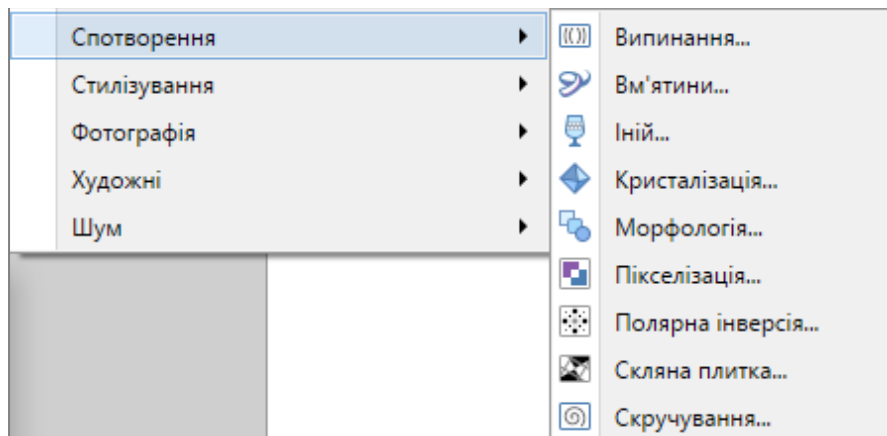


Рис. 407. Команди підменю Спотворення

ВИПИНАННЯ...

Цей ефект змушує частину шару або виділення збільшуватися або зменшуватися, якби зображення було вигнуте (Рис. 408).

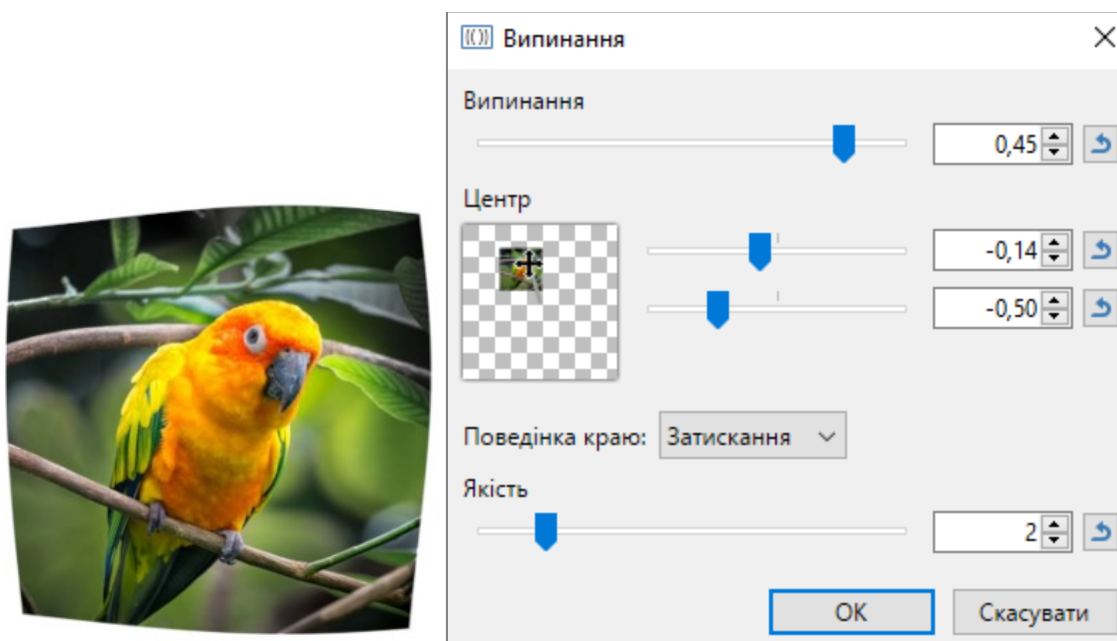


Рис. 408. Налаштування ефекту Випинання

Параметр **Випинання** задає силу ефекту.

Центр показує, звідки інтерпретуватиметься ефект. Центр можна переміщувати: використовуйте подвійний повзунок або перетягніть хрестик у вікні попереднього перегляду ескізів.

ВМ'ЯТИНИ...

Цей ефект перетворює зображення так, ніби на нього вплинули механічно, створюючи вм'ятини або виступи. Це може виглядати як деформація поверхні, коли певні частини зображення «прогинаються» всередину або «випираються» назовні. Часто використовується для створення об'ємних текстур, наприклад, для кнопок, металевих поверхонь, або інших матеріалів, що мають складну текстуру (Рис. 409).

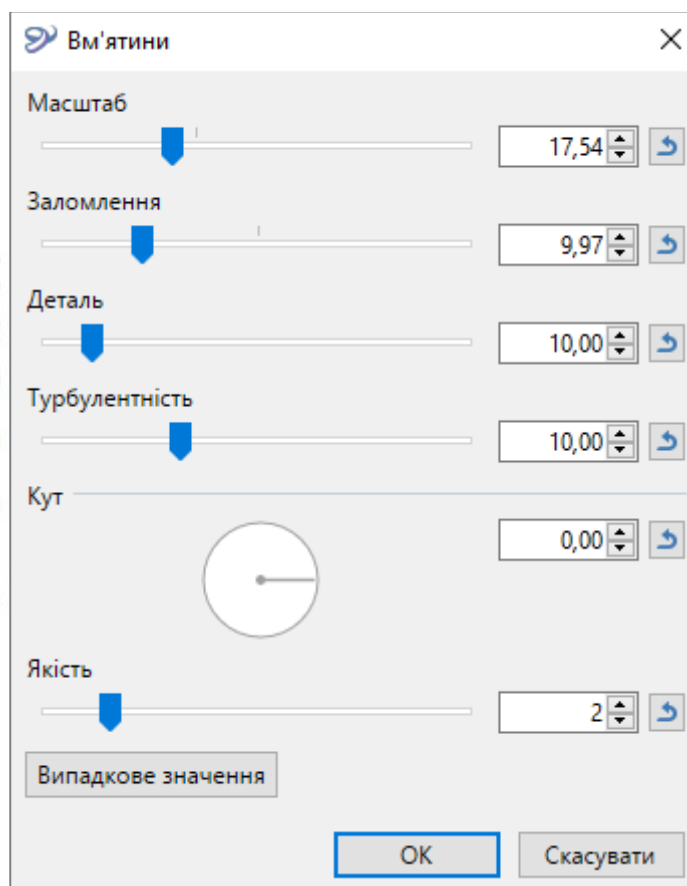


Рис. 409. Налаштування ефекту Вм'ятини

Має доволі багато параметрів налаштування:

Масштаб визначає, якого розміру будуть розірвані смужки. Низькі значення створюють дрібніший малюнок, а великі значення генерують великий.

Заломлення — це величина кривизни вм'ятин. Найменші значення створюють меншу турбулентність, а більші значення збільшують хаос.

Деталь — визначає область впливу ефекту. Менший розмір створить більш локалізовані вм'ятини.

Турбулентність — використовується для створення більш динамічного вигляду. Це додасть варіативності до спотворень, роблячи зображення більш живим.

Кут визначає напрямок спотворення.

Якість контролює гладкість країв. Вищі значення приведуть до гладкіших країв.

Випадкове значення рандомізує розрив смуг у новий візерунок.

ІНІЙ...

Цей ефект надає зображенню такого вигляду, ніби воно знаходиться по інший бік покритого інеєм скла (Рис. 410).

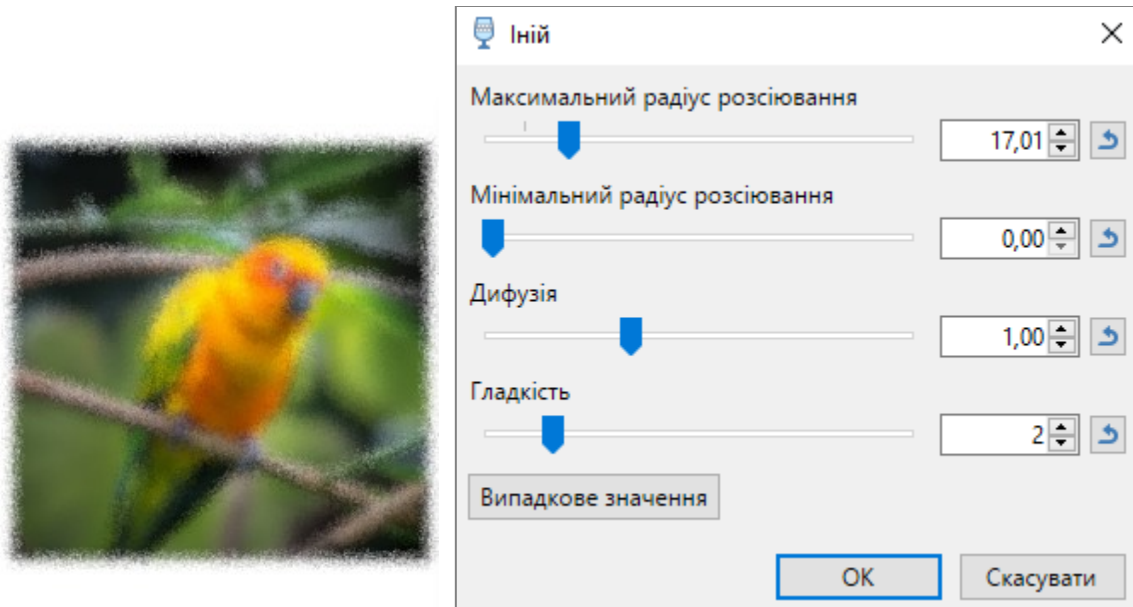


Рис. 410. Налаштування ефекту Іній

Повзунки **Максимального радіусу розсіювання** та **Мінімального радіусу розсіювання** керують величиною розсіювання.

Параметр **Максимального радіусу розсіювання** визначає, наскільки далеко може бути переміщений піксель із вихідного положення. Параметр **Мінімального радіусу розсіювання** визначає мінімальне зміщення, що застосовується.

Дифузія – цей параметр визначає, наскільки «пухким» або розмитим буде ефект інею. Вищі значення роблять текстуру більш розмитою і м'якою.

Гладкість – цей параметр контролює, наскільки рівним буде вигляд інею. Вищі значення призводять до більш однорідної текстури, тоді як нижчі значення створюють більше нерівностей.

КРИСТАЛІЗАЦІЯ...

Ефект, що перетворює зображення так, ніби воно зроблено з мозаїки складеної з різноманітних кристалічних елементів (Рис. 411).

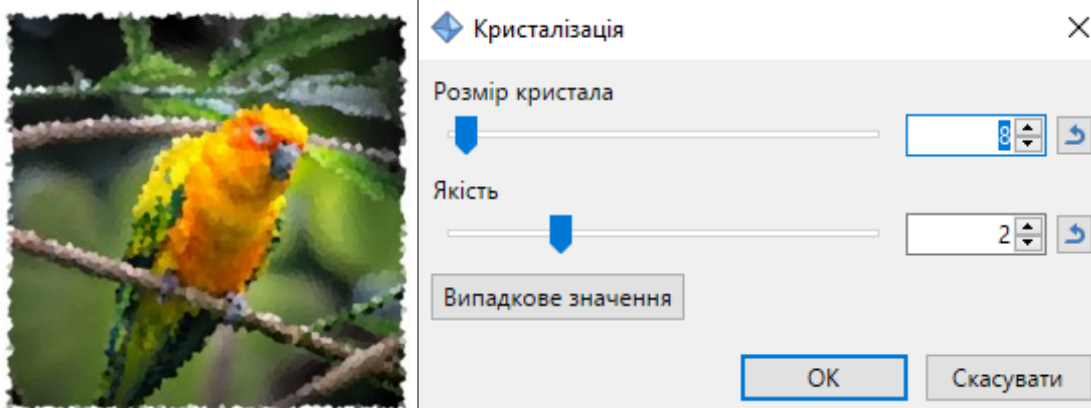


Рис. 411. Налаштування ефекту Кристалізація

Параметр **Розмір кристала** визначає розмір фігур.

Якість керує гладкістю країв. Вищі значення приведуть до гладкіших країв.

Кнопка **Випадкове значення** регенерує вихідний колір і форму комірок.

МОРФОЛОГІЯ...

Ефект **Морфологія** має два режими: стиснення та розширення.

Режим Розширення зменшує деталізацію зображення, видаляючи пікселі на межах різних областей. Відповідно до збільшення розміру вибірки втрачається все більше і більше деталей, і зображення збільшується в розмірі (Рис. 412.).

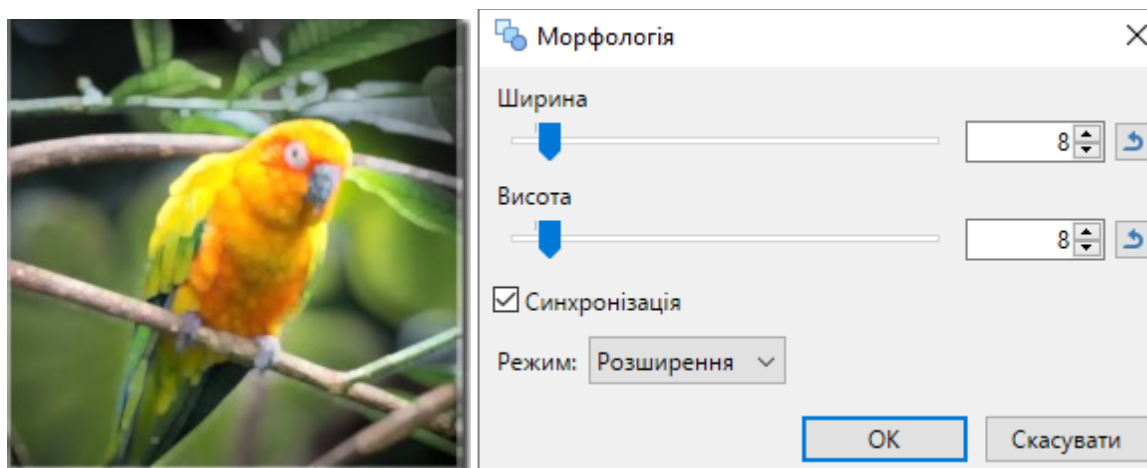


Рис. 412. Налаштування режиму Розширення ефекту Морфологія

Режим Стиснення стискає зображення. Відповідно до збільшення розміру вибірки деталі губляться і зображення зменшується у розмірі, начебто було застосовано зменшення розмиття (Рис. 413.).

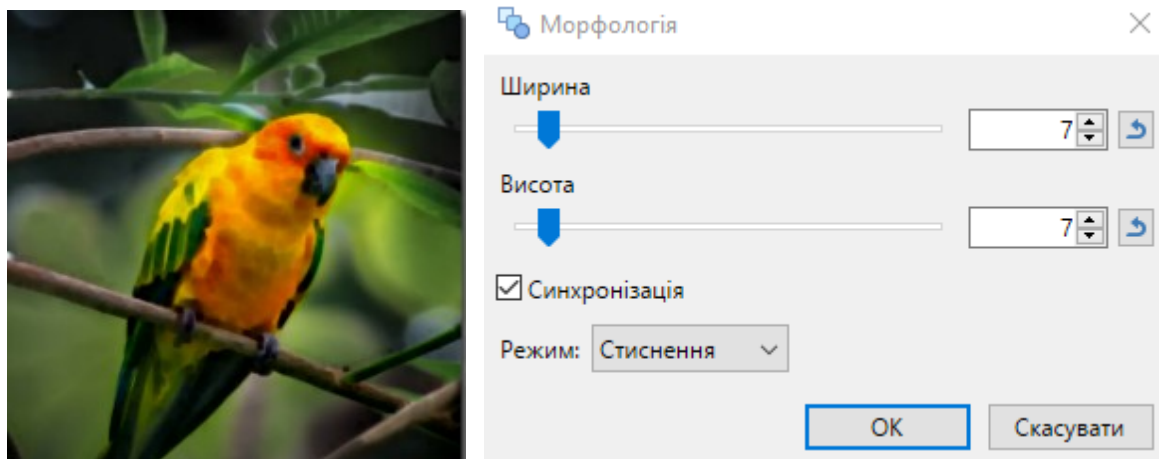


Рис. 413. Налаштування режиму Стищення ефекту Морфологія
Повзунки **Ширина** та **Висота** визначають розмір вибірки.

ПІКСЕЛІЗАЦІЯ

Цей ефект зменшує деталізацію зображення та створює ефект укрупнення пікселів (Рис. 414.).

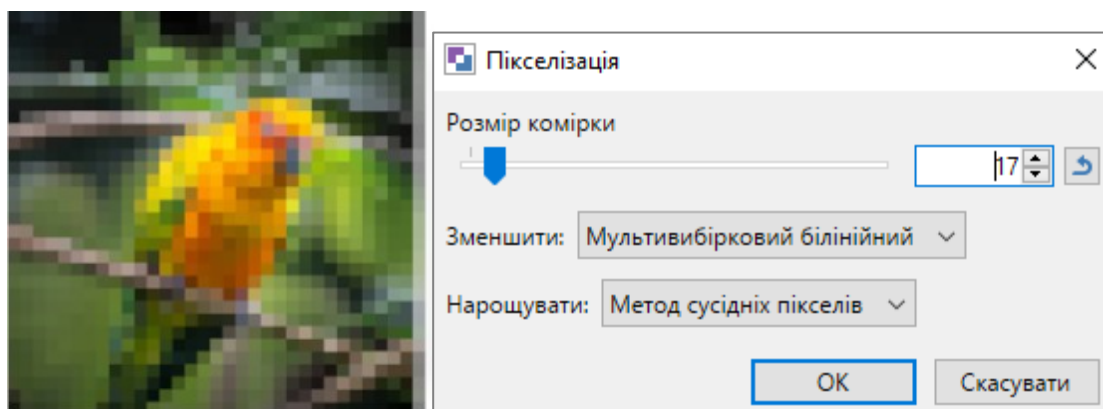


Рис. 414. Налаштування ефекту Пікселізація

Розмір комірки визначає розмір пікселів. Вищі значення створюють квадрати більшого розміру.

ПОЛЯРНА ІНВЕРСІЯ...

Ця команда застосовується для перетворення між декартовою та полярною системами координат. Результатом є радикальне позиційне перетворення, яке рекурсивно обгортає зображення навколо себе у виді нескінченно повторюваного шаблону (Рис. 415).

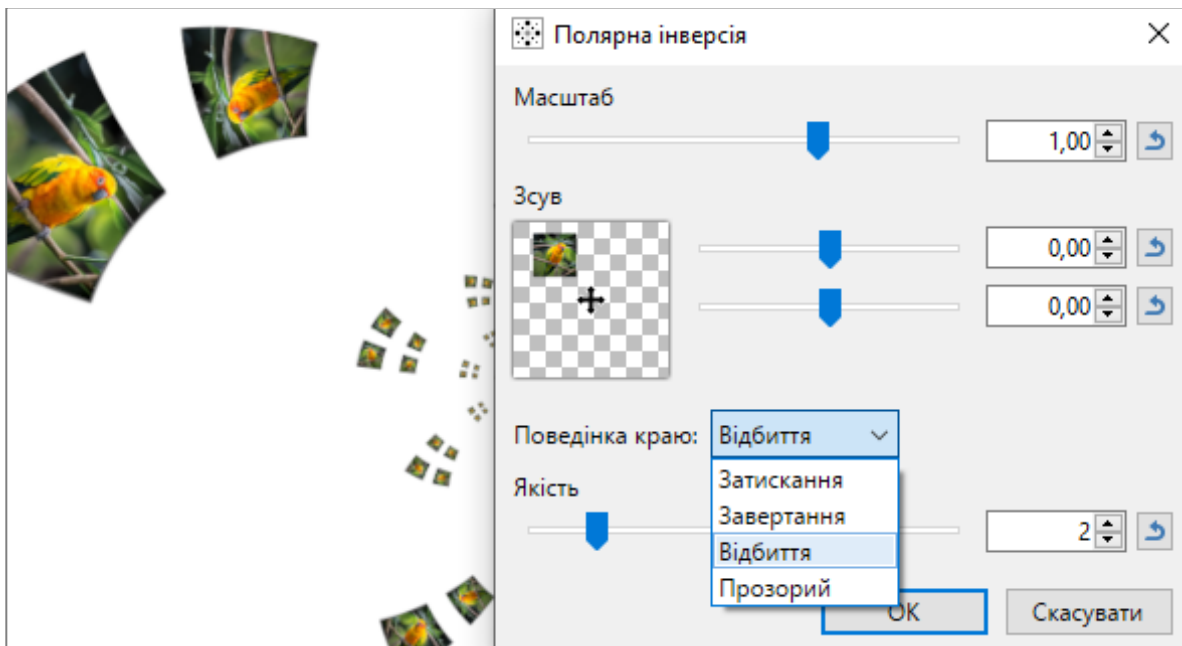


Рис. 415. Налаштування ефекту Полярна інверсія

Масштаб – це сила ефекту.

Зсув використовується для позиціонування центральної точки ефекту.

Поведінка краю (Затискання, Завертання, Відбиття, Прозорий) визначає, як поводить ся ефект.

Якість визначає кількість деталей. Нижчі налаштування надають більш спотвореного вигляду зміщеним пікселям, тоді як більш високі налаштування забезпечують більш плавний вигляд.

СКЛЯНА ПЛИТКА...

Цей ефект надає зображенню такого вигляду, ніби воно знаходиться позаду листа скла з візерунком із плиток. Форма плиток завжди квадратна (Рис. 416).

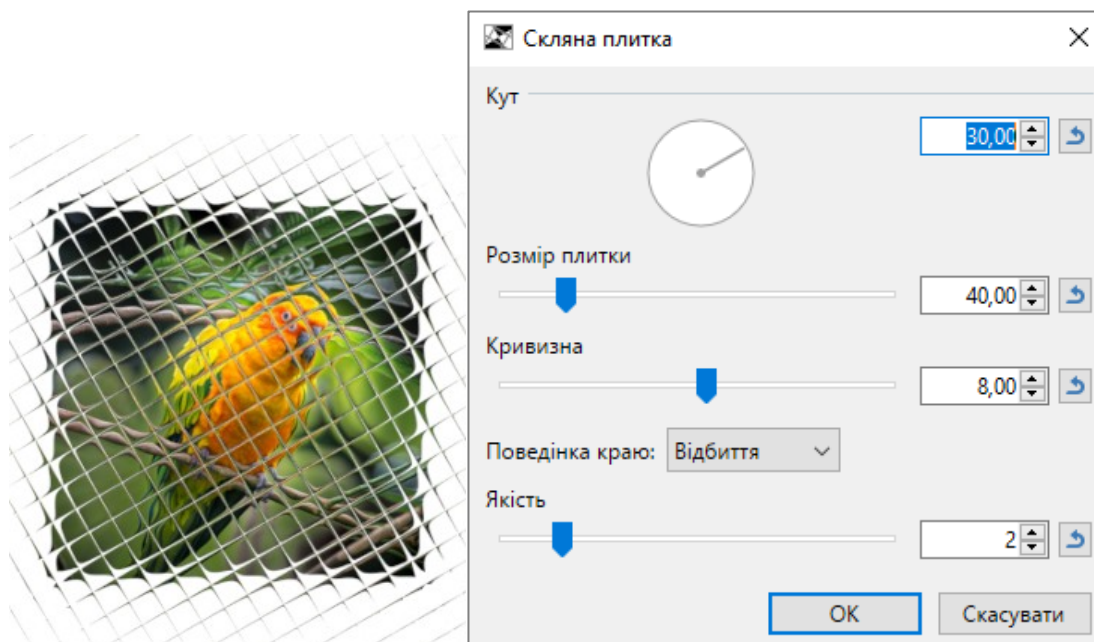


Рис. 416. Налаштування ефекту Скляна плитка

Кут керує обертанням або кутом ефекту. Кут, що дорівнює нулю або 180 градусам, надасть плиткам горизонтального вигляду.

Параметр **Розмір плитки** визначає розмір плиток.

Кривизна – це міра, що контролює величину викривлення країв плитки.

Якість визначає гладкість деталей.

СКРУЧУВАННЯ...

Цей ефект використовується для скручування частини зображення за годинниковою або проти годинникової стрілки (Рис. 417).

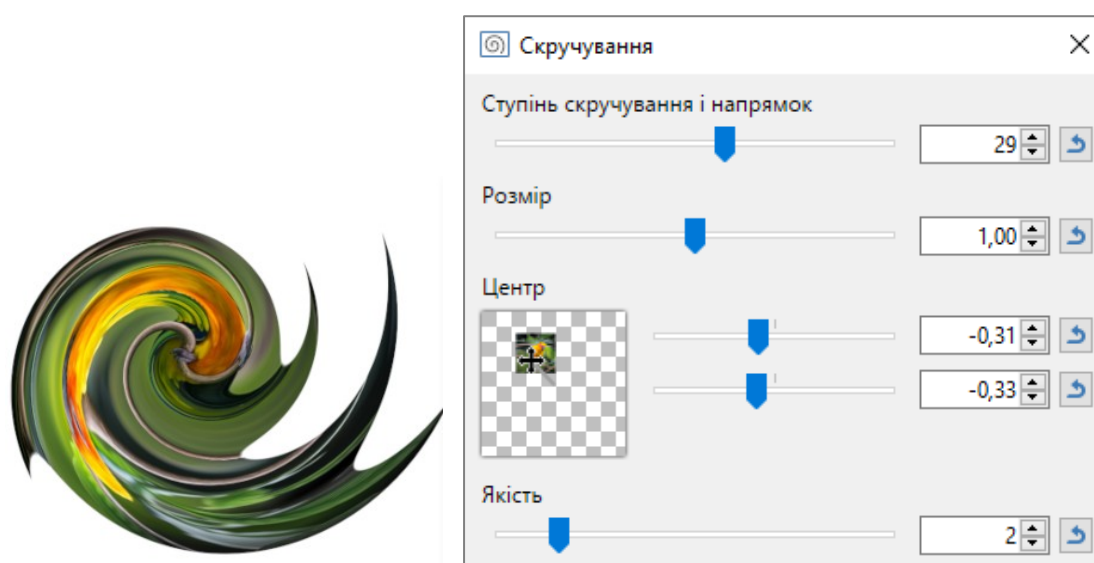


Рис. 417. Налаштування ефекту Скручування

Ступінь скручування та напрямок контролює силу скручування. Що більше число, то сильніше буде ефект. Додатні числа використовуються для повороту за годинниковою стрілкою та від'ємні проти.

Розмір – розмір області, на яку вплине ефект.

Центр – це точка, навколо якої відбувається скручування.

Параметр **Якість** визначає гладкість деталей. Нижчі налаштування надають більш нерівний вигляд, а вищі – плавніший.

СТИЛІЗУВАННЯ

Ця група ефектів (Рис. 418) корисна для пошуку меж об'єктів, або отримання спеціальної текстури для розробки ігор.

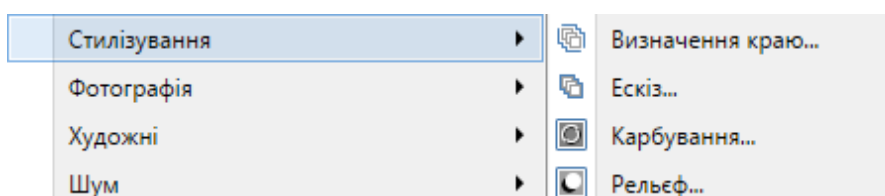


Рис. 418. Команди підменю Стилізування

ВИЗНАЧЕННЯ КРАЮ...

Цей ефект виділяє краї об'єктів у зображенні так, ніби світло було спрямоване на них під певним кутом (Рис. 419).

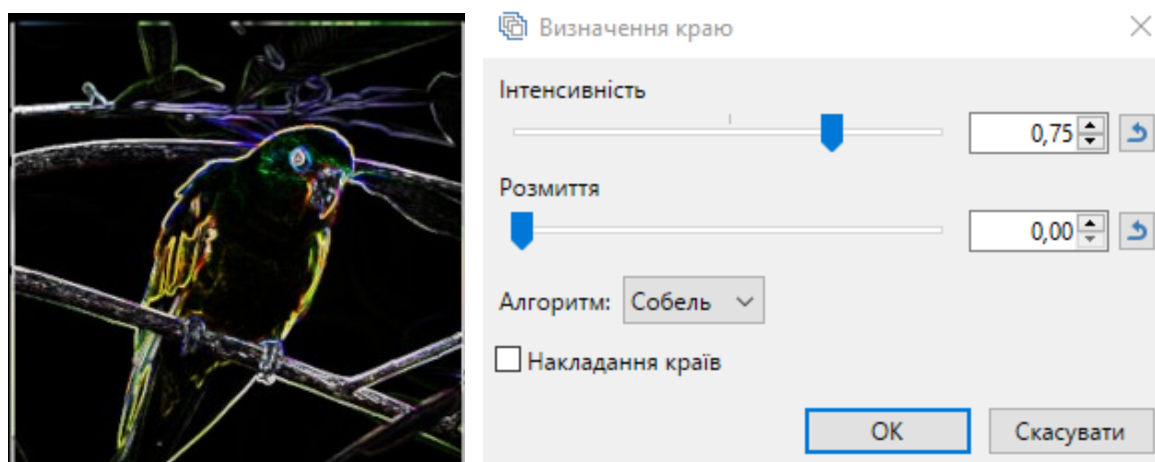


Рис. 419. Налаштування ефекту Визначення краю

Кут установлює видимий напрямок світла, що освітлює зображення.

ЕСКІЗ...

Цей ефект знаходить межі об'єктів зображення та виділяє їх (Рис. 420).

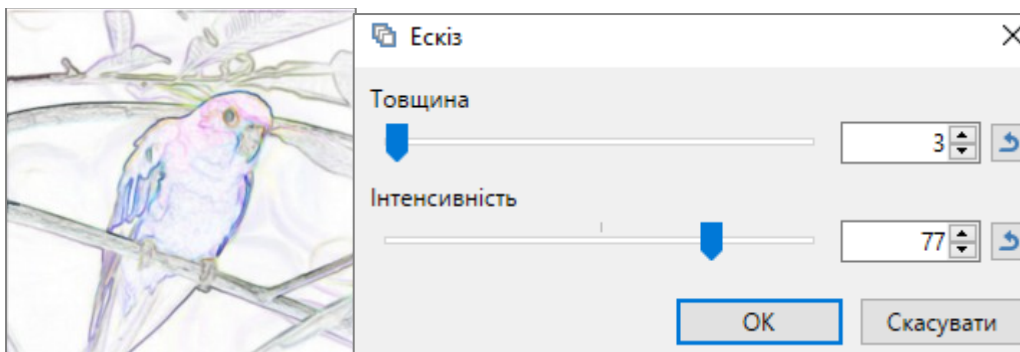


Рис. 420. Налаштування ефекту Ескіз

Товщина визначає товщину контуру. Нижчі значення призведуть до тоншого контуру. Великі значення створюють ефект розмиття.

