ОСВІТА
ВИДАВНИЧИЙ ДІМ

Ольга Коршунова, Ігор Завадський,
Зоряна Стасюк, Валентина Потієнко

ІНФОРМАТИКА

9 КЛАС

- Електронні таблиці (табличний процесор)
  - Базові поняття та призначення
    - Основне визначення
      - Програма для зберігання, опрацювання й аналізу даних у вигляді електронних таблиць
    - Функції та можливості
      - Наочне подання великих обсягів даних
      - Підтримка швидкого прийняття рішень на основі даних
      - Автоматизація обчислень, аналізу, візуалізації
    - Аркуші та книги
    - Структура даних
      - Рядок — запис / об’єкт
      - Стовпець — поле / параметр (ознака об’єкта)
      - Клітинка — значення параметру
    - Сфери застосування (будь-яка галузь, де потрібно працювати з даними)
  - Опрацювання та візуалізація даних
    - Сортування та фільтрація
      - Сортування
        - Певний порядок (за зростанням / спаданням)
        - Багаторівневе (за кількома стовпцями)
      - Фільтрація (відбір за умовами)
        - Прості фільтри (текстові, числові, за датою)
        - Розширені фільтри (декілька умов)
        - Правила виділення клітинок (більше, між, текст)
        - Правила відбору перших / останніх значень
    - Візуальний аналіз (умовне форматування)
      - Призначення: швидке виявлення тенденцій
      - Типи форматів
        - Гістограми (смужки даних)
        - Кольорові шкали (градації кольору, як теплова карта)
        - Набори піктограм
  - Аналіз та обчислення
    - Формули та посилання
    - Функції з умовами (агрегація)
      - Логічні функції
      - Функції для агрегування за умовою
        - COUNTIF / COUNTIFS (кількість)
        - SUMIF / SUMIFS (сума)
        - AVERAGEIF / AVERAGEIFS (середнє)
        - MINIFS / MAXIFS (екстремуми)
      - D-функції (робота з діапазоном як з базою даних)
        - DCOUNT, DSUM, DAVERAGE, DMIN, DMAX
        - Для складніших критеріїв

Ольга Коршунова, Ігор Завадський,
Зоряна Стасюк, Валентина Потієнко

Інформатика

Підручник для 9 класу
закладів загальної середньої освіти

Рекомендовано Міністерством освіти і науки України

Київ
Видавничий дім «Освіта»
2026

Рекомендовано Міністерством освіти і науки України
(наказ Міністерства освіти і науки України від 27.01.2026 № 110)

Підручник розроблено за модельною навчальною програмою
«Інформатика. 7–9 класи»
для закладів загальної середньої освіти

(авт. Завадський І. О., Коршунова О. В., Твердохліб І. А.)

Інтерактивний електронний додаток
(далі — е-додаток) до підручника розташовано за посиланням
https://smarttextbook.com.ua/informatyka-9/

Умовні позначки:

— Завдання рекомендуємо виконувати в парах чи у малих групах

— Завдання, які потрібно виконувати з використанням комп’ютера

Вітаємо!

Як ти вважаєш, **що важливіше — відповідь чи запитання?**
Поміркуймо!

> Чи розумна людина — дізнається з її відповідей.
> Чи мудра людина — дізнається з її запитань.
>
> *Наґіб Махфуз (Naguib Mahfouz), лауреат Нобелівської премії*

Запитання — це спосіб отримання інформації і водночас траєкторія думок людини. За допомогою запитань ми прокладаємо собі міст у невідоме та невизначене. Оскільки невизначеність і невідомість — це характерна риса сучасного світу, що стрімко змінюється, уміння ставити запитання є дуже актуальним, і його потрібно розвивати. Джерелом усіх запитань є подив, цікавість та допитливість. Ми ставимо запитання, шукаємо рішення і таким чином вирішуємо проблеми, навчаємося, розширюємо власні погляди й пізнаємо світ. Уміння ставити запитання допоможе кожному у професійній сфері й повсякденному житті, розвине креативність та критичність мислення.

Перед тобою перші сторінки підручника з інформатики для учнів 9 класу. Він допоможе тобі опанувати не тільки програму з інформатики, а й навчить учитися та самостійно здобувати інформацію. Кожний параграф у нашому підручнику розпочинається з проблемного запитання, відповісти на яке ти зможеш, опрацювавши зміст параграфа.

Як саме працювати з матеріалами підручника?

Спробуй виконати запропоновані завдання, під час пошуку рішень став запитання і знаходь на них відповіді в різних джерелах: у спілкуванні з однокласниками / однокласницями, у рубриці **«Запитання-відповіді»**, у мережі «Інтернет» тощо. Але... «Question everything!» («Усе став під сумнів!»), як чудово сказав Альберт Ейнштейн, перевіряй джерела інформації та свої міркування.

Перевірити ж власні навчальні досягнення ти зможеш, виконавши завдання з рубрики «Перевір себе».

Бажаємо цікавого та результативного навчального року!

Автори

P.S. А ти вже можеш відповісти на запитання на початку цієї передмови? 😉

РОЗДІЛ І

ОПРАЦЮВАННЯ ТАБЛИЧНИХ ДАНИХ

§ 1. Табличні дані в сучасному світі
§ 2. Сортування. Прості та розширені фільтри
§ 3. Візуальний аналіз даних
§ 4. Дані та аналітичні висновки
§ 5. Основи машинного навчання в електронних таблицях

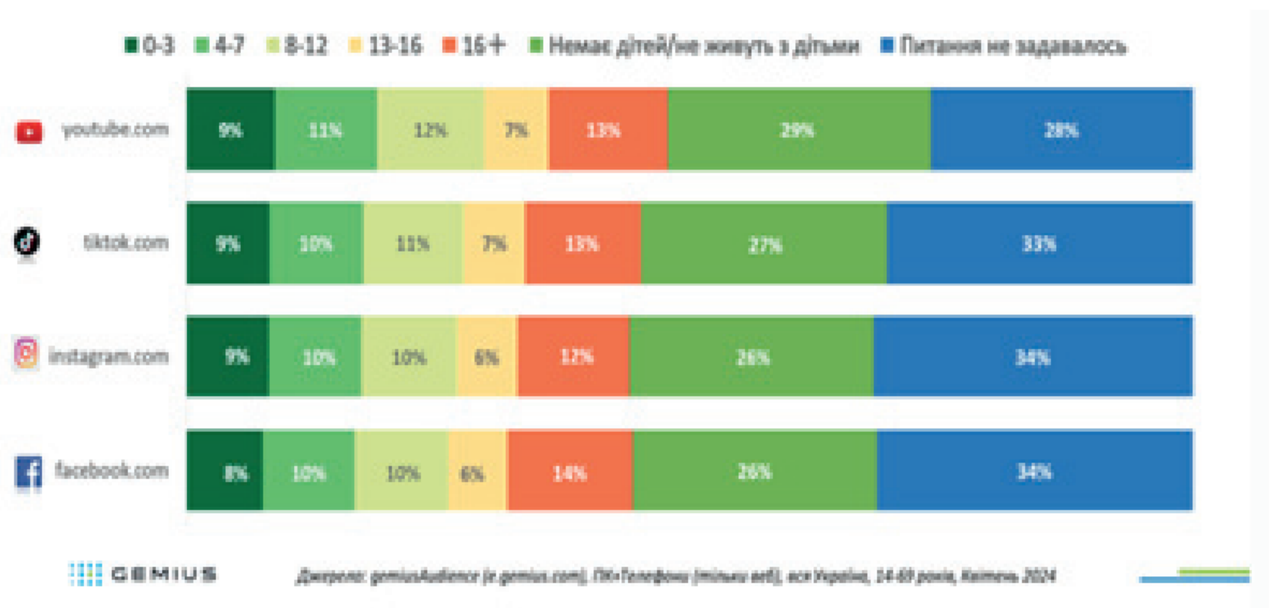
У розділі І більшість завдань та алгоритмів дій користувача подано на прикладі Google Таблиць, і тільки деякі доповнені прикладами з інтерфейсом Microsoft Excel.

§ 1. Табличні дані в сучасному світі

Чому табличне подання даних стало основою для прийняття рішень у сучасному світі?

Склад сімейної аудиторії в соціальних мережах за наявністю / проживанням дітей та їх віку

Квітень 2024 року



Джерело: https://vse.ee/cxff

Ми живемо у світі даних. Щодня ти бачиш цифри — у статистиці Instagram, у фітнес-додатках, у списках покупок, у щоденнику оцінок. Уміння працювати з даними — це як мати суперсилу: вона допомагає краще розуміти світ, швидше ухвалювати рішення і навіть уникати помилок.

Уяви, що твій старий телефон уже підводить, і настав час купити новий. У магазині сотні моделей — різні ціни, характеристики, виробники. В одного — крута камера, але слабка батарея. В іншого — суперпотужність, але ціна кусається. Як вибрати найкращий варіант за свої гроші? А тепер уяви, що ти вибираєш вид спорту: одні секції — дорого, інші — далеко, а треті дають надто велике навантаження на коліна. Як вибрати найкраще?

У таких ситуаціях рішення — не просто питання інтуїції. Потрібен **аналіз**. А щоб аналізувати, треба вміти правильно зібрати й подати інформацію. І саме тут стають у пригоді **електронні таблиці** — потужний інструмент, який використовують і школярі, і бізнесмени, і спортсмени.

Цей параграф нагадає тобі, як табличні процесори допомагають у розв'язанні реальних життєвих проблем. Опанувавши його, ти зможеш працювати з інформацією як справжній аналітик.

Завдання № 1

Розглянь хмару слів

об’єм двигуна 180 км/год 1,2 л Peugeot 108
BMW M135i xDrive
максимальна швидкість жовтий сірий

а Що серед наведених понять є об’єктом, параметром та значенням параметра? Про які об’єкти йдеться?

б Створи таблицю для зберігання відомостей про ці об’єкти. У таблиці має бути принаймні чотири стовпці, рядок заголовка та рядки трьох об’єктів.

в Заповни таблицю даними, знайшовши в інтернеті інформацію принаймні про три об’єкти.

Завдання № 2

Уявіть, що твоя родина планує придбати електромобіль і розглядає різні моделі. Визнач, які характеристики важливі для порівняння, і створи структуру таблиці для прийняття рішення. Урахуй не лише технічні параметри, а й екологічні та економічні чинники.

Об’єднайтесь у пари, порівняйте свої таблиці. Презентуйте найкраще спільне рішення.

Завдання № 3

Розглянь дані про енергоспоживання різних побутових приладів у таблиці та проаналізуй ситуацію.

Прилад	Потужність (Вт)	Час роботи на день (год)
Холодильник	200	12
Пральна машина	2000	1,5
Комп’ютер	300	6
LED-лампочка	10	8

а Сформулюй три запитання, відповіді на які можна отримати, аналізуючи ці дані. Поясни, чому табличне подання цієї інформації є зручнішим за текстовий опис.

б За допомогою якої формули можна визначити, скільки електроенергії витрачає кожен пристрій протягом доби? Де доречно розмістити результати її обчислення?

в Створи проаналізовану таблицю в табличному процесорі. Додай до неї стовпець для визначення добового обсягу споживання електроенергії кожним пристроєм. Введи у стовпець формулу, що визначає добове споживання холодильника, і скопіюй її в рядки, що відповідають іншим пристроям.

г Додай до таблиці рядок, у якому визнач середню потужність, середній час роботи та середній обсяг добового енергоспоживання зазначених у таблиці пристроїв. Для цього скористайся функцією AVERAGE.

Завдання № 4

Ти отримав / отримала завдання спланувати закупівлю продуктів для приготування святкового обіду. Є список продуктів, які потрібно купити. Ось що вдалося дізнатися.

Ціна (1 кг) картоплі в супермаркеті — 14,90 грн, на ринку — 13,50 грн, в онлайн-магазині — 15,00 грн.

Олія (1 л): ринок — 58 грн, супермаркет — 55 грн, в онлайн-магазині — 54,50 грн.

Філе курки (1 кг) — 132 грн в онлайн-магазині, 130 грн — у супермаркеті, 128 грн — на ринку.

Морква (1 кг): супермаркет — 10 грн, ринок — 9,80 грн, в онлайн-магазині — 11,20 грн.

Майонез (400 г): ринок — 36,50 грн, в онлайн-магазині — 39 грн, супермаркет — 37,90 грн.

Сир твердий (1 кг): супермаркет — 264 грн, в онлайн-магазині — 255 грн, ринок — 268 грн.

Батон: ринок — 18 грн, супермаркет — 17,50 грн, в онлайн-магазині — 17,80.

Сметана (400 г): в онлайн-магазині — 38,90 грн, супермаркет — 40 грн, ринок — 39,50 грн.

Огірки (1 кг): супермаркет — 55 грн, ринок — 52 грн, в онлайн-магазині — 54 грн.

Помідори (1 кг): ринок — 62 грн, супермаркет — 60 грн, в онлайн-магазині — 61,50 грн.

Цибуля (1 кг): 15 грн на ринку, 14,50 грн — у супермаркеті, 14,90 грн — в онлайн-магазині.

а Подай ці дані у вигляді таблиці.

б Додай стовпець, у якому за допомогою функції MIN визнач найвигіднішу ціну кожного товару.

в Обчисли загальну суму закупівлі, якщо купувати:
тільки в супермаркеті;
тільки на ринку;
тільки в онлайн-магазині;
ідеальний варіант — де кожен товар купують там, де він найдешевший.

г Візуалізуй дані за допомогою діаграми, дібравши найбільш відповідний її тип.

д Зроби висновок: який спосіб покупки найвигідніший?

Завдання № 5

Прийми рішення за допомогою табличних даних. Уяви, що тобі потрібно обрати найкращий варіант курсу англійської мови з трьох запропонованих. Є інформація про вартість, тривалість курсу, кількість занять, репутацію школи (за відгуками), близькість до дому.

Назва школи	Вартість (грн)	Тривалість (тижнів)	Кількість занять	Оцінка за відгуками (з 5)	Відстань до школи (км)
LinguaPro	5400	8	32	4,7	2,5
EnglishTime	4800	6	24	4,5	0,8
SmartTalk	5000	10	40	4,2	3,0

а З п’ятьох наявних критеріїв (вартість, тривалість, кількість занять, оцінка за відгуками та відстань) визнач два або три найважливіших для тебе.

б За кожним із важливих критеріїв визнач, яка школа є найкращою.

в Введи дані в електронну таблицю. Додай до таблиці стовпець «Вартість одного заняття» та обчисли дані в ньому за допомогою відповідної формули. Визнач, яка школа є найкращою за цим критерієм.

г Згідно з різними критеріями кращими є різні школи. Як з-поміж них все-таки вибрати одну? Запропонуй методику або формулу, за допомогою якої можна поєднати різні критерії та визначити найкращу школу.

Завдання № 6

Рекламна агенція створила таблицю для порівняння популярності месенджерів серед підлітків:

Месенджер	Кількість користувачів (млн)	Середній час використання (хв/день)	Рейтинг безпеки (1–10)
Telegram	950	85	9
WhatsApp	2800	70	7
Viber	260	45	8

На основі цієї таблиці агенція зробила висновок: *Telegram — найкращий месенджер для підлітків*. Об’єднайтесь у пари.

а Дайте відповіді на запитання, застосувавши навички критичного аналізу:
- Які запитання потрібно поставити щодо цих даних перед тим, як робити висновок?
- Чи об’єктивним є висновок? Якщо ні, то чому?
- Які важливі показники могли бути пропущені в таблиці?
- Як ті самі дані можна інтерпретувати, щоб зробити «переможцем» інший месенджер?

б Сформулюйте правила, яких варто дотримуватися під час аналізу табличних даних, щоб уникнути маніпуляцій та неправильних висновків. Скористайтеся рубрикою «Запитання-відповіді».

Завдання № 7

Для даних з попереднього завдання вигадайте формулу, яка виводила б для кожного месенджера підсумкову оцінку якості, що враховує всі три критерії.

а Введіть дані в електронну таблицю за зразком, а в останньому стовпці введіть та скопіюйте винайдену формулу.

б Який месенджер виявився найкращим згідно з підсумковою оцінкою? Порівняйте результати та формулу оцінювання з тими, які отримали однокласники / однокласниці.

в Обговоріть у класі результати. Запозичте найкращі ідеї одне в одного. Спробуйте вивести спільну найбільш об'єктивну формулу.

ЗАПИТАННЯ-ВІДПОВІДІ

Чому табличне подання даних є таким поширеним?

Табличне подання даних надзвичайно поширене в сучасному світі, тому що забезпечує ефективне, зручне й наочне представлення великих обсягів інформації. Крім того, таблична форма дає змогу легко автоматизувати обчислення, аналіз та візуалізацію даних за допомогою цифрових інструментів, таких як електронні таблиці. Якщо змінити вхідні дані в електронній таблиці, результати обчислень та діаграми зміняться автоматично.

Застосування таблиць охоплює найрізноманітніші галузі. У науці таблиці використовують для подання результатів експериментів, у спорті — для ведення статистики та рейтингів, в економіці — для фінансової звітності, в екології — для моніторингу стану довкілля, у медицині — для ведення історій хвороб та запису результатів обстежень. Практично в будь-якій сфері, де потрібно приймати рішення на основі даних, активно використовують таблиці.

Яке програмне забезпечення називають електронними таблицями?

Електронна таблиця — це комп'ютерна програма, що дає змогу зберігати й обробляти інформацію в табличній формі. Такі прикладні програми називаються ще табличними редакторами (процесорами).

Можливості електронних таблиць:

- створення й редагування електронних таблиць;
- створення багатотабличних документів;
- оформлення і друк електронних таблиць;
- побудова діаграм, їх модифікація і вирішення економічних завдань графічними методами;
- створення багатотабличних документів, об'єднаних формулами;

- робота з електронними таблицями як з базами даних: сортування таблиць, вибірка даних за запитами;
- створення підсумкових і зведених таблиць;
- використання під час побудови таблиць інформації із зовнішніх баз даних;
- створення слайд-шоу;
- вирішення оптимізаційних завдань;
- вирішення економічних завдань типу «якщо — то» шляхом підбору параметрів;
- розроблення макрокоманд, налаштування середовища під потреби користувача тощо.

Дані, що входять у таблиці, можна копіювати, видаляти, сортувати (тобто розташовувати клітинки, а також рядки і стовпці з них у певному порядку); виконувати дії над усією таблицею або її частиною і роздруковувати частково або повністю на папір; зміна даних, які пов'язані, миттєво призводить до змін у всьому документі.

Таблиці якої структури використовують найчастіше?

Найпоширенішою є така структура таблиці:

- кожен рядок відповідає окремому об'єкту;
- кожен стовпець відповідає певній властивості чи параметру;
- у кожній клітинці міститься значення певного параметра певного об'єкта.

Дата	Середня температура	Швидкість вітру	Хмарність	Опади
01.Чер	21	1,4	хмарно	дощ
02.Чер	23	1,6	мінлива хмарність	сухо
03.Чер	23	2,8	ясно	сухо
04.Чер	28	0,7	ясно	сухо
05.Чер	30	1,1	ясно	сухо
06.Чер	31	3,8	ясно	сухо
07.Чер	20	5,7	хмарно	гроза

У цілому таблиця такої структури містить інформацію про набір однотипних об'єктів. Об'єкти є однотипними, оскільки вони мають однаковий набір параметрів.

Як визначити, чи доцільно подавати інформацію у вигляді таблиці?

Для визначення доцільності табличного подання інформації аналізують тип і призначення даних. Як уже зазначалося, таблиця є ефективною формою подання, коли потрібно зіставити кілька об'єктів, явищ або подій, що мають спільні властивості. У такому разі кожен об'єкт розміщується в окремому рядку, а властивості — у стовпцях. Табличне подання доцільне також тоді, коли дані потрібно опрацювати, відсортувати, згрупувати чи візуалізувати. Натомість для опису подій, процесів або аналізу причинно-наслідкових зв'язків більш доречною буде текстова форма.

Отже, таблиця доцільна тоді, коли інформація має регулярну структуру, а її опрацювання потребує наочності, точності й зручності.

Як таблиці допомагають у прийнятті рішень?

Розміщення даних у рядках і стовпцях дає змогу виявляти закономірності, тренди, винятки та залежності, які часто залишаються непоміченими в неструктурованих даних. Наприклад, завдяки таблиці можна швидко помітити зв'язок між обсягом виробництва та прибутком підприємства і знайти той обсяг, що максимізує прибуток.

Однак часто мало просто розмістити дані в таблиці. Щоб отримати важливу інформацію, таблицю потрібно опрацювати. Наприклад, є таблиця з даними про ціни сотень товарів, а тобі потрібно вибрати три найдешевші. Для цього таблицю треба *відсортувати*, тобто розташувати її рядки в порядку зростання чи спадання значень певного параметра, у нашому випадку — ціни. Виконати сортування та багато інших корисних операцій можна в *табличному процесорі.*

Подання даних у вигляді таблиць зменшує ризик хибних висновків, що базуються лише на враженнях чи неповній інформації, і підвищує якість ухваленого рішення. Отже, таблиці виступають не просто форматом подання даних, а повноцінним інструментом для зваженого, обґрунтованого вибору.

Як розпізнати маніпуляції з табличними даними?

Під час аналізу таблиць варто звертати увагу на джерело даних, методику їх збору, повноту представленої інформації та об'єктивність висновків. Важливо ставити запитання: хто створював таблицю і з якою метою, чи всі важливі характеристики враховані, чи коректними є методи порівняння даних і підрахунку підсумків? Критичне мислення допомагає виявити упередження та неповну інформацію, а отже, уникнути неправильних висновків.

Розглянемо приклад. Уявімо, що виробник енергетичних напоїв створив таблицю для порівняння різних способів підняття працездатності:

Спосіб	Час застосування (хв)	Тривалість ефекту (год)	Доступність	Загальна оцінка
Енергетичний напій X	10	3	Висока	Відмінно
Кава	10	2	Середня	Добре
Фізичні вправи	45	4	Низька	Задовільно
Здоровий сон	480	8	Низька	Задовільно

На перший погляд таблиця видається об'єктивною, а енергетичний напій і справді перемагає. Однак критичний аналіз виявляє проблеми. По-перше, немає важливих критеріїв: вплив на здоров'я, вартість, побічні ефекти. По-друге, оцінка доступності фізичних вправ як «низької» сумнівна — адже рухатися можна будь-де і безкоштовно. По-третє, здоровий сон оцінено як малодоступний, хоча він доступний кожній людині щодня.

Виробник свідомо вибрав лише ті критерії, які вигідно представляють його продукт, та проігнорував негативні чинники енергетиків. Коректніша таблиця обов'язково містила б показники безпеки для здоров'я, довгострокові наслідки та вартість, що змінило б результати порівняння.

Як застосовувати відразу кілька критеріїв для вибору найкращого рішення?

Коли серед кількох об'єктів вибирають найкращий, то найчастіше аналізують значення багатьох параметрів. Результати можуть бути суперечливими: за одним параметром (критерієм) кращим буде один об'єкт, за іншим — інший. Щоб у цій ситуації все-таки зробити остаточний вибір, можна застосувати один із таких підходів:

- Вибрати серед критеріїв найважливіший, а решту проігнорувати.
- Створити формулу, що поєднує значення кількох параметрів. Наприклад, можна обчислити *середнє арифметичне зважене* їхніх значень. Припустимо, a, b і c — параметри, що оцінюються, причому a — найважливіший, а c — найменш важливий. Зіставимо параметрам вагові коефіцієнти: a — 0,5; b — 0,3; c — 0,2. Обчислимо для кожного об'єкта значення виразу $0{,}5a + 0{,}3b + 0{,}2c$ — це і буде загальна оцінка його «якості». Об'єкт з найвищою оцінкою можна вважати найкращим.

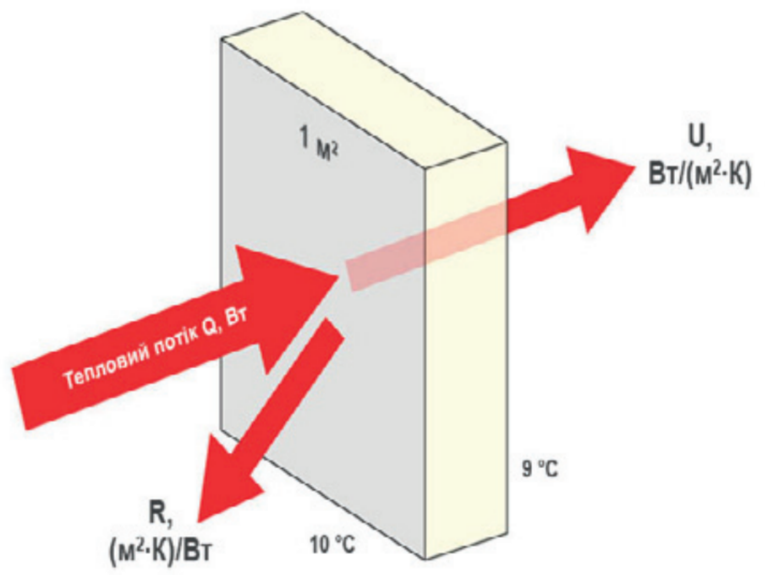
ПЕРЕВІР СЕБЕ

ЗНАННЯ — РОЗУМІННЯ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Наведи приклади даних, для яких доцільно використати табличну форму подання, а також приклади даних, для яких табличний спосіб організації є недоцільним.