Інтенсивність визначає силу ефекту. Думайте про це значення як про кількість пікселів, до яких буде застосовано ефект.

КАРБУВАННЯ...

У результаті роботи цього ефекту зображення перетворюється на чорно-білу картинку, де краї об'єктів представлені тривимірним підсвічуванням, схожим на реальне карбування (Рис. 421).

У результаті тиснення завжди виходить зображення в градаціях сірого незалежно від вихідних кольорів і поточних основних та додаткових кольорів.

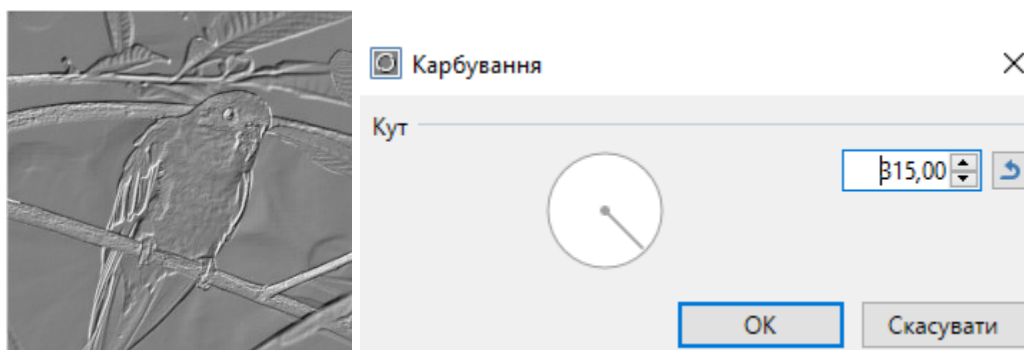


Рис. 421. Налаштування ефекту Карбування

У діалоговому вікні є один елемент управління **Кут**, який установлює видимий напрямок світла, що освітлює зображення.

РЕЛЬЄФ...

У Paint.NET ефект **Рельєф** можна застосувати до зображення для створення текстурованого вигляду, який імітує об'ємність (Рис. 422).

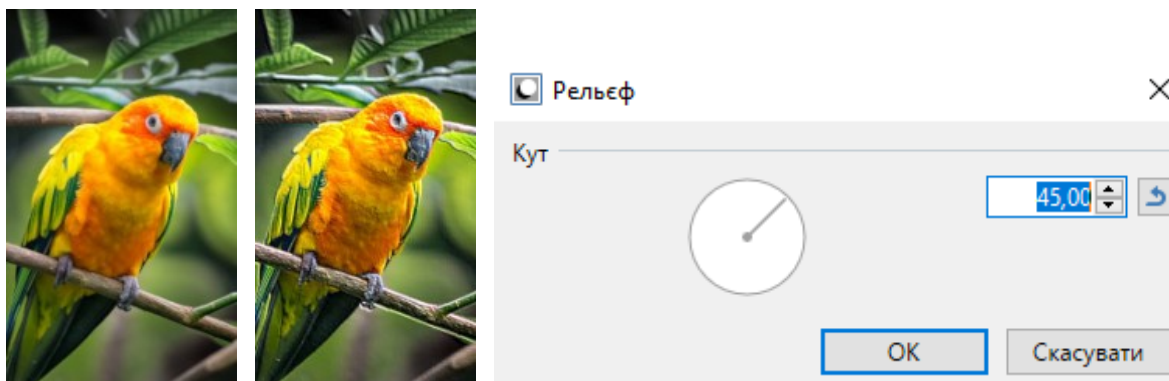


Рис. 422. Налаштування ефекту Рельєф

Параметр **Кут** установлює видимий напрямок світла, що освітлює.

ФОТОГРАФІЯ

Ця група ефектів (Рис. 423) зазвичай найбільш корисна під час роботи з фотографіями.

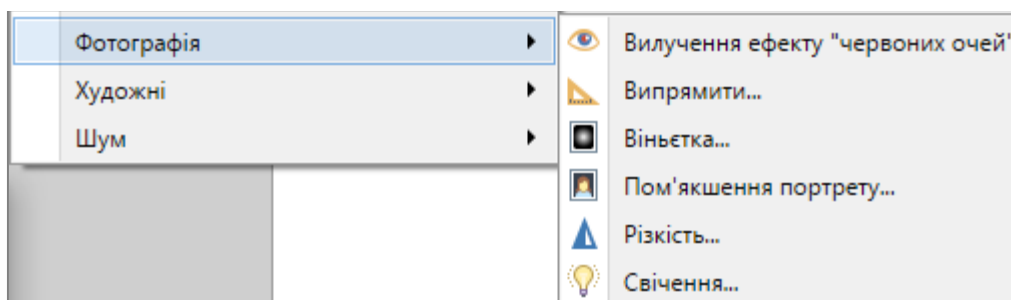
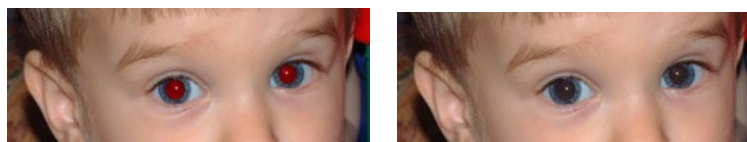


Рис. 423. Команди підменю Фотографія

ВИЛУЧЕННЯ ЕФЕКТУ «ЧЕРВОНИХ ОЧЕЙ»

Доволі часто на аматорських фотографіях очі деякої людини набувають червоного кольору. Ця команда корисна для усунення цього ефекту. Перед її використанням переконайтеся, що очі об'єкта, який обробляється, були обведені за допомогою одного з інструментів виділення (Рис. 424).



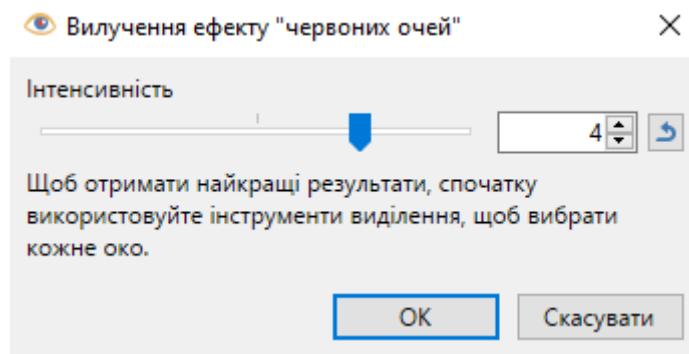


Рис. 424. Налаштування ефекту Вилучення ефекту «червоних очей»

ВИПРЯМИТИ...

У Paint.NET ефект «Випрямити» використовується для вирівнювання зображень, які були зняті під кутом (Рис. 425).

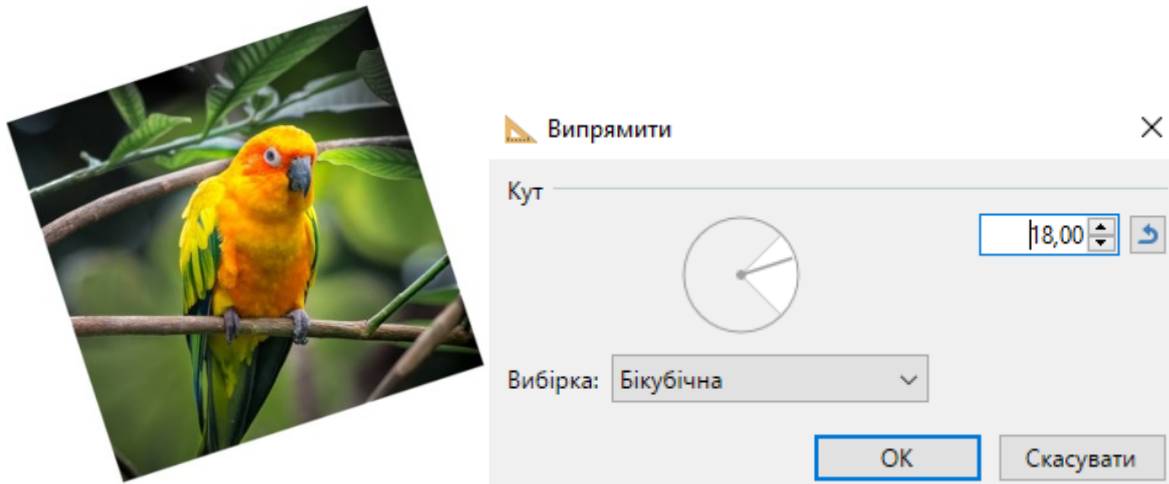


Рис. 425. Налаштування ефекту Випрямити

ВІНЬЄТКА...

Цей ефект використовується для створення темного кругового ореолу навколо країв зображення, який зазвичай спостерігається у портретній фотографії (Рис. 426).



Рис. 426. Приклад застосування ефекту Віньєтка

Має два основні параметри налаштування: Радіус і Щільність, а також можна налаштувати положення центру затемнення (Рис. 427).

Радіус визначає силу ефекту, контролюючи розмір потемніння ореолу.

Інтенсивність визначає, наскільки темною буде віньєтка. Що більше значення Щільності, то більше затемнення.

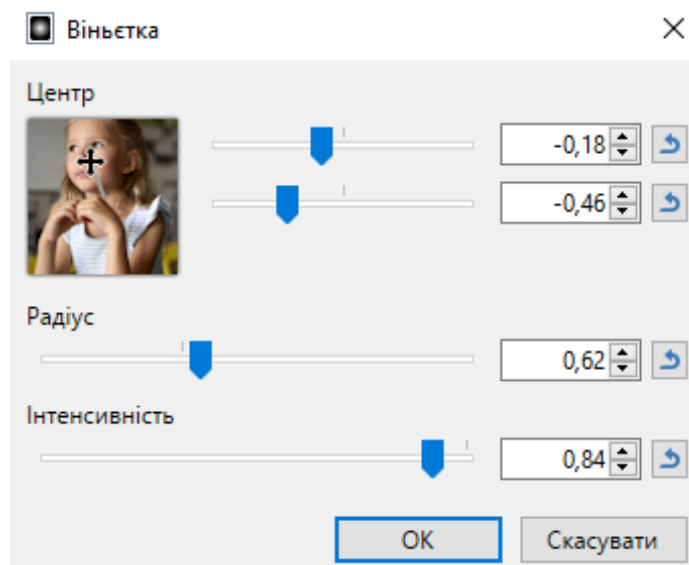


Рис. 427. Налаштування ефекту Віньєтка

Колір віньєтки не можна налаштувати. Він завжди буде чорним або відтінком сірого, якщо знижена Інтенсивність.

ПОРТРЕТНИЙ

Цей ефект корисний для додавання пом'якшення, освітлення або теплоти у фотографію (Рис. 428).

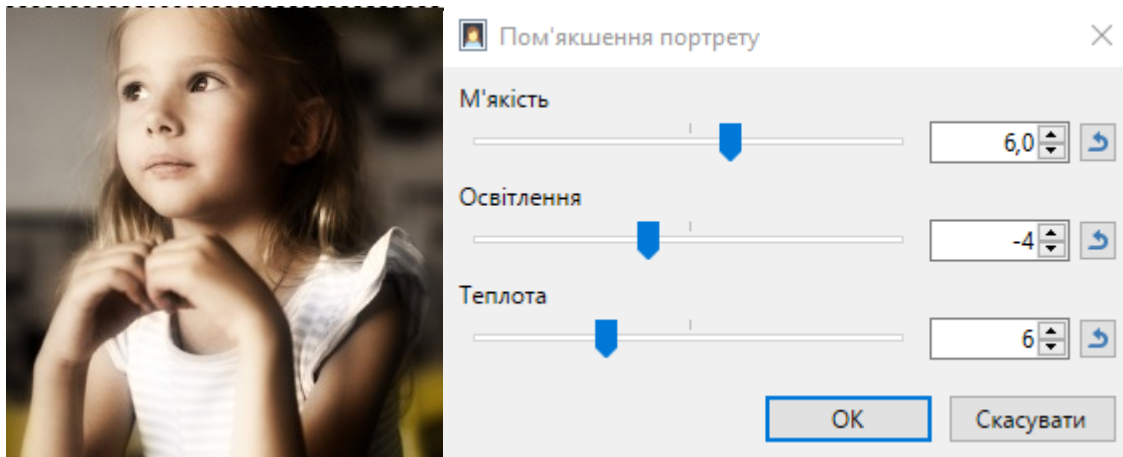


Рис. 428. Налаштування ефекту Портретний

У ньому є три параметри для редагування фотографій:

М'якість – це сила ефекту. Вищі налаштування збільшують ступінь пом'якшення, надаючи зображенню трохи пухнастішого вигляду.

Освітлення змінює інтенсивність освітлення. Вищі значення збільшують силу освітлення і дають яскравіше зображення.

Теплота визначає колір освітлення. Вищі значення збільшують тепло, роблячи відтінки шкіри золотистими. Нижчі значення роблять зображення холоднішим, зменшуючи кількість теплих колірних тонів.

РІЗКІСТЬ

Якщо фотографія (або будь-яке інше зображення) має надто м'який вигляд або деталі втрачені через розмитість, цей ефект можна використовувати для підвищення різкості меж (Рис. 429).

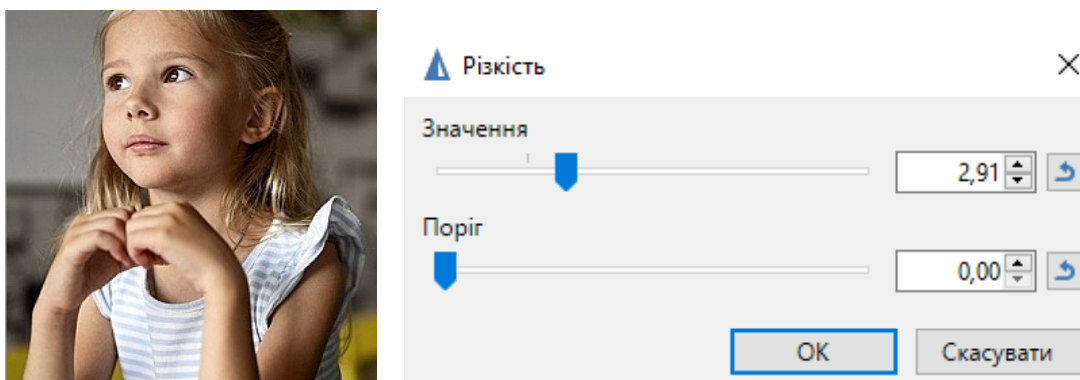


Рис. 429. Налаштування ефекту Різкість

Параметр **Значення** задає силу ефекту.

Поріг дає змогу налаштувати поріг, вище якого пікселі будуть піддаватися обробці. Зменшення значення порогу призведе до обробки більшої кількості пікселів, тоді як збільшення – до обробки лише найчіткіших.

СВІЧЕННЯ...

Цей ефект надає зображенню свого роду внутрішнього світіння, освітлюючи світліші тони та дозволяючи їм вторгтися у темніші. Ефект насичення кольору значно послаблюється. Загальний результат аналогічний фотографії, до якої було застосоване сильне заднє освітлення (Рис. 430).

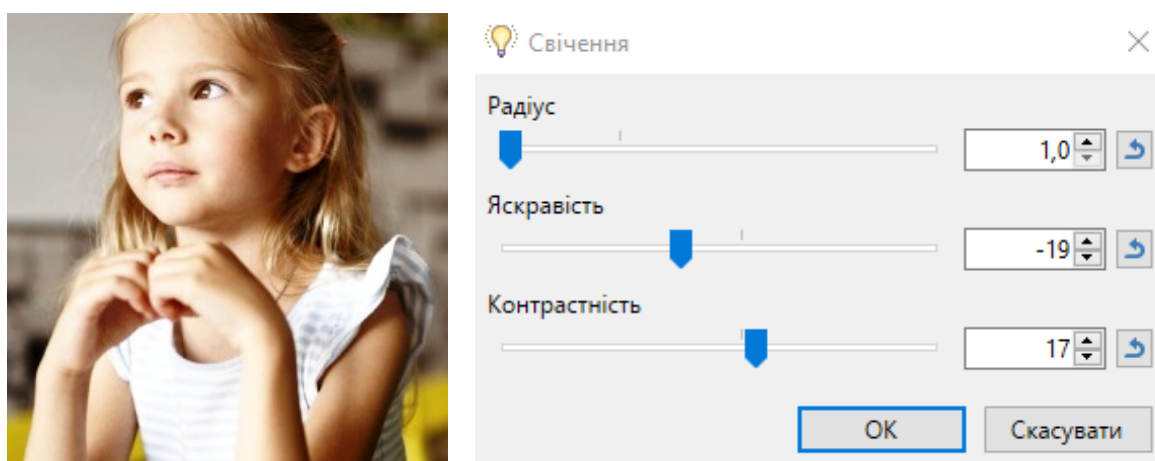


Рис. 430. Налаштування ефекту Свічення

Радіус визначає силу ефекту. Коли використовується **Свічення** для об'єкта, оточеного прозорими пікселями, **Радіус** визначає розмір затінення або розмиття, що відбувається по краях об'єкта. Коли використовується **Свічення** з елементами, оточеними непрозорими пікселями, **Радіус** визначає ступінь скосу.

Яскравість визначає інтенсивність віртуального висвітлення. При більш високих налаштуваннях яскравість оригіналу зменшується, а нижчі значення зображення затемнюють.

Контрастність діє так само, як і налаштування **Контрастність** у меню **Регулювання** → **Яскравість / Контрастність**. Зменшення значення контрастності звужує діапазон тонів кольорів, тому різниця між кольорами менша. Збільшення контрастності поліпшує кольори, але може також видаляти деякий діапазон кольорів, що призводить до меншої кількості яскравих кольорів.

ХУДОЖНІ

Ця група ефектів (Рис. 431) призначена для надання зображення виду, характерного для традиційних технік виконання зображень (малюнок пером, малюнок олівцем або картина маслом).

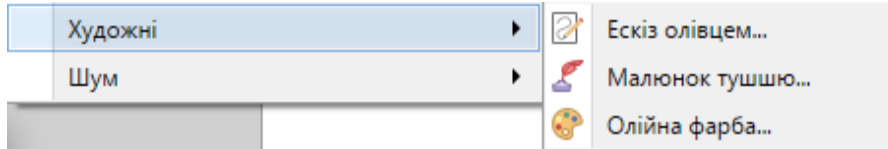


Рис. 431. Команди підменю Художні

ОЛІЙНА ФАРБА...

Цей ефект використовується для створення зображення, яке виглядає так, ніби воно було намальоване олією (Рис. 432).

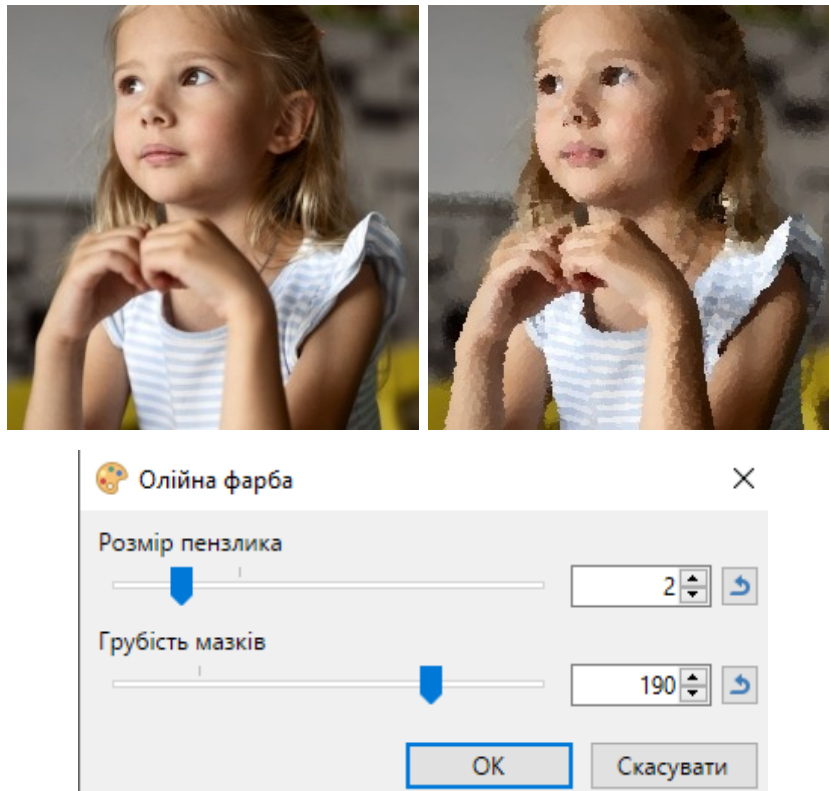


Рис. 432. Налаштування ефекту Олійна фарба

Розмір пензля визначає видимий розмір віртуального кінчика пензля. Великі значення призводять до грубішого вигляду, немов використовувався широкий пензлик. Найменші значення дають більш тонкий результат, начебто використовувався тонший пензлик.

Грубість мазків визначає, скільки деталей фіксується при перетворенні. Низькі значення дають менше деталей, а вищі значення забезпечують більшу кількість дрібних деталей.

ЕСКІЗ ОЛІВЦЕМ...

Цей ефект використовується для створення зображення, яке виглядає так, ніби воно було намальоване простим олівцем (Рис. 433.).

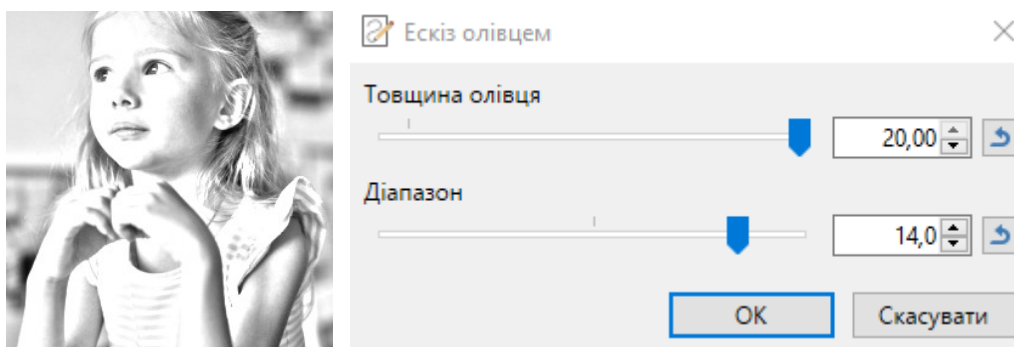


Рис. 433. Налаштування ефекту Ескіз олівцем

Товщина олівця визначає, наскільки гострим чи тупим буде віртуальний олівець. Великі значення призводять до більш товстого та м'якого вигляду ліній. Менші значення дають більш тонкий результат, якби використовувався гостріший і твердіший олівець.

Діапазон визначає, скільки деталей зафіксовано в ескізі. Як і значення **Грубість** в ефекті **Олійна фарба**, низькі значення дають менше деталей, а вищі значення показують менші деталі.

МАЛЮНОК ТУШШЮ...

Цей ефект використовується для створення зображення, яке виглядає так, ніби воно було намальоване тушшю (Рис. 434).

Параметр **Окреслення** впливає на товщину або силу краю. Великі значення призводять до більш товстої / жирної лінії, а менші значення – тоншої / світлішої лінії.

Параметр **Забарвлення** визначає, скільки кольорів використовується у зображенні. Низькі значення дають ненасичені зображення з меншою кількістю кольорів. Вищі значення будуть використовувати більше вихідного забарвлення.

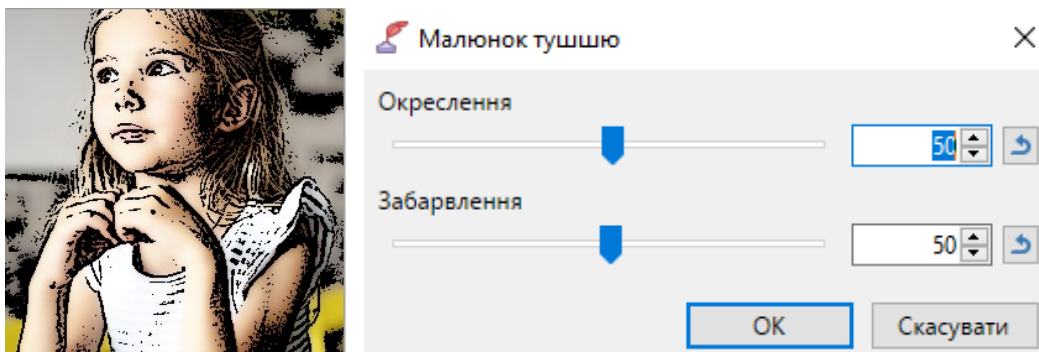


Рис. 434. Налаштування ефекту Малюнок тушшю

ШУМ

Підменю містить ефекти (Рис. 435), що використовуються для додавання або видалення шуму зображення.

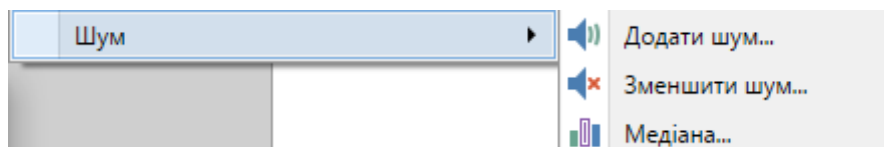


Рис. 435. Команди підменю Шум

ДОДАТИ ШУМ...

Ефект додає випадковий шум до зображення (Рис. 436). Ефект повинен застосовуватися до областей, що містять колір. Використання ефекту на прозорих областях не призводить до появи шуму в цих областях.

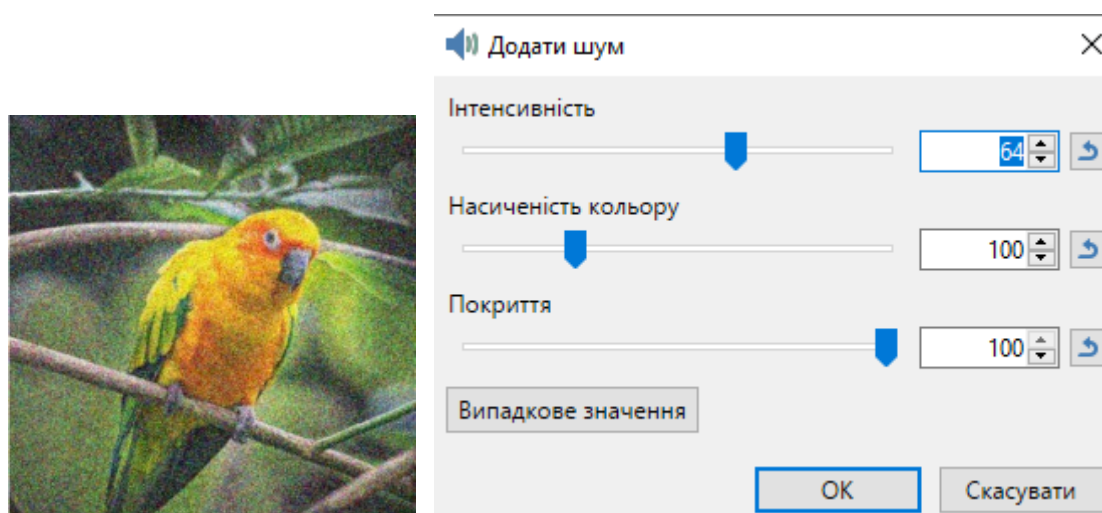


Рис. 436. Налаштування ефекту Додати шум

Інтенсивність визначає, наскільки палітра відрізнятиметься від оригіналу. Низькі настройки відображають шум за допомогою тонкої або приємної палітри,

високі значення роблять шум яскравими кольорами, менш схожими на початкову палітру.

Насиченість кольору – це інтенсивність забарвлення. При низьких налаштуваннях шум відображається у монохромному (чорно-білому) режимі. При високих налаштуваннях використовуються яскравіші кольори.

Покриття визначає, яка частина оригіналу буде вкрита ефектом. Вищі значення покривають більшу частину виділення чи шару.

ЗМЕНШИТИ ШУМ...

Цей ефект може бути корисним для зменшення шуму зображення, насамперед у фотографіях, без втрати деталізації та контрастності (Рис. 437).

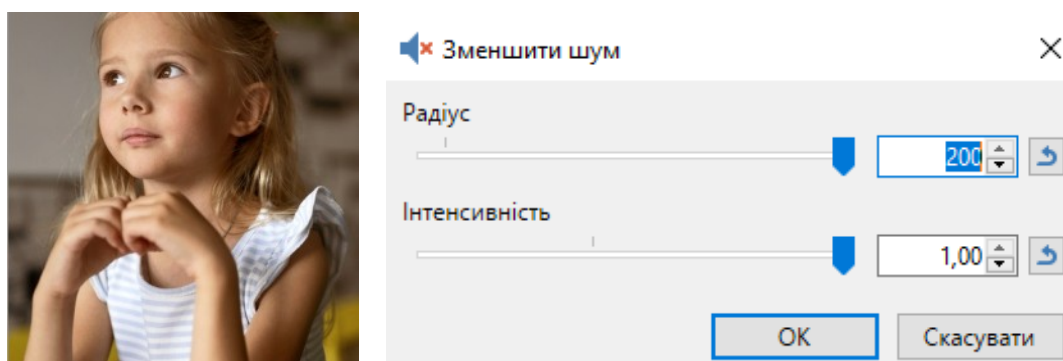


Рис. 437. Налаштування ефекту Зменшити шум

Радіус визначає, яку частину зображення буде вибрано. Низькі значення дають вибірку невеликих областей, а великі значення збільшують розмір вибірки. Розмір вибірки визначає, скільки пікселів буде порівнюватися, щоб побачити, чи є піксель, що розглядається, досить схожим, щоб вважатися шумом чи ні.

Інтенсивність задає силу ефекту. Великі значення видаляють більше шуму, також можуть мати легкий ефект освітлення.

МЕДІАНА...

У програмі Paint.NET інструмент «Медіана» (Median) використовується для фільтрації та зменшення шуму в зображеннях (Рис. 438). Він працює шляхом зміни пікселів на основі їхніх сусідів, усуваючи різкі відмінності між кольорами.

Медіана включена до цього підменю, оскільки її можна використовувати для зменшення шуму при низьких налаштуваннях. При більш високих налаштуваннях медіана діє як розмиття.

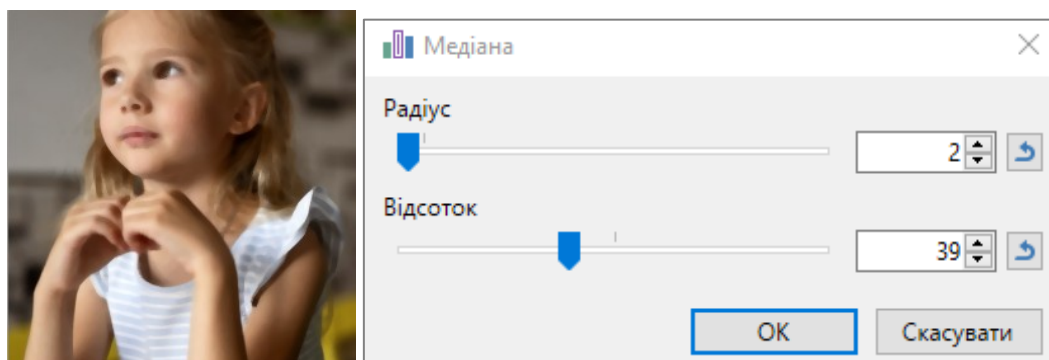


Рис. 438. Налаштування ефекту Медіана

Параметри, які можна налаштувати у фільтрі «Медіана», охоплюють:

Радіус – визначає радіус області навколо кожного пікселя, в межах якої аналізуватимуться сусідні пікселі. Що більший радіус, то ширша область впливу, що призводить до більшого згладжування та зменшення деталей на зображенні.

Відсоток – визначає, наскільки суворо фільтр замінюватиме пікселі. За замовчуванням, значення 50% означає, що кожен піксель змінюватиметься на середнє значення своїх сусідів (медіану). Якщо відсоток збільшити, заміщення пікселів стане менш агресивним.

ПЛАГІНИ ТА РОЗШИРЕННЯ

ВСТАНОВЛЕННЯ ДОДАТКОВИХ ІНСТРУМЕНТІВ І ФІЛЬТРІВ

1. Відвідайте сайти, які пропонують плагіни для Paint.NET, наприклад, Paint.NET Forum або [GitHub](#).
2. Виберіть потрібний плагін і завантажте файл. Зазвичай це ZIP-архів.
3. Відкрийте ZIP-архів і витягніть вміст у папку на вашому комп'ютері.
4. Знайдіть папку, де встановлено Paint.NET. Зазвичай це C:\Program Files\paint.net\ або C:\Program Files (x86)\paint.net\.
5. Відкрийте папку Effects або File Types (залежно від типу плагіна) та скопіюйте витягнуті файли в цю папку.

6. Якщо Paint.NET був відкритий під час копіювання файлів, закрийте та знову відкрийте програму, щоб вона оновила список доступних фільтрів та інструментів.

ОГЛЯД ПОПУЛЯРНИХ ПЛАГІНІВ ДЛЯ PAINT.NET

Aye Caramba! Цей плагін додає великий набір фільтрів, які можуть допомогти створити унікальні ефекти на зображеннях.

Boltbait's Plugin Pack. Це один із найбільших і найпопулярніших наборів плагінів для Paint.NET. Він охоплює чималу кількість інструментів і ефектів, таких як корекція яскравості та контрастності, ефекти розмиття, різкості, кольорові корекції тощо.

Color Replacement Tool. Дає змогу швидко та просто змінювати кольори на зображенні, що корисно під час редагування фотографій або ілюстрацій.

Curves+ – вдосконалена версія стандартного інструмента «Curves». Він дозволяє точніше керувати яскравістю, контрастністю, тінями та світлом на зображенні. Це чудовий інструмент для точної корекції фотографій.

Distort This! Плагін для спотворення та зміни зображень. Він охоплює кілька методів, які дають змогу створювати унікальні ефекти шляхом скручування, розтягування чи згинання зображення.

Dodge and Burn. Цей плагін додає до Paint.NET інструменти «Dodge» та «Burn», які дозволяють освітлювати або затемнювати окремі частини зображення. Ці інструменти дуже корисні для ретуші та художньої роботи.

Dpy's Plugins. Зазначені плагіни додатково розширюють функціонал Paint.NET і охоплюють: генерацію хмар, створення ефекту тривимірності на текстурях, додавання різних типів шуму до зображень.

Feather. Feather дозволяє згладжувати краї виділених об'єктів, що робить їх менш різкими та допомагає природніше виглядати при накладанні на інші зображення.

Gradient Mapping. Gradient Mapping дає змогу застосовувати градієнти до зображень для створення різних художніх ефектів. Це корисно для кольорової корекції або для створення особливих настроїв у фотографіях.

Liquify. Цей плагін додає до Paint.NET функцію «Liquify», схожу на ту, що є у Photoshop. Він дозволяє «розтягувати» і «деформувати» частини зображення, змінюючи їх форму без втрати якості. Особливо корисний при ретуші фотографій або створенні креативних художніх ефектів.

MadJik's Plugin Pack. Цей пакет містить інструменти для: HDR зображень, фільтрів для художнього стилю, ефектів розмиття та корекції кольору.

Nicolas's Plugins. Цей набір плагінів фокусується на: векторній графіці та шрифтах, застосуванні текстур та ефектів.

NormalMap Plus. Цей плагін дає змогу створювати нормальні карти (Normal maps) для використання у тривимірних проєктах. Він корисний для тих, хто працює з 3D-графікою або текстурами.

Paint.NET Plugin Pack. Цей плагін-пакет є набором найпопулярніших плагінів, які додають нові ефекти й інструменти. Він містить корисні ефекти для корекції кольорів, перетворення зображень та інших художніх операцій.

Pixelate. Цей плагін дозволяє легко пікселізувати зображення, що може бути корисним для створення артефактів або анонімізації особистих даних.

Seamless Texture Maker. Цей плагін полегшує створення безшовних текстур. Він автоматично редагує краї зображень так, щоб їх можна було з'єднувати між собою без видимих переходів.

ShapeMaker. ShapeMaker дає змогу створювати й додавати власні фігури в Paint.NET. Цей плагін корисний для створення складних графічних елементів, яких немає у стандартному наборі інструментів програми.

Smart Resize. Змінює розмір зображення без втрати якості, дозволяючи зберігати деталі.

Smudge. Плагін Smudge додає інструмент для розмиття та змішування кольорів на зображенні, що дозволяє створювати плавні переходи між елементами зображення або пом'якшувати деталі.

РОЗДІЛ 4

ІНТЕРАКТИВНІСТЬ, ПРОТОТИПУВАННЯ ТА ОПТИМІЗАЦІЯ ВЕБДИЗАЙНУ

4.1. ОСНОВИ ПРОТОТИПУВАННЯ ІНТЕРФЕЙСІВ

Прототипування – етап розроблення інтерфейсів, що дає змогу створити візуальне або інтерактивне представлення майбутнього програмного продукту.

Метою прототипування є: створення базової моделі інтерфейсу; перевірка зручності користування; перевірка ідей дизайнера, розробника й замовника погляду логіки та зручності використання; виявлення можливих проблем на ранніх етапах роботи.

Усе це зумовлює економію часу та ресурсів і дає змогу швидко та дешево протестувати різні концепції.

Є такі прототипи:

- Паперові прототипи.
- Вайрфрейми (Wireframes).
- Інтерактивні прототипи.
- Повноцінні прототипи.

Паперові прототипи – найпростіший тип прототипів, які створюються на папері та мають вигляд ескізів інтерфейсу. Використовуються на початкових етапах проєктування для обговорення концепції.

Wireframes (каркаси) – це схематичні зображення інтерфейсу користувача майбутнього сайту або мобільного додатку. Переважно **Wireframes-схеми** найчастіше мають **чорно-білий** колір. Квадрати, прямокутники й прості лінії позначають місце для меню, кнопок, тексту та зображень. Головна ідея Wireframes – швидко пояснити структуру сайту: куди користувач подивиться першим, як буде рухатися далі, а не прикрашати його (Рис. 439).

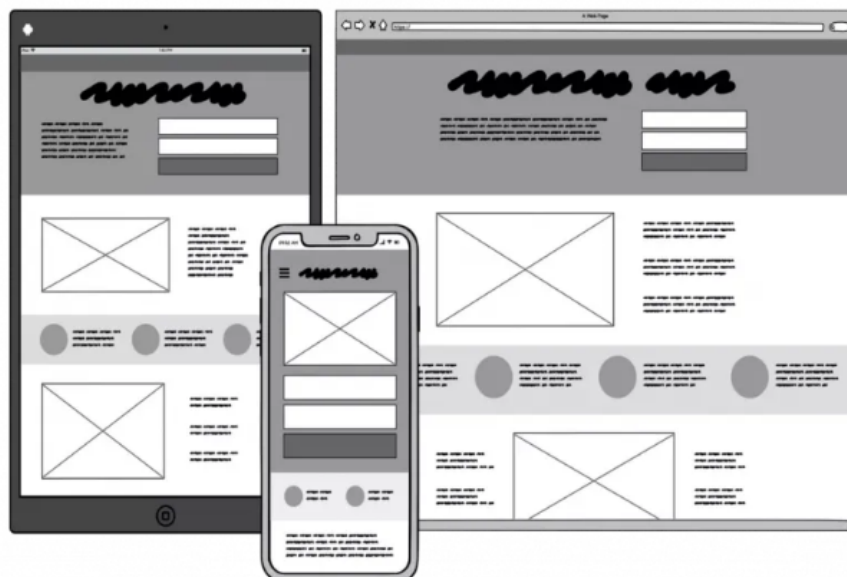


Рис. 439. Wireframes-схеми

Low-fidelity (Low-Fi) прототипи – це базові, спрощені моделі майбутнього продукту. Вони можуть бути створені і на папері, і у цифровому форматі, вони призначені для перевірки ключових ідей і функціональності без занурення в деталі.

Click-through прототипи – інтерактивні макети, створені в спеціальних графічних додатках. Прототипи дають змогу користувачам переходити між екранами та взаємодіяти з такими елементами інтерфейсу, як кнопки, вкладки тощо.

High-fidelity прототипи – високодеталізовані інтерактивні прототипи з максимально наближеною до реальної функціональністю. Такі прототипи охоплюють анімацію, введення даних і симуляцію реальних сценаріїв використання і використовуються для тестування та демонстрації готового продукту перед запуском розробки.

Кодовані прототипи – це прототипи, створені за допомогою коду (HTML, CSS, JavaScript). Такі прототипи можна використовувати як базову версію продукту, коли потрібно протестувати продукт в умовах, наближених до реальних.

Інтерактивні презентації – прототипи, створені за допомогою програм для презентацій (наприклад, PowerPoint або Keynote). Вони представляють переходи між слайдами, як різні екрани додатку. Такі прототипи дають змогу швидко створити уявлення про продукт. Використовуються на ранніх етапах розробки вебдодатку для представлення концепції інтерфейсу.

Інструменти для створення інтерактивних прототипів

Кожен інструмент для створення інтерактивних прототипів має свої переваги та підходить для різних етапів розроблення продукту:

- **Figma** – хмарний інструмент для спільної роботи, що дає змогу створювати інтерактивні каркаси та високодеталізовані прототипи.
- **Adobe XD** – інструмент для створення прототипів з інтерактивними переходами й анімаціями.
- **InVision** – платформа для створення інтерактивних прототипів і тестування із залученням користувачів.
- **Axure RP** – дає змогу створювати детальні інтерактивні прототипи з логікою та умовними переходами.
- **Proto.io** – платформа для створення інтерактивних високодеталізованих прототипів для мобільних додатків і вебсайтів.

4.2. АНІМАЦІЯ У FIGMA

Анімація у Figma – це не просто ефект для краси, а інструмент, який робить інтерфейс «живим» і зрозумілим користувачу. Завдяки прототипуванню можна показати, як поводитимуться елементи при натисканні, наведенні або зміні стану. Це допомагає клієнтам, розробникам і дизайнерам бачити не лише статичний макет, а й логіку взаємодії.

ПРИНЦИПИ SMART ANIMATE

Smart Animate – це «розумна» анімація, яка автоматично обчислює відмінності між двома кадрами (frames) і плавно переходить від одного стану до іншого. Наприклад, якщо на першому кадрі кнопка сірого кольору, а на другому – блакитна, Figma сама створить плавну зміну кольору, без додаткових налаштувань.

Smart Animate працює з усіма властивостями, які можна змінювати:

- положенням (x, y);
- розміром;
- кольором;

- прозорістю (opacity);
- скругленнями кутів;
- навіть зі зміною тексту чи іконки, якщо назви шарів збігаються.

Щоб Smart Animate спрацював правильно необхідно, щоб шари мали однакові назви в обох кадрах (наприклад, Button / Main у першому й другому), елементи мали однакову ієрархію всередині груп або компонентів. Також неможна змінювати порядок шарів між кадрами, інакше Figma не розпізнає, який об'єкт анімувати.

Практичний приклад: під час створення випадаючого меню можна анімувати плавне розкриття блоку. Замість того, щоб просто показати меню у другому кадрі, Smart Animate додасть рух: блок «сповзе» вниз, а іконка стрілки обернеться на 180°.

Оптимальна тривалість анімації Smart Animate – **300–400 мс**. Менше – рух стане надто різким, більше – сповільнить сприйняття.

ПЕРЕХІД МІЖ КАДРАМИ (FRAMES) І ЗАТРИМКА

Будь-яка анімація у Figma ґрунтується на **переході між двома кадрами**. Кожен кадр – це окремий стан інтерфейсу: наприклад, головне меню, сторінка профілю або картка товару після натискання.

Щоб задати перехід необхідно:

1. Вибрати елемент, який має реагувати на дію, наприклад кнопку.
2. Перейти у вкладку **Prototype** (праворуч від **Design**). Якщо вкладки **Prototype** немає, активуйте її через **View** → **Panels** → **Prototype** або скористайтесь комбінацією клавіш **Shift + E**. Якщо вкладка **Prototype** усе одно не з'являється, перевірте, який режим роботи включено – виконайте команду **View** → **Switch to Design**.
3. Перетягнути синю стрілку до кадру, куди має відбутися перехід (Рис. 440).

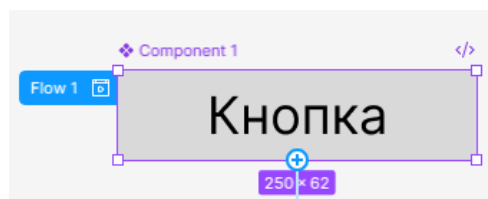


Рис. 440. При наведенні на маркер сторони з'являється синій плюсики, за який необхідно потягнути

4. У вікні **Interaction** вибрати тип переходу: **Smart Animate**, **Instant**, **Move In/Out**, **Push** тощо (Рис. 441).

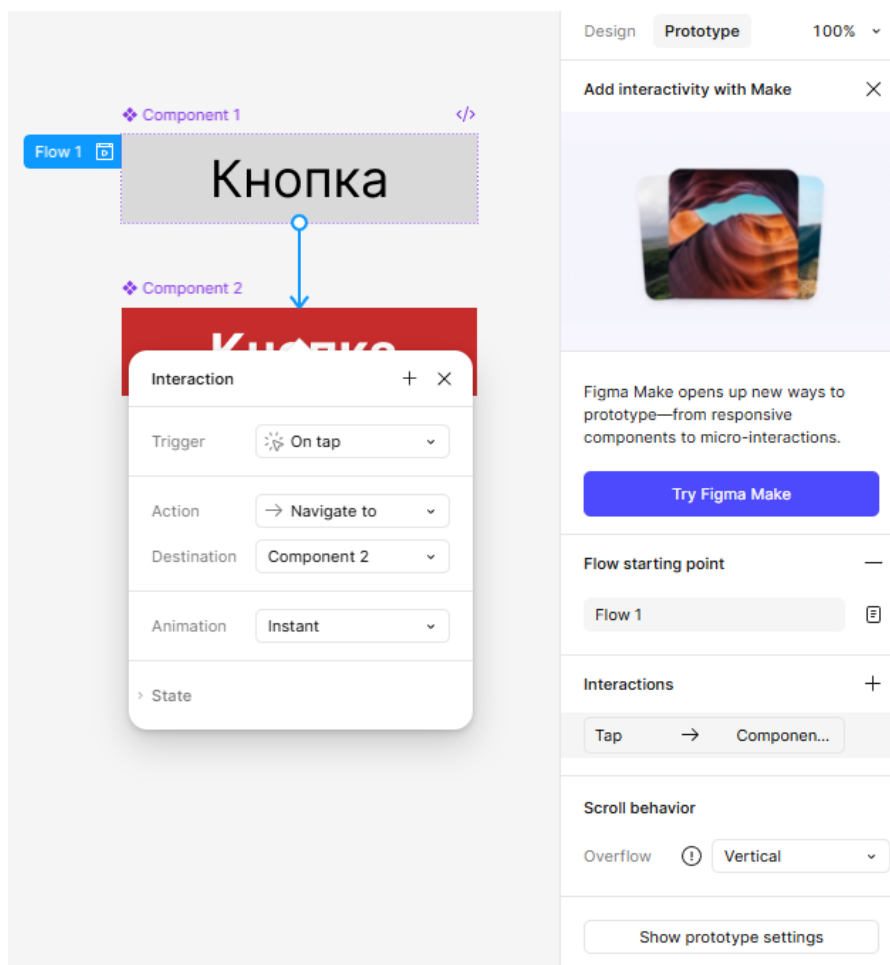


Рис. 441. Вікно *Interaction* із параметрами анімації

Далі можна налаштувати:

- **Тривалість (Duration)**. Наприклад, 300–400 ms – оптимально для UI.
- **Затримку (Delay)**. Наприклад, 100 ms – коли елемент має з’являтися не одразу, це дає змогу зробити рухи більш природними.
- **Easing (крива руху)**: Ease in/out або Ease out для природної динаміки.

КОМПОНЕНТНІ СТАНИ ТА МІКРОАНІМАЦІЇ

Figma дає змогу створювати **варіанти компонентів (Variants)** – наприклад, кнопка може мати стани **Default**, **Hover**, **Pressed**, **Disabled**. Коли вмикається режим прототипу, Figma автоматично розуміє, що це різні стани одного елемента, і плавно переходить між ними.

Такі **мікроанімації** допомагають користувачу відчувати зворотний зв'язок: кнопка трохи затемнюється при натисканні або «відпружинює» при відпусканні.

Це дрібниця, але саме вона створює відчуття завершеності та якісного інтерфейсу.

Для мікроанімацій краще обирати короткі тривалості – 100–200 мс. Цього достатньо, щоб користувач відчув рух, але не втратив концентрації.

АНІМАЦІЯ ПРИ ВЗАЄМОДІЇ (ON CLICK, WHILE HOVERING, AFTER DELAY)

Figma підтримує кілька типів тригерів (подій), які запускають анімацію:

- **On Click (при кліку мишкою)** – найпоширеніший варіант, коли перехід спрацьовує після натискання. Використовується для переходів між сторінками, відкриття модальних вікон або активації кнопок. Наприклад, натискання на картку товару відкриває сторінку з детальним описом.
- **While Hovering (при наведенні)** – використовується для hover-ефектів: зміна кольору кнопки, підсвічування елемента або додавання тіні, коли курсор наближається тощо.
- **After Delay (після затримки)** – анімація спрацьовує автоматично через певний час. Наприклад, splash-screen або автоперехід між слайдами.
- **While Pressing / On Drag** – використовується для імітації натискання або перетягування елементів. Наприклад, користувач тягне картку – вона плавно рухається за курсором.

Не вартує перевантажувати макет великою кількістю рухів – достатньо кількох точних мікроанімацій, щоб зробити інтерфейс живим.

ПРОТОТИПУВАННЯ ПЕРЕХОДІВ У FIGMA

Прототипування – це спосіб поєднати всі кадри у логічну послідовність і створити інтерактивну модель сайту чи застосунку.

Після налаштування переходів можна протестувати логіку руху між екранами. Для цього необхідно:

- Натиснути у верхньому меню **Present** (іконка ►). Figma відкриє прототип у режимі попереднього перегляду.
- Клікнути на елементи, щоб перевірити, чи працюють переходи.

У цьому ж режимі можна:

- тестувати взаємодію мишкою або клавішами;
- вимірювати швидкість переходів;
- демонструвати клієнту готовий сценарій роботи інтерфейсу.

Прототипування дає змогу ще до етапу розробки побачити, **як почуватиметься інтерфейс у реальному користуванні**. Водночас варто додавати стрілки назад (Back Interaction) – це зробить навігацію зрозумілішою під час демонстрації.

ЕКСПОРТ ІНТЕРАКТИВНИХ ПРОТОТИПІВ

Після створення прототипу ним можна **поділитися з іншими**. Для цього:

1. Натискають **Share** → **Copy link** → **Prototype**.
2. Надсилають посилання замовнику або команді. Користувач зможе переглядати інтерактивні переходи навіть без реєстрації у Figma в режимі перегляду.

Також можна вбудувати прототип у **презентацію, Behance чи Notion**, експортувати окремі екрани у PNG/JPG або записати демонстрацію екрану (screen capture) для відеопрезентацій.

4.3. МАКЕТИ ДЛЯ ВЕБСАЙТІВ І МОБІЛЬНИХ ЗАСТОСУНКІВ

СТРУКТУРА САЙТУ

Структура сайту – це логічна схема взаємного розташування сторінок сайту та переходів між ними. Виокремлюють такі основні види структур сайтів:

- **Лінійна структура** – сторінки розташовані у виді ланцюжка. Таку структуру рекомендується застосовувати на сайтах-презентаціях, портфоліо й інших

видах сайтів, які мають на меті ознайомити відвідувача з усіма сторінками у певній послідовності (Рис. 442).



Рис. 442. Лінійна структура

- **Лінійна з відгалуженнями** – використовується на сайтах, де кожен розділ може мати свої підрозділи, розташовані послідовно (Рис. 443). Наприклад, це може бути сайт для навчання за темами, розбитими на окремі уроки.

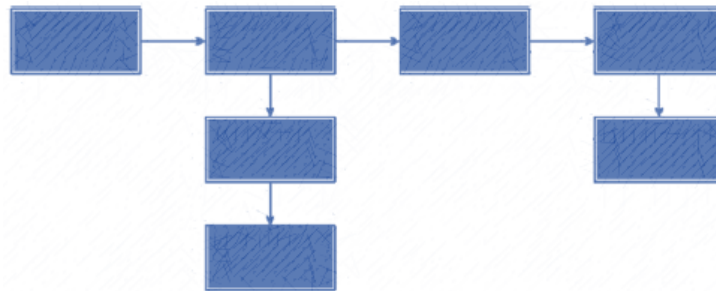


Рис. 443. Лінійна структура з відгалуженнями

- **Деревоподібна структура** – найбільш універсальний варіант. Головна сторінка є коренем, а розділи та підрозділи – гілками (Рис. 444). Такий підхід зручний для інтернет-магазинів, порталів, корпоративних сайтів.

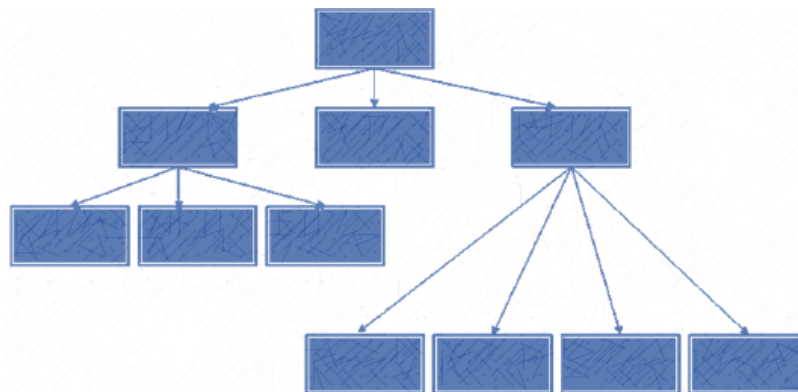


Рис. 444. Деревоподібна структура

- **Блокова структура** – на будь-яку сторінку можна перейти з будь-якої іншої завдяки численним посиланням (Рис. 445). Таку структуру рекомендується застосовувати для конкретного продукту / послуги, коли кожную сторінку можна використовувати як опис якоїсь окремої властивості та їх сукупності. Прикладом може бути вікіпедія та блоги з багатьма тегами.

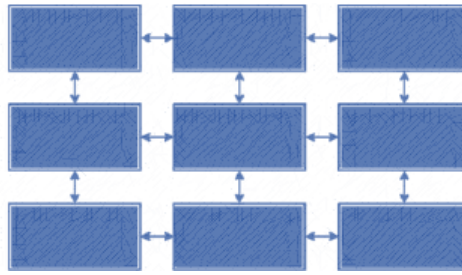


Рис. 445. Блокова структура

ЗОВНІШНЯ СТРУКТУРА САЙТУ

Зовнішня структура визначає розташування основних елементів сторінки: хедера, меню, контенту й футера. Вона має бути логічною, зручною для користувача та відповідати стилю сайту. Найбільш поширена класична структура подана на Рис. 446.



Рис. 446. Класична зовнішня структура сайту

- **Хедер** (шапка) – верхня частина сайту, в якій розташовуються логотип, назва сайту, головне меню та деякі додаткові елементи, характерні стилістиці сайту або його тематиці (слайдер, велике зображення, контакти, кнопка пошуку, кнопки переходу на інші мови, кнопка версії сайту для людей із вадами зору тощо).
- **Область меню навігації** – може бути розташована горизонтально, не входячи в область хедера, а може розташовуватись вертикально ліворуч або праворуч. Головне призначення – надати користувачам можливість швидко переміщуватися по сторінках сайту.

- **Блок контенту** – центральний блок, який пропонує цільову інформацію для користувача. Контент повинен бути читабельним і помітний, не варто змушувати користувача виконувати прокручування вниз, щоб побачити інформацію, яка його цікавить. Це відбувається, наприклад, коли зверху сторінки розташовується велике зображення, а інформація розміщена нижче.
- **Футер** (підвал сайту) – призначений для розміщення контактної інформації, інформації про розробників сайту, знаку копірайту, кнопок соціальних мереж. У футері допускається повторення навігації, розташування форм зворотного зв'язку, карти сайту, слайдера з логотипами партнерів тощо.

ТИПИ СТОРІНОК САЙТУ

Типи сторінок сайту визначають їхнє призначення, функціонал і контент, який вони надають користувачам. Різні типи сторінок виконують різні завдання в межах одного вебсайту, забезпечуючи комфортну навігацію. Виокремлюють такі основні типи сторінок:

- **унікальні сторінки** – це сторінки, що мають ексклюзивний графічний стиль із не повторювальними далі елементами, які не є наскрізними (наприклад, головна, «Про нас», «Контакти»);
- **стандартні сторінки** – це сторінки сайту, що використовують шаблони з однаковими графічними компонентами (наприклад, сторінки новин або товарів);
- **додаткові сторінки** – сторінки, що виконують окремі функції: сторінка 404, сторінка пошуку, подяка після надсилання форми тощо.

НАВІГАЦІЯ

Навігація – це система переходів між сторінками. Від того, наскільки вона зрозуміла, залежить зручність користування сайтом.

Критерії якісної навігації:

- **простота** – всі елементи інтерфейсу користувача повинні бути добре видимими та зрозумілими: чіткі іконки, прості форми, уникнення зайвих графічних елементів;

- **доступність** – із будь-якої сторінки сайту повинна бути зрозуміла користувачеві можливість перейти на необхідний розділ або повернутися на головну сторінку;
- **оптимізація для різних пристроїв** – елементи навігації мають бути адаптованими під різні пристрої;
- **сумісність із дизайном** – елементи навігації мають виділятися на тлі та відрізнятися від основного тексту, але водночас органічно вписуватися у загальний стиль сайту.

Основні види навігації сайту:

- **глобальна** – забезпечує перехід з/на будь-яку сторінку сайту. Сучасний дизайн не рекомендує включати пункт у меню «Головна», прийнято закріплювати посилання на головну сторінку за логотипом;
- **основна** – в меню розташовуються посилання на найбільш важливі розділи сайту;
- **тематична** – містить посилання на близькі за тематикою розділи. Наприклад, схожі публікації на сайтах новин. Також це можуть бути посилання під однією статтею з можливістю переходу до наступної або попередньої. Використовується в електронних виданнях для лінійного переходу, а також в електронних книгах;
- **рекламна (банерна)** – містить посилання на інші ресурси чи сторінки з пропозицією послуг чи продукції. Така навігація може бути текстовою або графічною;
- **мовна** – користувачеві пропонується вибрати мову, якою відображатиметься інформація. Використовується на сайтах, потенційна аудиторія яких є мультимовною, найчастіше на сайтах міжнародних організацій. Не рекомендується вбудовувати автоматичний перекладач Google;
- **вказівна («хлібні крихти»)** – показує користувачеві, в якій частині сайту він знаходиться. Зручна для великих порталів і сайтів із великою кількістю розділів;

- **географічна** – застосовується для великих порталів, які вимагають вказати країну, місто чи регіон, до якого належить потрібний користувачеві розділ;
- **пошукова** – передбачає форму пошуку та виведення матеріалів по введеному запиту.

Є такі типи навігації:

- **липка навігація** (sticky) – передбачає відображення навігаційного меню у фіксованому місці. Це може бути верхня частина сторінки з прокручуванням вниз або нижня з прокручуванням вгору;
- **приховане меню, що розкривається (гамбургер)** – припускає приховування меню за деякими графічними елементами (знаки «+», «≡») та розкриття його при натисканні на піктограму меню;
- **мега-меню** – список посилань на сторінки, що не випадає, а виглядає як великий каталог посилань (Рис. 447).