2. На сайті https://youkraina.com.ua/ знайди дані дослідження, які тебе зацікавили, та оформи їх у вигляді таблиці.

3. Інженери / інженерки досліджують ефективність різних матеріалів для теплоізоляції будинків. Відомо, що коефіцієнт теплопровідності мінеральної вати становить 0,04 Вт/(м·К)[1], пінопласту — 0,038 Вт/(м·К), екструдованого пінополістиролу — 0,031 Вт/(м·К).



Розроби структуру таблиці, яка допоможе порівняти ці матеріали за комплексом характеристик, важливих для вибору утеплювача. Врахуй, що інженеру потрібно знати не лише теплоізоляційні властивості, а й економічні показники (вартість матеріалу), конструктивні особливості (товщина шару, вага), експлуатаційні характеристики (стійкість до вологи, довговічність) та екологічні чинники. Поясни, чому для вирішення інженерних завдань варто використовувати електронні таблиці.

---

[1] Коефіцієнт теплопровідності вимірюється у ватах на метр-кельвін (Вт/(м·К)). Це означає, що він показує кількість теплоти в Джоулях, яка проходить через 1 квадратний метр площі матеріалу товщиною 1 метр за 1 секунду при різниці температур на протилежних сторонах в 1 кельвін.

4. У тебе є список з 12 моделей смартфонів, доступних у трьох магазинах твого міста.

Модель	Пам'ять (ГБ)	Акумулятор (мА·г)	Ціна в магазині № 1	Ціна в магазині № 2	Ціна в магазині № 3
Xiaomi Redmi A2	64	5000	4100	3900	4050
Samsung A14	128	5000	6100	5990	6150
iPhone SE	64	2018	13 500	13 100	13 250
Realme C55	128	5000	4800	4700	4750
Poco M5	128	5000	5200	5050	5120
Motorola G13	64	5000	4450	4350	4390
Infinix Hot 12	64	6000	4750	4600	4720
Nokia G21	64	5050	4980	4800	4890
Tecno Spark 10C	128	5000	4200	4100	4190
Vivo Y16	64	5000	4600	4500	4550
Honor X6	64	5000	4700	4650	4600
Oppo A17	64	5000	5100	4950	5000

1) Створи таблицю в Excel або Google Таблицях, ввівши дані за зразком, або завантаж дані з е-додатка (1.1).
2) Для кожної моделі знайди мінімальну ціну серед трьох магазинів.
3) Обчисли показник: вартість 1000 мА·г акумулятора для мінімальної ціни. Результати розмісти в новому стовпці.
4) Визнач три моделі з найменшою ціною за 1000 мА·г (найвигідніші батареї).
5) Побудуй діаграму для візуального порівняння цін у трьох магазинах. Добери найбільш доречний тип діаграми.

5. Оцінювання моделей смартфонів.

Створи таблицю оцінювання

У таблиці нижче наведено приклад порівняння трьох моделей смартфонів за критеріями: *ціна, камера, батарея, дизайн, швидкодія.* Кожен критерій оцінено за шкалою від 1 до 10. Підсумкова оцінка — це сума всіх балів.

Модель смартфона	Ціна	Камера	Батарея	Дизайн	Швидкодія	Сума балів
A	8	6	7	7	9	37
B	6	9	8	8	7	38
C	9	5	6	6	8	34

Створи подібну таблицю з реалістичними даними для п’яти смартфонів. Можна скористатися вигаданими моделями або тими, які тобі відомі.

Подумай і проаналізуй

Проаналізуй таблицю, яку ти створив / створила, та відповідай на запитання:

1) Чи правильно обирати найкращу модель смартфона лише за сумою балів?
2) Які проблеми можуть виникнути, якщо всі критерії вважаються однаково важливими?
3) Чи можуть різні користувачі / користувачки (наприклад, блогери, школярі, бізнесмени) оцінювати важливість критеріїв по-різному?
4) Запропонуй свій варіант більш справедливої оцінки — наприклад, признач кожному критерію **вагу** (у %), що покаже його важливість, і перерахуй бали.

Порівняй інші рейтинги

Знайди в інтернеті 2–3 рейтинги смартфонів або будь-якої техніки, де є таблиці з оцінками.

1) Які критерії там використано?
2) Який принцип обчислення загальної оцінки?
3) Як визначають переможця / переможницю?
4) Чи є відчуття упередженості (наприклад, коли враховано лише дизайн або ціну)?

Обговори результати з однокласниками / однокласницями. Напиши короткий висновок.

§ 2. Сортування. Прості та розширені фільтри

Як ефективно опрацьовувати однотипні дані?

Завдання № 1

Подані відомості оформи в одну таблицю Excel.

Горошкіна Марія
Стрибки у довжину (см) — **160**
Біг на 100 м (сек) — **20**
Метання диска (см) — **550**

Сокол Андрій
Стрибки у довжину (см) — **160**
Біг на 100 м (сек) — **16**
Метання диска (см) — **602**

Зарудний Андрій
Стрибки у довжину (см) — **173**
Біг на 100 м (сек) — **17**
Метання диска (см) — **580**

Ситник Слава
Стрибки у довжину (см) — **168**
Біг на 100 м (сек) — **20**
Метання диска (см) — **650**

Дерев'янко Павло
Стрибки у довжину (см) — **168**
Біг на 100 м (сек) — **18**
Метання диска (см) — **540**

Мороз Дарина
Стрибки у довжину (см) — **180**
Біг на 100 м (сек) — **17**
Метання диска (см) — **600**

Кожух Олексій
Стрибки у довжину (см) — **172**
Біг на 100 м (сек) — **18**
Метання диска (см) — **550**

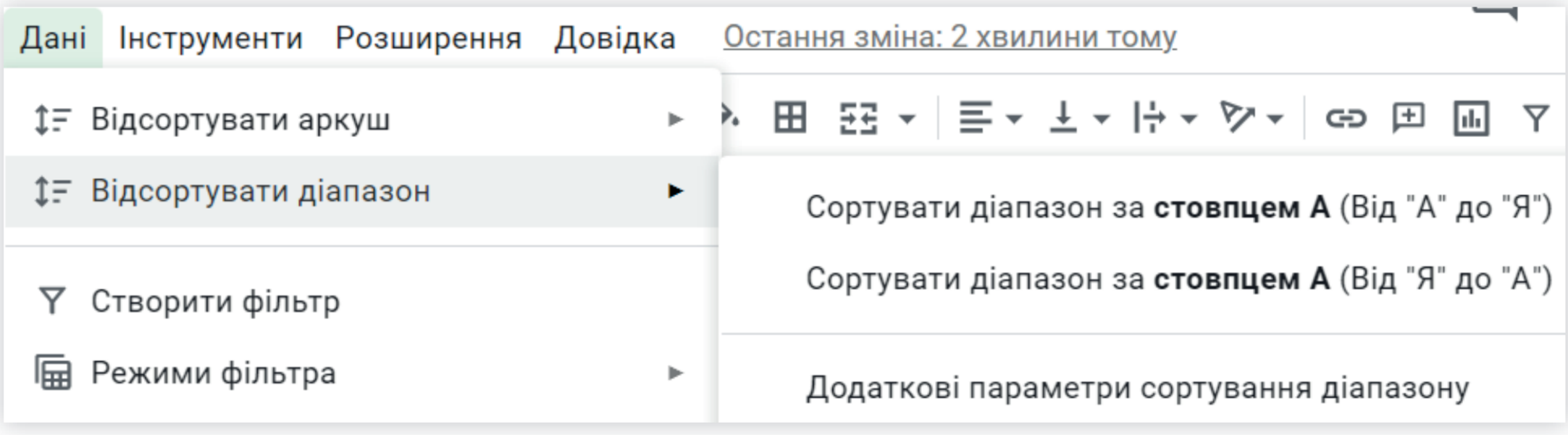
Голуб Софія
Стрибки у довжину (см) — **162**
Біг на 100 м (сек) — **19**
Метання диска (см) — **555**

- Скільки рядків у створеній таблиці? Які дані подані в кожному рядку?
- Скільки стовпців у створеній таблиці? Які дані подані в кожному стовпці?
- Доповни твердження.
  Кожен об’єкт має певний набір _____. У кожному _____ таблиці із цього завдання (крім заголовку) містяться відомості про об’єкт з такими властивостями: _____. Отже, всю _____ можна розглядати як набір відомостей про однотипні об’єкти.

Завдання № 2 Ознайомся з алгоритмами* та застосуй їх до створеної в попередньому завданні таблиці.

Алгоритм № 1

1. Обери діапазон клітинок електронної таблиці, які необхідно відсортувати.
2. Обери послідовно пункти меню «Дані» — «Відсортувати діапазон за стовпцем (напрям сортування)».



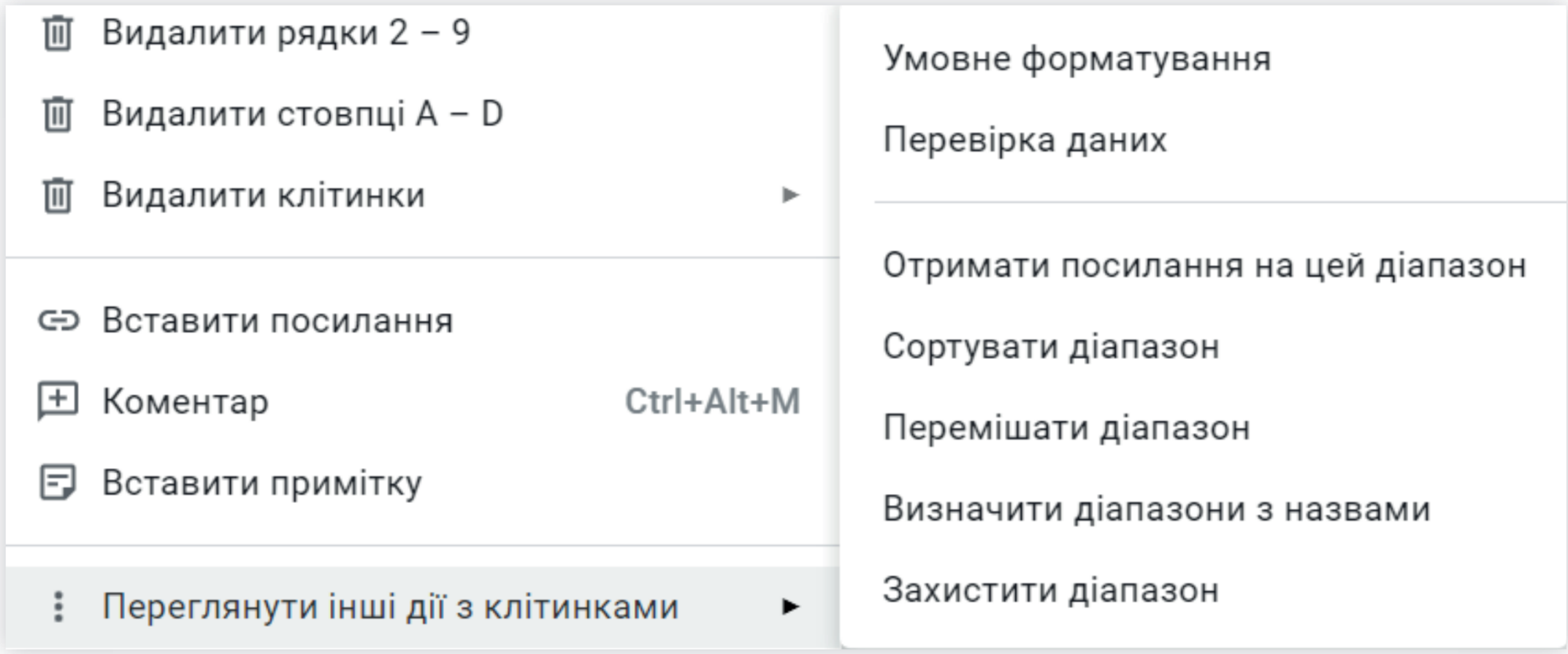
Увага!

Напрям Від «А» до «Я» — сортування за зростанням, а Від «Я» до «А» — сортування за спаданням.

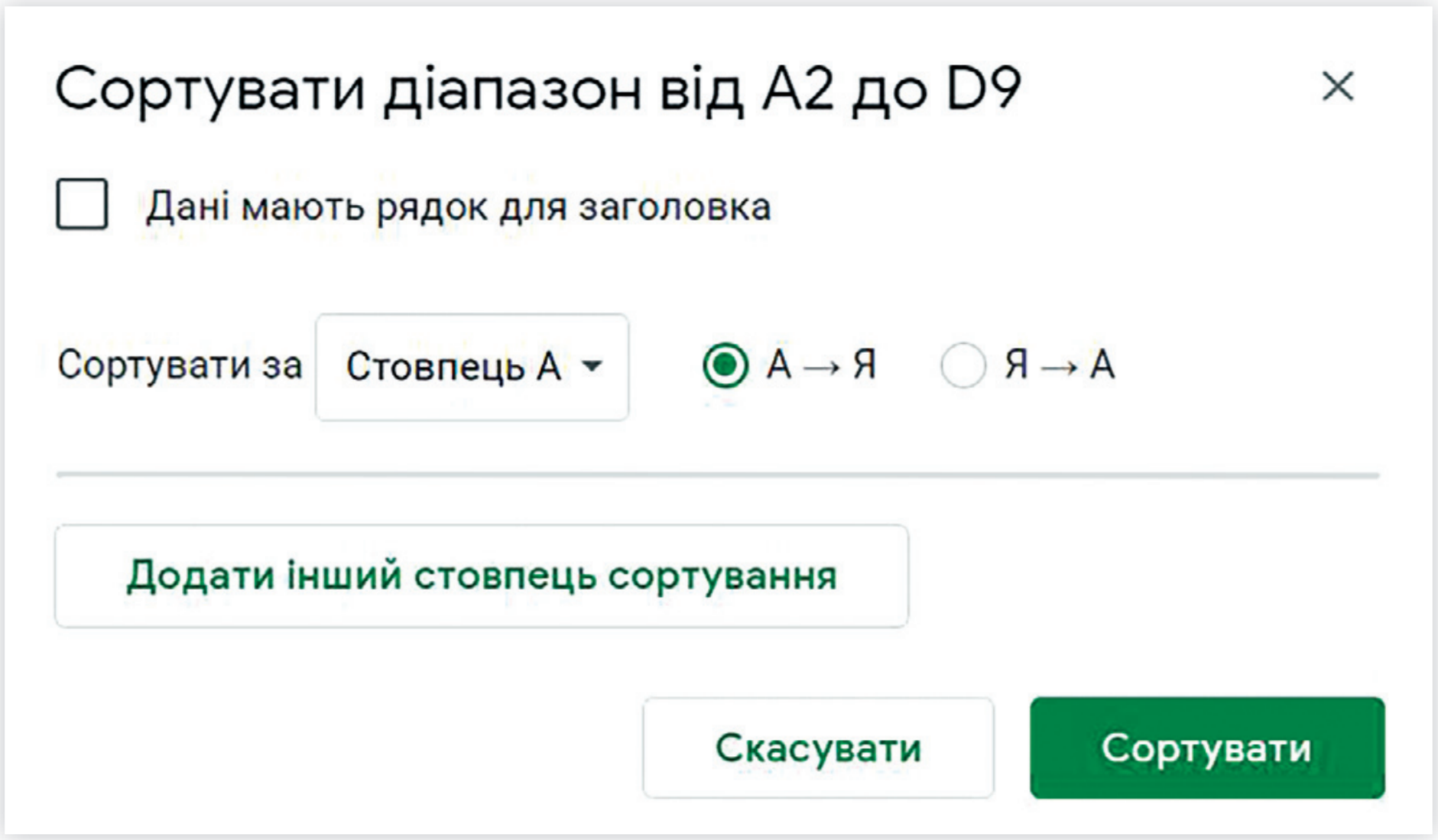
* Алгоритми дій користувача подано на прикладі Google Таблиць.

Алгоритм № 2

1. Обери діапазон клітинок електронної таблиці, який необхідно відсортувати.
2. Відкрий контекстне меню.
3. Послідовно обери пункти меню «Переглянути інші дії з клітинками» — «Сортувати діапазон».



4. Обери налаштування (стовпець, за яким відбуватиметься сортування; напрям сортування).
5. Натисни кнопку «Сортувати».



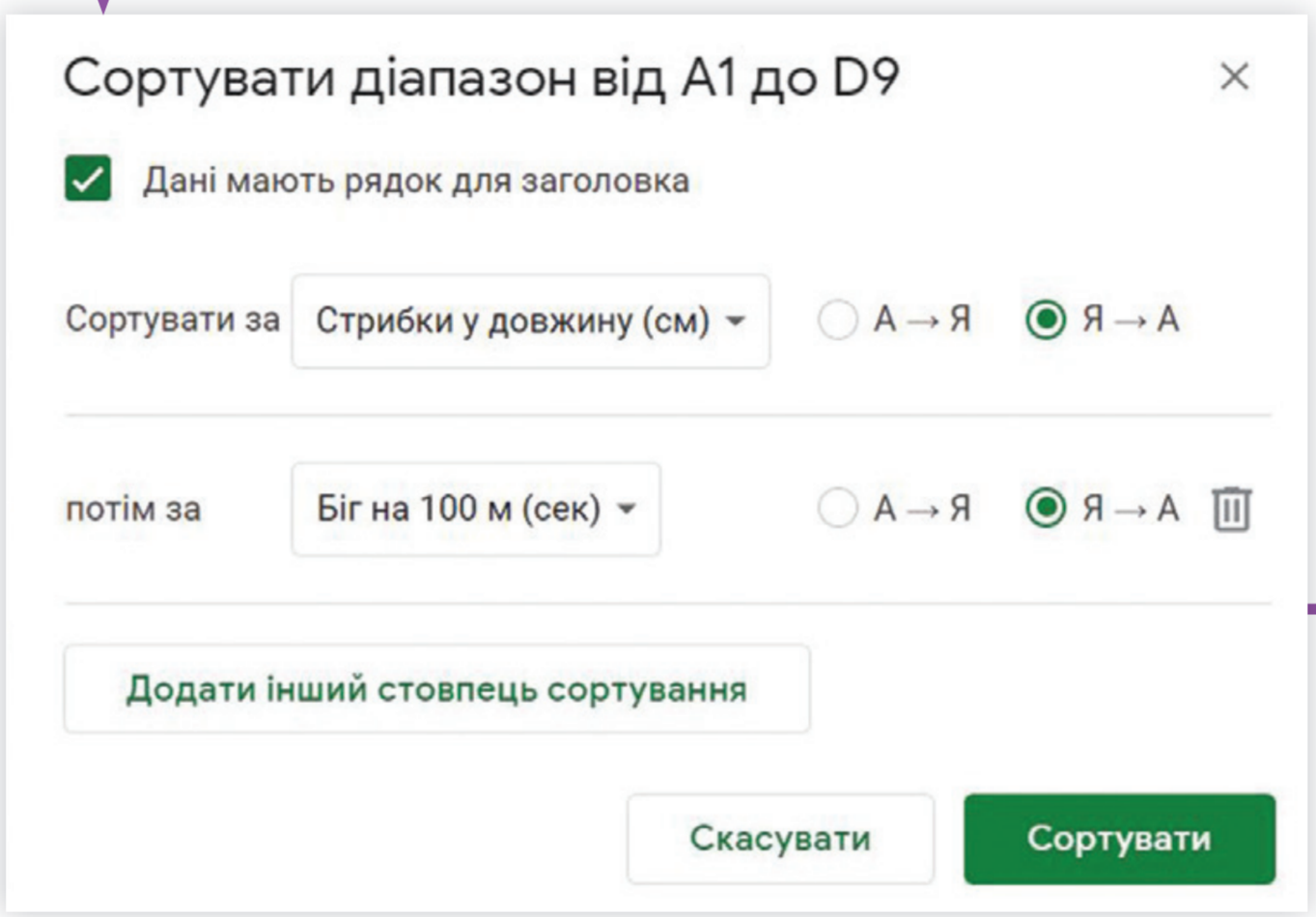
Завдання № 3

Порівняй результати двох сортувань.

Варіант № 1

БУЛО

	A	B	C	D
1	**Прізвище та ім’я**	**Стрибки у довжину**	**Біг на 100 м (сек)**	**Метання диска (см)**
2	Зарудний Андрій	173	17	580
3	Горошкіна Марія	160	20	550
4	Кожух Олексій	172	18	550
5	Мороз Дарина	180	17	600
6	Дерев’янко Павло	168	18	540
7	Сокол Андрій	160	16	602
8	Голуб Софія	162	19	555
9	Ситник Слава	168	20	650



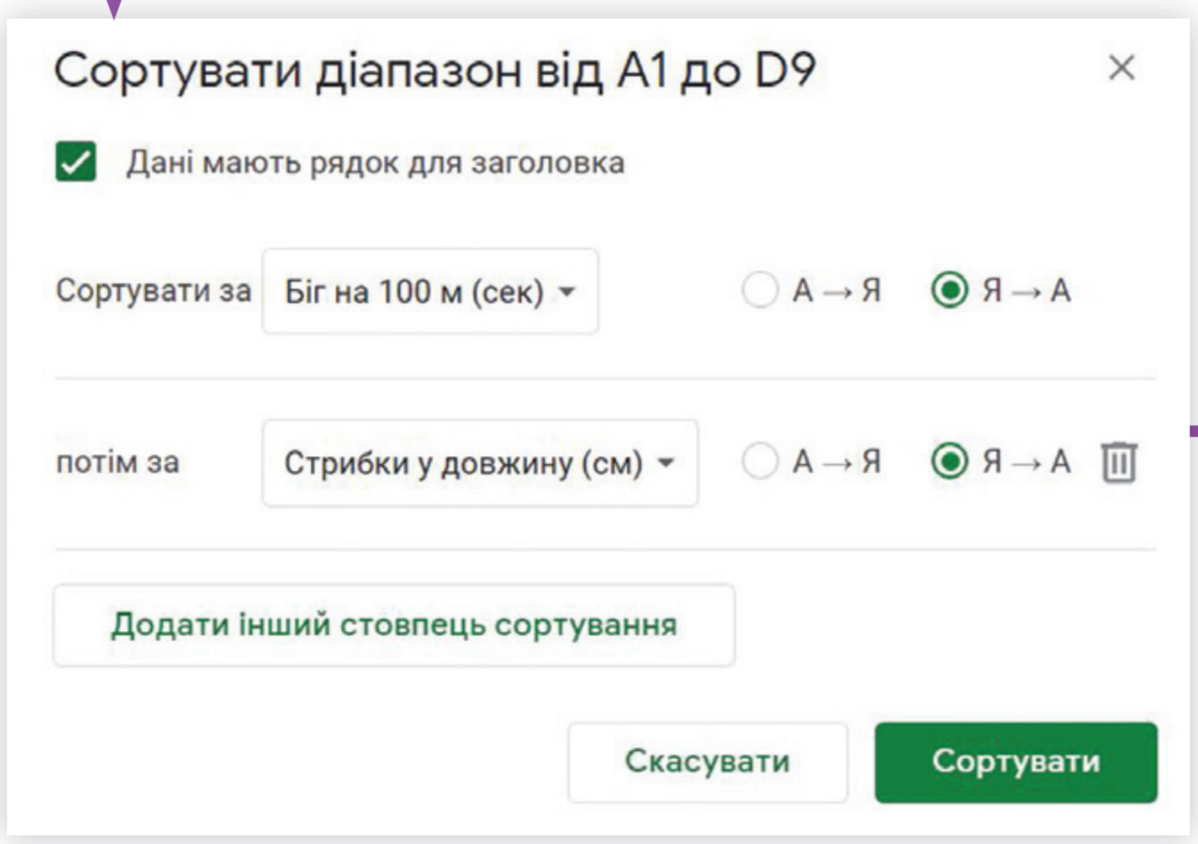
СТАЛО

	A	B	C	D
1	**Прізвище та ім’я**	**Стрибки у довжину**	**Біг на 100 м (сек)**	**Метання диска (см)**
2	Мороз Дарина	180	17	600
3	Зарудний Андрій	173	17	580
4	Кожух Олексій	172	18	550
5	Ситник Слава	168	20	650
6	Дерев’янко Павло	168	18	540
7	Голуб Софія	162	19	555
8	Горошкіна Марія	160	20	550
9	Сокол Андрій	160	16	602

Варіант № 2

БУЛО

	A	B	C	D
1	Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
2	Зарудний Андрій	173	17	580
3	Горошкіна Марія	160	20	550
4	Кожух Олексій	172	18	550
5	Мороз Дарина	180	17	600
6	Дерев’янко Павло	168	18	540
7	Сокол Андрій	160	16	602
8	Голуб Софія	162	19	555
9	Ситник Слава	168	20	650



СТАЛО

	A	B	C	D
1	Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
2	Ситник Слава	168	20	650
3	Горошкіна Марія	160	20	550
4	Голуб Софія	162	19	555
5	Кожух Олексій	172	18	550
6	Дерев’янко Павло	168	18	540
7	Мороз Дарина	180	17	600
8	Зарудний Андрій	173	17	580
9	Сокол Андрій	160	16	602

- Чому різняться результати?
- Опиши, як саме комп’ютер виконує багаторівневе сортування.
- Запропонуй свій підхід для визначення переможця / переможниці змагань. Продемонструй цей підхід, виконавши відповідні дії в електронній таблиці.

Завдання № 4

Уяви, що ти входиш до складу журі спортивного конкурсу.