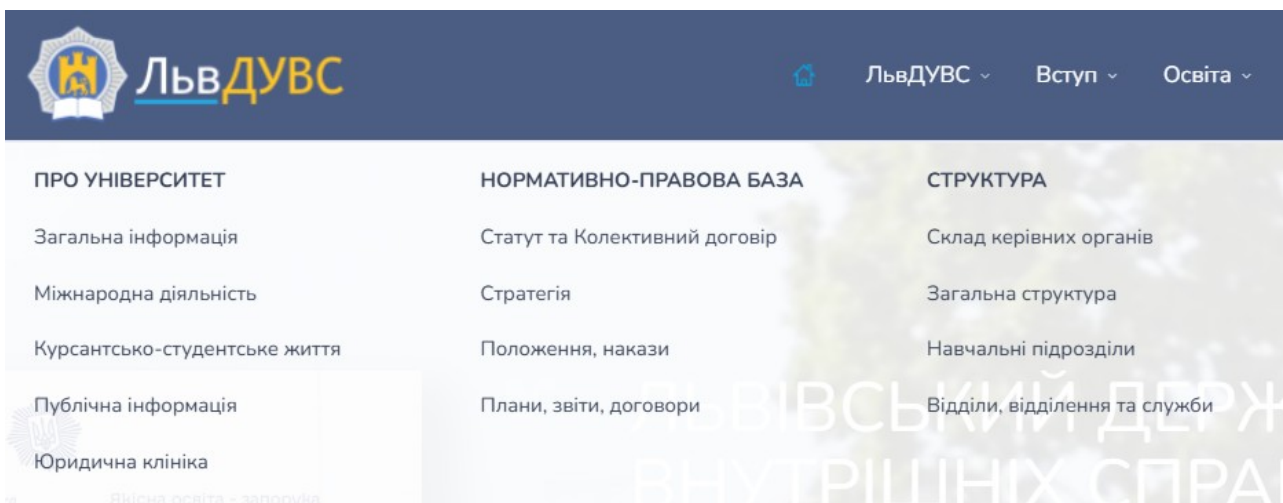


Рис. 447. Мега-меню

- **вертикальна / горизонтальна навігація** – використовується за наявності безлічі підрозділів у основних розділах;
- **каруселі** – візуальні блоки, при натисканні на які відкривається блок із контентом. Цей тип меню дає змогу заощаджувати місце та публікувати більше інформації на сторінці (Рис. 448);

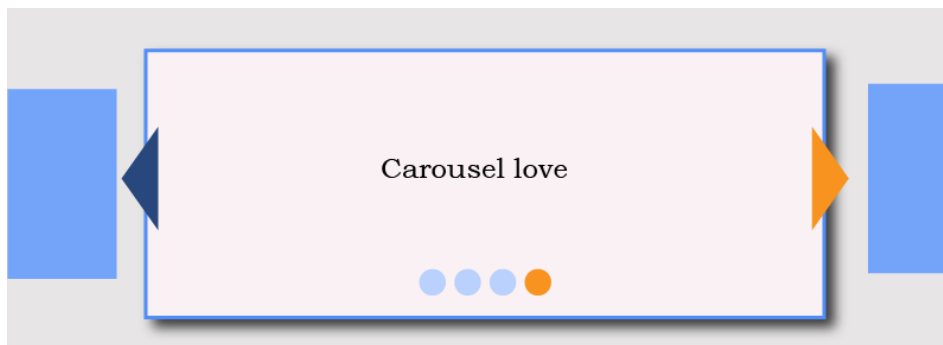


Рис. 448. Карусель

- **зміст** – містить посилання на розділи на головній сторінці;
- **кутові посилання** – використовуються для створення чіткої навігації з використанням знаків ">", "<", "^", "↓", "↑", "→", "←", для позначення напрямку або переходу між сторінками чи розділами (горизонтальна прокрутка між розділами, вертикальна навігації для сторінок із довгим контентом, кнопки переходу вперед або назад у галереях, статтях або каруселях);
- **точкова навігація** – це послідовність невеликих знаків у виді точок, які слугують посиланнями на інші розділи сайту або частини сторінки. Посилання підсвічуються, вказуючи поточну позицію користувача (див. Рис. 448);
- **універсальна навігація** – це навігація, яка є на всіх сторінках сайту. Використовується на великих порталах та інтернет-магазинах.

На практиці часто комбінують кілька типів навігації, щоб підвищити зручність користування.

4.4. КОМПОНЕНТИ ВЕБСТОРИНОК

Компоненти – це базові елементи будь-якого інтерфейсу: кнопки, поля введення, іконки, модальні вікна, списки, таблиці.

Компоненти інтерфейсу – це будівельні блоки будь-якої вебсторінки: елементи, з яких складаються сценарії взаємодії. Користувач не думає про «компоненти», він просто натискає, вводить, підтверджує. Завдання розробників – зробити так, щоб кожна взаємодія була очевидною, швидкою та безпомилковою.

Кнопки – найчастіший тригер дій. Вони запускають відправлення форми, перехід, збереження, видалення, оплату. Важливо, щоб основна дія на екрані була видимою одразу: контрастний колір, достатній розмір, зрозумілий текст на кшталт «Надіслати», а не абстрактне «Ок». Стани натискання, наведення та недоступності мають бути помітним без підказок, і обов’язково – фокус для клавіатурної навігації. Якщо кнопок кілька, їх доцільно пріоритезувати: головна – тягне на себе погляд, вторинні – спокійніші.

Іконки працюють як візуальна мова. Маленьке зображення кошика швидше пояснює додавання товару, ніж довге речення. Але іконка повинна підкріплювати зміст, а не замінювати його. Якщо значення може бути неочевидним, варто додати підпис або підказку – користувач не має здогадуватися, що означає «зірочка» чи «три точки». У багатьох випадках іконки інтерактивні: натиснув – стався перехід або відкрився стан. У такому разі вони мають і поводитися як кнопки: мати фокус, читабельну область натискання і зрозумілий стан наведення.

Форми – це міст між користувачем і даними. Людина залишає ім’я, пошту, адресу, вибирає варіанти, погоджується з умовами. Правильно побудована форма не має змушувати думати. Підписи мають бути поруч із полями, наявні приклади форматів заповнення, просто пояснені помилки із вказівками, що саме виправити. Текстові поля, перемикачі, радіокнопки, випадаючі списки – всі ці елементи мають бути достатнього розміру, із зрозумілим фокусом і логічним порядком переходу клавішею Tab. Якщо користувач помилиться – він не має бути покараний повторним заповненням, виправити має тільки поля із помилками.

Модальні вікна потрібні тоді, коли потрібно привернути всю увагу до одного питання. Наприклад, підтвердити дію, показати важливе повідомлення, попросити мінімальні дані. Вони мають з’являтися рідко, по справі й легко закриватися. Текст у модальному вікні має відповідати ситуації та не лякати зайвими деталями.

Списки допомагають показувати чимало однотипних об’єктів: новини, товари, контакти. Кожен елемент списку має мати однаковий порядок полів: заголовок, короткий опис, ціна або дата, і, за потреби, кнопка дії. Відступи мають бути також

однаковими. Якщо чимало даних, потрібно додавати пагінацію або ліниве завантаження.

Таблиці – найкращі для структурованих даних: порівняння, розцінки, замовлення. Їхня сила – у чітких заголовках, помірній кількості колонок і можливості швидко знайти потрібне. На мобільних таблиці не повинні ламатися, тому потрібно додавати або горизонтальну прокрутку з підказкою, або адаптивні картки з головними полями.

Усі ці елементи мають мати помітний фокус для клавіатурної навігації, читабельний контраст тексту, зрозумілі повідомлення про помилки, і послідовні кольори й відступи від компонента до компонента. Коли ці речі узгоджені, сторінка відчувається цілісною: кнопки виглядають і поведуться однаково, іконки говорять тією самою мовою, форми не лякають, а модальні вікна не заважають працювати.

ПРАВИЛА КОМПОЗИЦІЇ

Композиція – це взаємодія всіх елементів інтерфейсу, що дає змогу створити ефективний та естетичний дизайн сайту.

Основні характеристики композиції:

Цілісність – властивість композиції, коли жоден елемент не можна видалити, додати чи перемістити без втрати загальної ідеї. Всі елементи композиції повинні бути пов'язані стилем малюнка, вирівнюванням, кольорами, розмірами тощо (Рис. 449).

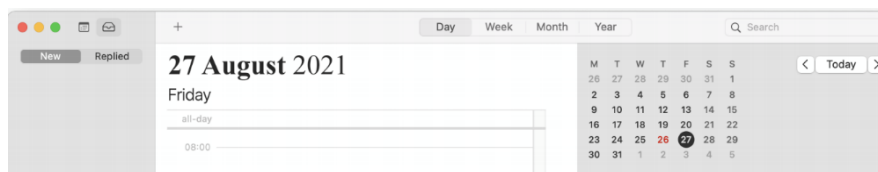


Рис. 449. Закон цілісності порушено недоречним шрифтом для відображення поточної дати (27 серпня)

Виразність – здатність композиції привертати увагу і точно передавати задум дизайнера. Виразність проявляється у використанні контрастів: за кольорами, світлом, розмірами. Наприклад, у вебдизайні контраст між заголовком і текстом дає змогу виокремити ключові ідеї.

Виявлення центру – це точка дизайну, що виражає основну ідею. Композиційний центр може збігатися з геометричним центром на перетині діагоналей (Рис. 450) або бути зміщеним за рахунок виділення кольором, розміром, формою або контрастом.

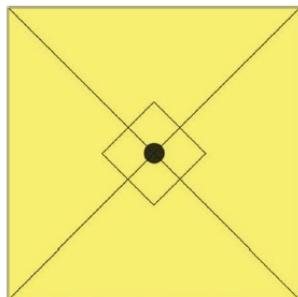


Рис. 450. Геометричний центр композиції

Композиційних центрів у композиції може бути кілька, тоді як геометричний центр один.

Гармонія – це злагодженість і баланс між усіма елементами дизайну, коли всі елементи доповнюють одне одного. Найяскравіший приклад – це навколишній світ, у якому всі елементи взаємопов'язані.

Основними інструментами гармонізації є два правила: симетрія та «золотий переріз».

Симетрія – засіб організації інтерфейсу з метою досягнення візуальної рівноваги.

В інтерфейсах найчастіше застосовують два типи симетрії: вертикальна осьова симетрія (симетрія відносно вертикальної лінії, проведеної через центр групи елементів) і діагональна осьова симетрія (симетрія відносно діагоналі). Симетрія зустрічається у всіх типах дизайну. Приклад симетричного дизайну зображено на Рис. 451.

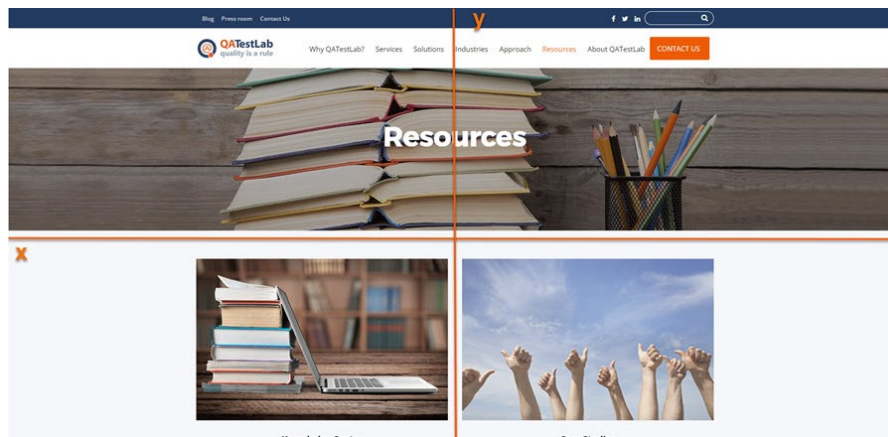


Рис. 451. Симетричний дизайн вебсайту

Асиметрія використовується для створення динамічного і не стандартного вигляду, але з обов'язковим дотриманням загального балансу. Лише досвідчені дизайнери здатні досягати асиметричної рівноваги, керуючи візуальною вагою окремих елементів. У асиметрії найкраще привертають увагу користувачів тільки в одній точці, акцентуючи на невеликій кількості елементів (Рис. 452).



Рис. 452. Приклад асиметрії у дизайні сайту

Найчастіше асиметричні методи проєктування використовують разом із елементами, які мають симетрію.

Пропорція – це співвідношення розмірів між частинами об'єкта або між різними елементами композиції. Вона визначає, як гармонійно сприймаються форма, масштаб і рівновага у дизайні. Наприклад, співвідношення ширини до висоти блоку, розміру зображення до тексту чи простору між елементами інтерфейсу.

Ще з античних часів архітектори та художники приділяли величезну увагу пропорціям. Вони вважали, що саме через правильне співвідношення частин можна

досягти краси й гармонії. Одним із найвідоміших відкриттів у цій сфері є «**золотий переріз**», який і сьогодні залишається універсальним орієнтиром для дизайнерів.

«Золотий перетин» – це співвідношення, що може виявлятися у всіх аспектах життя, від будови кісток людини до спірального розташування насіння соняшника та завитків раковин моллюсків, воно є в основі всіх біологічних структур і забезпечує гармонійне співвідношення елементів (Рис. 453).

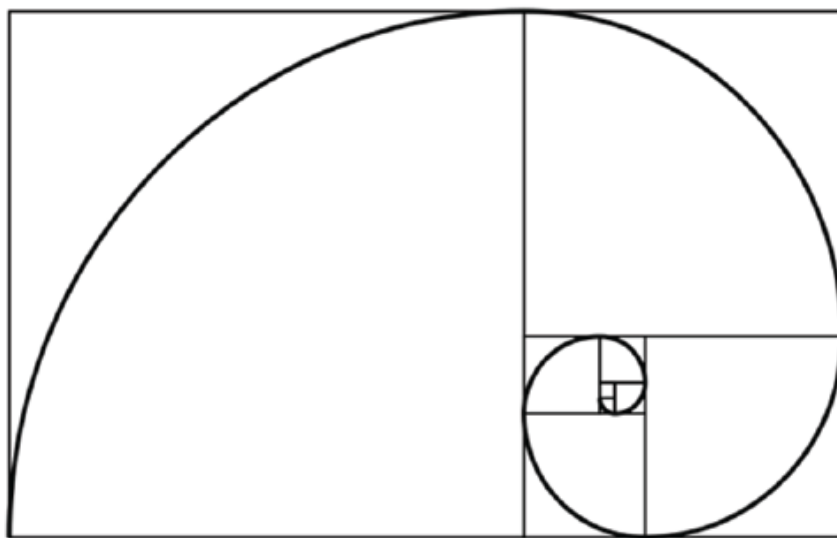


Рис. 453. Вигляд «золотого перерізу»

«Золотий переріз» можна отримати, якщо розділити відрізок на дві нерівні частини так, щоб відношення всього відрізка (у нашому випадку С) до більшої частини (В) дорівнювало відношенню більшої частини відрізка (В) до меншої (А).

У пропорціях це виглядатиме так:

$$\frac{C}{B} = \frac{B}{A} = \varphi \approx 0,618$$

У відсотковому співвідношенні – це розподіл будь-якої величини щодо 62% і 38%. Цю пропорцію називають **золотим числом** або φ (**фі**) (Рис. 454).

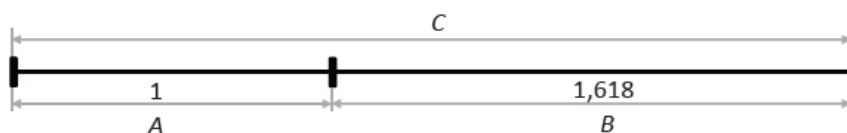


Рис. 454. Отримання «золотого перерізу»

У дизайні принцип золотого перерізу часто застосовується інтуїтивно – наприклад, при побудові сітки сторінки, розміщенні заголовків, зображень

або кнопок. Ці пропорції допомагають створювати візуальну рівновагу між елементами, коли жоден із них не переважає, але погляд природно спрямовується до центру композиції (Рис. 455).

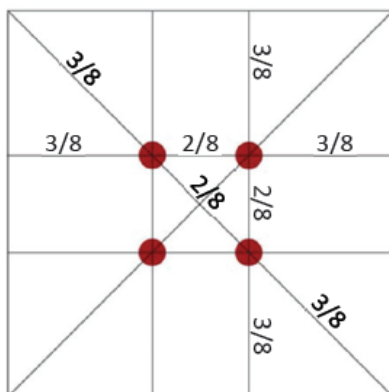


Рис. 455. Розміщення зорових центрів

На практиці для зручності дизайнери часто користуються **правилом третин** – спрощеним варіантом золотого перерізу. Воно ділить площину на три рівні частини по горизонталі та вертикалі. Будь-яка з чотирьох точок перетину цих ліній – чудове місце для розташування головного акценту, наприклад, фото, заголовка або кнопки дії (Рис. 456).

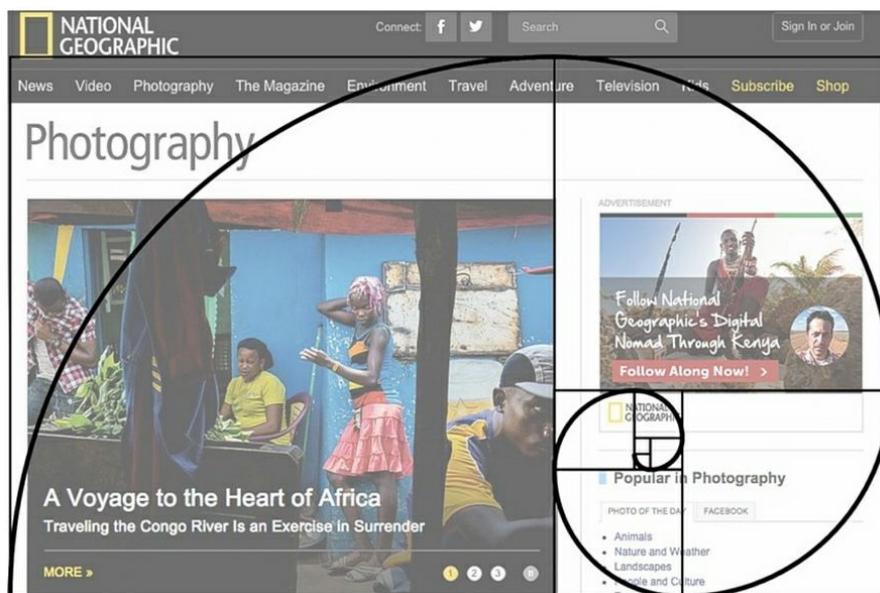


Рис. 456. Дизайн сайту, що побудований на основі «золотого перерізу»

4.5. ІНТЕРАКТИВНІСТЬ ТА АНІМАЦІЯ У ВЕБДИЗАЙНІ (GIF, SVG, CSS, JAVASCRIPT)

Сучасний веб – це не просто сторінки з текстом і зображеннями. Користувач очікує, що сайт «відповідатиме» на його дії, підказуватиме, що відбувається, і створюватиме відчуття живої взаємодії. Саме це і є **інтерактивність** – здатність інтерфейсу реагувати на людину. А **анімація** додає руху, часу й логіки у зміни на екрані. Разом вони роблять досвід користувача приємним, зрозумілим і більш «людським».

Анімація створює відчуття руху й часу, допомагає зрозуміти, що змінилося. Інтерактивність додає реакцію на дії користувача – натискання, наведення, прокрутку. Разом ці два інструменти формують позитивний досвід взаємодії з сайтом, коли він не просто показує інформацію, а спілкується з людиною.

Коли кнопка м'яко змінює колір при наведенні – це сигнал, що «на мене можна натиснути». Коли після відправки форми з'являється коротке повідомлення «успішно надіслано» – це підтвердження, що дія виконана. А коли щось пішло не так і поле заповнене неправильно, червоне підсвічування або підказка допомагають одразу зрозуміти, що виправити.

Без цього зворотного зв'язку сайт виглядає «мертвим». Користувач натискає кнопку – нічого не відбувається, і він починає сумніватися: чи спрацювало? Чи зависло? чи я зробив щось не так? Така невизначеність викликає роздратування та змушує закривати сторінку.

Головне правило – **усе має мати сенс**. Ефект заради ефекту швидко набридає. Але якщо кожна дія користувача викликає передбачувану реакцію – це створює відчуття контролю й довіри. Людина розуміє, що саме відбувається, чому і що буде далі.

Уявімо звичайну форму для введення електронної пошти. Користувач вводить неправильну адресу, натискає «Надіслати» – і нічого не відбувається. Екран «мовчить», форма просто підстрибує, і людина не розуміє, що не так.

Якщо ж те саме поле м'яко підсвічується червоним, а під ним з'являється коротке повідомлення – «Введіть дійсну адресу електронної пошти». Жодної паніки, жодних здогадок. Людина миттєво розуміє, що саме потрібно змінити.

Це й є хороші інтерактивність та анімація – не показні, не нав'язливі, а людяні. Вони не відволікають, а допомагають.

Усі ці реакції можна реалізувати різними способами:

- **CSS** – для плавних переходів, тіней, масштабування чи прозорості;
- **JavaScript** – для складніших сценаріїв: прокручування, затримок, інтерактивних графіків;
- **SVG-анімації** – для ліній, логотипів, іконок, які реагують на рух миші або торкання;
- **GIF** – для коротких демонстрацій або декоративних ефектів (але не бажано перевантажувати сторінку такими файлами).

GIF-АНІМАЦІЯ

Формат GIF – один із найстаріших способів «оживити» вебсторінку. Фактично це набір послідовних кадрів, що змінюються настільки швидко, що створюють ілюзію безперервного руху. Фактично, це маленький мультфільм, зібраний із кадрів, що повторюються по колу.

Головна перевага GIF-анімації – простота. Вона не потребує жодного програмування: достатньо підготувати кадри, зберегти їх у GIF-форматі – і готово. Такий файл відкриється у будь-якому браузері, навіть дуже старому, тому цей формат довго залишався стандартом для коротких банерів, логотипів і декоративних вставок.

Та з розвитком технологій у GIF з'явилися й помітні обмеження. Формат підтримує лише 256 кольорів, не має справжньої прозорості (видимі «сходинки» навколо об'єктів) і часто важить забагато, насамперед якщо кадрів багато або анімація довга. Через це завантаження сторінки може сповільнюватися.

Сьогодні GIF поступово поступається місцем сучаснішим рішенням: **WebM** та **SVG-анімації**, які дають краще стиснення і плавніший результат.

Попри це, GIF і досі має свою нішу. Його варто використовувати для **простих і коротких ефектів**, які не потребують великої кількості кольорів чи інтерактивності. Але потрібен невеликий, але емоційний штрих.

Наприклад, до анімації було статичне зображення чашки кави без жодного руху. Але варто додати легкий рух пари, що піднімається над чашкою і картинка ніби оживає. Ефект простий, однак створює атмосферу теплоти й затишку.

SVG-АНІМАЦІЯ

SVG (Scalable Vector Graphics) – це векторний формат зображення, який відтворює не пікселі, а математичні лінії. Завдяки цьому іконки й логотипи в SVG є чіткими на будь-якому екрані – від телефону до 4K-монітора. Але головна перевага SVG у тому, що його можна **анімувати просто в коді**.

Є кілька способів це зробити. Раніше використовували вбудований механізм SMIL, але тепер частіше застосовують анімацію через **CSS** або **JavaScript**, що дає більше контролю.

За допомогою SVG можна створити ефект «намальованої лінії»: коли логотип поступово «з'являється», ніби його малюють просто на очах. Для цього змінюють параметри **stroke-dasharray** і **stroke-dashoffset** у CSS, і лінія відкривається крок за кроком.

Отже, SVG-анімація ідеальна для логотипів, діаграм та іконок, де важлива чіткість і мала вага. Крім того, SVG можна робити інтерактивним – змінювати колір чи рух при наведенні курсора.

Наприклад, на сайті кав'ярні можна зробити так, щоб при завантаженні логотип плавно з'являвся з контурів – спочатку обводка чашки, потім димок, далі текст. Жодного відео, а ефект приємний, легкий і запам'ятовується.

АНІМАЦІЯ ЗАСОБАМИ CSS

CSS-анімація – це спосіб додати рух або плавний перехід між станами елементів, використовуючи лише стилі. Вона виконується прямо у браузері, без завантаження додаткових бібліотек чи JavaScript-коду. Її головна перевага –

простота й ефективність: ефекти виглядають плавно, займають мало ресурсів і не уповільнюють роботу сторінки.

CSS має два основні інструменти для створення руху – **transition** (перехід) і **@keyframes** (послідовність кадрів).

Перший відповідає за плавну зміну властивостей під час взаємодії (наприклад, наведення миші). Другий – за створення складніших ефектів із кількома фазами руху, які можуть повторюватися нескінченно або зупинятися після одного циклу.

Прості переходи (transition)

Перехід – це найпростіша форма анімації. Він реагує на події, коли елемент змінює свій стан. Щоб перехід працював, достатньо вказати, які саме властивості змінюються і за який час це має відбуватися.

Наприклад, маємо кнопку: у спокійному стані вона синя, а при наведенні – темніє. Без **transition** колір зміниться різко, але якщо додати одну стрічку коду, усе стає плавним:

```
button {  
  background: #2563eb;  
  transition: background 0.3s ease;  
}  
button:hover {  
  background: #1e40af;  
}
```

Коли користувач наводить курсор, браузер автоматично прораховує проміжні значення кольору й відтворює плавний перехід.

Можна задавати не лише колір, а й тінь, розмір, прозорість, нахил – майже будь-яку властивість, що підтримує анімацію.

Складні рухи (@keyframes)

Якщо потрібно складніший рух – наприклад, коливання або пульсацію – використовують **@keyframes**. Це набір ключових моментів руху, між якими браузер сам обчислює проміжні стани.

Ми визначаємо початок, середину і кінець анімації – і CSS робить решту. Наприклад, нехай кнопка злегка «дихає», збільшуючись і повертаючись до звичайного розміру.


```
@keyframes pulse {  
  0% { transform: scale(1); }  
  50% { transform: scale(1.1); }  
  100% { transform: scale(1); }  
}  
button {  
  animation: pulse 1.5s infinite;  
}
```

У цьому прикладі кнопка повторює рух безперервно. Ми можемо налаштовувати тривалість (1.5s), кількість повторів (**infinite** або конкретне число), напрямок (**alternate** – туди-назад) і навіть затримку перед початком.

Наприклад, **animation-delay: 0.3s;** відтермінує старт, щоб рух не виглядав занадто синхронним з іншими елементами.

CSS-анімації – це не лише краса. Вони допомагають користувачеві орієнтуватися.

Коли новий елемент з’являється плавно, око встигає зафіксувати його появу. Коли меню розгортається зверху вниз, користувач розуміє, звідки воно з’явилося і куди зникне.

Такі деталі роблять взаємодію логічною, навіть якщо людина не замислюється над цим свідомо.

Добре спроектована анімація завжди має смислову роль. Вона:

- підказує дію (підсвічує активну кнопку);
- підтверджує результат (плавна поява повідомлення «успішно надіслано»);
- полегшує сприйняття змін (м’яке оновлення списку).

Але потрібно пам’ятати, що не всі властивості однаково швидко обробляються браузером. Найкраще працюють **transform** і **opacity**: вони використовують апаратне пришвидшення і не змушують браузер перераховувати розкладку.

Натомість властивості типу **width**, **top**, **left**, **margin** змінюють геометрію сторінки і це може викликати «підгальмовування».

АНІМАЦІЯ ЗА ДОПОМОГОЮ JAVASCRIPT

JavaScript відкриває двері до справжньої динаміки. Якщо CSS дає змогу створювати простий рух, то JavaScript потрібен для складних сценаріїв – інтерактивних елементів, паралакс-ефектів, 3D-рухів, а також для узгодження анімації з діями користувача. За його допомогою можна керувати часом появи елементів, запускати рух лише тоді, коли об'єкт потрапляє у зону видимості, створювати паралакс-ефекти, інтерактивні карти та навіть мініігри.

У будь-якій анімації є одна проста ідея: у нас є об'єкт із певною властивістю (позиція, масштаб, колір, прозорість), і ми поступово змінюємо цю властивість у часі. JavaScript дає змогу керувати цим процесом вручну, кадр за кадром.

Колись для цього використовували метод `setInterval()`, який просто запускав функцію через певний інтервал часу. Але сучасні браузерери мають кращий інструмент – `requestAnimationFrame()`. Він працює розумніше: «просить» браузер оновити кадр тоді, коли екран уже готовий до нового зображення. Завдяки цьому рух виглядає плавно, без ривків і розсинхронізації.

Розглянемо приклад, в якому коло на сторінці плавно рухається праворуч:

html:

```
<div id="circle"></div>
```

css:

```
#circle {  
  width: 50px;  
  height: 50px;  
  background: crimson;  
  border-radius: 50%;  
  position: absolute;  
  left: 0;  
  top: 100px;  
}
```

js:

```
let x = 0;  
function move() {  
  x += 2;  
  circle.style.left = x + 'px';  
}
```

```

    if (x < 400) {
      requestAnimationFrame(move);
    }
  }
  move();

```

Щоразу, коли викликається **move()**, коло зміщується на 2 пікселі праворуч, а **requestAnimationFrame()** запускає наступний кадр тоді, коли браузер готовий його намалювати. У результаті отримуємо плавний рух без «стрибків». Як тільки координата **x** досягає 400, цикл зупиняється.

Справжня сила JavaScript – у здатності реагувати на поведінку людини. Наприклад, можна зробити, щоб елемент проявлявся лише після натискання кнопки.

Розглянемо приклад, коли користувач натискає кнопку, блок поступово «з’являється» – від прозорого до повністю видимого. Така дрібна деталь додає відчуття, що сайт «чує» людину та реагує на неї.

```

const box = document.querySelector('.fade');
const btn = document.querySelector('button');

btn.addEventListener('click', () => {
  let opacity = 0;

  function fadeIn() {
    opacity += 0.02;
    box.style.opacity = opacity;
    if (opacity < 1) requestAnimationFrame(fadeIn);
  }

  fadeIn();
});

```

JavaScript-анімації незамінні, коли потрібно більше, ніж просто плавний перехід. Наприклад:

- ефект паралаксу – коли фонові елементи рухаються повільніше за передній план, створюючи відчуття глибини;
- індикатори завантаження або лічильники, які оновлюються в реальному часі;

- інтерактивні карти, 3D-графіка (завдяки бібліотекам на кшталт Three.js);
- реакція на курсор, нахил пристрою чи дотик.

CSS може зробити красивий рух, але JavaScript дозволяє руху мислити – підлаштовуватися, змінювати швидкість, реагувати на події.

Створювати складну анімацію вручну – складне завдання, тому розробники часто користуються готовими **бібліотеками**, які економлять час і допомагають створювати складні ефекти без зайвої математики:

- **GSAP** (GreenSock Animation Platform) – інструмент для створення сцен, таймлайнів і синхронізації ефектів.
- **Anime.js** – бібліотека, яка підходить для плавних переходів, руху SVG-елементів і мікроанімацій.
- **Framer Motion** – рішення для React, що дає змогу описувати анімацію як частину компонента.

JavaScript-анімації надають чимало можливостей, але важливо пам'ятати про продуктивність. Події типу **scroll** чи **mousemove** викликаються дуже часто десятки разів на секунду. Якщо кожен виклик запускає складні обчислення, сторінка починає «тормозити». Щоб цього уникнути необхідно:

- використовувати `requestAnimationFrame()`, щоб рух був плавним і синхронізованим із частотою екрана;
- змінювати лише легкі для рендеру властивості – **transform** та **opacity**;
- не запускати одночасно занадто багато ефектів – краще один якісний, аніж п'ять дрібних, що «з'їдають» продуктивність.

ПРАКТИЧНІ ПОРАДИ

По-перше, будь-яка анімація має бути швидкою. Якщо користувач чекає, поки «дограє» ефект, це не приємність, а затримка. Оптимальна тривалість більшості рухів – від 0.2 до 0.6 секунди.

По-друге, рух має посилювати зміст. Не варто рухати все одночасно – очі не встигають фокусуватися. Краще виділяти лише ключові дії: натискання, появу нового блоку, зміну стану.

По-третє, варто використовувати властивості, які апаратно пришвидшуються (transform, opacity) і не впливають на розмітку. Це зменшує навантаження на процесор і забезпечує плавність навіть на слабких пристроях.

І не потрібно забувати про доступність. У багатьох операційних системах користувач може ввімкнути налаштування «зменшити рух». У CSS для цього є спеціальний медіавираз:

```
@media (prefers-reduced-motion: reduce) {  
  * { animation: none !important; transition: none !important; }  
}
```

Так ми поважаємо комфорт користувача, не позбавляючи його функціональності.

Отже, інтерактивність і анімація – це не прикраси, а важливі інструменти комунікації у вебдизайні. GIF додає прості циклічні рухи без коду. SVG дає векторну чіткість і гнучкість. CSS забезпечує плавні переходи й мікроефекти. JavaScript додає справжню динаміку та логіку реакцій. Разом вони створюють живий, сучасний веб, у якому рух і поведінка мають сенс. І коли кожен елемент реагує передбачувано, користувач відчувається впевнено, а сайт здається продуманим і професійним.

4.6. АНАЛІЗ ТА ОПТИМІЗАЦІЯ ДОСВІДУ КОРИСТУВАЧА

Після створення сайту робота дизайнера не закінчується. Гарний інтерфейс – це лише половина справи. Друга половина – зрозуміти, як ним реально користуються люди: де вони зупиняються, що шукають, на якому етапі губляться або залишають сторінку. Саме для цього є інструменти аналітики, тестування та оптимізації UX/UI.

ІНСТРУМЕНТИ ДЛЯ АНАЛІЗУ ДОСВІДУ КОРИСТУВАЧА GOOGLE ANALYTICS ТА HOTJAR

Якщо сайт працює, важливо розуміти не лише, як він виглядає, а й як люди ним користуються насправді. Інколи сторінка може бути гарною, але незручною, і тоді користувач просто закриває вкладку. Щоб цього не сталося, дизайнери та розробники використовують інструменти аналітики – вони показують, що відбувається «по той бік екрана».

Серед найпоширеніших і найзручніших інструментів – **Google Analytics** та **Hotjar**. Вони не конкурують, а чудово доповнюють один одного.

GOOGLE ANALYTICS

Google Analytics можна назвати «очима» власника сайту. Завдяки йому видно все: звідки приходять користувачі, які сторінки їм цікаві, скільки часу вони проводять на сайті і в який момент вирішують піти. Цей сервіс дає змогу зрозуміти, як люди знаходять ваш сайт – через пошук, рекламу, соцмережі чи з інших ресурсів. Також видно, на яких пристроях його відкривають – комп'ютері, планшетах чи телефоні.

Цифри, які показує аналітика, – це не просто статистика. Вони розповідають історію поведінки користувачів. Наприклад, якщо більшість відвідувачів залишає сторінку через кілька секунд, це може означати, що контент перевантажений, або кнопка «Купити» захована надто далеко. Google Analytics допомагає дизайнеру зрозуміти, які сторінки працюють добре, а які потребують змін, і дає змогу приймати рішення не навімання, а на основі фактів.

Перед тим, як почати збирати дані, потрібно створити обліковий запис на сайті analytics.google.com. Це безкоштовно і займає кілька хвилин. Після реєстрації можна додати свій сайт і система згенерує спеціальний код відстеження. Цей код потрібно вставити у вихідний код сайту, зазвичай у розділ `<head>`. Якщо сайт зроблено на CMS (наприклад, WordPress), достатньо встановити відповідний плагін – він зробить усе автоматично.

Після встановлення коду Google Analytics починає збирати дані. У розділі **Звіти** можна переглядати таку інформацію:

- Audience (Аудиторія) – показує кількість користувачів і їхні характеристики;
- Acquisition (Джерела трафіку) – звідки вони прийшли (пошук, соцмережі, реклама тощо);
- Behavior (Поведінка) – які сторінки відвідують найчастіше і як рухаються сайтом;
- Conversions (Конверсії) – наскільки ефективно сайт досягає цілей (наприклад, скільки людей заповнили форму або зробили покупку).

На вкладці **Real-time** можна побачити, як користувачі поведуться «в реальному часі», а саме скільки людей зараз на сайті, з яких сторінок вони почали перегляд і куди переходять далі.

Із часом ці дані можуть допомогти помітити закономірності: які сторінки приваблюють, а які відштовхують. На основі цього можна змінювати дизайн, тексти або структуру сайту.

HOTJAR

Якщо Google Analytics показує цифри, то **Hotjar** показує реальну поведінку користувачів. Це ніби «жива камера», що записує, як люди рухають курсор, куди натискають, доки прокручують сторінку і які елементи ігнорують.

Щоб почати, потрібно зареєструватися на hotjar.com і додати свій сайт. Так само, як у Google Analytics, Hotjar створить невеликий скрипт – його потрібно вставити у код сайту. Після цього сервіс почне фіксувати дії користувачів.

Перший інструмент, який варто спробувати, – **Heatmaps (теплові карти)**. Вони показують, які ділянки сторінки отримують найбільше кліків і уваги. Гарячі червоні зони – це місця, куди користувачі найчастіше натискають, а сині – ті, які залишаються без уваги. Якщо відображається, що люди клікають на зображення, яке не є посиланням, – це знак, що варто змінити дизайн або зробити його активним.

Другий інструмент – **Recordings (записи сесій)**. Тут можна подивитися короткі відео того, як користувач рухається сторінкою: куди наводить курсор, коли зупиняється, що намагається натиснути. Це дуже корисно, щоб зрозуміти, де саме людина заплуталася або здалася.

Також Hotjar має функцію **опитувань і відгуків**. Наприклад, можна поставити відвідувачам просте запитання: «Що б ви покращили на цій сторінці?» або «Чи легко було знайти потрібну інформацію?». Це короткі анкети, які з'являються на екрані ненав'язливо, але дають цінні підказки.

Ще одна корисна функція – **Funnels (воронки)**. Вона показує, на якому етапі користувачі найчастіше залишають сайт. Наприклад, можна побачити, що більшість відвідувачів додають товар у кошик, але не доходять до оплати. Це сигнал, що в інтерфейсі сторінки оформлення є щось, що заважає людині завершити покупку.

Google Analytics і Hotjar – це як два різні погляди на той самий сайт і дають найкращий результат у парі. Один показує, *де* є проблема, інший – чому вона виникає. Уявімо ситуацію: аналітика показує, що більшість користувачів залишає сторінку оформлення замовлення. Ми відкриваємо Hotjar і бачимо, що люди не помічають кнопку «Далі» або заплутуються в полі для заповнення адреси. Завдяки поєднанню цифр і спостережень можна швидко знайти причину і виправити її.

ТЕСТУВАННЯ КОРИСТУВАЦЬКИХ ІНТЕРФЕЙСІВ

Аналітика дає цифри, але, щоб зрозуміти реакцію людей, потрібне тестування. Будь-який інтерфейс, навіть найкрасивіший і найпродуманіший, потрібно перевіряти на реальних людях. Це як репетиція перед виставою. У дизайні така «репетиція» називається тестуванням користувацького інтерфейсу. Воно допомагає дізнатися, чи зручно користувачу виконувати завдання, чи розуміє він логіку сайту, і чи не викликають елементи інтерфейсу плутанини.

Тестування дає змогу подивитися на сайт очима звичайної людини, а не дизайнера чи розробника. У макеті може здаватися, що все зрозуміло, але в реальному житті користувач натискає зовсім не туди, де ми очікували. Кнопка може опинитися занадто далеко, форма – виглядати страшно, а посилання –

непомітним. Саме під час тестування проявляються ці «дрібниці», які потім вирішують, чи користувач залишиться на сайті, чи закриє його за кілька секунд.

Є чимало способів перевірити зручність інтерфейсу, але суть одна – показати продукт живим людям і подивитися, як вони ним користуються. Це може бути найпростіше юзабіліті-тестування, коли учасникам дають кілька завдань: знайти товар, оформити замовлення або зареєструватися. Дизайнер у цей час спостерігає, які дії викликають труднощі. Важливо не підказувати, не виправдовуватись – просто дивитися, як люди поведуться. Іноді навіть коротке спостереження відкриває очі на речі, які здавалися очевидними.

Ще один ефективний метод – А/В-тестування. Це, коли є два варіанти одного елемента: наприклад, синя кнопка й зелена, або коротка форма й довга. Користувачам випадковим чином показують різні версії, а потім аналізують, яка працює краще. Такі експерименти допомагають приймати рішення на основі фактів, а не інтуїції.

Інколи тестування проводять ще до запуску сайту – на рівні прототипів. Це дуже зручно, бо можна виявити проблеми ще до того, як витрачено час і ресурси на розробку. У Figma або інших інструментах робиться клікабельний макет, який імітує роботу реального сайту. Людина натискає, переходить між екранами, і дизайнер бачить, наскільки інтуїтивно вона орієнтується у структурі.

Тестування не обов'язково проводити в офісі. Зараз популярне віддалене тестування: користувачі виконують завдання вдома, а система записує відео з екрану, рухи курсора і навіть реакції. Це зручно, коли аудиторія з різних міст або країн. Окрім того, у «домашніх» умовах люди поведуться природніше, ніж у лабораторії, тому результати виходять точніші.

Щоб тестування було корисним, потрібно добре підготуватися. Спершу визначають, що саме потрібно перевірити: наприклад, чи бачать користувачі кнопку «Завантажити» або чи можуть без проблем знайти сторінку контактів. Потім обирають учасників, які максимально схожі на цільову аудиторію сайту – не дизайнерів, не колег, а звичайних людей, для яких продукт і створюється. Перед тестом готують прості сценарії – наприклад, «знайди товар і додай у кошик»

чи «заповни форму реєстрації». Далі все просто: дизайнер уважно спостерігає й фіксує кожну реакцію, кожне «де це?» або «чому не працює?». Саме ці моменти найцінніші.

Результати тестування часто відкривають очі на речі, які здавалося, давно відпрацьовані. Наприклад, користувачі можуть не помічати головну кнопку або плутатися в меню. Інколи виявляється, що форма занадто довга, а на мобільному її навіть не видно повністю. Або користувач натискає «Надіслати» і не розуміє, що сталося, бо сайт не дає жодного зворотного зв'язку.

Сучасні інструменти значно полегшують цей процес. У Figma можна створити прототип і протестувати його через платформу Maze. Hotjar записує відео реальних сесій і показує теплові карти кліків. Google Optimize допомагає швидко налаштувати A/B-тестування, а сервіси на кшталт Lookback чи UserTesting дозволяють спостерігати за користувачами в реальному часі. Це вже не теоретика – це практична аналітика, яка показує, що насправді відчуває людина під час роботи із сайтом.

ПОКАЗНИКИ ЕФЕКТИВНОСТІ UX/UI-ДИЗАЙНУ

Щоб оцінити якість дизайну об'єктивно, є кількісні показники.

Одним із базових показників є **час виконання завдання**. Він показує, скільки часу потрібно користувачеві, щоб знайти потрібну інформацію, оформити замовлення чи заповнити форму. Що швидше люди досягають своєї мети, то зручніший інтерфейс. Якщо користувач витрачає на просту дію кілька хвилин, варто задуматися над спрощенням структури або навігації.

Не менш важливий показник – **коефіцієнт успішності завдань**. Це частка людей, які без сторонньої допомоги змогли виконати поставлену дію. Наприклад, зареєструватися, оплатити товар чи знайти документ. Якщо успішність низька, це означає, що логіка інтерфейсу незрозуміла або кроків забагато.

Ще один важливий індикатор – **рівень помилок**. Кожна помилка користувача – це дзвіночок для дизайнера. Якщо люди часто вводять неправильний формат e-mail чи номеру телефону, значить форма не підказує, як саме потрібно вводити дані.

А якщо користувачі не помічають повідомлення про помилку, значить воно недостатньо виразне або з'являється в невдалому місці.

Окрім того, аналізують **поведінкові метрики**: кількість кліків, глибину перегляду, середній час на сторінці, відсоток відмов (bounce rate). Високий рівень відмов часто говорить, що сторінка не відповідає очікуванням або користувач не знаходить потрібне одразу. А якщо люди проводять на сайті багато часу, переходять між розділами – це хороший знак, що дизайн утримує увагу.

Окремо вимірюють **залученість користувачів**. Це можна зробити через відгуки, опитування або індекси, як-от **NPS (Net Promoter Score)** – показник, що показує, скільки людей готові рекомендувати сайт іншим.

Якщо користувачі повертаються знову і знову, взаємодіють з елементами, заповнюють форми, залишають коментарі – це означає, що досвід приємний і дизайн не заважає сприйняттю контенту.

Іноді проводять **емоційне тестування** – коли спостерігають за реакціями людей під час використання сайту або після нього. Якщо більшість користувачів описують досвід як «легкий», «зрозумілий», «приємний» – це найкраще підтвердження, що UX-дизайн спрацьовує так, як задумано.

ОПТИМІЗАЦІЯ ІНТЕРФЕЙСУ ЗА РЕЗУЛЬТАТАМИ АНАЛІТИКИ ТА ТЕСТУВАННЯ

Після того, як зібрано дані – цифри з Google Analytics, теплові карти з Hotjar, результати тестування з реальними користувачами – починається, мабуть, найцікавіша частина роботи UX/UI-дизайнера. Це **етап оптимізації**, коли дизайн не просто оцінюють, а вдосконалюють.

Аналітика і тестування самі собою не змінюють продукт – вони лише показують, де є проблема. А завдання дизайнера – зрозуміти причину й зробити так, щоб користувачу стало простіше, швидше і приємніше виконувати свої дії.

Робота починається з аналізу проблем. Наприклад, з аналітики видно, що користувачі масово залишають сторінку оформлення замовлення. На записах Hotjar видно, що вони зупиняються на полі «телефон». Це сигнал, що можливо форма

занадто складна, або формат введення неочевидний. Тут важливо не вгадувати, а шукати причини. Інколи проблема є на поверхні – кнопка невдала або надто далеко. А інколи – у психології: користувач не розуміє, навіщо запитують певні дані, або боїться вводити їх без гарантії безпеки.

Після визначення причини дизайнер пропонує **гіпотези**: що саме можна змінити, аби поліпшити ситуацію. Наприклад, скоротити форму, додати підказку, змінити колір кнопки або зробити повідомлення про помилку більш зрозумілим. Ці гіпотези потім перевіряються новими тестами – і так крок за кроком інтерфейс стає дедалі кращим.

Зазвичай оптимізація – це не масштабна переробка, а серія дрібних, але важливих поліпшень. Зміна контрасту кнопки, додавання мікроанімації під час наведення або правильна форма повідомлення про помилку можуть суттєво вплинути на поведінку користувача.

Буває, що навіть один маленький штрих – наприклад, заміна тексту «Надіслати» на «Отримати консультацію», збільшує кількість кліків у кілька разів.

Оптимізація UX – це постійний процес. Сайт або застосунок ніколи не бувають «ідеальними назавжди», бо змінюються звички користувачів, пристрої, навіть очікування. Те, що працювало рік тому, сьогодні може виглядати застаріло або незручно. Тому аналітика і тестування не закінчуються після запуску – вони тривають постійно, допомагаючи адаптувати продукт до нових умов.

ОПТИМІЗАЦІЯ ГРАФІКИ І HANDOFF

Після внесення функціональних покращень настає наступний важливий етап – **оптимізація графіки**. Навіть ідеально продуманий інтерфейс може працювати повільно, якщо зображення занадто «важкі», шрифти не оптимізовані, а кольори створюють надмірне навантаження на очі.

Менша вага ресурсів пришвидшує відображення вмісту сторінки, стабілізує верстку і піднімає Core Web Vitals (набір ключових метрик від Google, які показують, наскільки сайт відчувається швидким, стабільним і чуйним для користувача. Вони безпосередньо впливають на UX і враховуються в пошуку).

Користувач отримує простий ефект: сторінка відкрилася миттєво і не дратує. Норми контрасту беруть із рекомендацій WCAG 2.2, а синтаксис сучасних кольорових просторів – із CSS Color Module Level 4⁵.

ОПТИМІЗАЦІЯ ЗОБРАЖЕНЬ У ВЕБДОДАТКАХ

Зображення і графіка – це одна з частин будь-якої вебсторінки, що потребує найбільше ресурсу (найважча). Якщо вони завеликі чи збережені у неправильному форматі, сторінка завантажується повільно. Це призводить до того, що користувачеві доводиться чекати і він може просто закрити сторінку і більше не відвідувати сайт. Згідно з дослідженнями Google, затримка на 1 секунду може знизити кількість користувачів, що повернулися на сайт на 7–10%. Також пошукові системи знижують рейтинг сайту та витрачається на нього більше мобільного трафіку.

Водночас легка графіка зменшує навантаження на сервер і економить трафік, що важливо для мобільних пристроїв. Суттєвою є й користь для SEO. Пошукові системи враховують швидкість завантаження як один із факторів ранжування.

Тому для того, щоб сайт був якісним, зручним, конкурентоспроможним, доступним, працював швидко і без зайвих перешкод, графіка на його сторінках має бути оптимізована.

Щоб картинки на сайті не займали багато місця і водночас виглядали чітко, їх потрібно правильно підготувати, а саме вибрати найбільш підходящий формат.

Для іконок, логотипів чи схем найкраще підходить формат SVG – він залишається достатньо різким незалежно від масштабу.

Якщо ж йдеться про фотографії чи складні ілюстрації, варто використовувати такі формати, як WebP або AVIF, що мають значно менший розмір за звичні JPG чи PNG, але зберігають гарну якість.

Коли ж потрібна прозорість, рекомендовано обирати PNG-24 або WebP із підтримкою прозорого фону.

Просте правило: іконки та схеми – SVG, фото – WebP/AVIF, прозорість – PNG, фолбек – JPG.

Приклад. Порівняльна таблиця форматів JPG та WebP із однаковою якістю.

Формат	Розмір файлу	Якість зображення	Особливості
JPG	~200 Кб	Візуально чітке	Старіший формат, підтримується всюди, але файли більші за вагою
WebP	~60 Кб	Така сама чіткість, як у JPG	Сучасний формат від Google, файли у 2–3 рази менші, підтримує прозорість та анімацію

Окрім вибору формату зображення, його можна зберегти з різним рівнем стискання. Якщо виставити занадто високу якість, файл буде великий. Якщо занадто низьку – картинка стане «замиленою». Так для форматів WebP/AVIF оптимальною буде якість 60–80%, а для JPG – 70–85% (у прогресивному режимі, щоб зображення завантажувалося поступово). За вибору будь-якого формату потрібно перевіряти «на око», чи картинка виглядає добре і чи вона не завелика за розміром.

Різні пристрої потребують різних розмірів зображень. Наприклад, якщо пристрій має роздільну здатність 800×600, то немає сенсу завантажувати фото 4000×3000 рх. Тому в проєкті необхідно передбачити, які зображення завантажуватимуться на одних пристроях, а які на інших.

HTML має спеціальний механізм – **responsive images**, який дає змогу браузеру вибрати правильний розмір, економлячи трафік і забезпечуючи гарну якість. За це відповідають атрибути **srcset** і **sizes** у тегу ****, які допомагають браузеру вибрати оптимальний варіант залежно від розміру та роздільної здатності екрану:

- якщо екран маленький (телефон) – буде підвантажуватися версія 400 рх;
- якщо середній (планшет/ноутбук) – 800 рх;
- якщо великий монітор – 1200 рх.

```

```

Зменшити розмір зображень допомагають також спеціальні безкоштовні сервіси для стиснення, наприклад **TinyPNG**, **Squoosh** чи **ImageOptim**. Вони зменшують розмір файлу без помітної втрати якості завдяки видаленню зайвих даних із файлів.

Перевірити вагу файлів можна у сервісі **Chrome DevTools**. Для цього потрібно відкрити сайт, натиснувши F12, перейти на вкладку **Network** і встановити фільтр **Img**. З'явиться список з усіма зображеннями та їхніми розмірами. Це дасть змогу оцінити, які зображення достатньо легкі (100 Кб), а які потрібно оптимізувати (5 Мб).

Щоб сторінка відкривалася швидше, застосовують **Відкладене завантаження (Lazy loading)**. При **Lazy loading** зображення, які знаходяться нижче «першого екрану», не завантажуються одразу. Вони підвантажуються лише тоді, коли користувач прокрутить сторінку вниз.

```

```

Це пришвидшує відображення верхньої частини сторінки і робить сайт відчутно легшим.

ОПТИМІЗАЦІЯ ШРИФТІВ

Шрифти безпосередньо впливають на швидкість першого відображення тексту: якщо файл важкий або завантажується з затримкою, користувач бачить «порожні» блоки або мигтіння тексту, що погіршує досвід.

Для роздачі у вебі базовим форматом сьогодні є **формат вебшрифтів WOFF2**, який має менший розмір порівняно з форматами TTF чи OTF і підтримується майже всіма браузерами. Правильне підключення – це не тільки вибір формату, а й порядок властивостей і стратегія показу.

Для того, щоб показати текст системним шрифтом одразу, поки основний шрифт ще завантажується використовується параметр **font-display: swap;**. Це критично при повільному (мобільному) інтернеті – інакше користувач 2–3 секунди дивитиметься на «білу сторінку».

У прикладі нижче підключається змінний шрифт і одразу вмикається миттєвий показ тексту.

```
@font-face {  
    font-family: 'Inter';  
    src: url('/fonts/inter.woff2') format('woff2');  
    font-display: swap;  
}
```

Якщо сайт українською та англійською мовою, на ньому не потрібні ієрогліфи чи кирилиця розширених діапазонів. Тому варто зробити **subset** (залишити тільки потрібні символи). Обрізавши зайві символи, можна зменшити вагу шрифту у 2–3 рази. Можливі 2 варіанти subset: якщо сайт на Google Fonts та якщо свій шрифт.

Якщо сайт на Google Fonts, потрібно:

1. Зайти на Google Fonts.
2. Вибрати потрібний шрифт (наприклад, **Inter** або **Roboto**).
3. Додати параметр **&subset=latin,cyrillic** у посилання.
4. Додати **&display=swap**, щоб не було «порожнього тексту».

Приклад:

```
<link  
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;700&display=sw  
ap&subset=latin,cyrillic" rel="stylesheet">
```

Якщо сайт на **своєму шрифті** – найпростіше використати онлайн-сервіс. Для цього потрібно:

1. Завантажити шрифт.
2. Зайти на сервіс, наприклад: **Font Squirrel Webfont Generator** або **Transfonter**.
3. Вибрати галочку **Subset** та залишити тільки латиницю та кирилицю.
4. Завантажити готовий **.woff2** і підключити його у CSS:

```
@font-face {  
    font-family: 'MyFont';  
    src: url('/fonts/MyFont.woff2') format('woff2');
```

```
    font-display: swap;
}
```

Замість того, щоб підключати окремо шрифти Regular, Bold і Italic, можна взяти один **variable**-файл, у який всі стилі вже вбудовані. Це пришвидшує завантаження та суттєво скорочує кількість запитів. Наприклад:

```
body {
    font-family: 'InterVariable', sans-serif;
    font-weight: 300; /* Тонкий */
}
h1 {
    font-weight: 700; /* Жирний */
}
```

ОПТИМІЗАЦІЯ КОЛЬОРІВ

Кольори визначають, чи виглядає інтерфейс цілісним, чи дотримано контраст для читабельності, чи немає «візуального шуму».

Найпоширеніша проблема – «палітра з голови». Коли кожен розробник або дизайнер додає у код випадкові hex-значення, інтерфейс швидко втрачає цілісність. Вихід – система токенів: ми називаємо кольори за призначенням, а не за відтінком. Наприклад, замість того, щоб писати #1A73E8 у компоненті, ми використовуємо змінну `color-primary`, для тексту – `color-text`, для фону – `color-bg`. У корені оголошуємо значення, а компоненти посилаються лише на змінні. Приклад базової ініціалізації:

```
:root {
    --color-bg: #FFFFFF;
    --color-text: #0B1220;
    --color-muted: #5B6878;
    --color-primary: #2563EB;
    --color-primary-ink: #FFFFFF;
}
```

Усередині кнопки ми вже не думаємо про hex – ми працюємо зі змістом:

```
.button {
    background: var(--color-primary);
}
```



```
color: var(--color-primary-ink);  
}
```

Якщо деякий бренд змінює, наприклад, відтінок синього, потрібно оновити одне місце, і вся система підлаштовується автоматично.

Сучасні інтерфейси зазвичай мають світлу й темну теми. Тут логіка та сама: семантика стала, а значення змінюються. Ви тримаєте одні й ті самі токени, але для `data-theme="dark"` задаєте інші кольори. Це дає контроль і не множить стилі. Приклад темізації виглядає так:

```
:root[data-theme="light"] {  
  --color-bg:#FFFFFF;  
  --color-text:#0B1220;  
  --color-primary:#2563EB; }  
:root[data-theme="dark"] {  
  --color-bg:#0B0F17;  
  --color-text:#E6EAF2;  
  --color-primary:#60A5FA; }
```

Перемикач тем робиться в один-два рядки JavaScript, а ще можна підхоплювати системну тему користувача через `prefers-color-scheme`.

Обов'язковий елемент – контраст. За WCAG для звичайного тексту мінімальне співвідношення 4.5:1 між текстом і фоном, для великого тексту – 3:1. Тому світло-сірий на білому для абзаців – погана ідея, так само як пастельний синій на білому без посилення насиченості. Перевіряти контраст можна вбудованими інструментами браузера або будь-яким простим контраст-чекером. Потрібно пам'ятати також про стани елементів: одних лише кольорів для позначення помилок недостатньо. Варто додати іконку або текстове пояснення, щоб люди з порушенням сприйняття кольорів теж коректно сприймали сигнали.

Щоб отримати узгоджені стани наведення і натискання без ручного «підбору на око», користуються властивістю `color-mix()`. Наприклад, для кнопки можна зробити уточнення кольору на `hover`, змішавши основний колір із білим, а для `active` – із чорним. Це створює послідовну систему відтінків у всьому інтерфейсі й полегшує підтримку:

```
.button { --btn: var(--color-primary); background: var(--btn); }
.button:hover { background: color-mix(in srgb, var(--btn) 90%, white 10%); }
.button:active { background: color-mix(in srgb, var(--btn) 85%, black 15%); }
```

Доброю практикою є мати шкали кольорів на кшталт primary-50...900. Це пришвидшує прийняття рішень: наприклад, для бордерів беремо умовний primary-300, для фонів карток – primary-50, для заголовків – primary-900. Так не буде винаходитись відтінок щоразу наново і не будуть розмножуватись випадкові значення, які ускладнюють підтримку.

Також потрібно пам'ятати про видимий фокус для клавіатурної навігації. Це базова доступність. Простий і помітний контур дає відчутний плюс до юзабіліті:

```
:root { --focus: #22C55E; }
:focus-visible { outline: 3px solid var(--focus); outline-offset: 2px; }
```

Так користувач завжди розуміє, де зараз перебуває фокус, і може керувати інтерфейсом без миші.

Тепер з'єднаємо шрифти і кольори в одному невеликому прикладі. Уявімо кнопку «Оплатити». Ми підключили змінний шрифт, задали токени кольорів і оголосили стилі кнопки з плавними станами:

```
<button class="btn">Оплатити</button>
:root {
  --font-ui: "Inter Subset", "Inter Var", system-ui, -apple-system, "Segoe UI",
  Roboto, Arial, sans-serif;
  --bg:#FFFFFF; --ink:#0B1220; --muted:#5B6878; --primary:#2563EB;
  --primary-ink:#FFFFFF; --focus:#22C55E;
}
:root[data-theme="dark"] {
  --bg:#0B0F17; --ink:#E6EAF2; --muted:#94A3B8; --primary:#60A5FA; --primary-
  ink:#0B0F17;
}
body { font-family: var(--font-ui); background: var(--bg); color: var(--ink); }
.btn {
  font: inherit;
  padding:.75rem 1rem;
  border-radius:.75rem;
  border:1px solid color-mix(in srgb, var(--primary) 30%, var(--bg) 70%);
```

```

background: var(--primary);
color: var(--primary-ink);
transition: background .15s ease, transform .05s ease;
}
.btn:hover { background: color-mix(in srgb, var(--primary) 92%, white 8%); }
.btn:active { transform: translateY(1px); }
.btn:focus-visible {
  outline: 3px solid var(--focus);
  outline-offset: 3px;
}

```

Цей приклад демонструє ключові принципи: змінні шрифти для гнучкості та швидкості, токени для керування кольорами, автоматичні відтінки для станів і видимий фокус для доступності.

HANDOFF: ПЕРЕДАЧА ДИЗАЙНУ В РОЗРОБКУ

Handoff – процес, коли дизайн-проект переходить від дизайнера до розробника – не як красива картинка, а як набір чітких інструкцій і файлів, з яких зрозуміло, що саме і як реалізувати. Мета handoff – зробити так, щоб розробник не вгадував «на око», а мав точні формати, розміри, відступи, стилі й стани. Якщо handoff організований правильно, розробники витрачають менше часу на перепитування, роблять менше помилок і в результаті отримують продукт, який відповідає макету.

Експорт-гайд – це інструкція з підготовки документів для розробки. У ньому дизайнер відповідає на питання: у яких форматах, які саме розміри, як називати файли тощо. Наприклад, дизайнер може написати, що замість того, аби віддавати іконку в PNG, краще експортувати її у SVG, бо тоді вона є векторною, гнучко стилізується через CSS і мало важить. Для фотографій же доцільно підготувати одразу кілька розмірів у сучасному форматі WebP або AVIF – наприклад 400, 800 і 1200 пікселів по ширині, щоб фронтенд міг підключити srcset і віддавати користувачеві рівно стільки, скільки потрібно його екрану. Окремо варто зафіксувати параметри шрифтів і правила найменування:

коли кнопка називається `btn/primary/filled` у макеті, так само її має бути названо в коді, це економить години на пошук і синхронізацію.