Перед тобою — результати змагань учасників у трьох видах спорту: стрибки у довжину, біг на 100 м (сек) та метання диска (см). Журі оцінює результати так:

- У кожному виді спорту учасники / учасниці впорядковуються за результатами: перше місце отримує найкращий результат, восьме — найгірший.
- Потім усі місця додаються, і учасник з найменшою сумою посідає перше місце.

Твоє завдання — виконати запропоновані дії та знайти відповіді на запитання:

1. Розглянь таблицю з початковими результатами (таблиця 1).
2. Проаналізуй таблицю 2 — як змінено дані?
3. Проаналізуй таблиці 3, 4, 5 і 6 — вони показують проміжні етапи сортування:
   - За якими критеріями відбувалося сортування?
   - Чи змінюється порядок залежно від того, який показник урахову-ється першим?
   - Як це впливає на загальний результат?

1

Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
Мороз Дарина	180	17	600
Зарудний Андрій	173	17	580
Кожух Олексій	172	18	550
Ситник Слава	168	20	650
Дерев’янко Павло	168	18	540
Голуб Софія	162	19	555
Горошкіна Марія	160	20	550
Сокол Андрій	160	16	602

2

Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
Мороз Дарина	1	17	600
Зарудний Андрій	2	17	580
Кожух Олексій	3	18	550
Ситник Слава	4	20	650
Дерев’янко Павло	5	18	540
Голуб Софія	6	19	555
Горошкіна Марія	7	20	550
Сокол Андрій	8	16	602

3

Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
Сокол Андрій	8	16	602
Мороз Дарина	1	17	600
Зарудний Андрій	2	17	580
Кожух Олексій	3	18	550
Дерев’янко Павло	5	18	540
Голуб Софія	6	19	555
Ситник Слава	4	20	650
Горошкіна Марія	7	20	550

4

Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
Сокол Андрій	8	1	602
Мороз Дарина	1	2	600
Зарудний Андрій	2	3	580
Кожух Олексій	3	4	550
Дерев’янко Павло	5	5	540
Голуб Софія	6	6	555
Ситник Слава	4	7	650
Горошкіна Марія	7	8	550

5

Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
Ситник Слава	4	7	650
Сокол Андрій	8	1	602
Мороз Дарина	1	2	600
Зарудний Андрій	2	3	580
Голуб Софія	6	6	555
Кожух Олексій	3	4	550
Горошкіна Марія	7	8	550
Дерев’янко Павло	5	5	540

6

Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)
Ситник Слава	4	7	1
Сокол Андрій	8	1	2
Мороз Дарина	1	2	3
Зарудний Андрій	2	3	4
Голуб Софія	6	6	5
Кожух Олексій	3	4	6
Горошкіна Марія	7	8	7
Дерев’янко Павло	5	5	8

4. Створи таку таблицю. Виконай дії над даними, що відповідають змінам у таблицях **2** – **6**.
   - Порахуй суму місць кожного учасника, відсортуй їх і признач місця.

7

Прізвище та ім’я	Стрибки у довжину	Біг на 100 м (сек)	Метання диска (см)	Сумарний результат	Місце
Ситник Слава	4	7	1	12	
Сокол Андрій	8	1	2	11	
Мороз Дарина	1	2	3	6	
Зарудний Андрій	2	3	4	9	
Голуб Софія	6	6	5	17	
Кожух Олексій	3	4	6	13	
Горошкіна Марія	7	8	7	22	
Дерев’янко Павло	5	5	8	18	

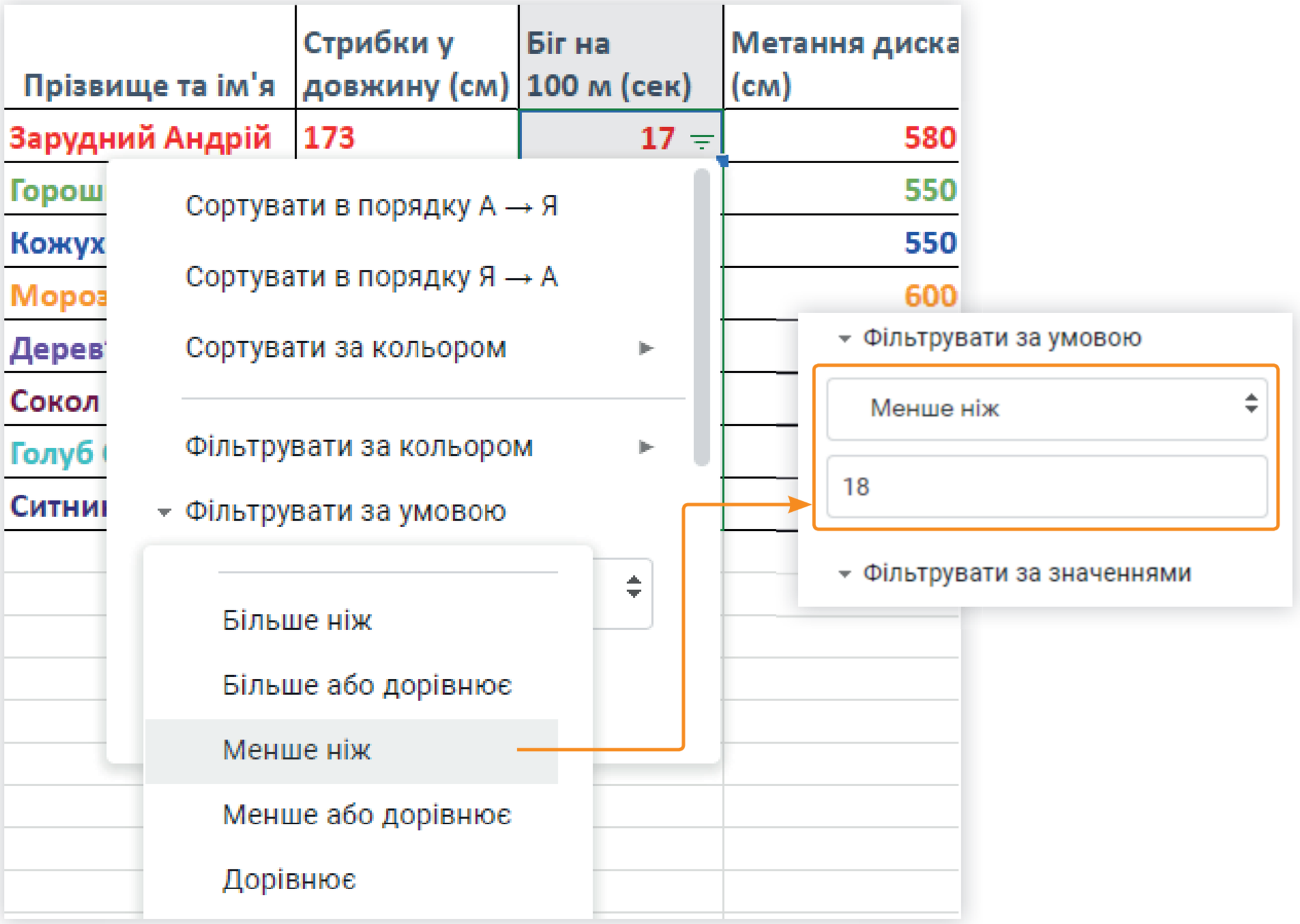
5. Підготуй відповіді на запитання:
   - Чи є така система підсумовування місць справедливою?
   - Чи відображає вона реальний рівень досягнень?
   - Як би ти пояснив / пояснила, чому один учасник з майже рівними результатами отримав суттєво інше місце?
6. Запропонуй інший підхід до визначення переможця / переможниці. Наприклад:
   - Використай реальні результати, а не лише місця.
   - Признач вагу кожному виду спорту (наприклад: біг важливіший — 50 %, метання — 25 %...).
   - Запропонуй власну формулу оцінювання.
   - Обґрунтуй, чому твій підхід точніше відображає успіх спортсменів.
7. Випробуй свій підхід на тих самих даних (електронна таблиця або обчислення вручну). Покажи, як зміняться підсумки.

Підказка:

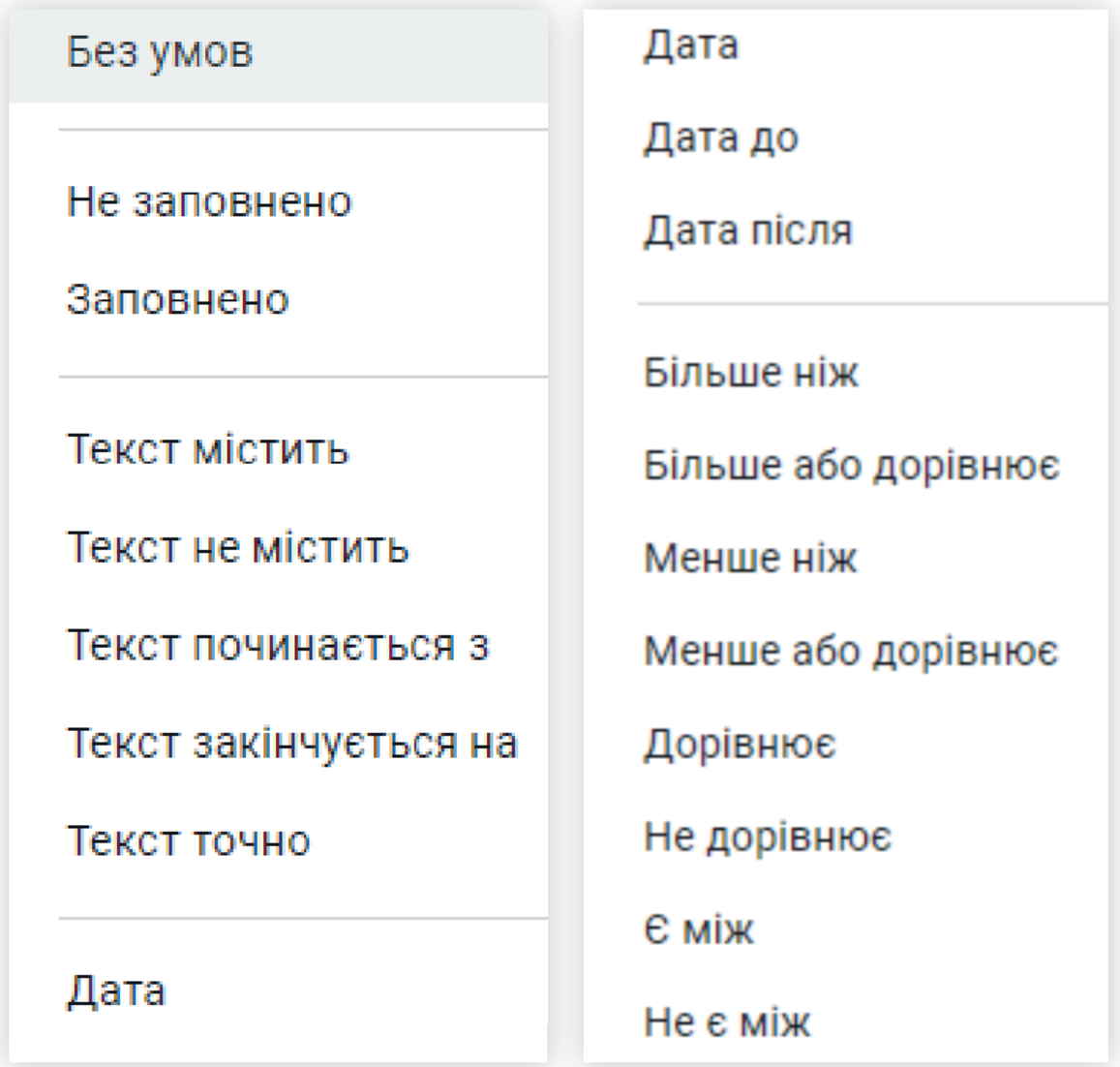
Подумай, чи однаково справедливо оцінювати учасників, якщо між їхніми результатами дуже велика або дуже мала різниця. Наприклад, стрибки на 179 см і 180 см — це майже однаково, але місця можуть бути друге і перше. Як урахувати цю деталь?

Завдання № 5

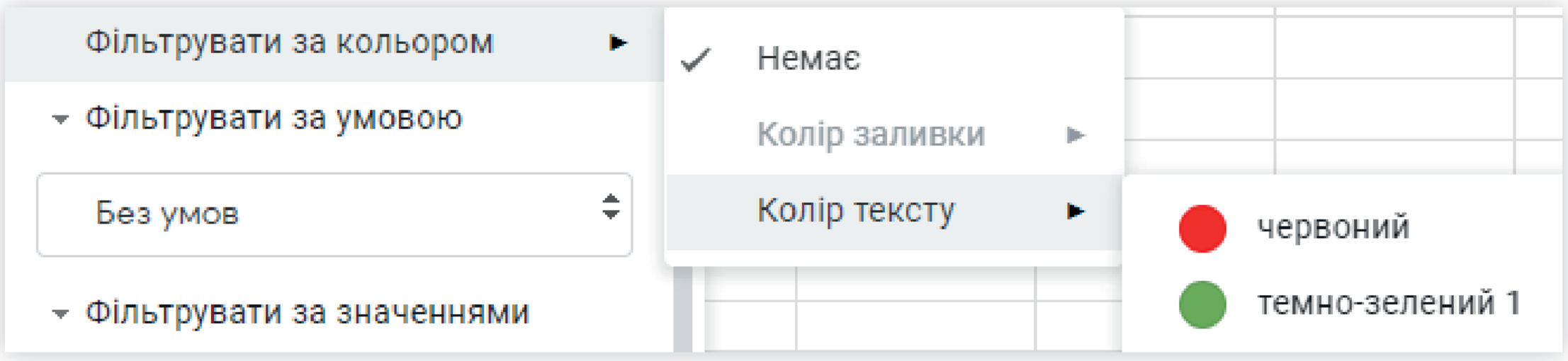
Розглянь фрагменти копій екрана (с. 25). Поясни, чим операція сортування відрізняється від операції фільтрування.



- Яким буде результат застосування зображеної дії до початкової таблиці спортивних досягнень?
- Наведи приклади використання поданих нижче умов фільтрування даних.



- Поясни, як працює поданий фільтр.



- Наведи приклади процесів фільтрації у житті.

Завдання № 6

Об’єднайтесь у пари, створіть чотири копії таблиці спортивних досягнень, а в них відберіть такі дані:

- учні / учениці, які пробігли 100 метрів менше ніж за 18 секунд;
- учні / учениці, які стрибнули далі ніж 1 метр 60 см;
- учні / учениці, які пробігли менше ніж за 18 секунд і стрибнули більше ніж на 1 метр 60 см;
- учні / учениці, які пробігли менше ніж за 18 секунд або стрибнули більше ніж на 1 метр 60 см.

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке сортування даних?

Сортування в табличному процесорі — це вбудована функція, за допомогою якої можна розташувати рядки таблиці в зручному порядку для їх подальшого аналізу.

Можна відсортувати дані в алфавітному порядку, а також за зростанням або спаданням значень, числових даних або дат.

Що таке багаторівневе сортування?

Інколи рядки таблиці потрібно впорядкувати за значеннями кількох стовпців одразу. Наприклад, є таблиця з даними про всіх учнів школи, а ми хочемо розташувати в алфавітному порядку прізвища учнів кожного класу окремо. Тоді таблицю потрібно відсортувати:

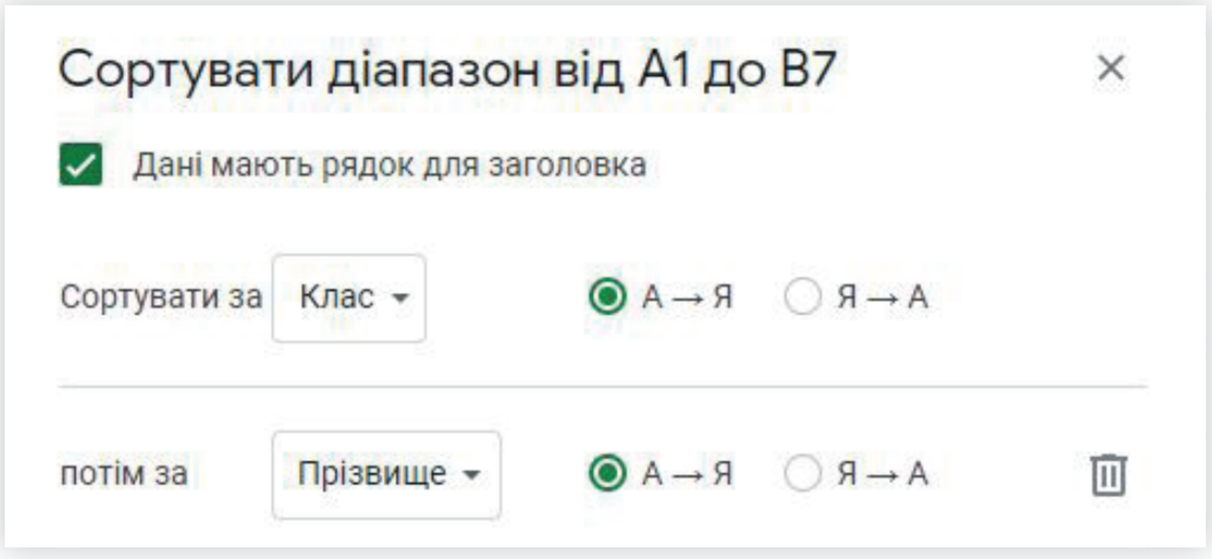
1) за класами; 2) за прізвищами учнів усередині кожного класу.

Це завдання можна виконати за допомогою однієї операції багаторівневого сортування.

Клас	Прізвище
9А	Арбузов Микола
9Б	Амосов Петро
9А	Беззубий Іван
9А	Дорошенко Петро
9Б	Березюк Дарина
9Б	Гусь Микита

→

Клас	Прізвище
9А	Арбузов Микола
9А	Беззубий Іван
9А	Дорошенко Петро
9Б	Амосов Петро
9Б	Березюк Дарина
9Б	Гусь Микита

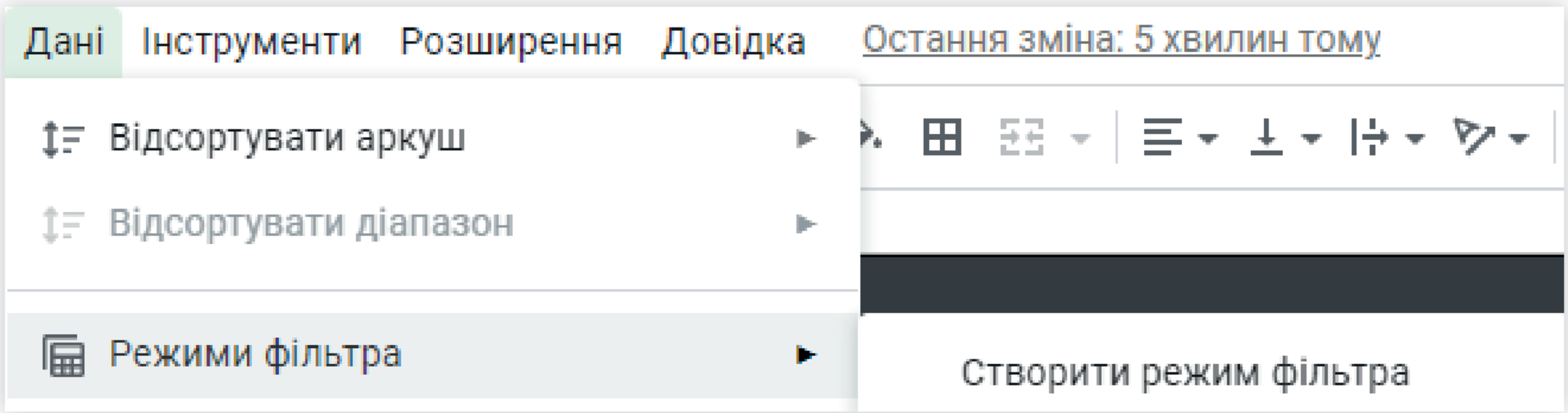


Що таке фільтрація даних?

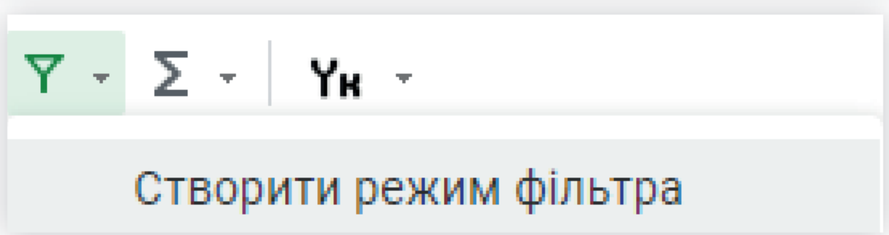
Фільтрація — це відбір тих рядків таблиці, значення у яких задовольняють певним умовам. У табличному процесорі фільтри бувають двох типів: прості (для простих умов) та розширені (для складніших умов).

Алгоритм створення простого фільтра:

1. Зробити поточною будь-яку клітинку електронної таблиці.
2. Увімкнути режим фільтрування.



або



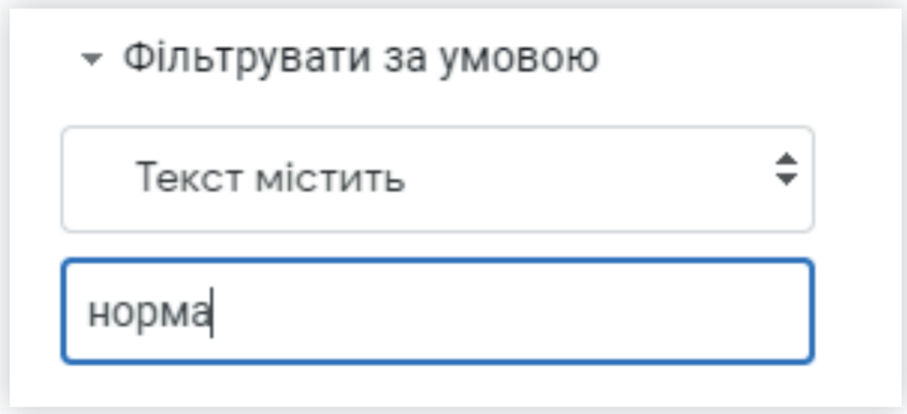
3. Обрати умову фільтрування.

Прості фільтри бувають такими:

- текстові (наприклад, чи містить текст клітинки певний рядок);
- числові (більше, менше, дорівнює, не дорівнює тощо);
- фільтри за датою (чи належить дата в клітинці певному часовому проміжку).

Приклад № 1

Прізвище та ім’я	Стрибки у довжину (см)	Стан виконання нормативу
Ситник Слава	168	норма
Горошкіна Марія	160	нижче норми
Голуб Софія	162	нижче норми
Кожух Олексій	172	норма
Дерев’янко Павло	165	нижче норми
Мороз Дарина	180	норма
Зарудний Андрій	173	норма
Сокол Андрій	160	нижче норми

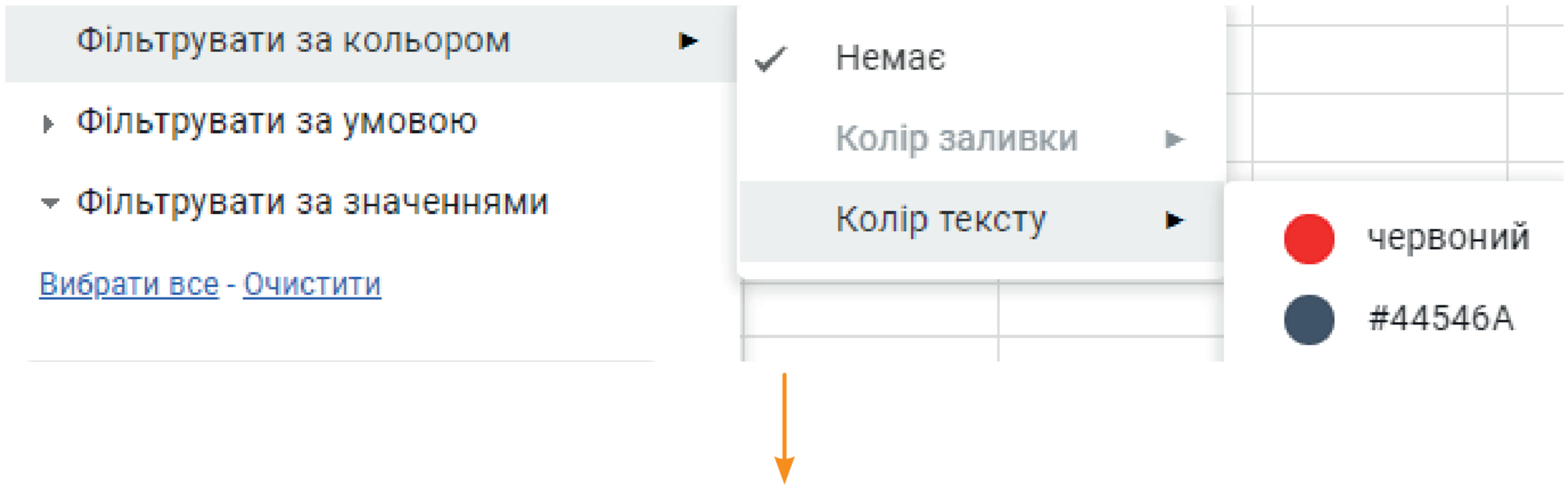


Прізвище та ім’я	Стрибки у довжину (см)	Стан виконання нормативу
Ситник Слава	168	норма
Кожух Олексій	172	норма
Мороз Дарина	180	норма
Зарудний Андрій	173	норма

Позначка, яка вказує, що до даних цього стовпця застосовано фільтр

Приклад № 2

Прізвище та ім’я	Стрибки у довжину (см)	Стан виконання нормативу
Ситник Слава	168	**норма**
Горошкіна Марія	160	нижче норми
Голуб Софія	162	нижче норми
Кожух Олексій	172	**норма**
Дерев’янко Павло	165	нижче норми
Мороз Дарина	180	**норма**
Зарудний Андрій	173	**норма**
Сокол Андрій	160	нижче норми



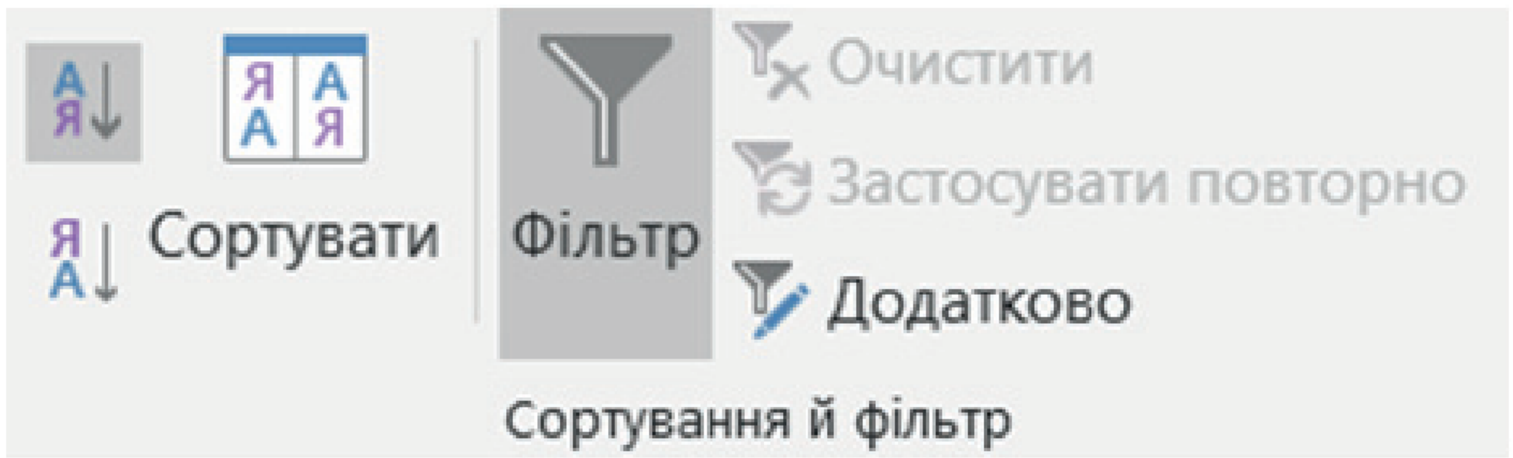
Прізвище та ім’я	Стрибки у довжину (см)	Стан виконання нормативу
Ситник Слава	168	**норма**
Кожух Олексій	172	**норма**
Мороз Дарина	180	**норма**
Зарудний Андрій	173	**норма**

Фільтрацію можна здійснювати послідовно для кількох стовпців електронної таблиці. Тоді кожний наступний фільтр буде застосований уже тільки для тих рядків таблиці, які відображаються після застосування всіх попередніх фільтрів.

Що таке розширений фільтр і коли його використовують?

Якщо в умові відбору рядків таблиці фігурує сполучник «або», то за допомогою простих фільтрів її реалізувати неможливо. Тоді стане в пригоді розширений фільтр.

На жаль, такої функції в Google Таблицях поки ще немає. У Microsoft Excel застосувати розширений фільтр можна за допомогою кнопки «Додатково» в області «Сортування й фільтр».



Щоб застосувати розширений фільтр, необхідно побудувати додаткову таблицю з критеріями відбору, яку ще називають діапазоном умов.

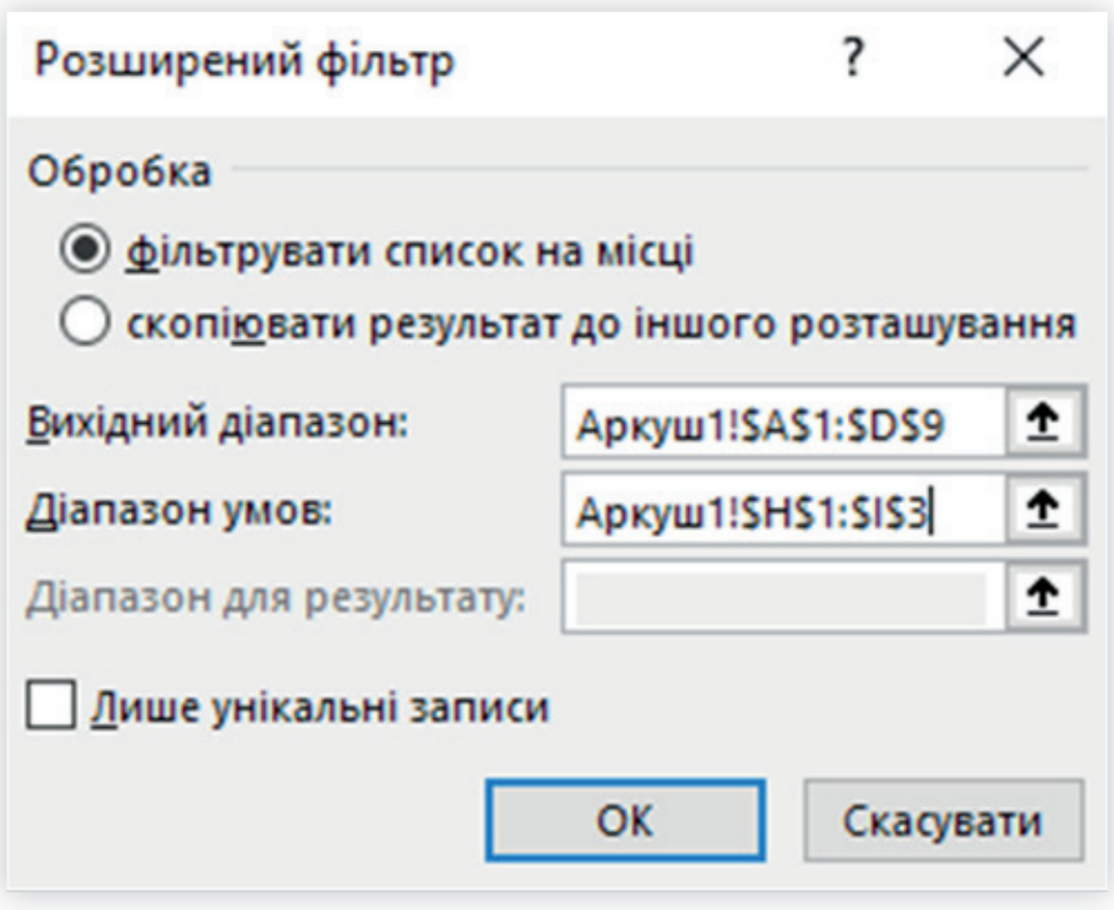
Діапазон умов створюють так:

- у перший рядок копіюють назви стовпців, за якими відбуватиметься фільтрація;
- умови, з'єднані сполучником «і», записують в одному рядку;
- умови, з'єднані сполучником «або», записують у різних рядках.

Наприклад, ось діапазон умов для відбору учнів, які пробігли менше ніж за 17 секунд або пробігли менше ніж за 20 секунд і при цьому метнули диск більше ніж на 550 сантиметрів:

Біг на 100 м (сек)	Метання диска (см)
<17	
<20	>550

Після того, як діапазон умов сформовано, потрібно натиснути кнопку «Додатково» і у вікні «Розширений фільтр» вказати діапазон, де розміщено фільтровану таблицю (вихідний діапазон) та діапазон умов.



ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Укажи, для вирішення яких завдань можна використати табличний процесор:
   а) ведення домашньої бухгалтерії;
   б) написання електронного листа другу;
   в) побудови графічного плану своєї квартири;
   г) ведення електронного класного журналу.
2. Наведи приклади завдань, які можна зручно вирішити за допомого вбудованих в Excel функцій сортування та фільтрування даних.
3. Скориставшись мережею «Інтернет», заповни таблицю даними.

Винахід	Рік винаходу	Винахідник
Спектральне розширення частот (основа Wi-Fi і Bluetooth)		
Комп’ютерна миша		
Друкарська машинка		
Пральна машина		
Шрифт Брайля		

Відсортуй дані в таблиці за власними критеріями.

4. Розглянь таблицю, до якої було застосовано багаторівневе сортування. Дай відповіді на запитання:
   - Відомості про які об’єкти містяться в таблиці?
   - Дані якого типу містяться в кожному стовпці?

Назва тістечка	Вартість	Вага
Кошик	27 грн	200
Ягідне	27 грн	180
Бісквітне	35 грн	200
Шоколадне	40 грн	180
Бізе	40 грн	120

   - За яким критерієм було відібрано тістечко (другий рядок таблиці)?

5. Відкрий е-додаток (2.1).
Завантаж файл **«Застосовуємо.xlsx»** та виконай завдання на кожному аркуші.

6. Створи подану таблицю.

	A	B	C	D	E	F	G
1	Початок сеансу	Кінотеатр	Назва фільму	Жанр	Ціна	Кількість	Всього
2	13:00	Оскар	Фокстер і Макс (2019)	фантастика	150	85	12750
3	10:00	Зірка	Клара та Чарівний Дракон (2019)	мультфільм	170	150	25500
4	16:00	Оскар	Тільки диво (2019)	фентезі	200	45	9000
5	13:00	Планета кіно	11 дітей з Моршина (2019)	комедія	150	7	1050
6	16:00	Зірка	Тільки диво (2019)	фентезі	200	12	2400
7	10:00	Зірка	Викрадена принцеса: Руслан і Людмила (2018)	мультфільм	120	78	9360
8	20:00	Оскар	Пекельна Хоругва, або Різдво Козацьке (2019)	фільм-казка	100	50	5000
9	16:00	Планета кіно	11 дітей з Моршина (2019)	комедія	150	120	18000
10	20:00	Зірка	Викрадена принцеса: Руслан і Людмила (2018)	мультфільм	120	90	10800

Виконай дії:

- Відсортуй фільми від найменшої ціни за один сеанс до найбільшої.
- Відсортуй фільми за кінотеатрами, а в кожному кінотеатрі — за часом показу, від найранішого до найпізнішого.
- Відбери:
  — сеанси, квитки на які коштують менше 170 грн;
  — фільми фентезі;
  — мультфільми, квитки на які коштують менше 150 грн;
  — фільми, квитки на які або коштують менше 150 грн, або продаються в кількості більше 50.
- Об’єднайтесь у пари. Ставте по черзі один одному завдання щодо сортування та фільтрування даних та виконуйте їх.

7. Аргументуй, чому структуровані у вигляді таблиць дані зручніші для опрацювання.

ЗАКОДУЄМО?

Склади алгоритм сортування даних у лінійній таблиці. Запрограмуй у будь-якому середовищі програмування та презентуй своє рішення.

§ 3. Візуальний аналіз даних

Як унаочнити дані для швидкого прийняття рішень?

Уяви, що ти аналізуєш різні показники успішності своєї команди в чемпіонаті. Як швидко виділити найкращі та найгірші значення серед сотень чисел? Як одним поглядом зрозуміти, де потрібна увага, а де все добре? Сучасні табличні процесори дають змогу не просто зберігати дані, а робити їх наочними — автоматично виділяти важливу інформацію кольорами, символами та візуальними ефектами. Умовне форматування перетворює звичайну таблицю на потужний інструмент візуального аналізу, який допомагає миттєво помічати закономірності, тенденції та проблемні місця.

Завдання № 1

Створи «розумну» таблицю для планування особистого часу протягом тижня, що містить відомості про час, який ти витрачаєш щодня на навчання, розваги, спорт, відпочинок, сон тощо. У таблиці мають автоматично:

- відображатися позначки щодо порушення режиму сну (менше 8 годин);
- виділятися дні з перевантаженням (понад 10 годин активності) та недостатньою активністю (менше 6 годин активності).

Завдання № 2

ІТ-компанія з України створила мобільний додаток та відстежує його рейтинги в різних країнах. Результати такі: Україна — 4,8; Польща — 4,6; Німеччина — 4,2; США — 4,0; Індія — 3,8; Бразилія — 4,4.

Створи таблицю з додатковими показниками — кількість завантажень та населення країни. Дані першого показника введи на власний розсуд, а другого — дізнайся з інтернету. Дізнайся з рубрики «Запитання-відповіді» про умовне форматування та застосуй:

- кольорові шкали для візуалізації рейтингів (від червоного до зеленого);
- гістограми для відображення кількості завантажень;

- піктограми для позначення ринків з високим, середнім та низьким потенціалом. Потенціал ринку визнач як відношення кількості завантажень до кількості населення країни. Що нижче це відношення, то вищий потенціал ринку.

Завдання № 3

Відкрий вебсторінку (е-додаток (3.1)), дізнайся про десять українських стартапів з найбільшими обсягами залучених у 2024 році інвестицій. Створи таблицю зі стовпцями назв стартапів та обсягів інвестицій. За допомогою умовного форматування виділи назви компаній, які залучили понад 5 мільйонів доларів інвестицій.

Завдання № 4

Школа проводить енергоаудит та вимірює споживання електроенергії в різних кабінетах протягом тижня. Потрібно виявити приміщення з найвищим та найнижчим споживанням.

Створи таблицю з даними про споживання електроенергії у 4–5 кабінетах протягом усіх робочих днів тижня. Введи дані згідно зі своїми припущеннями про енергоспоживання у твоїй школі. Застосуй умовне форматування, щоб:

- виділити червоним номери трьох кабінетів, що витратили найбільше електроенергії;
- за допомогою кольорових шкал створити «теплову карту» для візуалізації споживання в кожному кабінеті по днях тижня.

Завдання № 5

За допомогою умовного форматування визнач усі прості числа в діапазоні від 10 до 100.

1) За допомогою автозаповнення введи послідовність цілих чисел від 10 до 100.

2) Виділи введений діапазон і створи правило умовного форматування, яке виділятиме червоним числа, що не діляться на жодне із чисел 2, 3, 5, 7. Скористайся рубрикою «Запитання-відповіді», щоб дізнатися, як створити правило.

Простим називається число, що ділиться націло лише саме на себе та на одиницю. Щоб дізнатися, чи є простим число x, достатньо перевірити, чи ділиться воно хоча б на одне з простих чисел, що не перевищують $\sqrt{x}$. Отже, число від 10 до 100 є простим, якщо воно не ділиться на 2, 3, 5 або 7. Щоб перевірити, чи ділиться число a на число b, можна використати функцію MOD (остача від ділення): вираз MOD(a;b)<>0 буде істинним, якщо a не ділиться на b (тобто остача від ділення a на b не дорівнює 0).

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке умовне форматування і навіщо воно потрібне?

Умовне форматування — це автоматичне застосування візуальних ефектів (кольорів, шрифтів, границь, піктограм) до клітинок таблиці залежно від їхнього вмісту або заданих умов. Воно перетворює звичайну таблицю на потужний інструмент візуального аналізу, дозволяючи миттєво помічати тенденції, виділяти екстремальні значення та приймати швидкі рішення на основі даних.

Які основні типи умовного форматування існують?

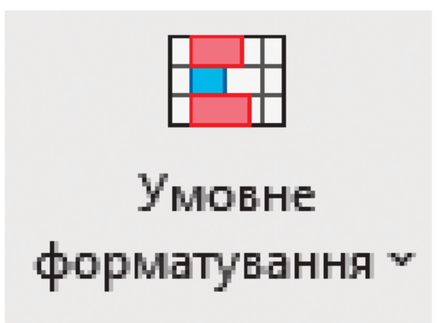
Основні типи включають правила виділення клітинок (більше, менше, між значеннями, рівно), правила першого та останнього значень (найбільші, найменші, вище середнього), гістограми (горизонтальні смужки всередині клітинок), кольорові шкали (градієнт від одного кольору до іншого) та набори піктограм (стрілки, світлофори, зірки). Кожен тип вирішує різні завдання візуалізації даних.

Як правильно вибрати колір для умовного форматування?

При виборі кольорів варто враховувати їх психологічний вплив та загальноприйняті асоціації: зелений зазвичай означає позитивні результати, червоний — проблеми або небезпеку, жовтий — попередження або нейтральний стан. Важливо забезпечити достатній контраст для читабельності та врахувати потреби людей з порушеннями кольорового зору, використовуючи додаткові візуальні елементи.

Як виконати умовне форматування?

Потрібно виділити діапазон з даними, на вкладці **Основне** натиснути кнопку **Умовне форматування** і вибрати з меню правило форматування.

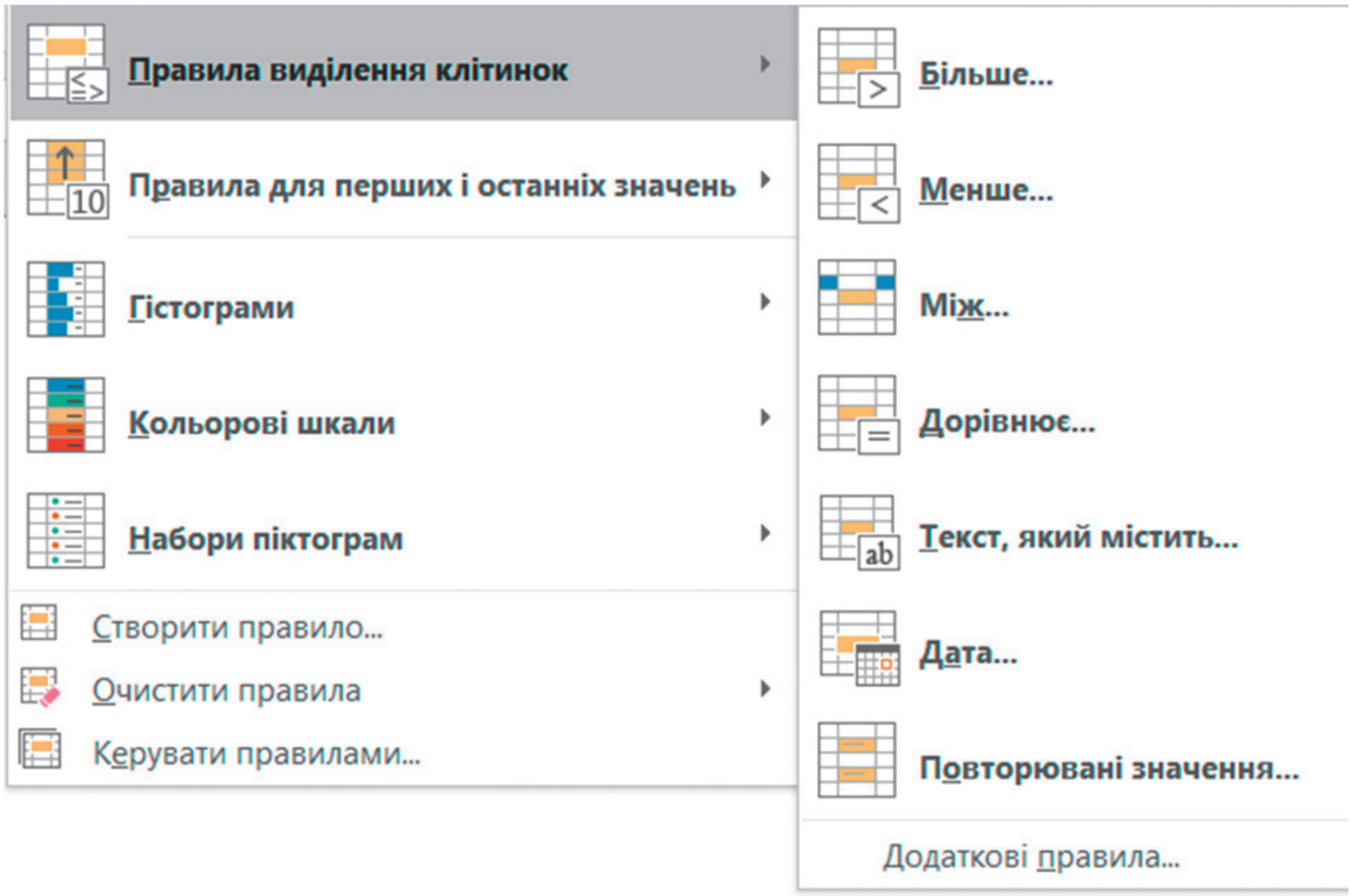


Які є види умовного форматування?

Усі сценарії умовного форматування поділено на такі категорії:

- правило виділення клітинок;
- правило відбору перших і останніх значень;
- гістограми;
- кольорові шкали;
- набори піктограм.

Правила виділення клітинок застосовують до тих клітинок, які порівнюються з певним значенням.



Більше... Якщо значення клітинки, до якої застосовується правило виділення, більше заданої величини, то починає діяти зазначений формат.

Менше... Форматує клітинки, у яких значення менше заданого порога.

Між... Форматування застосовується, якщо вміст клітинки перебуває у заданих межах.

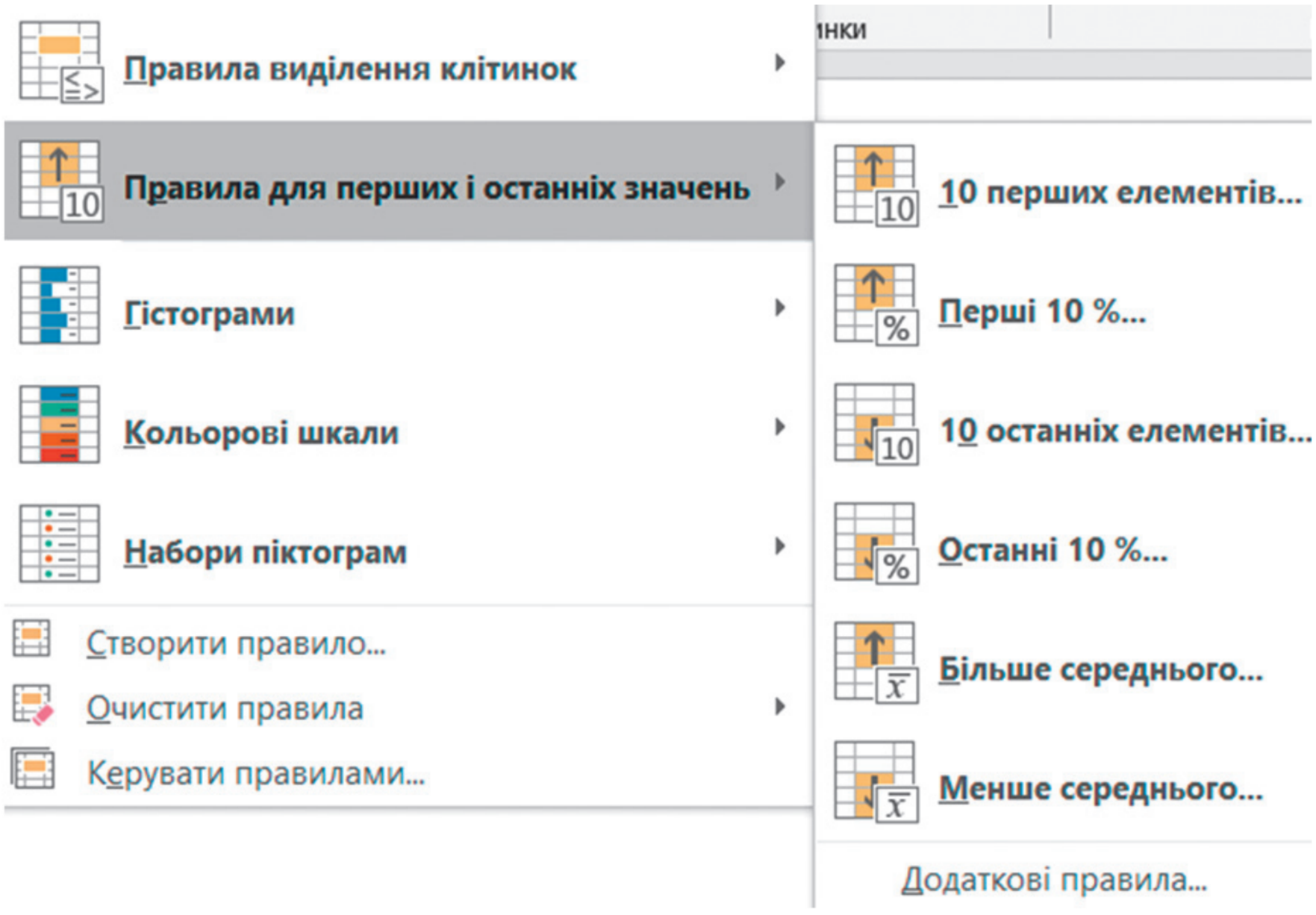
Дорівнює... Якщо значення або текст у клітинці збігається з умовою.

Текст, який містить... Якщо збігається тільки частина тексту (слово, код, комбінація символів тощо).

Дата... Можливість форматувати періоди часу, наприклад: сьогодні, вчора, останні сім днів, наступний місяць тощо.