Спес-лист – це «паспорт» елемента або цілої сторінки. У ньому коротко, але чітко фіксується геометрія (висота, ширина, відступи, пропорції), палітра (кольори тексту, фону, бордерів), типографіка (назва шрифту, розмір, міжряддя) і стани. Наприклад, для кнопки описують, що у стані «default» фон синій, у «hover» – темно-синій, у «disabled» – сірий, висота становить 44 пікселі, горизонтальні відступи – по 12 пікселів, а текст набраний Inter розміром 16 пікселів. Тут же коротко пояснюється, що відбувається у разі помилки: який текст показується, який колір індикатора, як виглядає підсвітка поля. Хороший spes-лист позбавляє розробника необхідності здогадуватися, і зменшує кількість повернень «на переробку».

Подивімося на практичний handoff. Дизайнер у Figma збирає макет сторінки, і паралельно готує активи: іконки експортує у SVG, фотографії – у WebP або AVIF одразу в кількох розмірах. Для ключових компонентів – кнопок, форм, карток – робить короткі spes-листи з конкретними значеннями. Далі він передає розробникові файл Figma та посилання на таблицю з параметрами. Розробник відкриває макет і в режимі Inspect бачить справжні значення відступів, кольорів і CSS-властивостей, а не намагається «виміряти лінійкою». У результаті код максимально точно відтворює дизайн.

Інструменти для handoff сьогодні доволі зручні. У самій Figma є режим Inspect, де розробник бачить розміри, кольори, тіні та CSS просто у макеті. Якщо команді потрібне додаткове структурування, підійдуть сервіси на кшталт **Zeplin** або **Avocode** – вони вміють імпортувати фрейми, розкласти стилі та формувати специфікації. Для зберігання spes-листів, експорт-гайдів і чек-листів зручно використовувати **Notion** або **Google Docs**: важливо, щоб у команди завжди було єдине місце, де всі зміни фіксуються і не губляться в чатах.

Після handoff обов'язково проводимо перевірку якості – і тут у нагоді стають **Lighthouse** і **AXE**. **Lighthouse** – це вбудований інструмент у Chrome, який проганяє

сторінку за чотирма напрямками: Performance, Accessibility, Best Practices і SEO, і видає звіт з оцінками від нуля до ста разом із конкретними рекомендаціями. По суті, він показує, наскільки швидко завантажується сторінка, чи немає критичних помилок у розмітці та безпеці, і чи коректно організовано базові SEO-речі. **AXE** – це автоматична перевірка доступності, яка виявляє такі проблеми, як відсутність підписів до полів, надто низький контраст тексту, відсутність видимого фокусу для клавіатурної навігації. Після прогону видається список помилок із короткими підказками, як їх виправити.

На завершення робиться короткий огляд проекту, щоб переконатися, що все узгоджено. Дивляться, чи оптимізовано зображення, чи вони у правильних форматах і чи підготовлені вони у кількох розмірах для різних екранів; чи всюди проставлені alt-тексти; чи підключено srcset; чи видно фокус під час навігації клавіатурою і чи проходить контраст тексту вимоги WCAG; чи показують перевірки Lighthouse стабільно високі бали у продуктивності, доступності та найкращих практиках. Коли handoff, експорт-гайди, спец-листи і перевірки у Lighthouse/axe працюють як одна система, сайт виходить швидким, точним у реалізації дизайну і доступним для всіх – саме таким, яким його задумав дизайнер і чого очікує користувач.

4.7. ДОСТУПНІСТЬ (WCAG 2.2)

Сучасний вебдизайн має бути не лише гарним, а й **доступним для всіх користувачів** – незалежно від їхніх фізичних можливостей чи способу взаємодії з технікою. Стандарт **WCAG 2.2 (Web Content Accessibility Guidelines)** допомагає зробити інтерфейси зрозумілими людям із різним рівнем зору, моторики або когнітивного сприйняття. Він сумісний із попередніми версіями, тому підходить для більшості сучасних сайтів і застосунків.

КОНТРАСТ

Одним із найважливіших принципів доступності є **достатній контраст між текстом і фоном**. Саме він визначає, чи зможе людина легко прочитати текст, побачити іконку або зрозуміти, який елемент є клікабельним.

Коефіцієнт контрастності (contrast ratio) – це результат порівняння яскравості двох кольорів від 1:1 (повна відсутність контрасту) до 21:1 (чорний на білому). Наприклад, білий текст на темно-сірому фоні має нижчий контраст, ніж чорний на білому. Перевірити цей показник можна у будь-якому **contrast checker** онлайн (Рис. 457).

Foreground Color: #f20d0d

Background Color: #cc0df2

Contrast Ratio: **1.01:1**

WCAG Compliance Results

ELEMENT TYPE	AA	AAA
Small Text	✗ Fail	✗ Fail
Large Text	✗ Fail	✗ Fail
UI Components	✗ Fail	✗ Fail

WCAG AA and AAA Results

SMALL sample text: 14pt (18.5px)

LARGE sample text: 18pt (24px)

Icons:

Рис. 457. Приклад поєднання двох кольорів із найменшим коефіцієнтом контрастності

ТЕКСТ

Звичайний текст має розмір 24px і нижче. Якщо він жирний, то до 19px. Для нього коефіцієнт контрастності має бути не нижче **4.5:1**.

Великий текст має розмір від **24 px** звичайний або від **19–20 px** **напівжирний**. Для нього достатньо коефіцієнту контрастності **3:1**.

Ці правила діють для будь-яких шрифтів і тем, незалежно від кольорової гами.

Якщо текст лежить на суцільному тлі для перевірки обирають колір тексту і колір фону та переконуються, що показник не нижчий за норму.

Якщо під текстом є зображення або градієнт, контраст усе одно має відповідати нормі. Щоб зберегти читабельність на таких фонах, корисно підкласти напівпрозору підкладку, додати легку тінь-«ореол» або підібрати відтінок тексту так, щоб він залишався достатньо темним або світлим на всіх ділянках (Рис. 458).

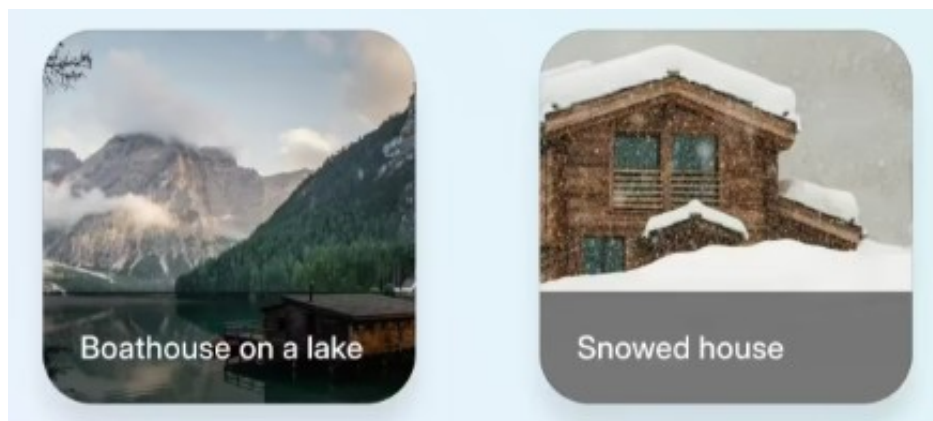


Рис. 458. Рисунок під текстом

ПОСИЛАННЯ

Колір не повинен бути єдиним способом показати, що це – посилання. Найпростіше рішення – **постійне підкреслення**. Якщо ви спираєтесь лише на колір, він має помітно контрастувати і з фоном, і з сусіднім текстом. При наведенні (hover) або фокусі (focus) обов’язково додається **додатковий сигнал** – наприклад, зміна кольору, підкреслення чи легка анімація (Рис. 459).

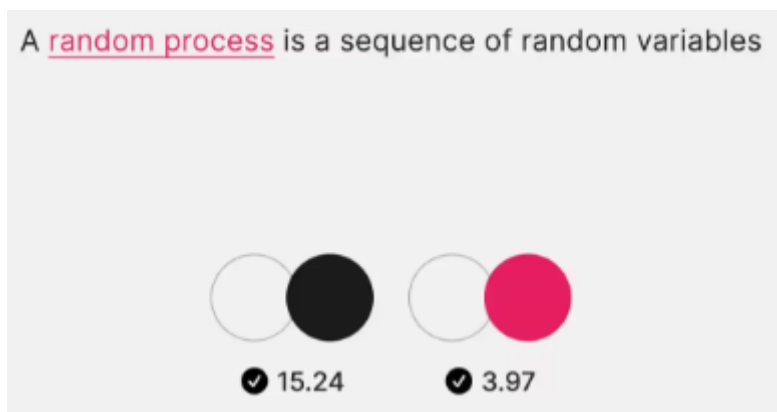


Рис. 459. Посилання в тексті

НЕТЕКСТОВИЙ КОНТРАСТ І КОМПОНЕНТИ

Ознаки, за якими користувач розпізнає інтерактивність (межі полів, іконки станів, тумблери), повинні бути помітні на тлі. Орієнтиром для таких елементів є **мінімальний коефіцієнт контрастності 3:1** із сусідніми кольорами. Для кнопок обов'язково потрібно перевіряти контраст підпису до фону кнопки; якщо ж саме рамка вказує на те, що елемент клікабельний, вона має бути достатньо контрастною (Рис. 460).

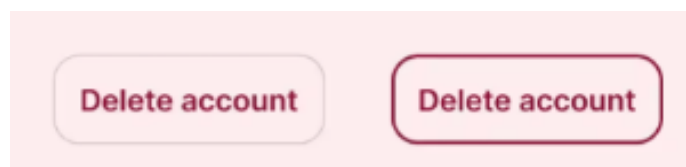


Рис. 460. Обидві кнопки на зображенні відповідають вимогам.

Однак кнопка праворуч більш візуально доступна для людей із вадами зору

Видимі межі важливі і для полів введення, чекбоксів, радіокнопок, і тоглів. Вони також повинні мати **мінімальний коефіцієнт контрастності 3:1**.

При роботі із складними компонентами, такими як тогли (Рис. 461), що складаються з декількох частин, перевіряти потрібно всі суміжні пари: іконка до ручки, ручка до доріжки, доріжка до фону сторінки тощо. Зручно завести токени і фіксувати цільові пари для кожного стану: звичайний, наведений, натиснутий, вимкнений.



Рис. 461. Приклади тоглів

ФОКУС-СТАНИ

Компоненти мають зберігати достатній контраст у всіх візуальних станах: звичайному, наведенні, натисканні та фокусі. Фокус має бути **видимим** і **не перекриватися** іншими шарами інтерфейсу. Кільце фокусу має мати достатню товщину та відчутний контраст до оточення; воно має залишатися у полі зору під час прокручування чи появи панелей (Рис. 461).

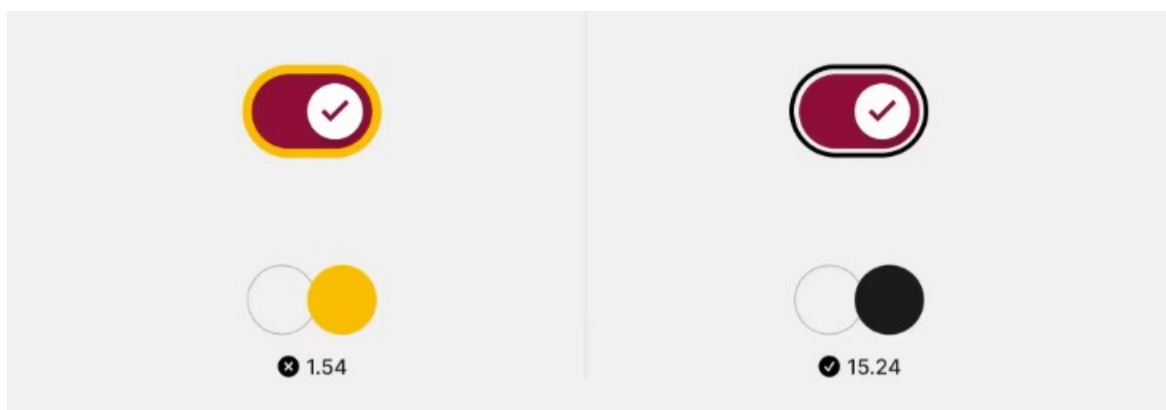


Рис. 462. Стан фокуса компонента. Зліва жовта обводка не має достатнього контрасту з фоном, а справа розташована на відстані від компонента й дуже добре контрастує зі сторінкою

Неактивні стани, що недоступні для взаємодії, не підпадають під ті самі вимоги контрасту, що активні елементи.

ВІЗУАЛЬНІ ЕЛЕМЕНТИ

Декоративні зображення (іконки, інфографіка, ілюстрації тощо) також не зобов'язані відповідати вимогам, якщо вони не несуть інформації і не використовуються для взаємодії. Водночас будь-який текст, який користувач повинен прочитати, має бути читабельним.

Особливу увагу варто приділити **діаграмам** – їхні кольори мають не лише контрастувати з фоном, а й бути достатньо різними між собою, щоб усі сегменти можна було розрізнити.

КЛАВІАТУРНА НАВІГАЦІЯ

Весь функціонал сайту повинен бути доступним **із клавіатури**. Користувач повинен мати змогу перейти, натиснути, відкрити або закрити будь-який елемент без миші. Порядок переходів (tab-послідовність) має бути логічним і відповідати природному порядку читання.

ARIA В МАКЕТАХ

ARIA (Accessible Rich Internet Applications) – це спеціальні атрибути, які допомагають читачам екрана розуміти структуру сторінки. Вони визначають ролі елементів, стани та властивості. Однак найкраща практика – спершу використовувати **нативні теги** (header/banner, nav, main, aside, footer) і лише потім додавати ARIA, коли це дійсно необхідно.

REDUCE MOTION

Деяким людям рух заважає: відволікає або навіть викликає дискомфорт. Браузери вміють передавати налаштування системи **«менше руху»** – сайт має це поважати і мати спокійний варіант для налаштування «менше руху»: без паралаксів і інших трансформацій.

ІНСТРУМЕНТИ І ПРОЦЕС

Найбільш надійний шлях – перевіряти контраст безпосередньо в макеті та фіксувати успішні пари у дизайн-системі як семантичні токени: «текст ↔ фон», «іконка ↔ фон», «фокус ↔ контекст». Так команда менше залежить від ручних розрахунків і не ризикує зламати доступність у дрібних правках.

Навіть ідеальний контраст не вирішує всіх завдань. Інтерфейс не повинен покладатися виключно на колір, щоб передати стан чи помилку. Там, де важлива інформація, потрібно додавати текст, іконку або інший візуальний маркер.

РОЗДІЛ 5

АДАПТИВНИЙ ДИЗАЙН

5.1. ОСНОВИ АДАПТИВНОГО ДИЗАЙНУ

Нині люди користуються інтернетом із різноманітних пристроїв – від великих настільних моніторів і ноутбуків до планшетів і смартфонів із зовсім маленькими екранами. Саме тому дизайн вебсайту має бути зручним і зрозумілим у будь-якому форматі. Цю задачу вирішує адаптивний (responsive) дизайн.

Адаптивний (responsive) дизайн – це підхід, за якого елементи інтерфейсу автоматично змінюють свої розміри, положення та пропорції залежно від ширини екрана. Тобто сайт «підлаштовується» під простір, який йому доступний.

Коли користувач відкриває сторінку на телефоні, контент не просто зменшується – він перебудовується: текст стає більшим і зручнішим для читання, меню перетворюється на «бургер», а зображення змінюють розташування, щоб не порушити логіку композиції. Все це робиться без створення окремої мобільної версії – достатньо правильно налаштованої сітки та медіазапитів у CSS.

Основна ідея responsive-підходу – **єдиний сайт для всіх пристроїв**, який виглядає і працює гармонійно незалежно від того, звідки на нього зайшли.

Адаптований (adaptive) дизайн – це схожий, але трохи інший підхід. У цьому разі створюють **кілька заздалегідь підготовлених варіантів** макету для різних розмірів екранів: наприклад, один для комп'ютера, один для планшета і один для смартфона. Коли користувач відкриває сайт, сервер або браузер «підтягує» ту версію, яка найкраще відповідає його пристрою.

Це дає змогу точніше контролювати зовнішній вигляд і поведінку елементів – кожна версія може мати свої особливості. Проте створення кількох макетів потребує більше часу і зусиль, а зміни в одному місці доведеться дублювати в інших версіях.

Ключова різниця між responsive та adaptive підходами – у **способі реагування на екран**:

- **Responsive дизайн** «тече», плавно перебудовуючись у межах будь-якої ширини екрана.
- **Adaptive дизайн** працює «ступінчасто» – підключає готові версії для певних розмірів.

Responsive – більш гнучкий і універсальний, тоді як adaptive – більш керований і точний. На практиці ці підходи часто комбінують: основна структура сайту створюється як responsive, а для критичних брейкпоінтів (наприклад, мобільного виду) – додаються окремі adaptive-варіанти компонентів.

Адаптивність сайту надзвичайно важлива. Користувачі очікують, що сайт буде працювати однаково добре скрізь – на роботі за монітором, у транспорті на телефоні чи вдома з планшета. Якщо інтерфейс «ламається» або текст занадто дрібний, людина просто закриє сторінку.

Окрім того, пошукові системи, зокрема Google, віддають перевагу сайтам із якісним адаптивним дизайном – це безпосередньо впливає на рейтинг у видачі.

У підсумку мета адаптивного дизайну проста: створити інтерфейс, який є зрозумілим, читабельним і приємним незалежно від того, з якого пристрою його відкривають.

Розглянемо набір практичних правил, що допомагають побудувати логічну, живу та гнучку структуру сторінки.

Основні принципи адаптивності:

1. **Гнучкі сітки (flexible grids).** Замість фіксованих пікселів використовують відсотки, одиниці **em**, **rem** або **vw/vh**. Це дає змогу блокам плавно змінювати ширину.
2. **Гнучкі зображення.** Зображення та відео мають масштабуватися разом із контейнерами. Для цього у CSS застосовують правило **max-width: 100%; height: auto;**, яке не дає медіа «вилізти» за межі блоку.

3. **Media-запити (media queries).** Це серце адаптивного підходу: дизайнер визначає ключові «точки зламу» (**breakpoints**), де макет має змінювати вигляд. Наприклад, при ширині до 768 пікселів навігаційне меню стає прихованим у «бургер», а колонки перетворюються на одну вертикальну стрічку.
4. **Адаптивний контент.** На різних екранах показують не лише різну компоновку, а й різний зміст. Наприклад, на мобільному може не бути другорядних блоків чи великих фонів – усе, що заважає швидко виконати дію, прибирається або спрощується.
5. **Тестування на реальних пристроях.** Жоден емулятор не передасть повністю поведінку інтерфейсу на телефоні. Тому завжди варто перевіряти сайт «у руках»: як працює скрол, натискання, видимість елементів і розміри шрифтів.

5.2. РОБОТА З АДАПТИВНІСТЮ У FIGMA

Figma – це не просто інструмент для створення макетів, а середовище, де можна побачити, як дизайн поводить себе на різних екранах. Завдяки системі **Auto Layout**, **Constraints** і **Resizing** Figma дає змогу моделювати адаптивну поведінку інтерфейсу ще до етапу верстки. Це допомагає дизайнеру не лише намалювати гарну сторінку, а й передбачити, як вона зміниться на планшеті чи смартфоні.

AUTO LAYOUT – ГНУЧКА СТРУКТУРА ЕЛЕМЕНТІВ

Auto Layout у Figma – це не просто технічна функція, а справжня логіка побудови гнучких інтерфейсів. Він допомагає дизайнеру не «розкладати» елементи вручну, а задавати їм правила поведінки: як вони розташовуються, реагують на зміни розміру й утримують потрібні відступи.

Тобто Auto Layout працює як невидимий каркас, який тримає структуру сторінки. Коли ви додаєте або видаляєте елемент, інші автоматично підлаштовуються – зберігають рівні проміжки, вирівнювання та порядок. Наприклад, якщо у кнопці змінюється довжина тексту, кнопка розтягується

настільки, наскільки потрібно, і не ламає дизайн. Якщо ж у картці додається ще один рядок тексту або іконка, всі сусідні елементи акуратно зміщуються вниз, зберігаючи пропорції.

Auto Layout дає змогу вказати напрямок розташування (по горизонталі або вертикалі), інтервали між елементами, зовнішні відступи, а також вирівнювання – по центру, правому або лівому краю. Це дає змогу створювати не просто красиві композиції, а справді гнучкі інтерфейси, які «живуть» разом із контентом.

Наприклад, шапку сайту – логотип, меню та кнопку – можна об'єднати в один Auto Layout-фрейм (Рис. 463.). Тоді, коли ви змінюєте ширину екрана, всі елементи є на своїх місцях: логотип – ліворуч, меню – по центру, а кнопка – праворуч. Відступи між ними зберігаються автоматично, і структура виглядає стабільно, як у реальному браузері.

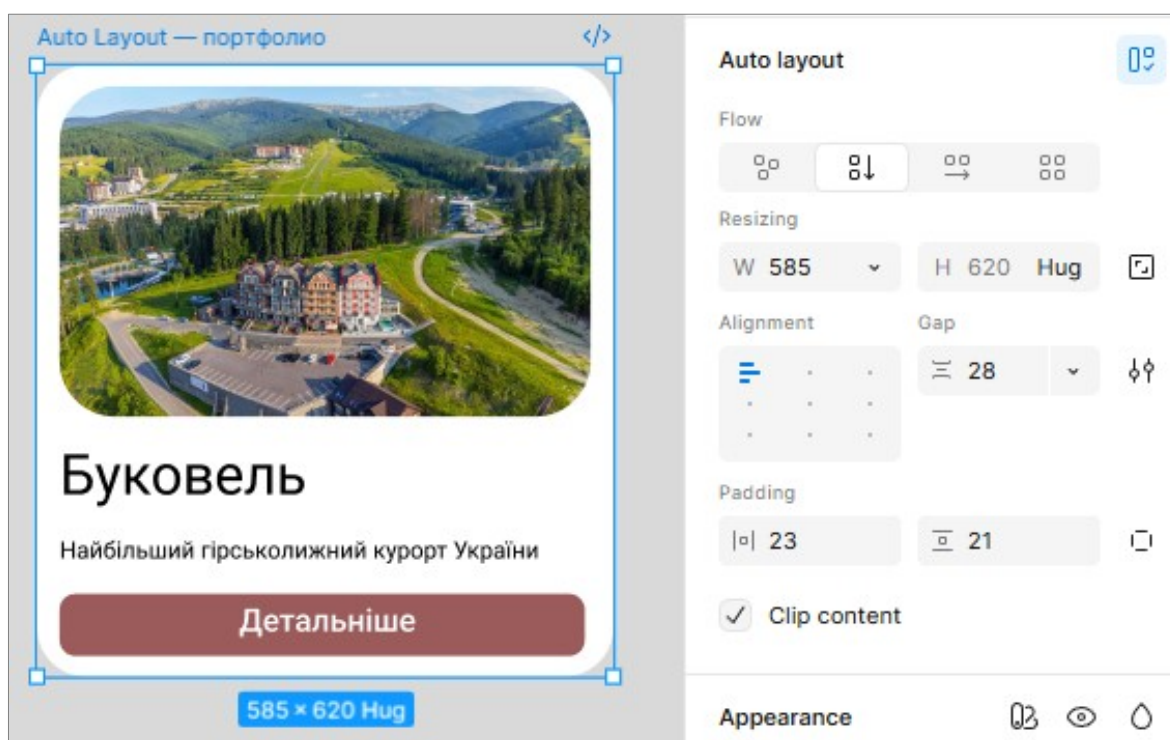


Рис. 463. Auto Layout-фрейм із властивостями

Отже, Auto Layout у Figma – це інструмент, який перетворює статичний макет на адаптивну систему, де все підпорядковується чітким правилам. Він економить час, зменшує кількість помилок і допомагає мислити як справжній розробник – у логіці взаємозв'язків, а не фіксованих розмірів.

Для створення Auto Layout потрібно декілька об'єктів – щонайменше два. Наприклад, це можуть бути:

- фрейм + фрейм;
- фрейм + текст;
- об'єкт + об'єкт.

Розглянемо використання двох фреймів. Створюємо їх, виділяємо та використовуємо кнопку **Toggle auto layout** або поєднання гарячих клавіш **Shift+A** (Рис. 464).

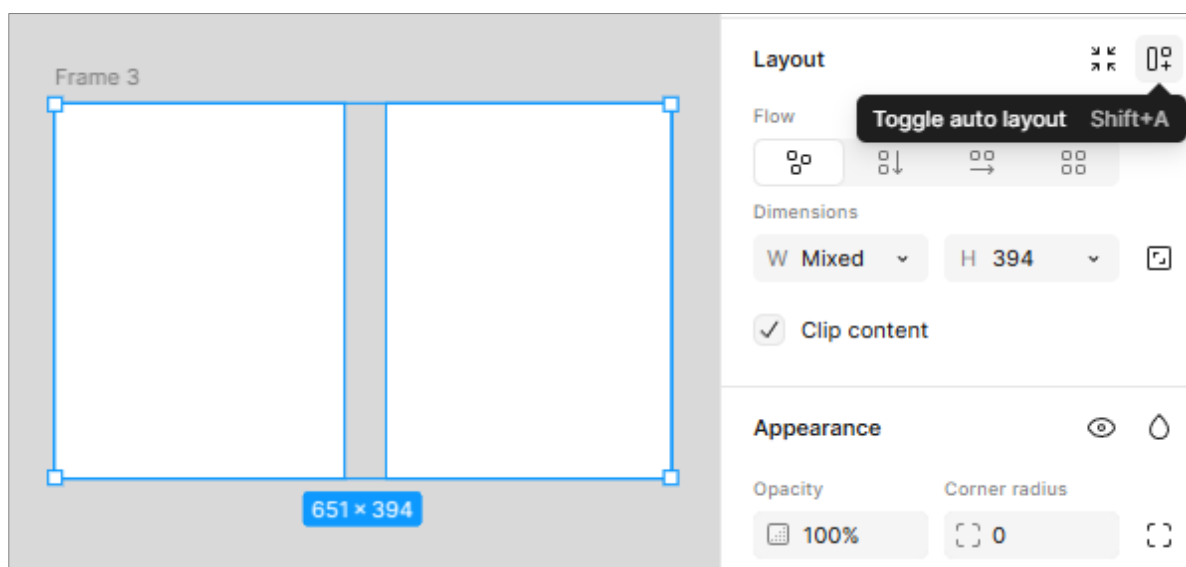


Рис. 464. Кнопка, розташована у верхньому правому кутку *Toggle auto layout*

Тепер можна перевірити, що Auto Layout додано. На панелі шарів з'явиться відповідна іконка, а кадри будуть об'єднані (Рис. 465).

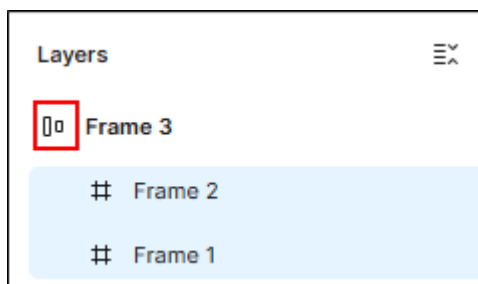


Рис. 465. Іконка *Auto Layout* на панелі шарів зліва

Якщо є кадр, усередині якого вже є об'єкт, можна застосувати Auto Layout до нього, а далі вже можна не турбуватися про ширину кадру, а налаштувати автоматичні відступи.

Тепер розберемо основні налаштування. Вони з'являються на правій панелі – при натисканні на елемент, до якого застосовано Auto Layout.

FLOW – ПОВЕДІНКА ЕЛЕМЕНТІВ У МЕЖАХ ФРЕЙМУ

Flow – це можливість задати, як поводитимуться об'єкти, якщо вони не поміщаються у визначений розмір фрейму. Тобто ця функція керує тим, **як елементи реагують на обмеження простору**: залишаються вільними, розташовуються вертикально, горизонтально чи за сіткою.

FREEFORM (ВІЛЬНЕ РОЗТАШУВАННЯ)

У цьому режимі елементи можна розміщувати всередині фрейму як завгодно – без будь-яких правил Auto Layout. Просто тягнете об'єкти мишею і ставите їх у потрібне місце. Це зручно, коли потрібно створити вільну композицію або колаж, але не підходить для адаптивних макетів, де важливо зберігати відступи й логіку вирівнювання (Рис. 466).

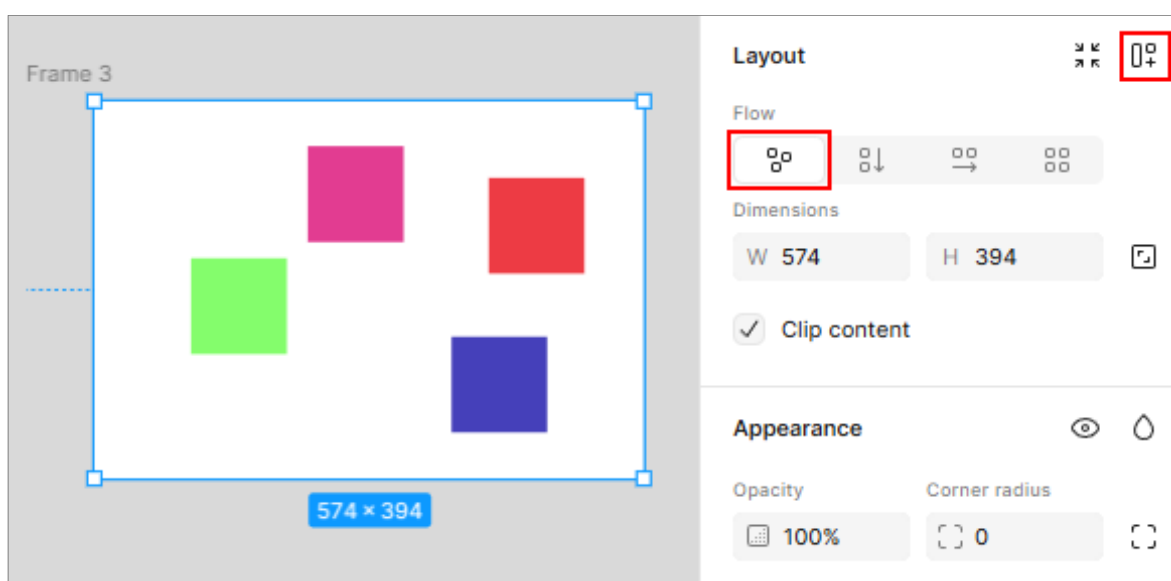


Рис. 466. Об'єкти можна розташовувати вільно усередині кадру

VERTICAL (ВЕРТИКАЛЬНЕ РОЗТАШУВАННЯ)

Усі елементи вибудовуються **один під одним** – від верху до низу. Це найпоширеніший варіант для списків, карток, форм або будь-якого контенту, який природно розташовується в стовпчик. Відстань між елементами регулюється параметром **Gap** (Рис. 467).