Повторювані значення... Виділяються клітинки з однаковим умістом. Хороший спосіб знайти дублікати (повтори). У налаштуваннях можна вибрати й обернений варіант — виділити тільки унікальні значення.

Правила відбору перших і останніх значень виділяють найбільші або найменші значення. Допомагають аналізувати дані, показуючи пріоритети і «слабкі місця».



10 перших елементів... Форматується певна кількість найбільших значень (можна задати 5, 20 тощо).

Перші 10 %... Виділяються 10 % максимальних значень. Частку можна змінити.

10 останніх елементів... Форматується певна кількість найменших значень.

Останні 10 %... Найменші 10 % або інша частка від усіх елементів.

Більше (менше) середнього... Форматуються всі значення, які більші (менші) за середнє арифметичне.

Гістограми дають можливість у кожну клітинку із числом додати стовпець лінійної гістограми, розмір якого визначається щодо максимального значення у виділеному діапазоні.

Це допомагає візуалізувати невеликий набір даних без використання окремих діаграм.

Прізвище та ім'я	Стрибки у довжину (см)
Ситник Слава	168
Горошкіна Марія	160
Голуб Софія	162
Кожух Олексій	172
Дерев’янко Павло	165
Мороз Дарина	180
Зарудний Андрій	173
Сокол Андрій	160

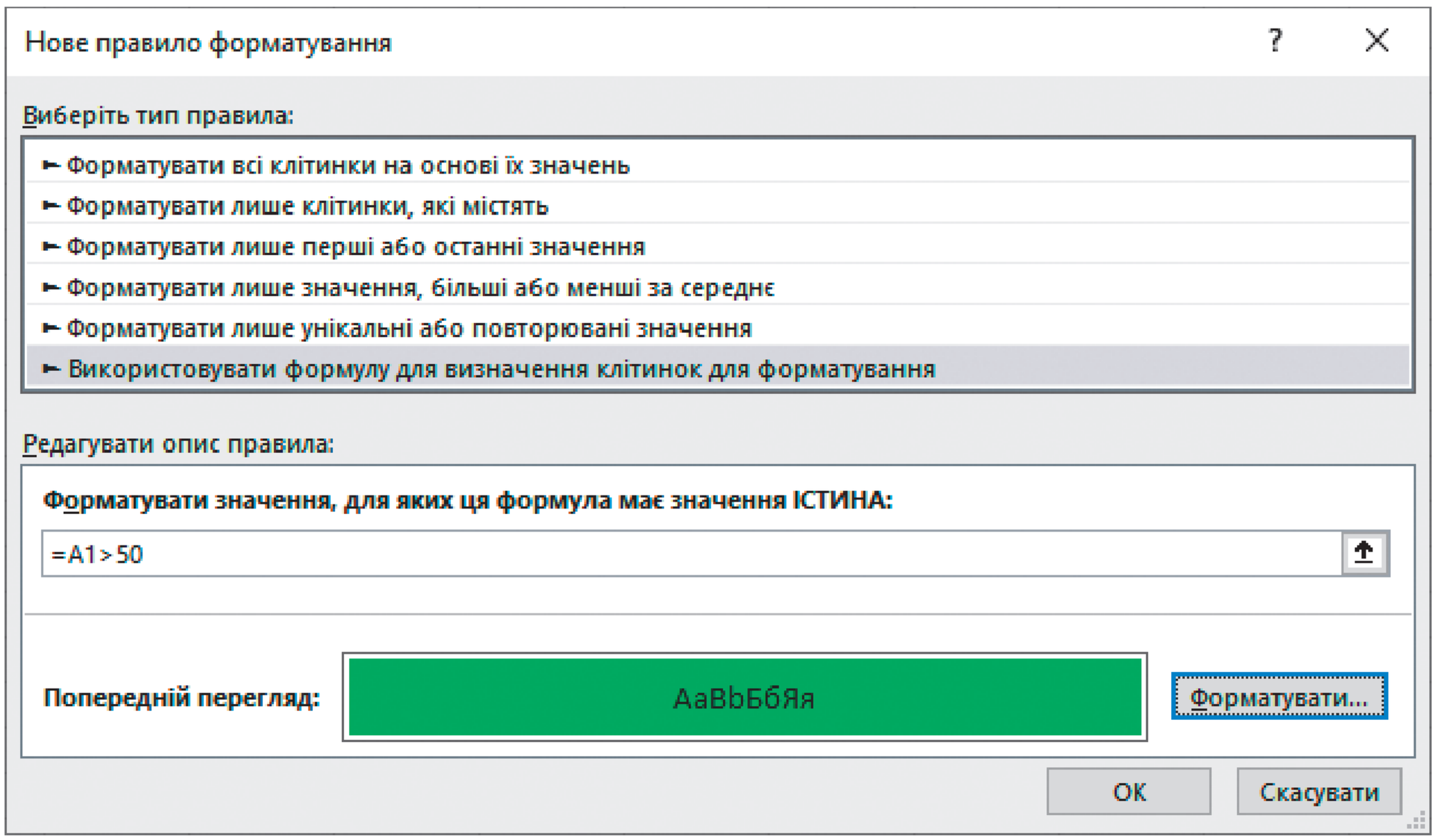
Кольорові шкали визначають максимальне і мінімальне значення в діапазоні й форматують кожну клітинку за кольором, який відповідає положенню значення між мінімальним та максимальним, зображаючи щось на зразок теплової карти.

Набори піктограм — кожній клітинці присвоюється позначка відповідно до обраного стилю.

Як відформатувати клітинки, значення яких відповідають довільній умові?

Якщо потрібно визначити формат клітинок, у яких значення відповідають довільній умові, наприклад вони непарні, більше 10, перебувають у діапазоні між 20 і 50 тощо, виконай такі дії:

1) Виділи діапазон клітинок.
2) У меню умовного форматування вибери пункт «Створити правило», а потім — пункт «Використовувати формулу для визначення клітинок для форматування».
3) У полі опису правила введи символ «=», а після нього — логічний вираз для верхньої лівої клітинки діапазону, наприклад =**A1>50**, і натисни кнопку «ОК».



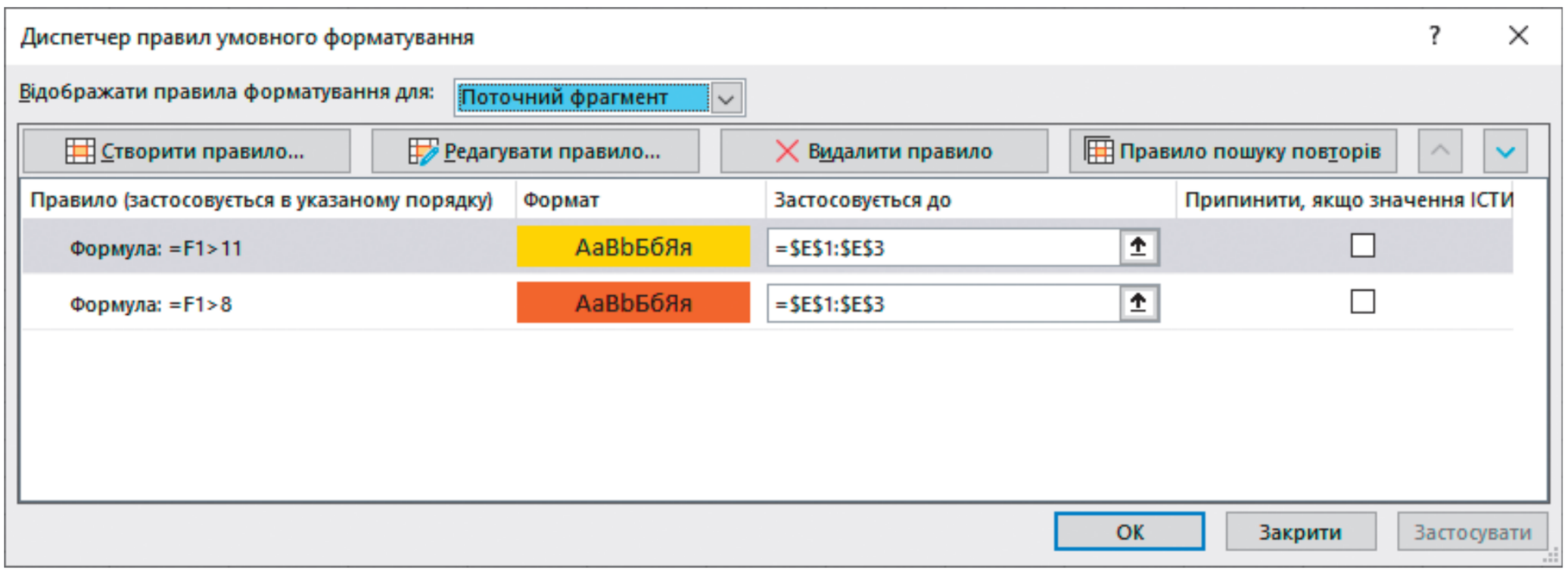
Формулу перевірки буде застосовано до всіх клітинок діапазону. Адреси в ній змінюватимуться за правилами копіювання формул.

! *Увага!*

Діапазон можна форматувати на основі правил, застосованих до іншого діапазону. Наприклад, можна виділяти кольором назви об'єктів у стовпці А на основі умов, накладених на значення властивостей цих об'єктів, що містяться у стовпцях В, С та ін. У такому разі слід виділити стовпець А, а у формулі, що визначає правило, ввести посилання на клітинки інших стовпців.

Як змінити правила умовного форматування?

Щоб змінити вже введені правила умовного форматування для діапазону, треба виділити цей діапазон, вибрати в меню умовного форматування пункт «Керувати правилами» і відредагувати чи видалити певне правило у вікні «Диспетчер правил умовного форматування».



! *Увага!*

До діапазону може застосовуватися кілька правил, які можуть суперечити одне одному. У такому разі пріоритет мають правила, розташовані вище. Порядок правил визначають за допомогою кнопок ^ та v у вікні «Диспетчер правил умовного форматування».

ПЕРЕВІР СЕБЕ

ЗНАННЯ — РОЗУМІННЯ — АНАЛІЗ — ВЛАСНІ ДУМКИ

1. Назви основні типи умовного форматування та опиши призначення кожного з них. Які кольори традиційно використовують для позначення позитивних, негативних та нейтральних показників у таблицях?
2. Поясни, чому умовне форматування є ефективним інструментом для аналізу великих обсягів даних. Наведи три конкретні приклади ситуацій із твого життя, де умовне форматування допомогло б швидше приймати рішення. Обґрунтуй вибір прикладів.

3. Розглянь таблицю результатів контрольної роботи з математики у дев'ятих класах:

Клас	Кількість учнів	Середній бал	Кількість оцінок 10–12	Кількість оцінок 1–3	% успішності
9-А	28	7,2	8	3	89 %
9-Б	25	6,8	5	5	80 %
9-В	30	8,1	12	1	97 %

Запропонуй систему умовного форматування для цієї таблиці, яка допоможе:

- швидко визначити найуспішніший клас;
- виділити проблемні показники;
- порівняти рівень знань у різних класах.

Обґрунтуй вибір кольорів, діапазонів значень та типів форматування, що допоможуть учителю та адміністрації школи в аналізі результатів.

4. Деякі критики стверджують, що за допомогою умовного форматування можна маніпулювати сприйняттям даних та спонукати до неправильних висновків. Інші вважають його необхідним інструментом для ефективного аналізу інформації в сучасному світі.

Вислови свою думку із цього питання:

- Чи погоджуєшся з тим, що умовне форматування може бути маніпулятивним? Наведи приклади.
- Як можна використовувати умовне форматування етично та об'єктивно?
- Яких правил варто дотримуватися під час створення візуалізацій даних для широкої аудиторії?
- Як розвиток навичок критичного мислення допомагає протистояти маніпуляціям через візуальне оформлення даних?

Підкріпи свою позицію конкретними аргументами та прикладами з власного досвіду або спостережень.

5. Ознайомся з інформацією в е-додатку (3.2) та вислови свою думку.

Дослідження.

Відкрий е-додаток (3.3). Виконай завдання.

Відкрий е-додаток (3.4). Виконай завдання.

§ 4. Дані та аналітичні висновки

Які висновки можна зробити на основі інформації, що міститься у великих електронних таблицях?

Уяви, що твій клас провів масштабне дослідження екологічної свідомості учнів школи, опитавши 200 респондентів. Тепер у тебе є таблиця із сотнями відповідей: вік, стать, клас, ставлення до сортування сміття, використання громадського транспорту, економія води тощо.

Як із цього «моря» цифр видобути справді важливі висновки? Як дізнатися, скільки учнів позитивно ставляться до сортування сміття? А скільки серед них дев'ятикласників? Як представити результати так, щоб вони вплинули на шкільну політику?

Сучасні інформаційні системи щодня обробляють величезні обсяги таких даних — від аналізу поведінки користувачів у соціальних мережах до дослідження ефективності лікування в медицині. Уміння правильно аналізувати дані, зокрема за допомогою електронних таблиць, стає ключовою навичкою для будь-якого дослідника, незалежно від сфери діяльності.

Кетрін Джонсон (англ. Katherine Johnson) — математикиня NASA, яка вручну обчислювала траєкторії космічних польотів. Їхня точність стала критично важливою для місії «Меркурій». Цікаво, тоді вже були електронні таблиці? Об'єднайтесь у пари і поміркуйте, як би її обчислення могли виглядати в електронних таблицях.

Завдання № 1

Учнівський парламент твоєї школи провів опитування про популярність спортивних секцій серед 150 учнів. Ось частина результатів.

Учень	Клас	Стать	Секція	Відвідує (так/ні)	Рейтинг (1–5)
Анна К.	8	Ж	Волейбол	Так	5
Петро М.	9	Ч	Футбол	Так	4
Софія Л.	8	Ж	Танці	Ні	3
…	…	…	…	…	…

Крок 1 Створи таблицю з даними, що стосуються 20 учнів різних класів. Використай реальні або правдоподібні дані (у межах зазначених діапазонів), дотримуючись логіки задачі.

Крок 2 Використай функцію **COUNTIF** для відповіді на запитання:

- Скільки дівчат та хлопців відвідують секції?
- Скільки учнів дев'ятих класів взяли участь в опитуванні?
- Скільки учнів виставили рейтинг не менше 4?

Створи окрему таблицю статистики за зразком:

Категорія	Кількість
Дівчата в секціях	=COUNTIF(...)
Хлопці в секціях	=COUNTIF(...)
Учні / учениці 9-х класів	=COUNTIF(...)
Високий рейтинг (4–5)	=COUNTIF(...)

Завдання № 2

Шкільна їдальня веде облік продажів обідів. Потрібно проаналізувати прибутковість різних страв, якщо прибутковість становить 10 % від ціни страви.

День	Страва	Ціна	Продано порцій	Категорія
Понеділок	Борщ	25	45	Перша страва
Понеділок	Котлета	35	40	Друга страва
Вівторок	Суп	20	50	Перша страва
...	...	...	...	...

Крок 1 Завантаж з електронного додатка (е-додаток (4.1)) таблицю з даними. Обчисли у стовпці E дохід, ввівши формулу в клітинку E2 і скопіювавши її в клітинки нижче.

Крок 2 Використай **SUMIF** для обчислення:

- кількості проданих порцій перших страв;
- загального доходу від других страв;
- загального доходу за понеділок.

Створи звіт за зразком.

Показник	Сума (грн)
Дохід від перших страв	=SUMIF(...) доповни формулу
Дохід від других страв	=SUMIF(...) доповни формулу
Дохід за понеділок	=SUMIF(...) доповни формулу
Дохід за вівторок	=SUMIF(...) доповни формулу

Завдання № 3

Завантаж з електронного додатка (е-додаток (4.2)) базу даних класу з інформацією про захоплення учнів.

№	Прізвище	Ім’я	Клас	Захоплення	Час на тиждень (год)	Витрати (грн/міс)
1	Петренко	Анна	9-А	Малювання	6	500
2	Іваненко	Петро	9-Б	Програмування	10	800
…	…	…	…	…	…	…

Використай функції для роботи з базою даних для обчислення:

- кількості учнів 9-х класів;
- кількості учнів, які витрачають на захоплення або більше 1000 грн на місяць, або більше 10 годин на тиждень.

Для виконання цього завдання використай формулу **DCOUNT** (з групи **D-функцій** в Excel: DCOUNT, DSUM, DAVERAGE тощо).

Функція **DCOUNT** в Excel — це функція баз даних, яка підраховує кількість клітинок із числовими значеннями в стовпці, що відповідають заданим критеріям. Вона використовується для аналізу даних у таблиці, коли потрібно порахувати елементи, які відповідають певним умовам.

Синтаксис:
=DCOUNT(база_даних; поле; критерій)

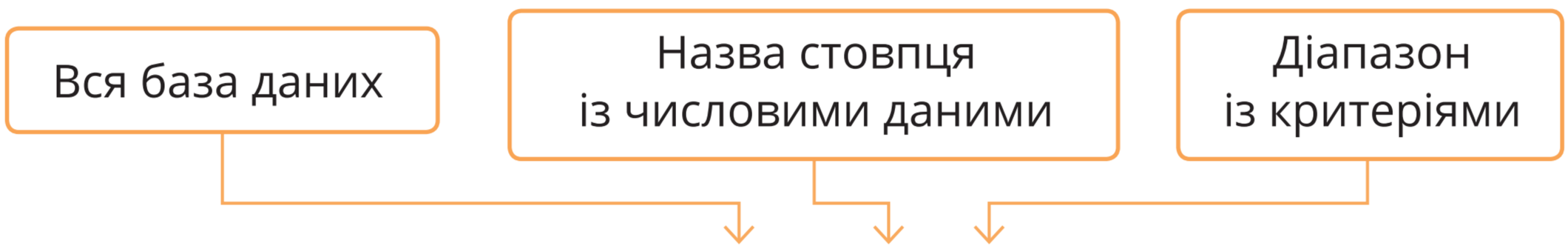
Аргументи:

1. база_даних — діапазон клітинок із таблицею, включно з рядком заголовків, наприклад: A1:G26.
2. поле — стовпець, за яким виконується підрахунок. Можна вказати:
   - назву стовпця в лапках (наприклад, "№", "Витрати (грн/міс)");
   - або номер стовпця (наприклад, 2 — це другий стовпець у діапазоні).

Важливо!

Функція рахує тільки числові значення у вибраному полі. Якщо в стовпці текст (наприклад, «Клас»), то результат буде 0. Тому в цій задачі для аргументу поля потрібно взяти назву стовпця із числовими даними, наприклад «№» (ми просто вказуємо будь-яке заповнене ЧИСЛОВЕ поле, щоб порахувати кількість рядків).

3. критерій — діапазон з умовами відбору (мінімум 1 заголовок стовпця + умова під ним).



Приклад формули: =DCOUNT(A1:G26;"№";I5:I8)

Для цієї задачі (обчислення кількості учнів 9-х класів) діапазон створених критеріїв буде мати вигляд:

Клас
9-А
9-Б
9-В

I5

I8

Виконай обчислення та перевір результати.

I	J	K	L
кількість учнів 9 класів		кількості учнів, які витрачають на захоплення або більше 1000 грн. на місяць, або більше 10 годин на тиждень.	
20		1	

Таблиці критеріїв

Клас		Витрати (грн/міс)	Час на тиждень (год)
9-А		>1000	
9-Б			>10
9-В			

- Зміни рішення, використавши для створення критеріїв (обчислення кількості учні 9-х класів) символи підстановки (див. «Запитання-відповіді»).

Завдання № 4

Спортивний клуб школи веде статистику результатів змагань.

Спортсмен	Вид спорту	Місце	Вік	Стать	Дата змагань
Анна П.	Плавання	1	15	Ж	15.03.2024
Олег К.	Легка атлетика	3	16	Ч	20.03.2024
…	…	…	…	…	…

Крок 1 Завантаж з електронного додатка таблицю з 20 записами (е-додаток (4.3)).

Крок 2 Добери самостійно функції для обчислення:

- найкращого місця серед дівчат;
- найгіршого місця серед хлопців у плаванні;
- середнього віку призерів (тих, хто посіли перше–третє місця).

Створи аналітичну таблицю.

Показник	Результат
Найкраще місце дівчат	?
Найгірше місце хлопців у плаванні	?
Середній вік призерів (перше–третє місця)	?

Завдання № 5

Повернімося до прикладу на початку параграфа. Твій клас здійснив дослідження екологічної свідомості. Завантаж з електронного додатка таблицю (е-додаток (4.4)).

Респондент	Клас	Вік	Стать	Сортує сміття	Економить воду	Використовує багаторазові речі	Економить електроенергію	Екобали
1	8	14	Ч	ні	так	так	ні	
2	9	15	Ж	так	так	так	так	
…	…	…	…	…	…	…		

В останньому стовпці «Екобали» для кожного учня / учениці підрахуй кількість екопрактик, яких він / вона дотримується: «Сортує сміття», «Економить воду», «Використовує багаторазові речі», «Економить електроенергію». За кожне «так» нарахуй 1 бал.

Визнач:

- кількість учнів, які сортують сміття;
- кількість учнів із високими екобалами (наприклад, 3–4 із 4);
- середній екобал дівчат та хлопців окремо;
- середній вік екосвідомих учнів (які і економлять воду, і сортують сміття);
- найвищий екобал серед учнів восьмих класів;
- найнижчий екобал серед тих, хто не сортує сміття;
- номер респондента з найвищим екобалом серед дівчат.

Зроби висновок: Який портрет найбільш екосвідомого учня?

ЗАПИТАННЯ-ВІДПОВІДІ

Чим відрізняються звичайні агрегатні функції від агрегатних функцій з умовами?

Агрегатні функції — це функції, які узагальнюють набір значень в одне підсумкове.

Приклади агрегатних функцій в Excel:

- SUM — сума;
- AVERAGE — середнє;
- MIN, MAX — мінімум / максимум;
- COUNT, COUNTA — кількість чисел / непорожніх клітинок;
- умовні варіанти: SUMIF/SUMIFS, AVERAGEIF/AVERAGEIFS, COUNTIF/COUNTIFS;
- для роботи з базою даних: DSUM, DAVERAGE, DCOUNT тощо.

Звичайні агрегатні функції (SUM, AVERAGE, MIN, COUNT тощо) обчислюють підсумкову величину (суму, середнє, мінімум, кількість тощо) для всіх значень у заданому діапазоні. Функції з умовами обчислюють підсумкову величину лише для тих значень, що відповідають певному критерію відбору.

Якщо критерій простий, наприклад «значення дорівнює певному рядку» або «значення більше якогось числа», то застосовують такі функції, як SUMIF, AVERAGEIF, MINIFS, COUNTIF тощо. Якщо критерій стосується кількох параметрів об’єкта, то застосовують функції для роботи з базою даних — DCOUNT, DSUM, DMIN тощо.

Найчастіше функції для обчислення підсумків за критерієм застосовують тоді, коли рядки таблиці містять інформацію про об’єкти, а стовпці — про їхні параметри. Тоді підсумки (суму, середнє тощо) можна обчислювати для одного параметра, а відбирати об’єкти — на основі значень інших параметрів.

Який синтаксис та призначення агрегатних функцій з умовами — COUNTIF, SUMIF, MINIFS тощо?

COUNTIF(діапазон; критерій)

Обчислює кількість значень у діапазоні, що відповідають критерію. Наприклад, формула =COUNTIF(D2:D8;">10") обчислює кількість більших 10 значень у діапазоні D2:D8.

SUMIF(діапазон відбору; критерій; діапазон підсумовування)

Діапазон відбору та діапазон підсумовування — це, як правило, стовпці однієї таблиці. Відбираються ті рядки, у яких значення в діапазоні відбору задовольняють критерію. У відібраних рядках підсумовуються значення з діапазону підсумовування.

Наприклад, формула =SUMIF(A2:A8;"Samsung";D2:D8) обчислює суму значень з діапазону D2:D8, яким відповідає значення «Samsung» у діапазоні A2:A8.

Діапазон відбору

Діапазон підсумовування

	A	B	C	D
1	**Марка**	**Модель**	**Вартість**	**Кількість на складі**
2	Apple	iPhone 17 Pro	59 500,00 ₴	3
3	Xiaomi	Redmi 14C	4 400,00 ₴	12
4	Apple	iPhone 17 Pro Max	75 000,00 ₴	44
5	Samsung	Galaxy S25 Ultra	40 000,00 ₴	25
6	Apple	iPhone 17 Air	42 000,00 ₴	9
7	Xiaomi	Xiaomi 15 Ultra	57 000,00 ₴	6
8	Samsung	Galaxy A16 5G	6 600,00 ₴	15

Відібрані рядки та значення, що підсумовуються

AVERAGEIF(діапазон відбору; критерій; діапазон підсумовування)

Функція діє так само, як і SUMIF, але замість підсумовування значень обчислюється їхнє середнє.

MINIFS(діапазон обчислення мінімуму; діапазон відбору; критерій)

Функція визначає мінімальне серед тих значень у діапазоні обчислення мінімуму, для яких відповідні значення в діапазоні відбору.

Наприклад, формула =MINIFS(C2:C8;A2:A8;"Apple") обчислює мінімальне значення в тих рядках діапазону C2:C8, для яких у клітинках діапазону A2:A8 міститься значення «Apple».

	A	B	C	D
1	**Марка**	**Модель**	**Вартість**	**Кількість на складі**
2	Apple	iPhone 17 Pro	59 500,00 ₴	3
3	Xiaomi	Redmi 14C	4 400,00 ₴	12
4	Apple	iPhone 17 Pro Max	75 000,00 ₴	44
5	Samsung	Galaxy S25 Ultra	40 000,00 ₴	25
6	Apple	iPhone 17 Air	42 000,00 ₴	9
7	Xiaomi	Xiaomi 15 Ultra	57 000,00 ₴	6
8	Samsung	Galaxy A16 5G	6 600,00 ₴	15

MAXIFS(діапазон обчислення мінімуму; діапазон відбору; критерій)

Функція діє так само, як і MINIFS, але замість мінімуму визначається максимум.

Що таке бази даних і чим вони відрізняються від електронних таблиць?

База даних — це організована колекція структурованих даних, яка зберігається та керується спеціальними системами управління базами даних (СУБД). На відміну від електронних таблиць, бази даних призначені для роботи з великими обсягами інформації, забезпечують надійність, безпеку та одночасний доступ багатьох користувачів.

Основні відмінності:

- обсяг даних: бази даних ефективно працюють із мільйонами записів, тоді як таблиці обмежені сотнями тисяч рядків;
- структура: у базі даних, як правило, є багато пов’язаних таблиць, а електронні таблиці одна від одної незалежні;
- безпека: бази даних мають системи авторизації та контролю доступу;
- цілісність: СУБД забезпечують автоматичну перевірку коректності даних та запобігають їхньому дублюванню.

Отже, електронна книга не є повноцінною базою даних, хоча табличний процесор може забезпечувати деяку подібність електронної таблиці до таблиці бази даних, зокрема виконувати деякі типові для бази функції з відбору даних.

Який синтаксис та призначення функцій для роботи з базами даних — DGET, DSUM, DMIN тощо?

Усі функції для роботи з базами даних мають однаковий синтаксис:
ІМ’Я_ФУНКЦІЇ(уся таблиця; поле; діапазон критерію)

Ім’я функції визначає, яка дія виконуватиметься над даними:
DGET — вибір значення;
DSUM — підсумовування значень;
DAVERAGE — обчислення середньої арифметичної;
DMIN — визначення мінімуму;
DMAX — визначення максимуму;
DCOUNT — кількість значень.

Ця дія застосовується до значень у стовпці з назвою *поле*, що містяться у рядках, які відповідають *критерію*. Критерій — це окрема область на аркуші, у якій записують умови відбору рядків таблиці за такими самими правилами, що й для розширених фільтрів:

- у першому рядку вказують назви стовпців, на значення в яких накладають умови;
- у наступних рядках записують власне умови;
- умови, що записані в одному рядку, з’єднуються сполучником «І», а умови з різних рядків — сполучником «АБО».

Наприклад, формула =DSUM(A1:D8;D1;F1:G2) для зображеної електронної книги працюватиме так:

- Уся таблиця («база даних») міститься в діапазоні A1:D8. У рядку 1 розташовані заголовки стовпців (полів).
- Критерій розміщено в діапазоні F1:G2. Він читається так: «вибрати смартфони, вартість яких перевищує 10 000 грн і кількість на складі перевищує 20 шт.». Умови з’єднано сполучником «і», оскільки їх розташовано в одному рядку. Смартфонів, що задовольняють обом цим умовам, усього два: Apple iPhone 17 Pro Max і Samsung Galaxy S25 Ultra.
- Підсумовуються значення поля D1, тобто кількості на складі, але лише з тих рядків, що задовольняють умові, — ці значення виділено синім. Їхня сума становить 69 — це і є результат.

	A	B	C	D	E	F	G
1	**Марка**	**Модель**	**Вартість**	**Кількість на складі**		**Вартість**	**Кількість на складі**
2	Apple	iPhone 17 Pro	59 500,00 ₴	3		>10000	>20
3	Xiaomi	Redmi 14C	4 400,00 ₴	12			
4	Apple	iPhone 17 Pro Max	75 000,00 ₴	44			
5	Samsung	Galaxy S25 Ultra	40 000,00 ₴	25			
6	Apple	iPhone 17 Air	42 000,00 ₴	9			
7	Xiaomi	Xiaomi 15 Ultra	57 000,00 ₴	6			
8	Samsung	Galaxy A16 5G	6 600,00 ₴	15			

Наведемо інші приклади критеріїв:

Вартість
<5000
>10000

смартфони, вартість яких або менша за 5000 грн, або більша за 10 000 грн

Марка	**Вартість**
Apple	>50000
Apple	<30000

смартфони Apple, вартість яких або менша за 30 000 грн, або більша за 50 000 грн

Марка	**Вартість**
Samsung	>20000
A*	>60000

усі смартфони Samsung, а також смартфони, назва яких починається з літери «А», вартістю більше 60 000 грн

В умові, що перевіряє рівність значення текстовому рядку, можна використати символи підстановки. Так, «» означає будь-яку послідовність символів, а «?» — будь-який один символ. Наприклад, умова *а* означає будь-який текст, що містить літеру «а».*

Увага!

Функція DGET працює лише якщо критерію відповідає тільки один рядок. Інакше вона видає помилку #НОМЕР!

ПЕРЕВІР СЕБЕ

ЗНАННЯ — РОЗУМІННЯ — СТВОРЕННЯ — АНАЛІЗ

1. Поясни відмінність між звичайними агрегатними функціями (SUM, AVERAGE, COUNT) та агрегатними функціями з умовами (SUMIF, AVERAGEIF, COUNTIF).

2. «Прочитай» зображені критерії для таблиці смартфонів з рубрики «Запитання-відповіді».

а)

Вартість	Вартість
>5000	<10000
>20000	<50000

б)

Виробник	Вартість	Кількість на складі
Xiaomi	<10000	
		>30

в)

Кількість на складі	Вартість
<10	>50000
>30	<10000

3. Накресли таблиці критеріїв для вибору таких смартфонів:
а) смартфони Apple, що є на складі у кількості принаймні 10 шт.;
б) смартфони, які або дорожчі за 20 000 грн, або є на складі в кількості більше 20 шт.;
в) смартфони, яких на складі або менше 3 шт., або більше 30 шт.;
г) смартфони Apple, які або дорожчі за 60 000 грн, або дешевші за 45 000 грн.

4. Твоя школа досліджує ефективність різних форм навчання під час карантину. Є дані про 100 учнів: форма навчання (онлайн / змішана / очна), клас (8–11), предмет, оцінка за семестр, кількість пропусків, час на виконання домашніх завдань. Сформулюй п’ять запитань, на які можна відповісти за допомогою вивчених функцій з умовами або функцій для роботи з базою даних. Для кожного запитання вкажи, яку саме функцію варто використати, який вигляд матиме діапазон критерію (якщо він потрібний), та поясни логіку застосування функції.

Приклад: «Яка середня оцінка учнів дев’ятих класів, що навчаються онлайн» — використовуємо DAVERAGE з критерієм, що поєднує дві умови.

5. За умовою попереднього завдання створи таблицю з відомостями про 10 учнів. Дай відповіді на сформульовані запитання за допомогою обраних тобою функцій.

6. Дирекція школи отримала таблицю з результатами опитування про якість харчування:

Клас	Кількість учнів	Оцінка якості (1–5)	Щоден-но їдять в їдальні	Скарги за місяць	Витрати на обід (грн)
8-А	28	3,2	15	8	45
8-Б	25	3,8	20	3	45
9-А	30	2,9	12	12	50
9-Б	27	4,1	24	2	50
10-А	24	3,5	18	5	55
11-А	22	4,3	21	1	55

Завдання для аналізу:

1. Критично проаналізуй представлені дані:
   - Які показники можуть бути пов’язані між собою?
   - Чи є суперечливі дані? Якщо так, наведи приклади.
   - Яких важливих показників немає в таблиці?
2. Використовуючи функції з умовами, обчисли:
   - Середню оцінку якості серед класів із високою (більше 19 учнів щодня) та низькою (менше 19 учнів щодня) відвідуваністю.
   - Загальну кількість скарг від класів з оцінкою нижче та вище 3,5 бала.
   - Середні витрати учнів, які оцінили якість вище та нижче 4 балів.
3. Зроби висновки:
   - Як впливає відвідуваність на оцінку якості?
   - Як пов’язані оцінка якості та кількість скарг?
   - Чи є зв’язок між витратами на обід та оцінкою якості?
4. Сформулюй гіпотези та перевір їх:
   - Чи впливає ціна обіду на оцінку якості?
   - Чи є зв’язок між кількістю скарг та відсотком учнів, що їдять в їдальні?
   - Чи відрізняється ставлення до їдальні у молодших та старших класів?
5. Запропонуй план покращення ситуації:
   - На основі аналізу даних визнач три найважливіші проблеми.
   - Для вирішення кожної проблеми запропонуй конкретні заходи.
   - Опиши, які додаткові дані потрібно зібрати для більш точного аналізу.
6. Етичні питання аналізу даних:
   - Чи можна публікувати результати опитування із зазначенням класів?
   - Як забезпечити об’єктивність висновків?
   - Які ризики неправильної інтерпретації даних?

§ 5. Основи машинного навчання в електронних таблицях

Як навчити комп’ютер самостійно знаходити закономірності в даних?

Щоранку, відкриваючи Spotify або YouTube, ти бачиш персональні рекомендації музики чи відео. Як платформа «знає», що тобі сподобається? Netflix пропонує фільми, які ідеально підходять твоїм смакам. Amazon радить товари, які ти справді хочеш купити. Instagram показує саме той контент, на якому ти затримуєш увагу.

За всім цим стоїть **машинне навчання** — технологія, завдяки якій комп’ютери самостійно знаходять закономірності в даних і роблять прогнози. Це не магія, а математика та алгоритми, які можна зрозуміти і навіть створити власними руками.

Звісно, справжні системи машинного навчання створені за допомогою спеціалізованих мов програмування, використовують потужні алгоритми та опрацьовують мільйони записів даних. Але щоб зрозуміти, як це працює, не обов’язково писати код. Базові ідеї можна змоделювати за допомогою електронних таблиць — простого, зручного та наочного інструмента.

Чи відомо тобі, що основи машинного навчання можна опанувати просто в Excel або Google Таблицях? Сьогодні ми створимо власну рекомендаційну модель і навчимо комп’ютер розпізнавати музичні жанри — усе це за допомогою звичайних формул. Це буде модель справжнього процесу — спрощена, але цілком реалістична.

Завдання № 1

Як навчити комп’ютер «вгадувати» вподобання.

Виконай кожен крок цього завдання практично.

Ситуація: Твій друг створює музичний стрімінговий сервіс* і просить тебе розробити систему рекомендацій. У тебе є дані про те, як різні користувачі оцінюють пісні (за шкалою від 1 до 5).

Потрібно спрогнозувати яку оцінку Данило поставить пісні E.

---

* Стрімінговий сервіс — це платформа, яка надає користувачам безперервний доступ до медіаконтенту (відео, музики, книг) у реальному часі через інтернет, без необхідності його завантажувати.

Користувач	Пісня A	Пісня B	Пісня C	Пісня D	Пісня E
Анна	5	2	4	1	3
Богдан	4	1	5	2	4
Віка	3	3	2	5	1
Данило	5	2	4	1	?

Для перегляду відеопояснення Завдання 1 відкрий е-додаток (5.1).

Ідея методу

*Щоб передбачити оцінку, яку Данило поставить пісні **E**, порівнюємо його музичні вподобання з уподобаннями інших користувачів. Що більш схожі оцінки, то більшу вагу матиме прогноз користувача. Остаточна оцінка визначається як зважене середнє оцінок інших користувачів для пісні **E**.*

Крок 1 **Введення даних**

У таблицю вносимо оцінки користувачів за пісні **A–E** (у Данила немає оцінки для пісні **E** — її потрібно спрогнозувати).

	A	B	C	D	E	F	G
1	Користувач	Пісня A	Пісня B	Пісня C	Пісня D	Пісня E	
2	Анна	5	2	4	1	3	
3	Богдан	4	1	5	2	4	
4	Віка	3	3	2	5	1	
5	Данило	5	2	4	1		
6							

Крок 2 **Визначення показників схожості**

Нижче створюємо додаткову таблицю, у якій для кожної пісні A–D обчислюємо схожість між оцінкою Данила та оцінками інших користувачів.

За формулою:

=1-|оцінка Данила - оцінка іншого користувача| / розмір шкали.

Потрібно ввести формулу в ліву верхню клітинку додаткової таблиці та скопіювати її в усі клітинки таблиці.

Введені формули:

K11

	A	B	C	D	E	F
1	Користувач	Пісня A	Пісня B	Пісня C	Пісня D	Пісня E
2	Анна	5	2	4	1	3
3	Богдан	4	1	5	2	4
4	Віка	3	3	2	5	1
5	Данило	5	2	4	1	
6						
7	Схожість оцінки з Данилом	Пісня A	Пісня B	Пісня C	Пісня D	
8	Анна	=1-ABS(B$5-B2)/4	=1-ABS(C$5-C2)/4	=1-ABS(D$5-D2)/4	=1-ABS(E$5-E2)/4	
9	Богдан	=1-ABS(B$5-B3)/4	=1-ABS(C$5-C3)/4	=1-ABS(D$5-D3)/4	=1-ABS(E$5-E3)/4	
10	Віка	=1-ABS(B$5-B4)/4	=1-ABS(C$5-C4)/4	=1-ABS(D$5-D4)/4	=1-ABS(E$5-E4)/4	

Отримані дані:

Схожість оцінки з Данилом	Пісня A	Пісня B	Пісня C	Пісня D
Анна	1,00	1,00	1,00	1,00
Богдан	0,75	0,75	0,75	0,75
Віка	0,50	0,75	0,50	0,00

Проаналізуй введені формули та отримані дані. Дай відповіді на питання.

- Що означають рівні схожості 1 та 0?
- Що обчислює функція ABS?
- Чому виконується ділення саме на число 4?
- Для чого введено знак $ в адресі B$5?

Крок 3 Середня схожість

Для кожного користувача знаходимо середнє значення його схожості з Данилом за всіма піснями A–D:

	A	B	C	D	E	F
1	Користувач	Пісня A	Пісня B	Пісня C	Пісня D	Пісня E
2	Анна	5	2	4	1	3
3	Богдан	4	1	5	2	4
4	Віка	3	3	2	5	1
5	Данило	5	2	4	1	
6						
7	Схожість оцінки з Данилом	Пісня A	Пісня B	Пісня C	Пісня D	Середнє
8	Анна	1,00	1,00	1,00	1,00	1,00
9	Богдан	0,75	0,75	0,75	0,75	0,75
10	Віка	0,50	0,75	0,50	0,00	0,44

Середнє
=AVERAGE(B8:E8)
=AVERAGE(B9:E9)
=AVERAGE(B10:E10)

Ці середні значення і є **вагами** для подальших розрахунків.

Крок 4 Прогнозування за допомогою функції SUMPRODUCT

Щоб урахувати як оцінку користувача, так і ступінь її подібності до оцінки Данила, обчислюємо зважене середнє. Для цього застосовуємо формулу:

=SUMPRODUCT(F2:F4;F8:F10)/SUM(F8:F10)

	A	B	C	D	E	F
1	Користувач	Пісня A	Пісня B	Пісня C	Пісня D	Пісня E
2	Анна	5	2	4	1	3
3	Богдан	4	1	5	2	4
4	Віка	3	3	2	5	1
5	Данило	5	2	4	1	2,9428571
6						
7	Схожість оцінки з Данилом	Пісня A	Пісня B	Пісня C	Пісня D	Середнє
8	Анна	1,00	1,00	1,00	1,00	1,00
9	Богдан	0,75	0,75	0,75	0,75	0,75
10	Віка	0,50	0,75	0,50	0,00	0,44

Функція SUMPRODUCT перемножує відповідні елементи двох діапазонів і додає результати.

Отже, ми враховуємо не тільки саму оцінку, але і її вагу, тобто наскільки відповідним є прогноз користувача прогнозу Данила.

$$(\text{оцінка Анни} \times \text{вага Анни}) + (\text{оцінка Богдана} \times \text{вага Богдана}) + \\ + (\text{оцінка Віки} \times \text{вага Віки})$$

SUM(F8:F10) Обчислює суму всіх ваг, щоб звести результат до діапазону від 1 до 5.

Остаточний результат — це зважене середнє:

$$\frac{\sum (\text{оцінка} \times \text{вага})}{\sum (\text{вага})}$$

Що таке зважене середнє?

Середнє арифметичне — це відоме нам «звичайне середнє». Ми додаємо всі значення і ділимо суму на їхню кількість.

Наприклад, якщо три учні отримали оцінки 5, 3 і 4, то середнє буде (5 + 3 + 4) / 3 = 4.

Але іноді не всі значення однаково важливі. Тоді використовують ***зважене середнє****.*

Приклад із життя

Уявімо, що в школі підсумкова оцінка з предмета складається з:

- *контрольної роботи (вага 60 %);*
- *двох домашніх завдань (вага — по 20 %).*

Учень отримав:

- *контрольна = 5;*
- *домашні = 4 і 3.*

ЗВАЖЕНЕ СЕРЕДНЄ

Звичайне середнє

5 3 4

$$\frac{5+3+4}{3} = 4$$

Зважене середнє

$5 \times 0{,}6$

$3 \times 0{,}2$

$4 \times 0{,}2$

$$\frac{5 \times 0{,}6 + 3 \times 0{,}2 + 4 \times 0{,}2}{0{,}6 + 0{,}2 + 0{,}2} = 4$$

Звичайне середнє:
(5 + 4 + 3) / 3 = 4.

Зважене середнє:
(5 × 0,6) + (4 × 0,2) + (3 × 0,2) = 3 + 0,8 + 0,6 = 4,4.

Тобто результат ближчий до оцінки контрольної, бо вона «важливіша».

У нашій задачі:

- *Оцінки пісні E іншими користувачами — це значення.*
- *Схожість із уподобаннями Данила — це вага кожного значення.*

Наприклад:

- *вподобання Анни та Данила схожі повністю (вага = 1), тому оцінка Анни 3 «важить» найбільше.*
- *Вподобання Богдана схожі трохи менше (вага 0,75), його оцінка 4 враховується, але меншою мірою.*
- *Віка має низьку схожість (вага 0,44), її оцінка 1 впливає на результат ще менше.*

Формула зваженого середнього:
(3 · 1 + 4 · 0,75 + 1 · 0,44) / (1 + 0,75 + 0,44) ≈ 2,94

Висновок

Зважене середнє — це спосіб обчислення середнього, де різним значенням надається різна важливість (вага).
У рекомендаційних системах середнє зважене дає змогу більше довіряти тим користувачам, чиї вподобання більш схожі на наші.

Завдання № 2

Виконай кожен крок цього завдання практично. Як навчити комп'ютер розпізнавати жанр музики?

Ситуація: Уяви, що ти працюєш над створенням музичного застосунку. Одне з твоїх завдань — навчити систему автоматично визначати жанр пісні на основі її характеристик, таких як темп, настрій, інструменти чи кількість вокалу.

Дані:

Назва пісні	Темп (BPM)	Гуч-ність	Енергійність	Ритміч-ність	Жанр
«Beat Drop»	128	8	9	8	EDM*
«Sunset»	65	4	3	2	Інді**
«Rock On»	140	9	8	6	Рок
«Chill Out»	75	3	2	4	Інді
«Bass Boom»	130	9	9	9	EDM
«Guitar»	120	7	7	5	Рок

Для перегляду відеопояснення до завдання 2 відкрий е-додаток (5.2).

Крок 1 Аналіз закономірностей

Обчисли середні значення для кожного жанру.

Жанр	Середній темп	Середня гучність	Середня енергійність	Середня ритмічність
EDM	=AVERAGEIF()	=AVERAGEIF()	=AVERAGEIF()	=AVERAGEIF()
Інді	=AVERAGEIF()	=AVERAGEIF()	=AVERAGEIF()	=AVERAGEIF()
Рок	=AVERAGEIF()	=AVERAGEIF()	=AVERAGEIF()	=AVERAGEIF()

Поміркуй, яка формула буде в кожній клітинці, якщо AVERAGEIF() — це функція в Excel, яка обчислює середнє тих значень, що відповідають певній умові.

---

* EDM (англ. Electronic Dance Music) — це термін для широкого спектра електронної танцювальної музики, яка створюється для клубів, фестивалів та вечірок.

** Незалежна музика, також відома як інді-музика.

AVERAGEIF(діапазон; умови; діапазон_усереднення)

Функція AVERAGEIF має такі аргументи:

- діапазон — обов'язковий аргумент. Одна або кілька клітинок, що містять числа для яких потрібно обчислити середнє значення;
- умови — обов'язковий аргумент. Умова у вигляді числа, виразу, посилання на клітинку або тексту, що визначає, для яких клітинок потрібно обчислити середнє значення. Наприклад, умова може мати таку форму: "32", ">32", "яблука" або B4;
- діапазон_усереднення — необов'язковий аргумент. Фактичний набір клітинок для обчислення середнього значення. Якщо пропущено, використовується діапазон, вказаний у першому аргументі.

Наприклад, щоб порахувати середній темп для жанру EDM, формула буде:

=AVERAGEIF(стовпець_жанрів; **"EDM"**; стовпець_темпу)

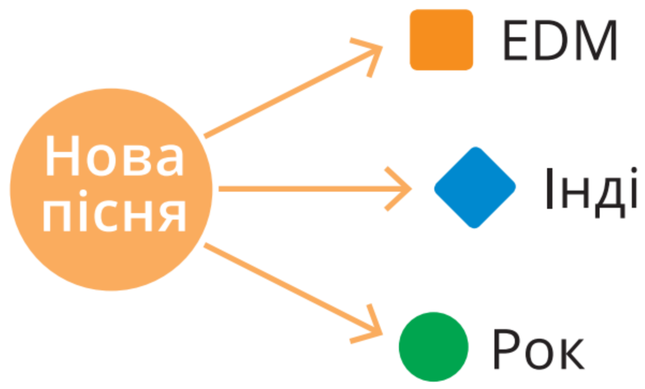
Крок 2 Застосування методу класифікації

Нова пісня для класифікації:

- Темп: 125 BPM
- Гучність: 7
- Енергійність: 8
- Ритмічність: 7

Потрібно визначити її жанр.

Метод «найближчого сусіда»



Метод найближчого сусіда (Nearest Neighbor) — це простий спосіб класифікації нових об'єктів на основі вже відомих прикладів. Його ідея полягає в тому, що схожі об'єкти мають належати до одного класу (у нашому прикладі класами є жанри EDM, Інді, Рок).

Як це працює:

1. Є база даних прикладів — у нашому випадку пісні, для яких відомі характеристики (темп, гучність, енергійність, ритмічність) і жанр.
2. Є новий об'єкт для класифікації — пісня, для якої ми знаємо характеристики, але не знаємо жанр.
3. Вимірюємо відстань (схожість) між новою піснею та кожним жанром. Для цього обчислюємо різницю між їхніми характеристиками (наприклад, за допомогою формули Евкліда).

> ***Евклідова відстань*** *— це відстань між двома точками у багатовимірному просторі, що вимірюється як довжина найкоротшого шляху між ними. Обчислюється за теоремою Піфагора як квадратний корінь суми квадратів різниць координат точок. Це найпростіший спосіб вимірювання відстані, який широко використовується в геометрії, машинному навчанні та інших галузях для кількісної оцінки подібності або відмінності між даними.*
>
> *У нашому прикладі ми маємо чотири ознаки:*
> *темп, гучність, енергійність, танцювальність.*
>
> *Тоді відстань між новою піснею x=(x1,x2,x3,x4) і прикладом пісні з першої таблиці y=(y1,y2,y3,y4) становить:*
>
> $$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 + (x_4 - y_4)^2}.$$
>
> *Що менше* ***d*** *— то більша схожість.*

4. Знаходимо найближчого сусіда — той жанр, відстань від якого до нашої нової пісні найменша.
5. Присвоюємо новій пісні такий самий жанр, як у найближчого сусіда.

Вигляд таблиці:

	A	B	C	D	E	F	G	H	I	J
1	**Назва пісні**	**Темп (BMP)**	**Гучність**	**Енергійність**	**Танцю-вальність**	**Жанр**				
2	«Beat Drop»	128	8	9	8	EDM				
3	«Sunset»	65	4	3	2	Інді				
4	«Rock On»	140	9	8	6	Рок				
5	«Chill Out»	75	3	2	4	Інді				
6	«Bass Boom»	130	9	9	9	EDM				
7	«Guitar»	120	7	7	5	Рок				
8	**Нова пісня**	125	7	8	7					
9										
10		**Середні значення кожного жанру**								
11	**Жанр**	**Темп (BMP)**	**Гучність**	**Енергійність**	**Танцю-вальність**	**Обчислення відстані**				
12	EDM	129	8,5	9	8,5	=SQRT((B$8-B12)^2+(C$8-C12)^2+(D$8-D12)^2+(E$8-E12)^2)				
13	Інді	70	3,5	2,5	3					
14	Рок	130	8	7,5	5,5	5,34				

SQRT(число)

Отримані дані:

Середні значення кожного жанру					
Жанр	**Темп (BMP)**	**Гучність**	**Енергійність**	**Танцю-вальність**	**Обчислення відстані**
EDM	129	8,5	9	8,5	4,64
Інді	70	3,5	2,5	3	55,53
Рок	130	8	7,5	5,5	5,34

Отже, ми з’ясували, що нова пісня, імовірно, належить до жанру EDM.