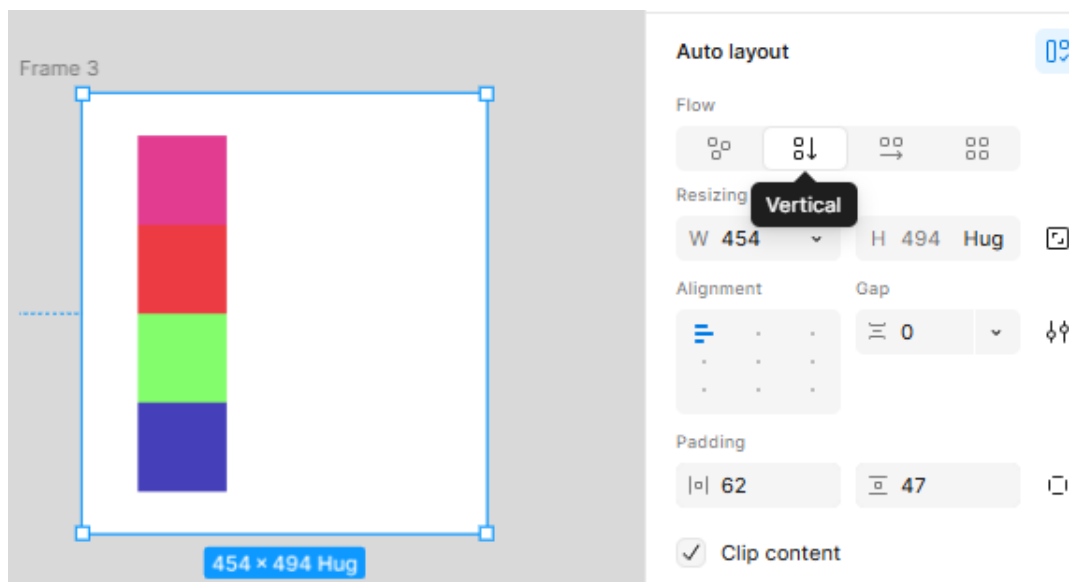


Рис. 467. Об'єкти автоматично вишиковуються в стовпчик

HORIZONTAL (ГОРИЗОНТАЛЬНЕ РОЗТАШУВАННЯ)

Елементи вишиковуються **зліва направо**. У цьому режимі з'являється додаткова опція – **Wrap (перенесення)**. Вона визначає, що робити, коли в рядку не вистачає місця:

- якщо перенесення вимкнене – елементи виходять за межі фрейму (Рис. 468);
- якщо увімкнене – вони автоматично переносяться на наступний рядок (Рис. 469).

Це корисно для тегів, карток або блоків із великою кількістю повторюваних елементів.

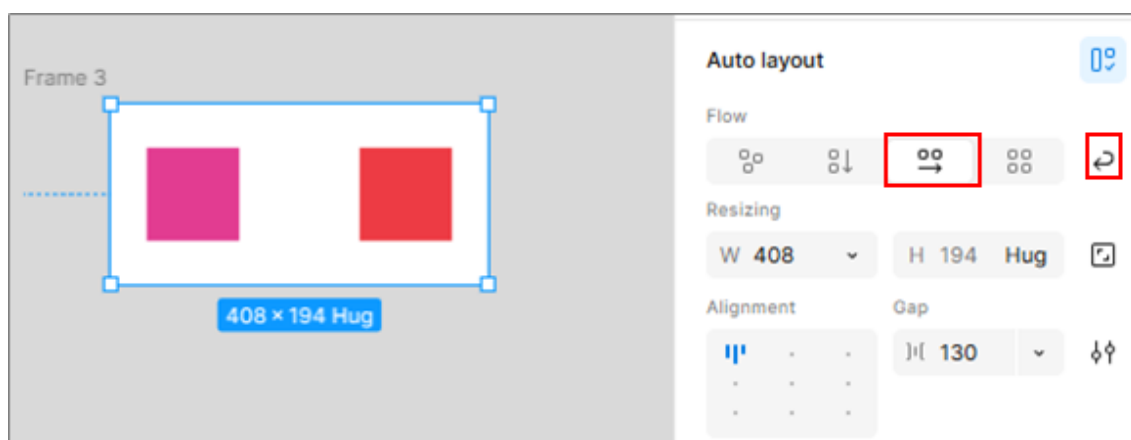


Рис. 468. Горизонтальне розташування з вимкненим перенесенням

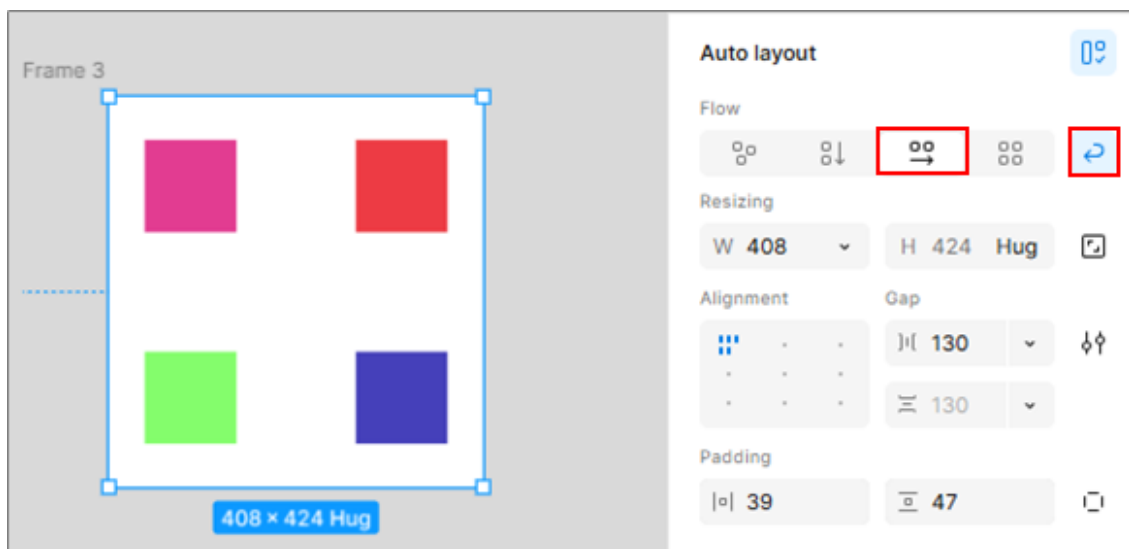


Рис. 469. Горизонтальне розташування з увімкненим перенесенням

GRID (РОЗТАШУВАННЯ ЗА СІТКОЮ)

Режим, який дає змогу елементам автоматично підлаштовуватись під **сітку**. У властивостях з'являється додаткове меню **Grid**, де можна налаштувати кількість колонок, відстані між ними й поведінку елементів при зміні розміру контейнера. Завдяки цьому створювати сітки карток або галереї зображень стає набагато швидше – об'єкти самі займають потрібне місце (Рис. 470).

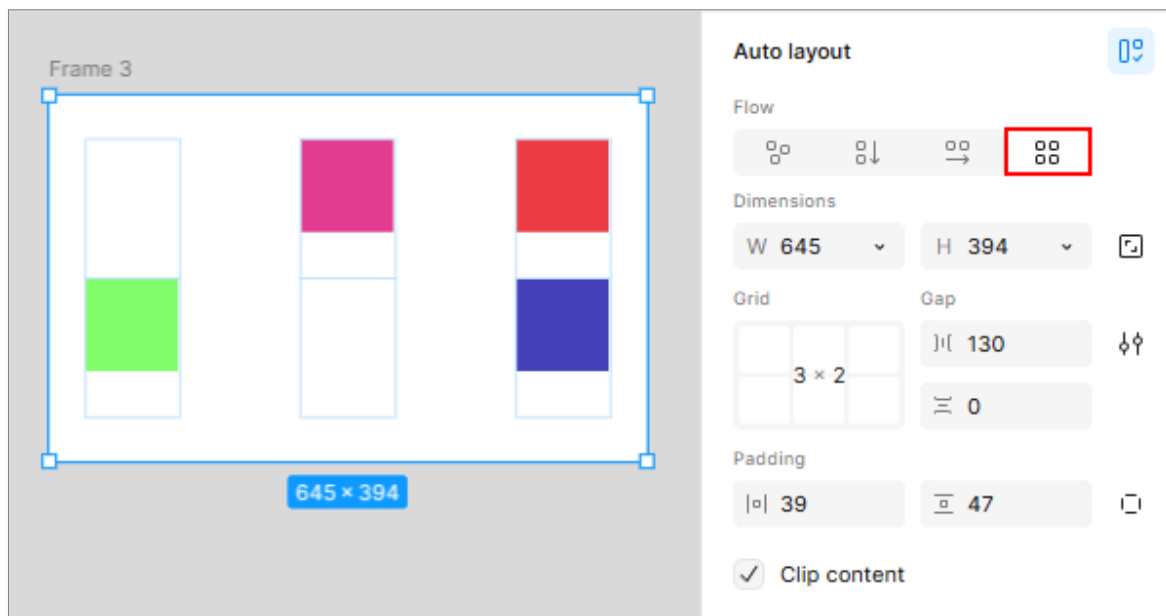


Рис. 470. При зміні сітки елементи підлаштовуються до неї

RESIZING – КЕРУВАННЯ РОЗМІРАМИ

У Figma будь-який елемент має параметри **W (ширина)** і **H (висота)**. Для Auto Layout доступно два основні варіанти (Рис. 471):

- **Fixed (фіксований розмір)** – ширина та висота не змінюються, незалежно від контенту (наприклад, логотип або іконка, що завжди має залишатися одного розміру);
- **Hug contents (підлаштування під вміст)** – розміри автоматично змінюються відповідно до вмісту та падингів (наприклад, бейдж або кнопка, де довжина залежить від довжини тексту).

Якщо, наприклад, збільшити відстань між елементами всередині фрейму з режимом **Hug**, його загальна висота теж автоматично зросте. Це забезпечує гнучкість і дає змогу швидко змінювати контент без ручного підганяння.

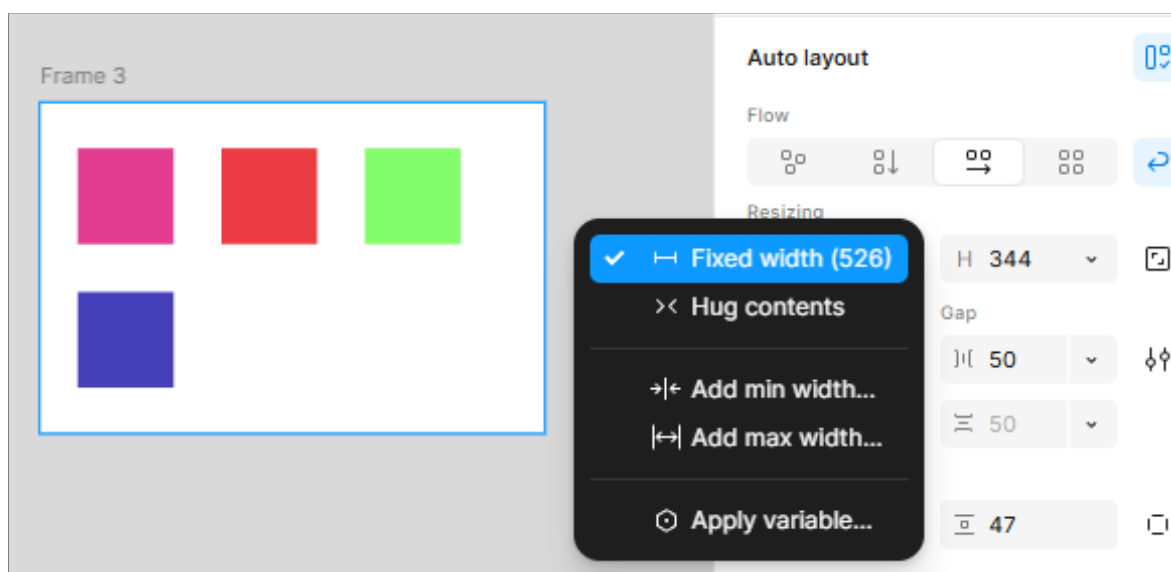


Рис. 471. Підстроювання під контент (hug): при зміні відстані між квадратами кадр збільшується або зменшується

GAP – ВІДСТАНІ МІЖ ЕЛЕМЕНТАМИ

Gap задає проміжки між об'єктами всередині Auto Layout. Його можна налаштувати **по горизонталі й по вертикалі** окремо. Наприклад, якщо встановити 100 пікселів, то всі елементи в Auto Layout будуть мати саме такий рівномірний відступ один від одного (Рис. 472).

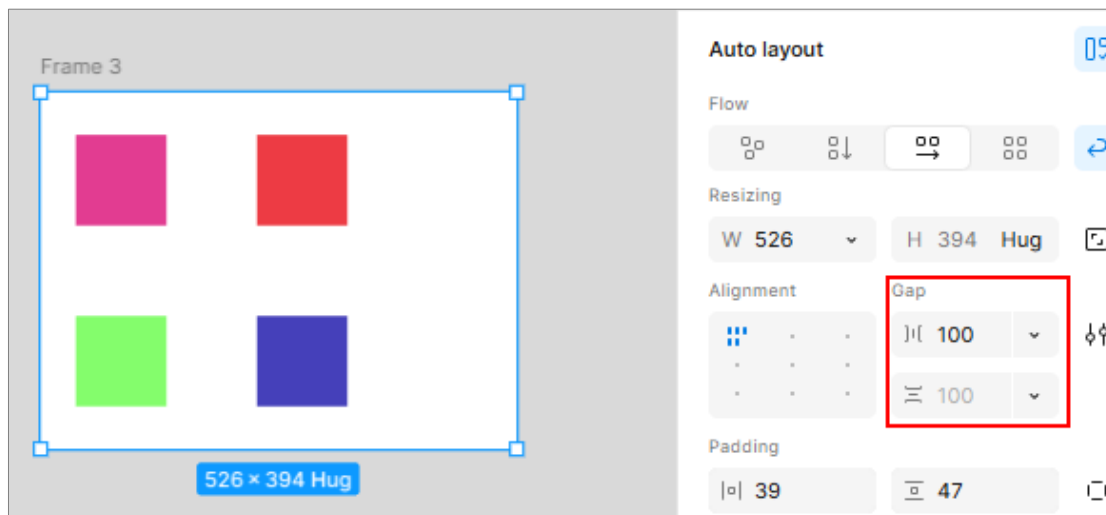


Рис. 472. Вертикальна та горизонтальна відстань між квадратами всередині автолейауту – 100 пікселів

ALIGNMENT – ВИРІВНЮВАННЯ

Alignment керує тим, як об'єкти розташовуються всередині контейнера: по лівому або правому краю, по центру, зверху або знизу. Ця функція працює лише тоді, коли активне горизонтальне або вертикальне розташування (**Flow**).

Якщо контейнер налаштовано на **Hug contents**, можна вказати, як розміщати елементи з різною висотою – наприклад, вирівняти по центру, притиснути до верху чи донизу (Рис. 473).

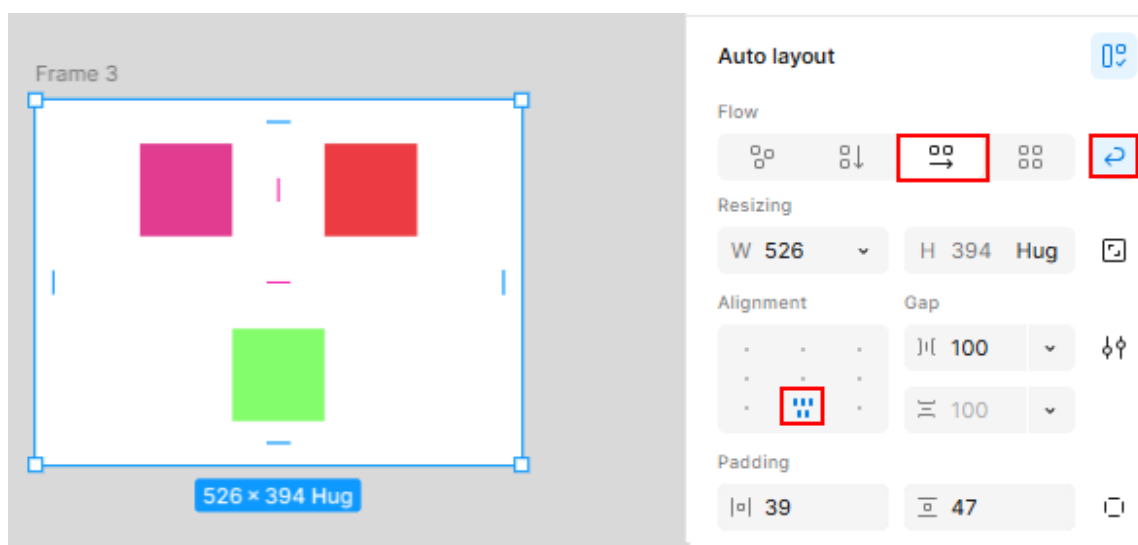


Рис. 473. Якщо вибрано перенесення (*hug*), *alignment* дозволить вибрати, як розташовувати неоднакову кількість елементів

PADDING – ВНУТРІШНІ ВІДСТУПИ

Padding визначає відстань від країв фрейму до вмісту. Її можна налаштовувати одразу для всіх сторін або окремо – зверху, знизу, ліворуч і праворуч.

Завдяки цьому, додаючи нові елементи, ви не витрачаєте час на ручне виставлення відступів – вони одразу з’являються в межах заданих правил (Рис. 474).

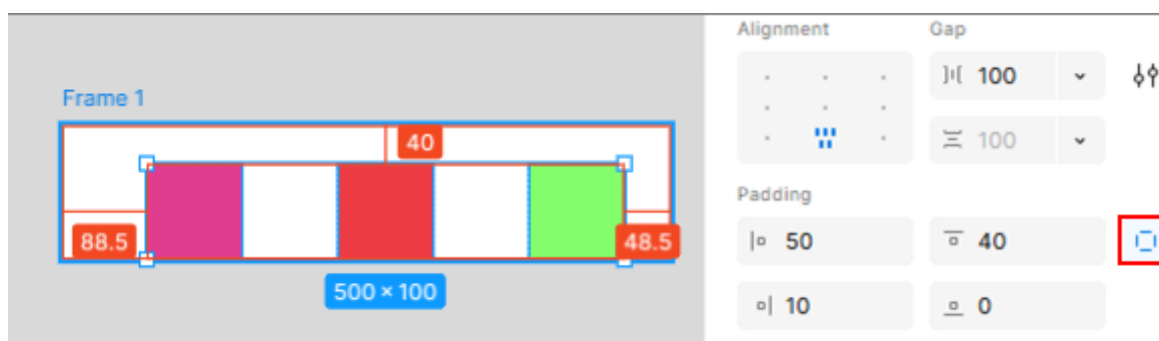


Рис. 474. Налаштування відступів від меж фрейму

ДОДАТКОВІ НАЛАШТУВАННЯ AUTO LAYOUT

Для Auto Layout передбачено декілька додаткових функцій (Рис. 475).

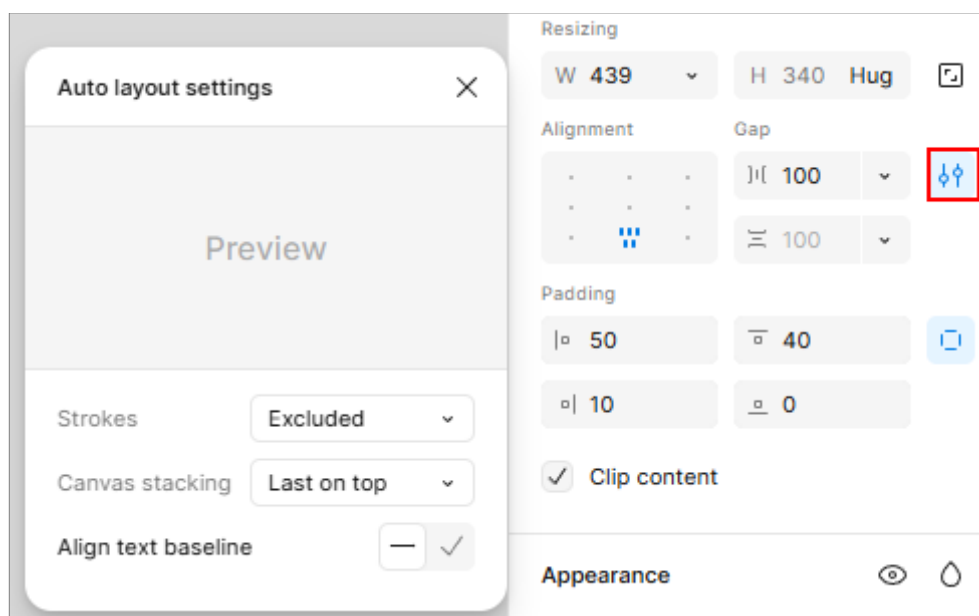


Рис. 475. Додаткові налаштування Auto Layout

- **Strokes** – керує обводкою контейнера. Можна обрати, щоб вона була всередині або ззовні фрейму.

- **Canvas stacking** – визначає порядок накладання елементів, якщо відступи між ними мають від’ємне значення (наприклад, коли аватари «наїжджають» один на одного). Можна вказати, який об’єкт буде зверху – перший чи останній.
- **Align text baseline** – вирівнює текстові блоки по базовій лінії шрифту. Це особливо помітно, коли поруч є кілька елементів із різними розмірами тексту.

IGNORE AUTO LAYOUT – ФІКСОВАНІ ЕЛЕМЕНТИ

Буває, що потрібно розмістити елемент поверх інших – наприклад, бейдж зі знижкою або іконку «додати в обране». У таких випадках допомагає функція **Ignore Auto Layout**, яка вмикає **абсолютне позиціонування** (Рис. 476). Об’єкт не реагує на загальні правила фрейму та є в заданій точці. Це зручно для декоративних елементів, наліпок, бейджів або фонів.

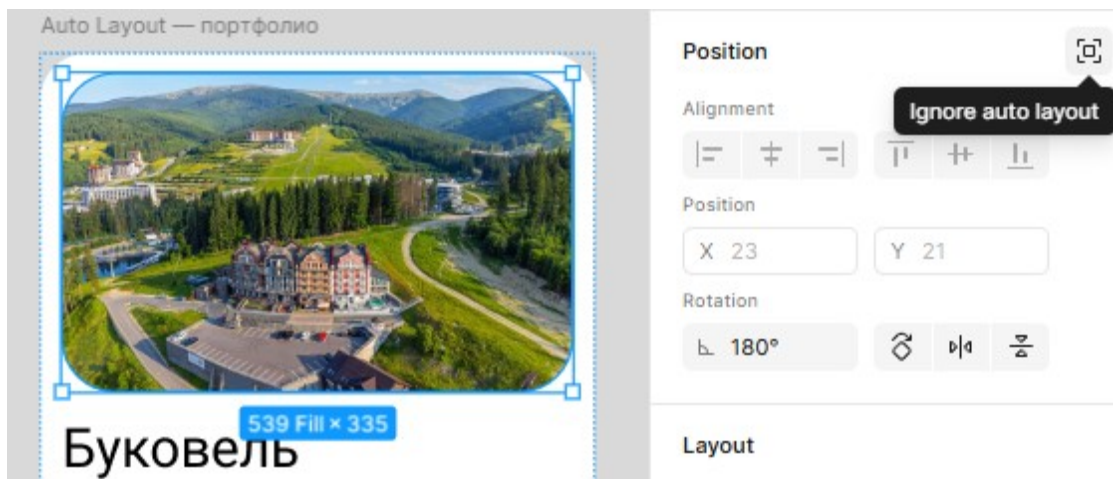


Рис. 476. Абсолютне позиціонування дозволило додати плашку
безпосередньо на фотографію

Для того, щоб вимкнути **Auto Layout** необхідно клацнути на ту саму кнопку, що використовується для його активації, або натиснути **Shift + Alt + A**.

Щоб повністю перетворити елементи на звичайні об’єкти, використовують комбінацію клавіш **Ctrl + Shift + G**.

CONSTRAINTS – ЗАКРІПЛЕННЯ ПОЗИЦІЇ ЕЛЕМЕНТІВ

Другий важливий інструмент – **Constraints** (обмеження).

Обмеження – це індивідуальні налаштування кожного шару в фреймі, які допомагають контролювати, як цей шар поводитиметься при зміні розмірів фрейму.

Вони визначають, до чого «прив'язаний» елемент: до лівого краю, правого, центру чи одразу до двох сторін. Завдяки цьому при зміні розміру фрейму кнопки, зображення або текстові блоки залишаються в логічних місцях – так, ніби це вже справжня адаптивна сторінка.

Наприклад, якщо закріпити логотип зліва і кнопку праворуч, то при збільшенні ширини фрейму вони будуть на своїх місцях, а простір між ними розтягуватиметься. Без обмежень усі елементи «поїхали» б при зміні розміру, і дизайнеру довелося б малювати кілька варіантів однієї шапки – для кожного типу екрана окремо.

ОБМЕЖЕННЯ ДЛЯ ФІГУР ЗА ЗАМОВЧУВАННЯМ

Коли ми створюємо шар (shape) усередині фрейму або переносимо вже наявний об'єкт у фрейм, до нього автоматично застосовуються два обмеження – **Left** і **Top**.

Це означає, що при зміні розміру фрейму об'єкт буде «прив'язаний» до верхнього лівого кута і не буде рухатись або змінювати розміри при зміні розміру фрейму. Переглянути обмеження можна в панелі **Constraints** (Рис. 477).

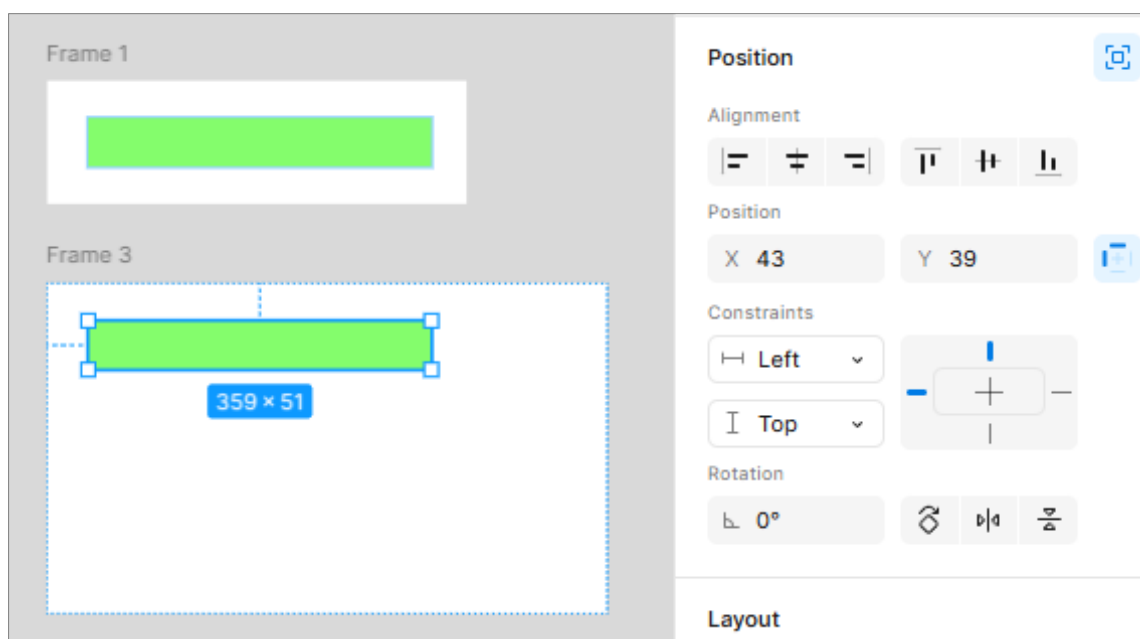


Рис. 477. Обмеження для фігур за замовчуванням

СТВОРЕННЯ «ГУМОВОЇ» ШАПКИ

Тепер уявімо, що потрібно створити шапку, яка при зміні ширини фрейму буде **розтягуватись по горизонталі**, але збереже висоту. Для цього потрібно виділити прямокутник і у блоці **Constraints** додати обмеження **Right**. Тепер активні три напрямки: **Left, Top і Right**. Якщо збільшити ширину фрейму, шапка автоматично розтягнеться, зберігаючи відступи від країв, а її висота буде незмінною (Рис. 478).

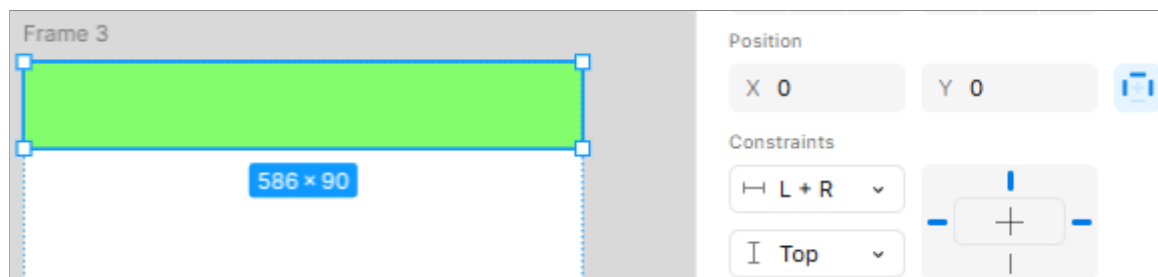


Рис. 478. Встановлення обмежень для створення «гумової шапки»

СТВОРЕННЯ САЙДБАРУ ІЗ «ГУМОВОЮ» ВИСОТОЮ

Ще один класичний приклад – **бічна панель (sidebar)**, яка має автоматично тягнутися вниз, зберігаючи ширину та прив'язку до лівого краю. Для її створення необхідно, щоб були активні напрямки: **Left, Top і Bottom**. Це зручно для вебдизайну, де висота сторінки часто залежить від кількості контенту.