Зверніть увагу!

Точність цього методу буде залежати від кількості даних. Що більше даних, то точніше результат.

А якщо даних буде достатньо багато, то визначати «на око» найменшу відстань буде складно. Застосуємо для автоматичного визначення жанру формулу:

```
=XLOOKUP(MIN(F12:F14);F12:F14;A12:A14)
```

	A	B	C	D	E	F
1	**Назва пісні**	**Темп (BMP)**	**Гучність**	**Енергійність**	**Танцю-вальність**	**Жанр**
2	«Beat Drop»	128	8	9	8	EDM
3	«Sunset»	65	4	3	2	Інді
4	«Rock On»	140	9	8	6	Рок
5	«Chill Out»	75	3	2	4	Інді
6	«Bass Boom»	130	9	9	9	EDM
7	«Guitar»	120	7	7	5	Рок
8	**Нова пісня**	125	7	8	7	**EDM**
9						
10	**Середні значення кожного жанру**					
11	**Жанр**	**Темп (BMP)**	**Гучність**	**Енергійність**	**Танцю-вальність**	**Обчислення відстані**
12	EDM	129	8,5	9	8,5	4,64
13	Інді	70	3,5	2,5	3	55,53
14	Рок	130	8	7,5	5,5	5,34

Пояснення:

- **MIN(F12:F14)**
  знаходить найменше значення серед відстаней.
- **=XLOOKUP(MIN(F12:F14);F12:F14;A12:A14)***
  шукає в діапазоні F12:F14 значення виразу MIN(F12:F14) і повертає відповідне значення з діапазону A12:A14.

Крок 3 **Визначення точності**

1. Створи тестову вибірку з п’яти нових пісень з відомими жанрами та характеристиками.
2. Визнач їхні жанри за розглянутим вище алгоритмом.
3. Порівняй результати з реальними жанрами.
4. Обчисли точність як відсоток правильних відповідей.

---

* Перший параметр функції LOOKUP — це значення або формула, результат якої ви шукаєте у вказаних далі діапазонах (масивах). У даному випадку формула — це пошук мінімального значення.

Завдання № 3

Побудова моделі прогнозування популярності

Створи модель, яка передбачає, чи стане пісня хітом, на основі її характеристик.

Дані про популярні та непопулярні пісні:

Пісня	Темп	Гучність	Енергійність	Танцювальність	Тривалість (сек)	Хіт?
Song1	120	6	7	8	180	Так
Song2	90	3	4	3	240	Ні
Song3	128	8	9	9	210	Так
…	…	…	…	…	…	…

Порада:

- Визнач середнє значення кожної характеристики хітів і не хітів.
- Обчисли Евклідові відстані від нової пісні до класів хітів і не хітів.
- Визнач, яка відстань менша і на основі цього зроби висновок, чи стане пісня хітом.

Завдання № 4

Міні-проєкт «Розумний плейлист»

Створи власну рекомендаційну систему для шкільної вечірки.

Етапи роботи:

1. Збери дані: Опитай не менше 10 однокласників / однокласниць про їхні улюблені пісні та попроси оцінити музичні характеристики. Можеш також додати пісні на власний розсуд.
2. Опрацюй дані, створивши:
   - Список пісень з характеристиками.
   - Систему класів пісень (енергійна, спокійна, для танців тощо). Вкажи, якому класу належить кожна пісня.
   - Таблицю, де для кожного користувача обчислюватиметься середнє значення кожної характеристики його улюблених пісень. Це буде інша система класифікації пісень, не за настроєм, а за вподобаннями користувача.
3. Реалізуй алгоритми:
   - Класифікації нових пісень за настроєм.
   - Автоматизованого створення плейлистів.
4. Протестуй систему:
   - Дай рекомендації кільком друзям.
   - Збери відгуки про точність.

5. Презентуй результати:
   - Продемонструй, як працює твоя система.
   - Поділися найцікавішими закономірностями.
   - Обговори з класом обмеження системи та можливості покращення.

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке машинне навчання простими словами?

Машинне навчання — це спосіб навчити комп'ютер робити висновки на основі прикладів, а не чітких інструкцій.

Якщо звичайна програма виконує лише те, що ми їй наказали, то в машинному навчанні ми даємо комп'ютеру багато даних (наприклад, оцінки пісень або фото тварин), і він шукає закономірності.

Потім, коли з'являється новий приклад (нова пісня чи нове фото), комп'ютер може передбачити результат: який це жанр, чи стане пісня хітом, яка тварина зображена тощо.

Простими словами: машинне навчання — це коли комп'ютер «вчиться на досвіді», подібно до того, як учні роблять висновки після кількох прикладів на уроці.

Які основні типи завдань розв'язує машинне навчання?

1. **Класифікація** — віднесення об'єктів до категорій (жанр музики, спам чи не спам).
2. **Регресія** — прогнозування числових значень (ціна, рейтинг, кількість переглядів).
3. **Кластеризація** — групування схожих об'єктів (наприклад, сегментація клієнтів на основі подібності вподобань).
4. **Рекомендаційні системи** — пропозиція релевантного (того, що може сподобатися) контенту.

Чому важливо розуміти основи машинного навчання сучасному учню / учениці?

Машинне навчання вже зараз впливає на твоє життя через:
- персоналізовані рекомендації в соцмережах;
- системи пошуку та перекладу;
- голосові помічники і чат-боти;
- системи безпеки та розпізнавання облич.

Розуміння основ машинного навчання допомагає:
- критично оцінювати алгоритмічні рекомендації;
- розуміти можливості та обмеження штучного інтелекту (ШІ);
- готуватися до професій майбутнього;
- приймати обґрунтовані рішення про використання технологій.

Які слабкі сторони алгоритмів машинного навчання?

1. **Якість даних.** Якщо дані недостовірні або містять помилки, то й результати будуть неправильними.
2. **Упередженість.** Якщо дані відображають стереотипи користувачів, то й алгоритм їх повторюватиме.
3. **Перенавчання.** Модель може «вивчити напам'ять» приклади, але не зрозуміти загальних правил. Тоді вона працюватиме добре тільки на старих прикладах, а на нових — ні.
4. **«Чорна скринька».** Іноді складно пояснити, чому комп'ютер зробив саме такий вибір чи прогноз.

   Етичні питання:
   - Чи справедливо дозволяти алгоритмам вирішувати щось важливе про людей?
   - Як зробити так, щоб робота ШІ була зрозумілою і прозорою для всіх?
   - Як захистити особисті дані користувачів від зловживань?
5. **Залежність від умов.** Якщо зміняться обставини (наприклад, прийдуть інші користувачі чи з'являться нові тренди), модель може стати некоректною.

Як і для чого використовують функцію SUMPRODUCT?

Функція SUMPRODUCT обчислює суму добутків відповідних значень у двох або більше масивах (діапазонах). Це означає, що вона перемножує відповідні елементи в кожному масиві, а потім додає всі отримані добутки.

Приклад використання

Уявіть, що у вас є список товарів, кількості товарів та ціни.

	A	B	D
1	Товар	Кількість	Ціна
2	Футболка M	5	100 грн
3	Футболка L	3	120 грн
4	Футболка XL	2	150 грн

Щоб обчислити загальну вартість, ви можете використати формулу =SUMPRODUCT(B2:B4;D2:D4). Вона виконає такі обчислення:

$$(5 \times 100) + (3 \times 120) + (2 \times 150) = 500 + 360 + 300 = 1160 \text{ грн.}$$

Як і для чого використовують функцію XLOOKUP?

Синтаксис функції є таким: XLOOKUP(формула;масив1;масив2). Масив1 і масив2 — це вертикальні або горизонтальні діапазони однакового розміру. Функція шукає результат вказаної як перший аргумент

формули у масиві1 і повертає значення з відповідної клітинки у масиві2. Функція дуже зручна, коли потрібно знайти значення в одному стовпці таблиці і отримати значення зі знайденого рядка, але іншого стовпця.

Наприклад, щоб у зображеній таблиці знайти постачальника цукерок, можна скористатися функцією =XLOOKUP("цукерки";B2:B4;A2:A4). Результатом застосування цієї функції буде значення з клітинки А3, тобто «Наші ласощі».

	A	B
1	Постачальник	Товар
2	Оріон	хліб
3	Наші ласощі	цукерки
4	Оріон	вода

ПЕРЕВІР СЕБЕ

ЗНАННЯ — РОЗУМІННЯ — АНАЛІЗ — СТВОРЕННЯ

1. Поясни, що таке машинне навчання, одним реченням.
2. Назви дві відмінності між звичайним середнім і зваженим середнім.
3. Що означає «найближчий сусід» у класифікації?
4. Для чого використовують AVERAGEIF і DCOUNT?
5. Чому Евклідова відстань підходить для порівняння пісень за кількома ознаками?
6. Розглянь таблицю з даними про прогноз температури в певній місцевості, отриманими від трьох різних метеостанцій. Порівнявши їх зі справжніми значеннями температури за 1–7 листопада, визнач:

   1) коефіцієнти достовірності (ваги) показів кожної метеостанцій;

   2) прогноз температури в даній місцевості на 8 листопада, як середнє арифметичне зважене прогнозів трьох метеостанцій.

	01.Лис	02.Лис	03.Лис	04.Лис	05.Лис	06.Лис	07.Лис	08.Лис
Метеостанція 1	+3	0	-2	-4	+1	+7	+6	+4
Метеостанція 2	+2	-1	-2	-5	0	+6	+5	+6
Метеостанція 3	+3	+1	0	-2	-5	+2	+6	+3
Справжня температура	+2	0	-1	-3	0	+3	+6	?

7. Є два способи оцінити діяльність учня:
   А) яку частку завдань він виконав (скільки з усіх);
   Б) середній бал за виконані завдання (наскільки якісно зроблено).

Який спосіб оцінювання — А чи Б — ти обереш, якщо учень пройшов:

1) Курс із багатьма короткими вправами.
2) Курс із трьома великими проєктами.
3) Придумай свою змішану формулу оцінювання діяльності учня, що враховує обидва способи А і Б.

Наведи короткий приклад обчислення оцінки за власною формулою (підстав числа і порахуй).

8. Учень або учениця має такі результати:
   - контрольна робота — **9 балів** (вага **50 %**);
   - проєкт — **8 балів** (вага **30 %**);
   - тести — **10 балів** (вага **20 %**).

   Обчисли підсумкову оцінку (зважене середнє).
9. За критеріями Солодкість — Кислотність — Газованість лимонад має оцінку (7, 3, 8), а айран (2, 7, 1). Визнач, до чого ближчий напій із критеріями (6, 4, 7) — лимонаду чи айрану?
10. За критеріями Солодкість–Кислотність–Газованість є дані про 3 кисломолочні продукти з оцінками (2,9,2), (4,8,1), (5,8,3), а також 3 прохолоджувальні напої з оцінками (9,3,9), (2,1,9), (6,4,2). Визнач, до якого з цих двох класів скоріше належить напій з оцінками (3,3,8).

Додатково

Simple ML for Sheets — це безкоштовне розширення для Google Таблиць, яке створила команда TensorFlow. Воно дає змогу виконувати машинне навчання без написання коду. Користувач може взяти таблицю з даними й навчити просту модель просто в інтерфейсі Google Sheets.

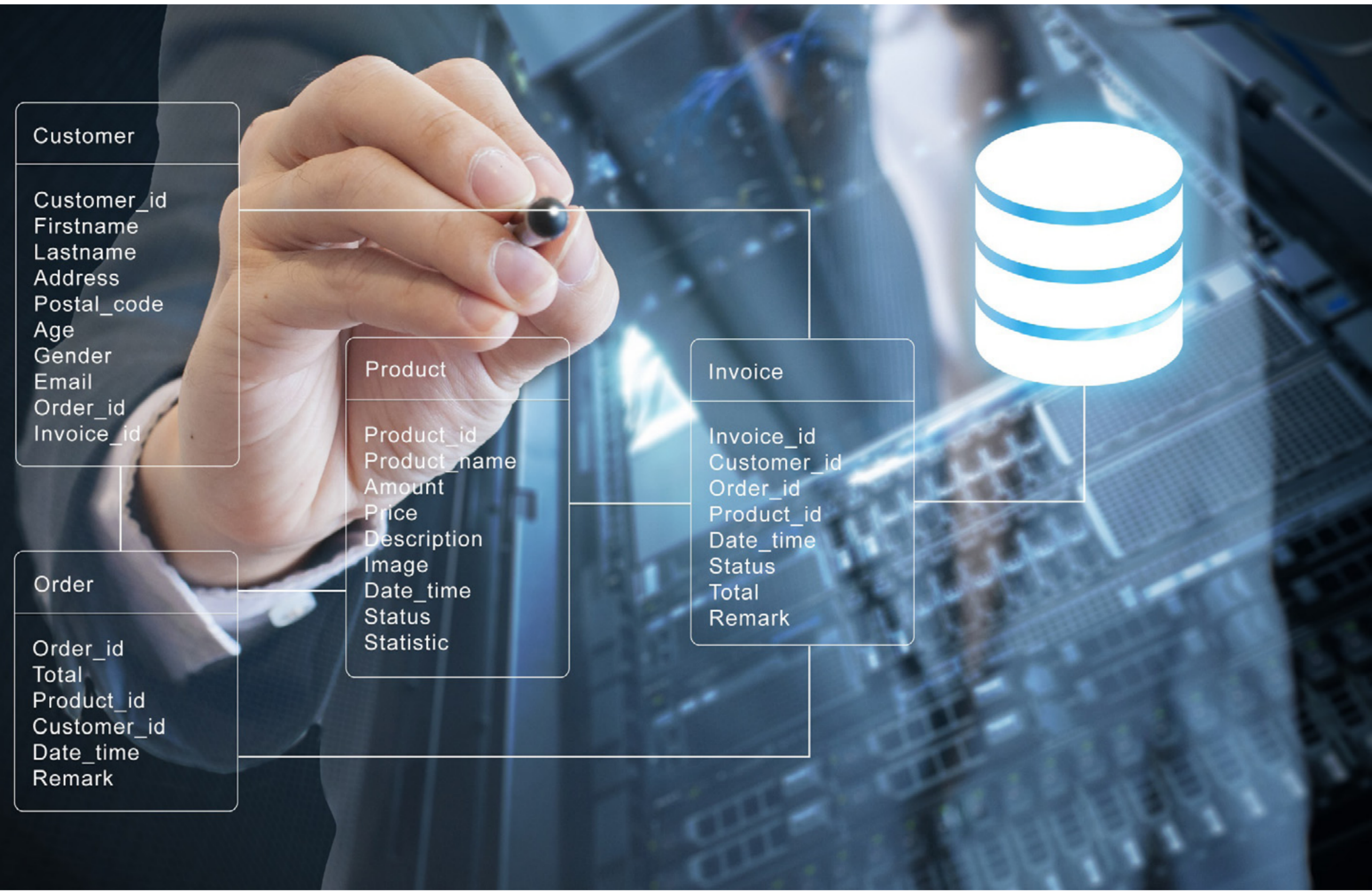
Який алгоритм роботи з розширенням, наприклад для розв’язання завдання «Передбачити значення для порожніх клітинок»?

1. Встановити доповнення Simple ML for Sheets із Google Workspace Marketplace (https://workspace.google.com/marketplace/).
2. Відкрити таблицю з даними.

   У меню обрати: Extensions → Simple ML for Sheets → Start.
3. На правій панелі обрати пункт Predict missing values та вказати стовпець, у який потрібно додати прогноз, а також стовпці з даними, які впливають на прогноз.
4. Натиснути кнопку Predict — і у таблиці з’являться нові стовпці з прогнозами.

РОЗДІЛ ІІ

БАЗИ ДАНИХ. СИСТЕМИ КЕРУВАННЯ БАЗАМИ ДАНИХ



§ 6. Основні поняття баз даних

§ 7. Ознайомлення із системою керування базами даних

§ 8. Обмеження цілісності

§ 9. Вибір даних

§ 6. Основні поняття баз даних

Чому для збереження табличних даних може бути недостатньо однієї електронної таблиці?

Як ти вже знаєш, таблиця — це зручний спосіб зберігання відомостей про набори однотипних об’єктів. Кожен рядок у таблиці відповідає певному об’єкту, а стовпець — певній властивості об’єктів. А якщо таких наборів кілька? Очевидно, для їх зберігання потрібно кілька таблиць! І ці таблиці зазвичай будуть взаємопов’язані. Такі системи взаємопов’язаних таблиць називаються реляційними (табличними) базами даних. І їх ми вивчатимемо в курсі інформатики в 9 класі.

Завдання № 1

1) Відомості про які об’єкти зберігаються в зображеній таблиці?
2) Наведи приклади об’єктів, їхніх властивостей та значень властивостей із зображеної таблиці.

Номерний знак	Модель	Рік випуску	Маршрут	Довжина (км)	Кількість зупинок
AA3129KH	CityLAZ-12	2010	9	7,3	18
AA1684PT	Богдан А144	2017	9	7,3	18
AI4124KK	CityLAZ-12	2012	9	7,3	18
AA3312TB	МАЗ 103	2001	9	7,3	18
AI2917KT	Богдан А144	2017	12	5,4	14
AA4409KH	CityLAZ-12	2012	12	5,4	14

Завдання № 2

У зображеній вище таблиці зберігаються відомості як про автобуси, так і про маршрути, якими ці автобуси курсують. Таке зберігання в одній таблиці інформації про різнотипні об’єкти має недоліки. Спробуй визначити, які саме, відповівши на запитання.

1) У скількох рядках містяться відомості про маршрут № 9 і про маршрут № 12?

2) Припустимо, що маршрут № 9 продовжили. Тепер його довжина становить 9 км і він має 20 зупинок. Значення у скількох клітинках таблиці потрібно оновити, щоб відобразити ці зміни?
3) Припустимо, що відомості про маршрут № 9 у деяких рядках оновили, а в деяких — залишили старими. Поясни, чому в такому випадку відомості про маршрут № 9 буде втрачено.
4) Поясни, що означає принцип ненадлишковості даних і чому через надлишковість дані може бути втрачено. Можеш скористатися рубрикою «Запитання-відповіді».

Завдання № 3

Для уникнення надлишковості таблицю потрібно поділити на дві: «Автобуси та Маршрути».

Номерний знак	Модель	Рік випуску
AA3129KH	CityLAZ-12	2010
AA1684PT	Богдан A144	2017
AI4124KK	CityLAZ-12	2012
AA3312TB	МАЗ 103	2001
AI2917KT	Богдан A144	2017
AA4409KH	CityLAZ-12	2012

Маршрут	Довжина (км)	Кількість зупинок
9	7,3	18
12	5,4	14

Однак тепер незрозуміло, який автобус яким маршрутом курсує. Нижче описано два способи, якими це можна вказати. Один із цих способів значно кращий за інший. Обери кращий спосіб і поясни, які недоліки має гірший. Можеш скористатися рубрикою «Запитання-відповіді».

- До таблиці «Автобуси» додати стовпець «Номер маршруту», де вказувати маршрут, яким курсує автобус.
- До таблиці «Маршрути» додати стовпець «Номер автобуса», де вказувати номери автобусів, що курсують цим маршрутом.

Завдання № 4

1) Поясни, що таке ключ таблиці.
2) Для кожної з наведених нижче таблиць укажи, які стовпці можуть бути ключовими.
3) Чому стовпець «Прізвище» в таблиці «Люди» не можна вважати ключовим, хоча значення в ньому не повторюються?

Люди

№ паспорта	Прізвище	Ім’я	Рік народження
КМ 381421	Круглєня	Дмитро	2007
РТ 905129	Жук	Ольга	2006
РС 140418	Кашпур	Ольга	2006
НС 781392	Шпитько	Микита	2006

Ноутбуки

Виробник	Модель	Процесор	Серійний номер
Asus	Asus Vivobook 15	AMD Ryzen 7	FNBL34-701835
Acer	Aspire 7	Intel Core i5	XD-05-41895632
Asus	E502M	Intel Byt-M 2Core	FMXD77-394777
HP	1U9R5EA	AMD Ryzen 5	31280540037

Автобуси

Номерний знак	Модель	Рік випуску
AA3129KH	CityLAZ-12	2010
AA1684PT	Богдан А144	2017
AI4124KK	CityLAZ-12	2012
AA3312TB	МАЗ 103	2001
AI2917KT	Богдан А144	2017
AA4409KH	CityLAZ-12	2012

Маршрути

Номер	Довжина (км)	Кількість зупинок
9	7,3	18
12	5,4	14

Завдання № 5

Для кожної пари зображених таблиць опиши зв’язок, який можна між ними організувати, а також поясни, як саме це зробити.

Вулиці

Назва	Довжина (км)	Тип
Вереснева	1,1	вулиця
Ремонтний	0,3	провулок
Шевченка	2,3	бульвар
Перемоги	12	проспект

Будинки

Номер	Кількість поверхів	Кількість квартир
11	16	127
9	16	127
8	9	104
8а	3	14

Планети

Назва	Маса (кг)	Радіус (км)
Венера	4,869E+24	6052
Земля	5,974E+24	6371
Марс	6,419E+23	3376
Юпітер	1,899E+27	69911

Супутники

Назва	Маса (кг)	Радіус (км)
Місяць	7,347E+22	1737
Титан	1,345E+23	2575
Іо	8,921E+22	1821
Ганімед	1,482E+23	2634

Ноутбуки

Виробник	Модель	Екран
Apple	MacBook Air M1	13,3
Asus	M513IA-BQ533	15,6
Lenovo	IdeaPAD 3	15,6
Dell	Latitude 5411	14

Процесори

Виробник	Модель	Швидкість, Мгц
Intel	i5	4100
Intel	i7	4700
AMD	Ryzen 9	3700
Apple	M1	3200

ЗАПИТАННЯ-ВІДПОВІДІ

Чому зберігання даних у таблиці може бути надлишковим?

Надлишковість означає багаторазове повторення тієї самої інформації. На малюнку наведено приклад таблиці, у якій відомості про номер квартири, її площу та поверх повторюються три рази.

Номер квартири	Площа квартири (кв. м)	Поверх	Прізвище	Ім’я	Стать
41	39	1	Огороднік	Юлія	ж
41	39	1	Огороднік	Євген	ч
41	39	1	Огороднік	Микола	ч
87	77	9	Чубенко	Іван	ч
87	77	9	Чубенко	Інна	ж

Найпоширенішою причиною надлишковості є подання в одній таблиці відомостей про різнорідні об’єкти. Наприклад, у зображеній вище таблиці подано відомості і про квартиру, і про її мешканців. Через це відомості про квартиру доводиться повторювати стільки разів, скільки в ній мешкає людей.

Які недоліки має надлишкове зберігання даних?

Основні недоліки надлишковості даних:

- Витрачається зайва пам’ять для зберігання даних.
- У разі змінення даних доводиться вносити зміни багаторазово.
- Виникає можливість ситуації, коли про один і той самий об’єкт будуть зберігатися суперечливі відомості. Наприклад, у зображеній таблиці квартира № 41 має площу і 39 кв. м, і 65 кв. м водночас, розташована водночас і на 1-му, і на 5-му поверсі.

Номер квартири	Площа квартири (кв. м)	Поверх	Прізвище	Ім’я	Стать
41	39	1	Огороднік	Юлія	ж
41	39	1	Огороднік	Євген	ч
41	39	1	Огороднік	Микола	ч
87	77	9	Чубенко	Іван	ч
87	77	9	Чубенко	Інна	ж

Отже, надлишковість призводить до суперечливості даних, а суперечливі дані — це втрачені дані. Адже із зображеної таблиці неможливо видобути інформацію про справжню площу квартири № 41 і поверх, на якому вона розташована. Цю інформацію втрачено.

Як уникнути надлишкового зберігання даних?

Найчастіше причиною надлишковості є зберігання відомостей про різнотипні об’єкти в одній таблиці. Тоді цю таблицю потрібно поділити. У кожній з отриманих у результаті таблиць мають зберігатися відомості лише про один тип об’єктів.

Квартири та люди

Номер квартири	Площа квартири (кв. м)	Поверх	Прізвище	Ім’я	Стать
41	39	1	Огороднік	Юлія	ж
41	39	1	Огороднік	Євген	ч
41	39	1	Огороднік	Микола	ч
87	77	9	Чубенко	Іван	ч
87	77	9	Чубенко	Інна	ж

Квартири

Номер квартири	Площа квартири (кв. м)	Поверх
41	39	1
87	77	9

Люди

Прізвище	Ім’я	Стать
Огороднік	Юлія	ж
Огороднік	Євген	ч
Огороднік	Микола	ч
Чубенко	Іван	ч
Чубенко	Інна	ж

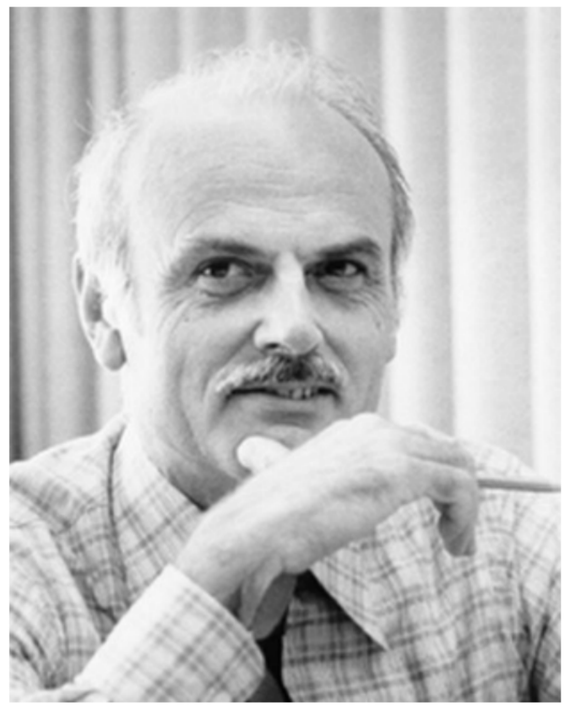
Що таке база даних?