Рис. 479. Встановлення обмежень для створення сайдбару

СТВОРЕННЯ ФОНОВОГО ФОТО

При розробці сторінок сайту дизайнери для фону часто використовують фотографії. Спершу на думку спадає ідея залити растровим фоном сам фрейм,

але на практиці це виявляється незручно, тому що часто виникає потреба з клавіатури копіювати стиль заливки фону та переносити його в інші фрейми. Значно легше перетягнути з робочого столу фотографію у вікно Figma і вписати її у фрейм, а потім зняти на ній всі обмеження, затиснувши клавішу **Shift**. Це означає, що фото буде масштабуватися пропорційно при зміні фрейму, а горизонтальне і вертикальне обмеження будуть стояти у позиції **Scale**.

ФІКСАЦІЯ МОДАЛЬНОГО ВІКНА ПО ЦЕНТРУ

Щоб модальне вікно завжди було по центру незалежно від розміру фрейму необхідно вирівняти його по центру та в блоці **Constraints** встановити обмеження **Center, Center**. З'явиться пунктирний хрест – він показує, що шар зафіксовано в центрі. Якщо фрейм збільшити, вікно залишиться точно по центру.

Для того, щоб при розтягуванні макета все не «поїхало», необхідно затиснути **Ctrl**, програма тимчасово ігноруватиме обмеження. Це дуже зручно, коли у макеті багато шарів, і налаштовувати кожен окремо займає багато часу.

5.3. ТИПОГРАФІКА ТА ШРИФТИ В АДАПТИВНОМУ ДИЗАЙНІ

ОСНОВНІ ПРИНЦИПИ АДАПТИВНОЇ ТИПОГРАФІКИ

Типографіка – це не просто вибір гарного шрифту. Вона формує візуальний ритм інтерфейсу, впливає на сприйняття інформації та створює емоційний настрій користувача. Саме через текст користувач найчастіше взаємодіє з цифровим продуктом: читає кнопки, повідомлення, заголовки чи інструкції. Тому грамотна робота з типографікою – це основа зрозумілого, приємного та професійного дизайну.

У контексті адаптивного дизайну роль типографіки зростає в рази. Якщо на великому екрані текст може «дихати» і мати багато простору, то на мобільному пристрої кожен піксель має значення. Тут важливо не лише зберегти читабельність, а й підтримати візуальну гармонію. Користувач повинен комфортно сприймати

інформацію незалежно від того, чи читає він сайт на ноутбучі, планшеті або телефоні.

Адаптивна типографіка допомагає зробити дизайн гнучким і цілісним. Вона дозволяє текстовим елементам змінювати розміри, відступи й пропорції разом із контейнером або екраном, не втрачаючи логіки та структури. Це означає, що дизайнер повинен не лише підібрати гарний шрифт, а й продумати, як він поводитиметься за різних умов.

Завдяки продуманій типографіці інтерфейс виглядає не «складеним із блоків», а єдиним, живим і зручним середовищем для читання. Добре налаштована система шрифтів забезпечує користувачу комфорт і довіру, а продуктові – впізнаваність і професійність.

Розглянемо **основні принципи адаптивної типографіки**:

ГНУЧКІСТЬ (FLUID TYPOGRAPHY)

Гнучка типографіка – це підхід, за якого розмір тексту автоматично змінюється залежно від ширини екрана. Якщо раніше дизайнер задавав фіксовані значення – наприклад, «заголовок 32 px, текст 16 px», – то сьогодні це вже не працює: один і той самий текст має виглядати добре і на великому моніторі, і на екрані смартфона.

Суть гнучкої типографіки в тому, щоб текст «дихав» разом із макетом. Коли користувач зменшує вікно браузера, розміри шрифтів зменшуються плавно, без різких стрибків. Коли екран розтягується – текст збільшується пропорційно, зберігаючи гармонію. Це створює відчуття природності, ніби дизайн «живий» і реагує на простір.

Для цього замість пікселів використовують **відносні одиниці виміру** – такі як **em**, **rem**, **vw**, **vh**. Вони дають змогу системі самій обчислювати, яким має бути розмір тексту залежно від ширини або висоти екрана. Ще зручніше – використовувати сучасну CSS-функцію **clamp()**, яка задає мінімальне, ідеальне та максимальне значення. Наприклад:

```
font-size: clamp(1rem, 2vw + 0.5rem, 2rem);
```

Це означає, що текст ніколи не стане меншим за 1rem і не більшим за 2rem, але всередині цього діапазону буде плавно масштабуватися.

Так дизайнер зберігає контроль над виглядом тексту на всіх пристроях, а користувач отримує однаково комфортне читання – без «дрібного» тексту на мобільному чи надто великих заголовків на ноутбучі.

ІЄРАРХІЯ

Ієрархія у типографіці – це система, яка допомагає користувачеві швидко зорієнтуватися в тексті. Вона створює «візуальний порядок»: що головне, що другорядне, а що допоміжне. Якщо дизайн побудований грамотно, користувач ще до того, як прочитає текст, **інтуїтивно розуміє, де заголовок, де підзаголовок, а де основний зміст.**

У адаптивному дизайні це особливо важливо, адже при переході з великого екрана на мобільний усі елементи зменшуються, і простір для тексту стає обмеженим. Завдання дизайнера – **зберегти логіку сприйняття**, навіть коли розміри шрифтів змінюються. Наприклад, на десктопі заголовок може бути великим і просторовим, а на смартфоні – меншим, але все одно виразним і контрастним щодо основного тексту.

Щоб цього досягти, використовують **типографічну шкалу** – систему пропорцій між розмірами шрифтів. Замість довільних значень (наприклад, 14, 23, 31 піксель) застосовують узгоджене співвідношення, наприклад 1.25 або 1.333. Тоді всі розміри текстів – від підписів до великих заголовків – виглядають гармонійно і масштабуються природно на різних екранах (Рис. 480).

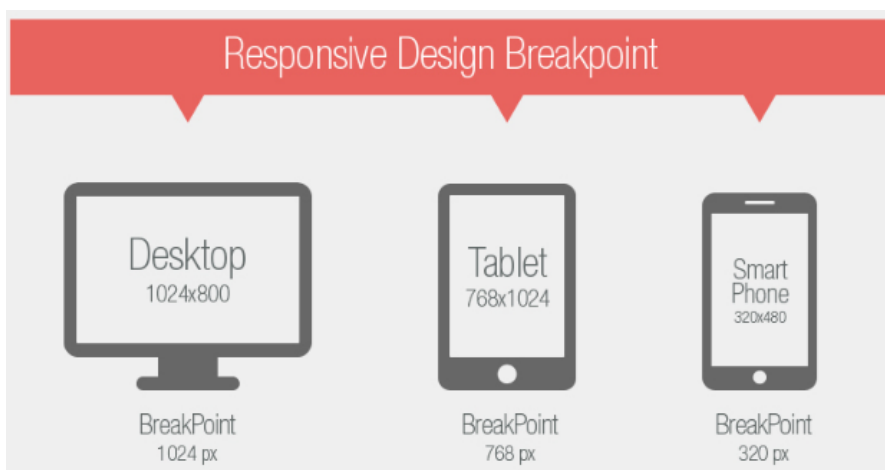


Рис. 480. Типографічна ієрархія допомагає зберегти логіку сприйняття навіть тоді, коли текст адаптується під різні розміри екранів

КОНТРАСТ І КОЛІР

Контраст і колір – це те, що робить текст не просто видимим, а **зручним для читання**. Навіть найкращий шрифт втратить сенс, якщо його складно розрізнити на фоні або очі швидко втомлюються.

Основна мета – **щоб текст було легко сприймати у будь-яких умовах**: на яскравому сонці, у темній кімнаті, на екрані ноутбука чи смартфона. Саме тому дизайнери працюють не лише з формою букв, а й з контрастом – співвідношенням між кольором тексту і фоном.

За стандартами **WCAG (Web Content Accessibility Guidelines)**, контраст між текстом і фоном повинен бути не менше **4.5:1** для основного тексту і **3:1** для великих заголовків. Якщо цей баланс порушено, очі швидко втомлюються, користувач мимоволі перестає читати і просто «пролистує» сторінку.

Інша важлива річ – **кольоровий контекст**. Кольори можуть сприйматися по-різному залежно від освітлення або темної / світлої теми інтерфейсу. Наприклад, світло-сірий текст на білому тлі майже непомітний, а яскраво-білий на чорному – створює ефект «миготіння». Тому хороший дизайнер завжди тестує свої кольори в різних умовах – на різних екранах, у режимах «day/night» і при різному рівні яскравості.

Ще один нюанс – **емоційне сприйняття кольору**. Холодні відтінки (сині, сірі) створюють відчуття технологічності й спокою, теплі (жовті, помаранчеві) – дружності та енергії. Головне – не переборщити: у тексті важлива не кількість кольорів, а чистота контрасту і зручність для очей (Рис. 481).

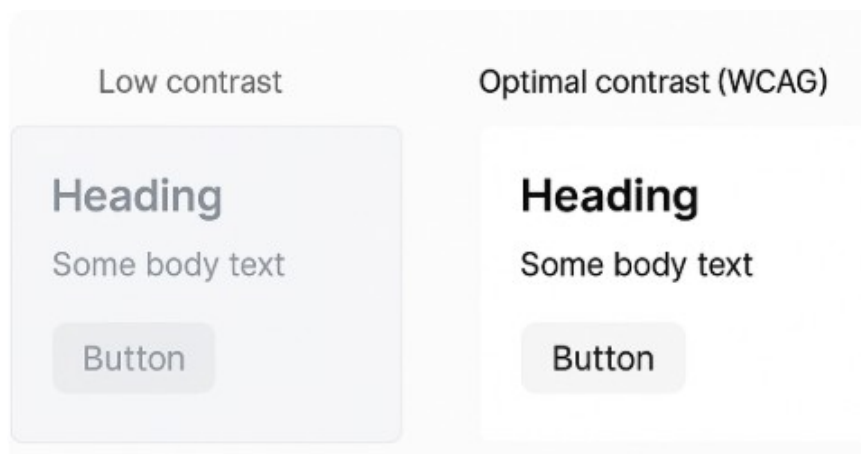


Рис. 481. Приклади написів різного контрасту

МІЖРЯДКОВИЙ ІНТЕРВАЛ (LINE-HEIGHT) І ТРЕКІНГ

Комфортне читання залежить не лише від розміру шрифту, а й від повітря між рядками та літерами. Занадто щільний текст перевантажує зір, а надто великий – розриває структуру.

Міжрядковий інтервал (line-height) – це відстань між рядками тексту. Якщо вона занадто мала, рядки «злипаються», і очам складно перескакувати з одного на інший. Якщо занадто велика – текст виглядає розірваним і втрачає ритм. Золоте правило – **баланс між щільністю та повітрям**. Для основного тексту зазвичай використовують коефіцієнт від **1.4 до 1.6**, а для заголовків – трохи менший, щоб вони виглядали компактно.

У **адаптивному дизайні** line-height має бути не статичним, а підлаштовуватися під розмір шрифту та ширину екрана. На мобільних пристроях, де текст стає вузьким, інтервал варто трохи збільшити – так очам легше читати. На широких моніторах, навпаки, його можна зменшити, щоб не створювалося відчуття порожнечі між рядками.

Трекінг (letter-spacing) – це відстань між літерами. Він допомагає зберегти чіткість і ритм тексту, насамперед коли змінюється розмір екрана або застосовується нестандартний шрифт. Для великих заголовків корисно трохи **збільшити трекінг** – це додає відчуття простору й підкреслює форму букв. Для звичайного тексту, навпаки, **занадто великий трекінг** може зробити читання незручним – слова ніби «розсипаються».

Головне правило – **читабельність понад усе**. Міжрядковий інтервал і трекінг повинні не просто виглядати гарно, а допомагати читачеві сприймати інформацію природно, без напруги. Це особливо важливо в інтерфейсах, де користувач швидко «сканує» поглядом екран – поганий ритм тексту одразу сповільнює сприйняття.

МІКРОТИПОГРАФІКА

До мікротипографіки належать дрібні деталі – правильні лапки, тире, апострофи, дефіси, неочевидні відступи. Ці нюанси створюють професійне враження. В адаптивному дизайні вони особливо важливі, бо навіть дрібні помилки стають помітними на малих екранах.

Дизайнер має стежити за переносами, уникати «висячих» прийменників і дбати, щоб текст не обрізався при ресайзингу.

Наприклад: «**Сьогодні – гарна погода**» читається набагато природніше, ніж "**Сьогодні - гарна погода**".

Важливе значення має й оптичне вирівнювання. Навіть, коли текст формально є по центру, він може виглядати зміщеним через форми літер. Тому досвідчені дизайнери коригують положення вручну – на кілька пікселів, щоб досягти візуального балансу. Ці корекції ледь помітні, але саме вони створюють гармонійне сприйняття.

У таблицях і формах важливо дбати про цифри. Якщо вони не вирівняні по розрядах, зчитування даних ускладнюється. Для таких випадків використовують **моноширинні цифри**, у яких усі символи мають однакову ширину. Це робить фінансову або статистичну інформацію впорядкованою й професійною.

Не менш важливі й пробіли. Вони формують логічні зв'язки між елементами. Наприклад, у записі **20 °C** варто залишати тонкий або нерозривний пробіл, щоб значення не розривалося при адаптації тексту. У навігаційних рядках на кшталт «**Головна / Контакти**» пробіл додає легкості й структури.

UX-WRITING ТА КОНТЕНТ-ДИЗАЙН

Тексти в інтерфейсі – це не просто слова. Це **підказки, рішення і настрій** одночасно. UX-writing – це мистецтво писати коротко, зрозуміло й з турботою про користувача. Завдання контент-дизайнера – зробити так, щоб людина ніколи не губилася в інтерфейсі й завжди розуміла, що відбувається.

Хороший UX-текст говорить із користувачем людською мовою. Замість «Помилка 404» він пояснить: «Сторінку не знайдено. Можливо, її перемістили або видалили». Такий підхід створює довіру і робить взаємодію теплішою.

У контент-дизайні важлива **структура**: кожне слово має свою роль. Заголовки привертають увагу, підзаголовки допомагають орієнтуватися, кнопки дають чіткі дії – «Надіслати», «Зберегти», «Продовжити». Не варто перевантажувати інтерфейс довгими фразами – краще, коли все сказано просто й у два-три слова.

Ще один принцип – **послідовність тону**. Якщо інтерфейс «говорить» із користувачем доброзичливо й спокійно, цього стилю треба дотримуватись усюди. Тон текстів – це частина бренду, його голос. Він може бути офіційним, нейтральним або емоційним, але головне – не змінюватися від екрана до екрана.

І нарешті – **узгодженість тексту і дизайну**. Якщо кнопка виглядає активною, вона має вести до дії; якщо з'являється повідомлення про помилку, воно повинно не лише констатувати, а й пояснювати, як виправити ситуацію. Текст і візуальна мова мають працювати як одне ціле.

Типографіка і UX-writing – це дві сторони одного процесу: одна відповідає за візуальний ритм, інша – за сенс. Разом вони створюють інтерфейс, який не просто гарно виглядає, а й говорить із користувачем зрозумілою мовою.

5.4. ТЕСТУВАННЯ АДАПТИВНОСТІ

Адаптивний дизайн не можна вважати завершеним, доки його не перевірено в дії. **Тестування адаптивності** – це етап, коли дизайнер або команда перевіряють, як інтерфейс виглядає і працює на різних пристроях, розмірах екранів і в різних браузерах. Головне питання тут одне: **чи буде сайт зручним і зрозумілим, незалежно від того, з чого його відкривають.**

Перше, що потрібно зробити, – **переглянути макет у різних розмірах екранів.** У Figma це можна зробити просто, змінюючи ширину фрейму або використовуючи готові пресети – наприклад, Desktop (1440 px), Tablet (768 px), Mobile (375 px). Під час цього варто спостерігати, як поведуться блоки, чи не «ламається» сітка, чи є відступи логічними. Добре, коли елементи плавно підлаштовуються під простір, а не стискаються чи зникають.

Наступний крок – **перевірка типографіки.** Шрифти не мають ставати надто дрібними або роз'їжджатися. Якщо на мобільному екрані текст виглядає надто тісно, варто збільшити міжрядковий інтервал або відступи. Тестування допомагає знайти такі дрібні деталі, які неможливо помітити в одному фіксованому макеті.

Окремо перевіряється **навігація та клікабельні зони.** На великому екрані користувач працює мишею, а на телефоні – пальцем, тому кнопки мають бути доволі великими (щонайменше 44×44 пікселі) і зручними для натискання. Також потрібно переконатися, що випадаючі меню, поля введення і спливаючі вікна не перекривають основний контент.

Ще один важливий аспект – **швидкість і поведінка контенту.** Зображення, відео чи інтерактивні елементи мають завантажуватись без затримок. Якщо картинки надто великі, їх варто оптимізувати під кожен тип екрана. Це не лише покращує досвід користувача, а й впливає на SEO – пошукові системи оцінюють швидкість завантаження.

Тестування адаптивності не обмежується лише дизайнерськими інструментами. Після верстки варто перевірити сайт у **реальних умовах** – на смартфоні, планшеті, ноутбучі. Так можна побачити те, чого не видно в Figma: наприклад, як поводить

меню при скролі, як працюють посилання чи як відображаються кольори на різних екранах.

І нарешті – **зворотний зв’язок від користувачів**. Те, що дизайнер вважає зручним, не завжди сприймається так само людьми, які користуються сайтом щодня. Кілька реальних тестів – навіть серед друзів чи колег – можуть показати, що десь кнопка занадто близько до іншої, а десь текст складно прочитати.

Тестування адаптивності – це не фінальна формальність, а **невід’ємна частина дизайну**, яка гарантує, що результат працює в реальному світі. Адже справжній хороший інтерфейс – це не той, що гарно виглядає у Figma, а той, що є зручним на будь-якому екрані.

5.5. РОБОТА З ОНЛАЙН-РЕСУРСАМИ: ШРИФТИ, ІКОНКИ, ШАБЛони

Сучасний дизайнер не створює все «з нуля». Інтернет надає величезну кількість інструментів і бібліотек, які допомагають працювати швидше, якісніше й системніше. Головне – **уміти правильно добирати ресурси**, щоб вони не лише прикрашали дизайн, а й відповідали стилю проекту, технічним вимогам і авторським правам.

ШРИФТИ

Підбір шрифту – одна з найважливіших частин будь-якого інтерфейсу. Щоб текст виглядав професійно, варто використовувати **перевірені джерела**, де шрифти мають відкриті ліцензії або чітко вказані умови використання.

Популярні платформи:

- **Google Fonts** – головне безкоштовне сховище. Тут можна фільтрувати шрифти за мовами, накресленнями, товщиною лінії. Кожен шрифт можна протестувати у власному тексті та одразу завантажити або підключити через CSS.

- **Adobe Fonts** – професійна бібліотека, інтегрована з Creative Cloud. Зручно для комерційних проєктів: усі шрифти мають чіткі ліцензії й автоматично активуються у Figma, Photoshop чи Illustrator.
- **Fontshare, Velvetyne, Fontsquirrel** – спільноти з відкритими шрифтами, часто з унікальним характером і підтримкою кирилиці.

Коли йдеться про українські тексти, важливо **перевіряти наявність кирилиці**. Навіть популярні шрифти можуть не мати літер «ї», «є», «г». У Google Fonts можна ввімкнути фільтр **Cyrillic Extended** – це допоможе уникнути сюрпризів під час верстки.

При виборі шрифтів передусім потрібно орієнтуватися на **читабельність і контраст**. Для великих текстів підходять прості гротески або гуманістичні шрифти – наприклад, **Inter, Roboto, Source Sans 3**. Для акцентів чи заголовків можна використати характерніші – **Playfair Display, Raleway, Space Grotesk**.

Важливо, щоб шрифт мав **кілька накреслень** (Regular, Medium, Bold, Italic). Це дає змогу створювати чітку ієрархію та виділяти потрібні акценти. У дизайні інтерфейсів достатньо трьох рівнів контрасту: звичайний текст, напівжирний заголовок і дрібні підписи або допоміжна інформація.

Ще один момент – **технічна оптимізація**. Деякі шрифти гарно виглядають у друці, але на екрані рендеряться нерівномірно. Варто вибирати ті, що створені саме для цифрового використання: вони мають збалансовані пропорції, не «пливуть» у пікселях і зберігають чіткість навіть на Retina-дисплеях.

Не всі шрифти можна використовувати у комерційних проєктах безкоштовно. Деякі мають ліцензію **Personal Use Only**, інші – **Open Font License (OFL)** або **Commercial Use**. Завжди читай умови перед завантаженням – це не лише питання законності, а й професійної репутації.

Якщо планується передавати макети замовнику, краще використовувати шрифти з відкритими ліцензіями (наприклад, із **Google Fonts** або **Fontshare**). Так можна уникнеш ситуації, коли клієнт не може відкрити проєкт через відсутність платного шрифту.

ІКОНКИ

Іконки – це мова інтерфейсу. Вони допомагають користувачу швидше орієнтуватися і зменшують навантаження на текст. Але тільки тоді, коли виконані в **єдиному стилі**.

Інтернет пропонує сотні бібліотек і наборів іконок. Головне – вибрати ті, що відповідають стилю вашого проєкту та не порушують авторських прав.

- **Figma Community** → **Icon sets** – найзручніше місце для дизайнерів. Тут можна знайти тисячі наборів, створених іншими дизайнерами, і одразу імпортувати їх у свій проєкт без додаткових плагінів.
- **Flaticon** – одна з найбільших баз безкоштовних іконок. Підтримує формати SVG, PNG, EPS. Можна редагувати колір, розмір і завантажувати окремі піктограми чи цілі пакети.
- **Noun Project** – це бібліотека, де кожна іконка має автора. Велика частина доступна за умови посилання на автора, або за платною ліцензією для комерційних цілей.
- **Lucide Icons** і **Feather Icons** – легкі, мінімалістичні колекції, ідеальні для сучасних інтерфейсів у стилі «quiet minimalism» або «Japandi digital».
- **Material Symbols** – офіційна бібліотека від Google, зручна для проєктів у стилі Android або коли потрібна сумісність із Material Design.

Іконки мають бути не просто красивими, а **функціональними**. Кожна з них повинна чітко передавати сенс дії чи об'єкта. Якщо користувач змушений здогадуватися, що означає значок, – це сигнал, що варто змінити піктограму.

Найголовніше правило – **стильова єдність**. Не можна змішувати, наприклад, лінійні іконки з об'ємними або кольоровими. Це руйнує візуальну гармонію інтерфейсу. Якщо набір піктограм не має потрібної іконки, краще створити її вручну, зберігши товщину ліній, пропорції та стиль решти елементів.

Ще один важливий аспект – **розмір і контраст**. Іконки мають бути достатньо великими, щоб залишатися читабельними навіть на мобільних пристроях. Найпоширеніший розмір – 24×24 або 32×32 пікселі. Варто також перевіряти, чи не зливаються вони з фоном і чи зберігається контраст при зміні темної або світлої теми.

Чимало іконок поширюються за відкритими ліцензіями – наприклад, **Creative Commons** або **MIT License**. Це означає, що їх можна використовувати у своїх проєктах, але часто потрібно **вказати автора** або не використовувати для комерційних цілей без дозволу.

Якщо іконки завантажені з платних сервісів (UI8, Noun Project, Iconfinder Pro), важливо зберегти документи про ліцензію. Це не лише формальність – це частина професійної відповідальності дизайнера перед клієнтом.

Поради з практики:

- Використовуйте **один набір іконок на проєкт** – це створює візуальну єдність.
- Зберігайте **єдину товщину лінії** (наприклад, 1.5 pt або 2 pt).
- Уникайте надмірної деталізації – дрібні елементи губляться при зменшенні.
- Якщо іконка не має очевидного значення, додайте **текстову підказку** або **лейбл**.
- Заздалегідь продумайте **стани іконок**: активний, неактивний, наведений, натиснутий.

ШАБЛони

Шаблони значно пришвидшують роботу, насамперед на старті проєкту. Це можуть бути **готові UI-кити, фреймворки або сітки**, які можна адаптувати під свій стиль. Вони допомагають не витратити час на базові елементи – кнопки, картки, модальні вікна – і зосередитися на головному: логіці й естетиці.

Нині є безліч платформ, де можна знайти якісні шаблони для будь-яких цілей – від лендінгів до мобільних додатків.

- **Figma Community** – головне джерело для дизайнерів. Тут є тисячі готових макетів сайтів, UI-кітів, компонентів, шаблонів презентацій, і навіть повні дизайн-системи. Важливо, що все це можна одразу редагувати у Figma, не виходячи з робочого середовища.
- **UI8** – платформа з преміум-шаблонами та UI-китами. Всі матеріали створюють професійні дизайнери, тож вони придатні для комерційного використання.

- **Creative Market** та **Envato Elements** – маркетплейси з великим вибором шаблонів, які підходять для різних програм: Figma, Sketch, Adobe XD, PowerPoint тощо.
- **Dribbble** і **Behance** – джерела натхнення. Тут часто можна знайти безкоштовні демо-файли або посилання на шаблони від авторів.

Шаблон – це **не готове рішення**, а лише **відправна точка**. Він може показати вдалі пропорції, структуру або логіку побудови сторінки, але справжня цінність у тому, як дизайнер його адаптує під свої потреби.

Коли ви завантажуєте готовий макет, не варто обмежуватись заміною тексту та зображень. Краще **розібрати шаблон** – подивитись, як автор побудував сітку, як працює Auto Layout, які стилі кольорів і тексту використовуються, як вирішена адаптивність. Це не просто економія часу, а **ефективне навчання**.

Також варто пам'ятати, що не всі шаблони зроблені якісно. Іноді вони перевантажені декоративними елементами, мають хаотичні назви шарів або відсутні стилі. Тому перед початком роботи важливо навести порядок: створити власну систему стилів, кольорів і компонентів.

Навіть якщо шаблон безкоштовний, потрібно уважно читати умови використання. Часто вони вимагають вказувати автора або забороняють комерційне використання без ліцензії. Якщо планується клієнтський проект, найкраще брати шаблони з відкритою або комерційною ліцензією (наприклад, на UI8 або Envato).

Неетично видавати чужий шаблон за власний. У професійному середовищі цінується не копіювання, а **вміння адаптувати** готовий макет під власні задачі, додаючи унікальність і продуманість.

Отже, шаблони – це інструмент не для лінощів, а для **розумного пришвидшення процесу**. Вони допомагають побачити хороші рішення, швидше стартувати й зрозуміти логіку побудови професійних макетів. Але справжня майстерність проявляється тоді, коли дизайнер не просто використовує шаблон, а **переосмислює його під власний стиль і завдання**.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. All WCAG 2.2 Understanding Docs. URL: <https://www.w3.org/WAI/WCAG22/Understanding/> (дата звернення: 19.10.2025).
2. Staiano F. Designing and Prototyping Interfaces with Figma – Second Edition: Elevate your design craft with UX/UI principles and create interactive prototypes. 2nd ed. 2023. 412 p.
3. Green T., Brandon K. UX Design with Figma: User-Centered Interface Design and Prototyping with Figma. Etobicoke, ON, Canada, 2024. 564 p.
4. Sanchez A. E. Newbies Guide to UI/UX Design Using Figma: Fundamentals of User Interface (UI) and User Experience (UX) Design Using Figma to Create User-centric Interfaces and Interactive Prototypes. Paperback, 2024. 108 p.
5. Стофер Б., Верхувен Є., Фасс Дж., Ревелл Т. Design and Digital Interfaces: Designing with Aesthetic and Ethical Awareness. Bloomsbury, 2021. 398 с.
6. Figma: The Collaborative Interface Design Tool. URL: <https://www.figma.com/> (дата звернення: 19.10.2025).
7. Шабала Є. Є. Комп'ютерна графіка та моделювання: конспект лекцій. Київ : КНУБА, 2022. 108 с.
8. Бородавка Є. В., Терентьев О. О. Комп'ютерна графіка : навчальний посібник. Київ : КНУБА, 2023. 132 с.
9. Шеховцов А. В., Полетаєва Г. Н., Крючковський Д. О. Комп'ютерні технології для дизайнерів : навчальний посібник. 2024. 318 с.
10. Пастернак І. І., Костик А. Т. Інструментальні засоби веб-технологій : навчальний посібник для студентів другого (магістерського) рівня вищої освіти спеціальності 123 «Комп'ютерна інженерія». Львів : Видавництво «Магнолія 2006», 2024. 197 с.
11. Paint.NET. URL: <https://paintnet.org.ua/> (дата звернення: 19.10.2025).
12. Design Resources. The Best Tools and Plugins for Designers. URL: <https://www.toools.design/> (дата звернення: 19.10.2025).

13. Awwwards. Best Web Design Trends and Inspiration. URL: <https://www.awwwards.com/websites/> (дата звернення: 19.10.2025).
14. Webflow. Top Web Design Tools and Resources 2025. URL: <https://webflow.com/blog/web-design-tools> (дата звернення: 19.10.2025).
15. HubSpot. Website Design Resources and Free Templates. URL: <https://www.hubspot.com/resources/website-design> (дата звернення: 19.10.2025).
16. Interaction Design Foundation. UX Design Glossary and Learning Resources. URL: <https://www.interaction-design.org/> (дата звернення: 19.10.2025).

АВТОРСЬКИЙ ВНЕСОК

Навчальний посібник **«Проектування користувацьких інтерфейсів та комп'ютерної графіки»** є результатом спільної роботи авторів. Кожен із них зробив внесок у структуру, зміст і практичну частину видання. Матеріал розроблено з урахуванням сучасних тенденцій у галузі цифрового дизайну, вебтехнологій і візуальних комунікацій.

Розподіл авторства за розділами:

- Розділ 1. Основи комп'ютерної графіки та вебдизайну – *Магеровська Т. В.*
- Розділ 2. Векторна графіка у вебдизайні – *Магеровська Т. В.*
- Розділ 3. Растрова графіка у вебдизайні – *Магеровська Т. В.*
- Розділ 4. Інтерактивність, прототипування та оптимізація вебдизайну –
Магеровський Д. В.
- Розділ 5. Адаптивний дизайн – *Сеник В. В.*

МАГЕРОВСЬКА Тетяна Валеріївна,
кандидат фізико-математичних наук, доцент

СЕНИК Володимир Васильович,
кандидат технічних наук, доцент

МАГЕРОВСЬКИЙ Дмитро Васильович

Проектування користувацьких інтерфейсів та комп'ютерної графіки

Навчальний посібник

Редагування *Галина Ялечко*
Макетування *Галина Шушняк*

Формат 60×84/8. Умовн. друк. арк. 47,43.

Зам № 95-25.

Львівський державний університет внутрішніх справ
Україна, 79007, м. Львів, вул. Городоцька, 26.

Свідоцтво про внесення суб'єкта видавничої справи до державного реєстру
видавців, виготівників і розповсюджувачів видавничої продукції

ДК № 2541 від 26 червня 2006 р.