База даних — це така структура даних, яка гарантує їхню узгодженість і мінімізує можливість втрати інформації. Щоб забезпечити ці властивості, проєктувальники баз даних керуються, як правило, принципом *мінімальної надлишковості*.

Які бази даних називаються реляційними?

Реляційними називаються бази даних, у яких усі дані зберігаються в таблицях. Слово «реляційний» походить від англійського relation. Цей термін перекладається як «відношення» і позначає математичний об'єкт, який найзручніше зображувати у вигляді таблиці.

Реляційна модель даних була запропонована в 1970 р. британським математиком Е. Коддом. У 1980-х роках бази даних реляційного типу стали домінуючими і залишаються такими дотепер.



Едгар Кодд

ЦЕ ЦІКАВО

У 1960-х роках Стефані Ширлі керувала компанією, яка виконувала замовлення з обробки даних для уряду Великої Британії. Сьогодні такі завдання ми виконуємо самостійно за допомогою електронних таблиць і функцій агрегації.

Які ще бувають різновиди баз даних, крім реляційних?

У «дореляційну» епоху найпоширенішими були ієрархічні та мережеві бази даних. Дані в цих базах зберігалися не в таблицях, а у вигляді інших структур, елементи яких вказували один на один. (Зауважте, що в таблицях ніяких «вказівників» немає.) Перевагою ієрархічних та мережевих систем була можливість забезпечувати швидший доступ до даних, а недоліком — складність роботи з ними. Зі зростанням потужності комп'ютерів ця перевага стала не дуже суттєвою, однак останнім часом, коли виникла потреба опрацьовувати величезні обсяги даних з інтернету, питання швидкодії знову виходить на передній план і новітні нереляційні підходи до зберігання даних поступово здобувають своє «місце під сонцем».

Як співвідносяться дані та елементи таблиць?

У кожному рядку таблиці зберігаються відомості про певний об'єкт, а таблиця в цілому містить інформацію про набір *однотипних об'єктів*. Кожен стовпець відповідає певній властивості таких об'єктів, а клітинка — значенню властивості для конкретного об'єкта.

Ноутбуки

Виробник	Модель	Екран
Apple	MacBook Air M1	13,3
Asus	M513IA-BQ533	15,6
Lenovo	IdeaPAD 3	15,6
Dell	Latitude 5411	14

Що таке ключ таблиці?

Ключем називається стовпець або набір стовпців таблиці, значення у яких є унікальними, а отже, можуть ідентифікувати об’єкти. Унікальність означає, що значення не можуть повторюватися (не лише для конкретного наповнення таблиці, а й для будь-якого можливого наповнення).

Що означає зв’язок «один-до-багатьох» між таблицями?

Між таблицями A і B існує зв’язок «один-до-багатьох», якщо:

1) кожному об’єкту з таблиці A може відповідати будь-яка кількість об’єктів із таблиці B;
2) кожному об’єкту з таблиці B може відповідати не більше одного об’єкта з таблиці A.

Наприклад, у кожній квартирі може мешкати різна кількість людей, але кожна людина проживає лише в одній квартирі.

Як створити зв’язок «один-до-багатьох»?

Для цього потрібно в таблицю, що розташована з боку «багато», додати стовпець і записати в ньому значення ключів відповідних об’єктів іншої таблиці.

Наприклад, у таблицю «Люди» додаємо стовпець «Квартира» і вказуємо в ньому номери квартир, де люди проживають. Ці номери мають бути значеннями з ключового стовпця таблиці «Квартири».

Прізвище	Ім’я	Стать	Номер квартири
Огороднік	Юлія	ж	41
Огороднік	Євген	ч	41
Огороднік	Микола	ч	41
Чубенко	Іван	ч	87
Чубенко	Інна	ж	87

Номер квартири	Площа квартири (кв. м)	Поверх
41	39	1
87	77	9

ПЕРЕВІР СЕБЕ

ЗНАННЯ — РОЗУМІННЯ — АНАЛІЗ — ВЛАСНІ ДУМКИ

1. Які елементи таблиці відповідають поняттям «об’єкт», «властивість об’єкта», «значення властивості об’єкта»?
2. Чому саме той стовпець, дані у якому не повторюються, може бути ключовим?
3. Поясни, що означає фраза «таблиці A і B пов’язані зв’язком один-до-багатьох».

4. Чому під час моделювання зображеного зв’язку «один-до-багатьох» між таблицями A і B додатковий стовпець створюють саме в таблиці B, а не A?
5. Між якими парами зображених таблиць можна створити зв’язки «один-до-багатьох»? Опиши ці зв’язки словами та поясни, чому їх можна охарактеризувати як «один-до-багатьох».

Країни

Назва	Площа (кв. км)
Україна	603 548

Міста

Назва	Населення (осіб)
Київ	2 954 700

Ріки

Назва	Довжина (км)
Дніпро	2285

Моря

Назва	Площа (кв. км)
Чорне	422 000

6. Очевидно, що одна людина може прочитати багато книжок. Однак зв’язок між таблицями «Люди» та «Книжки» не є зв’язком типу «один-до-багатьох». Поясни чому.
7. Наведи приклади сутностей, крім розглянутих у цьому параграфі, між якими можна було б встановити зв’язок «один-до-багатьох».
8. Який існує зв’язок між традиційним значенням слова «ключ» і значенням цього терміна в базах даних?
9. Поясни переваги баз даних над електронними таблицями.

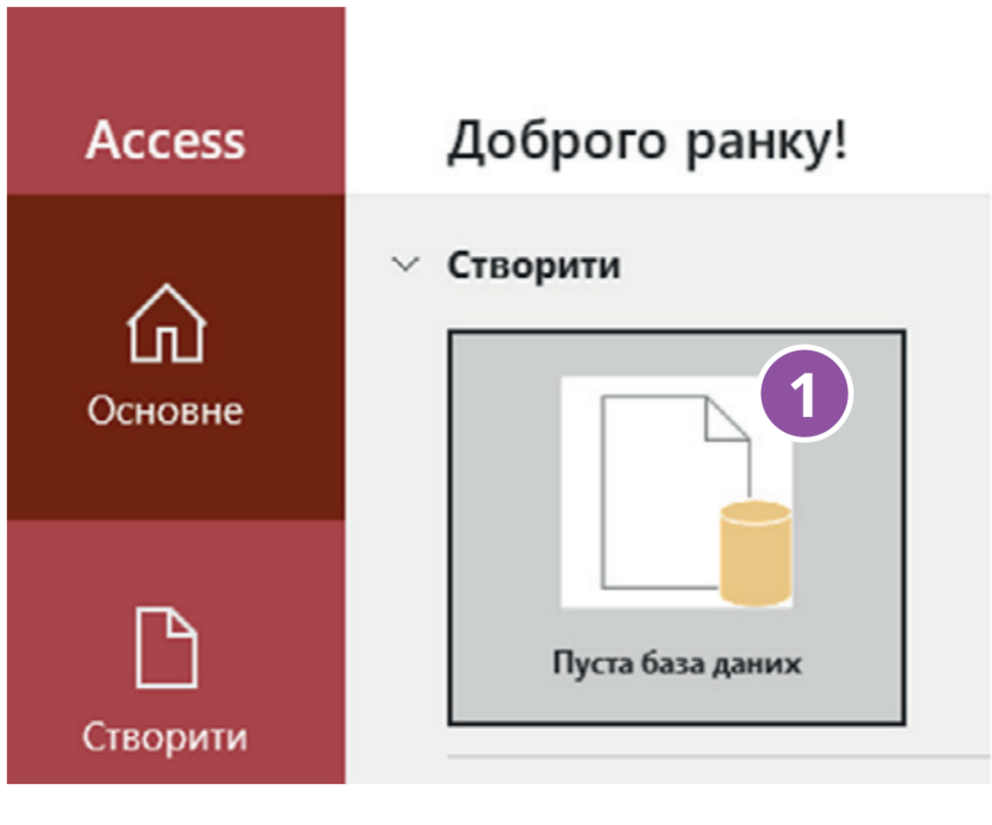
§ 7. Ознайомлення із системою керування базами даних

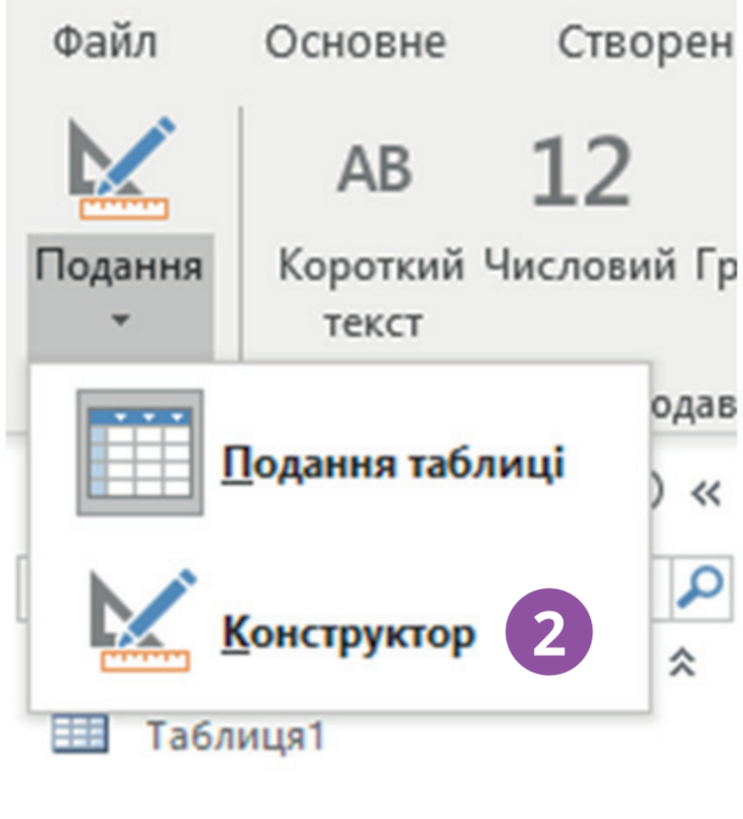
У чому основна відмінність системи керування базами даних від табличного процесора?

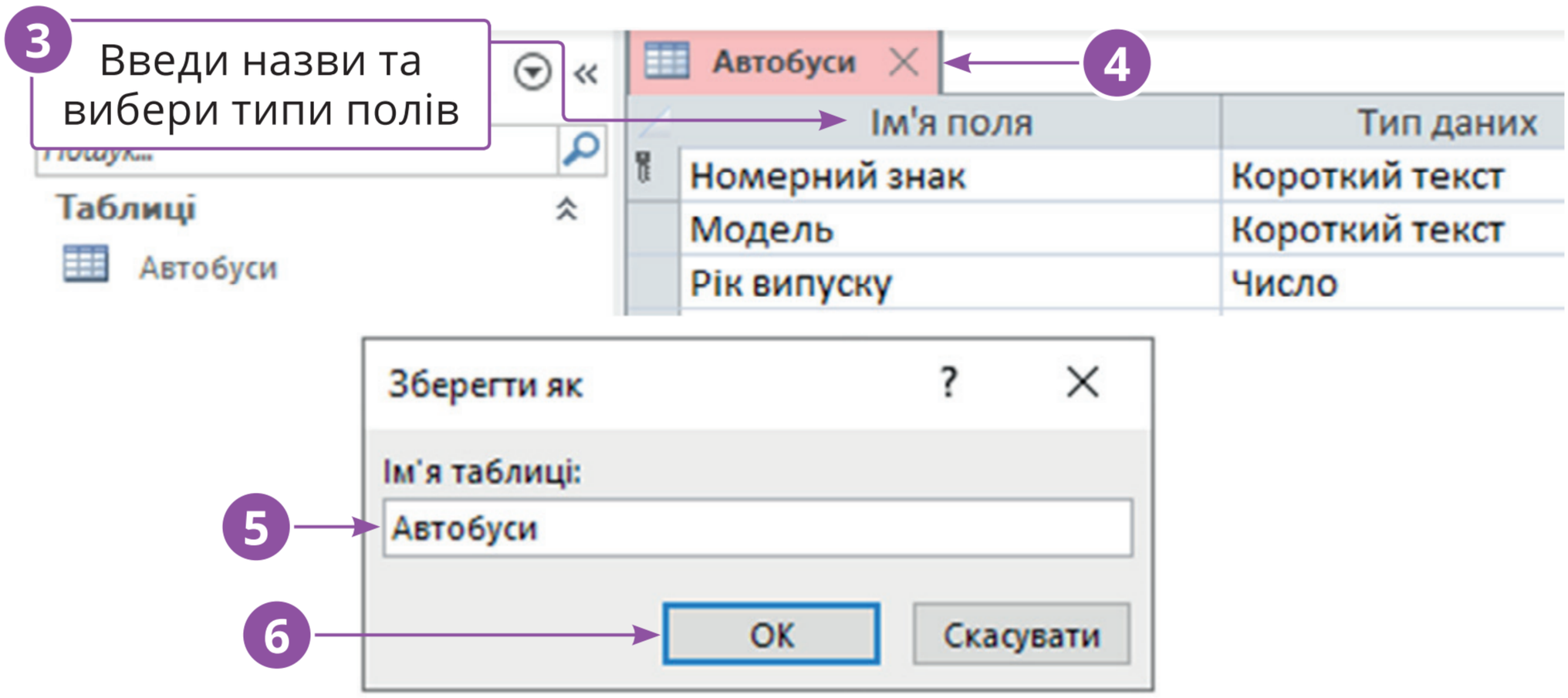
На попередньому занятті ми з'ясували, коли необхідно зберігати дані в багатьох таблицях, тобто в табличній базі даних, а також спроєктували просту базу даних із двох таблиць. Сьогодні ми реалізуємо цей проєкт у спеціальній програмі, призначеній для роботи з такими структурами, — системі керування базами даних (СКБД) Microsoft Access.

Завдання № 1

Запусти СКБД Microsoft Access. Керуючись зображеною послідовністю дій, створи нову базу даних, а в ній — таблицю «Автобуси» з полями «Номерний знак», «Модель», «Рік випуску». Добери доречні типи даних для полів.

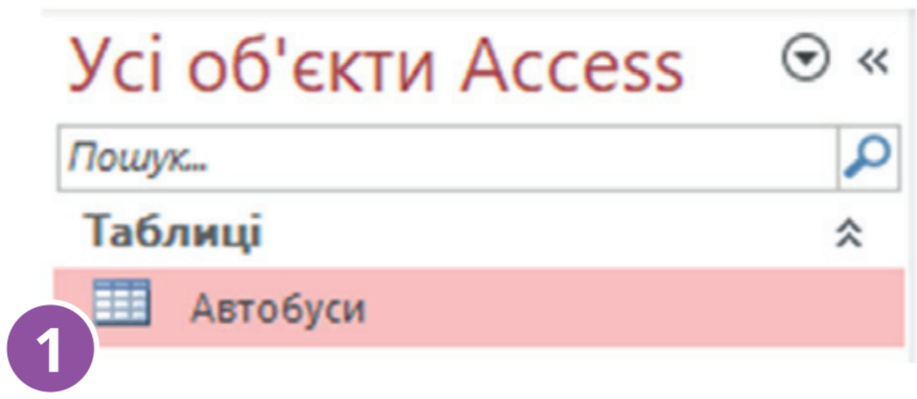


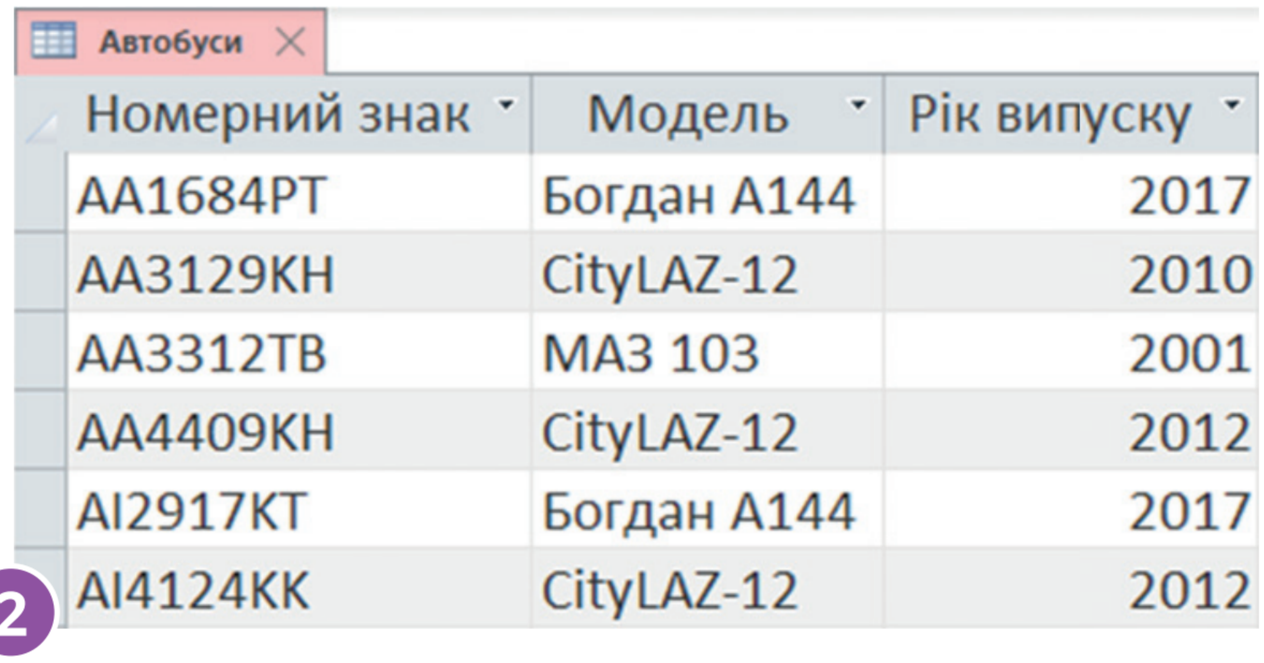




Завдання № 2

Відкрий таблицю «Автобуси», двічі клацнувши її значок в області «Усі об'єкти Access». Введи в таблицю «Автобуси» відомості про шість автобусів.





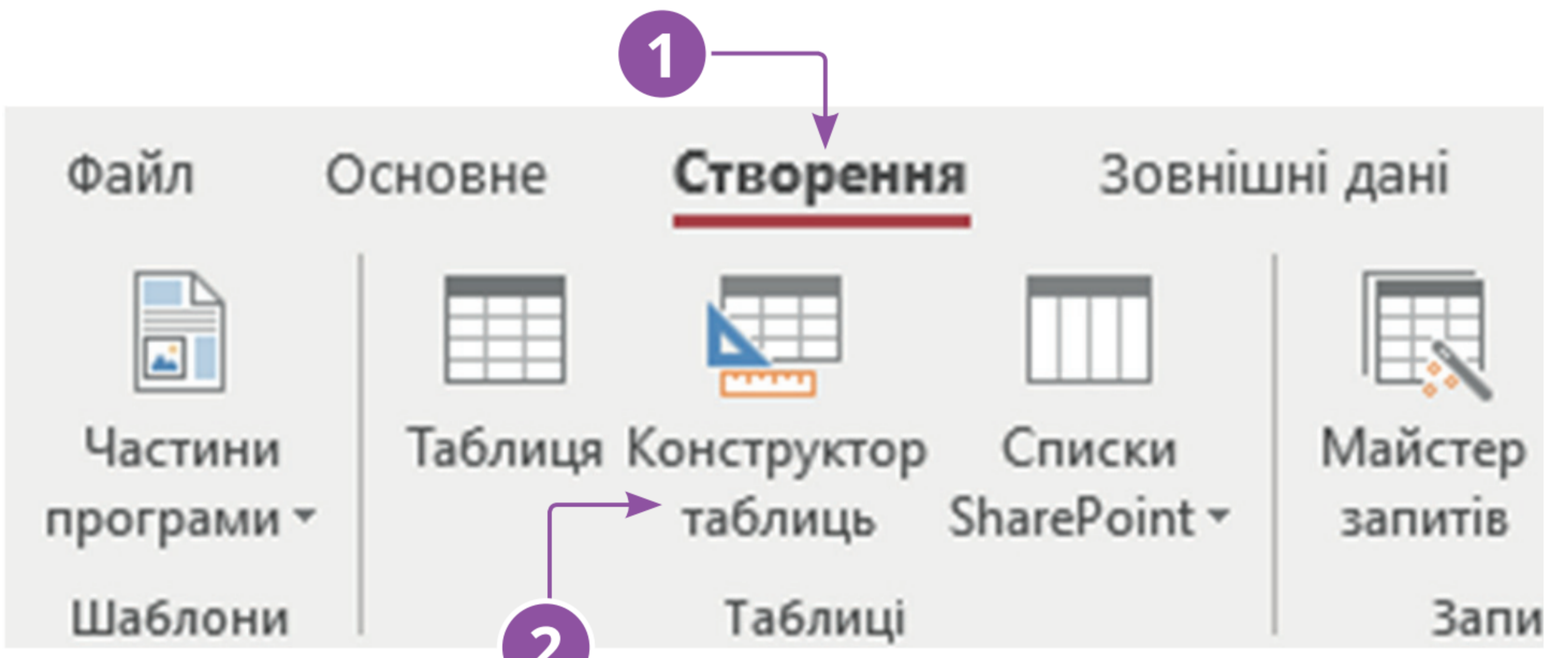
Завдання № 3

Скориставшись рубрикою «Запитання-відповіді» та самою СКБД, дай відповіді на запитання.

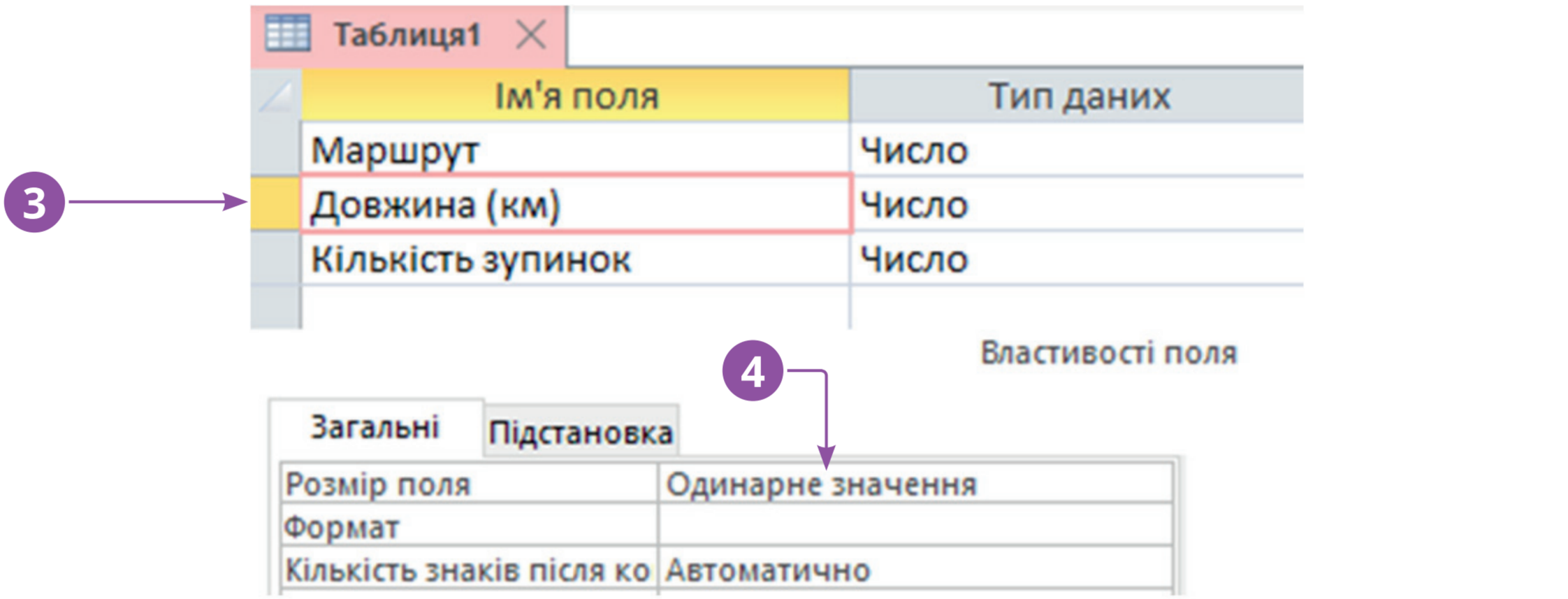
1. Яким елементам таблиці відповідають поняття «поле» і «запис»?
2. Чим відрізняється режим конструктора таблиці від режиму редагування даних?
3. Як перейти в режими конструктора таблиці та редагування даних?
4. Які типи даних можна надати полям таблиці в Microsoft Access?

Завдання № 4

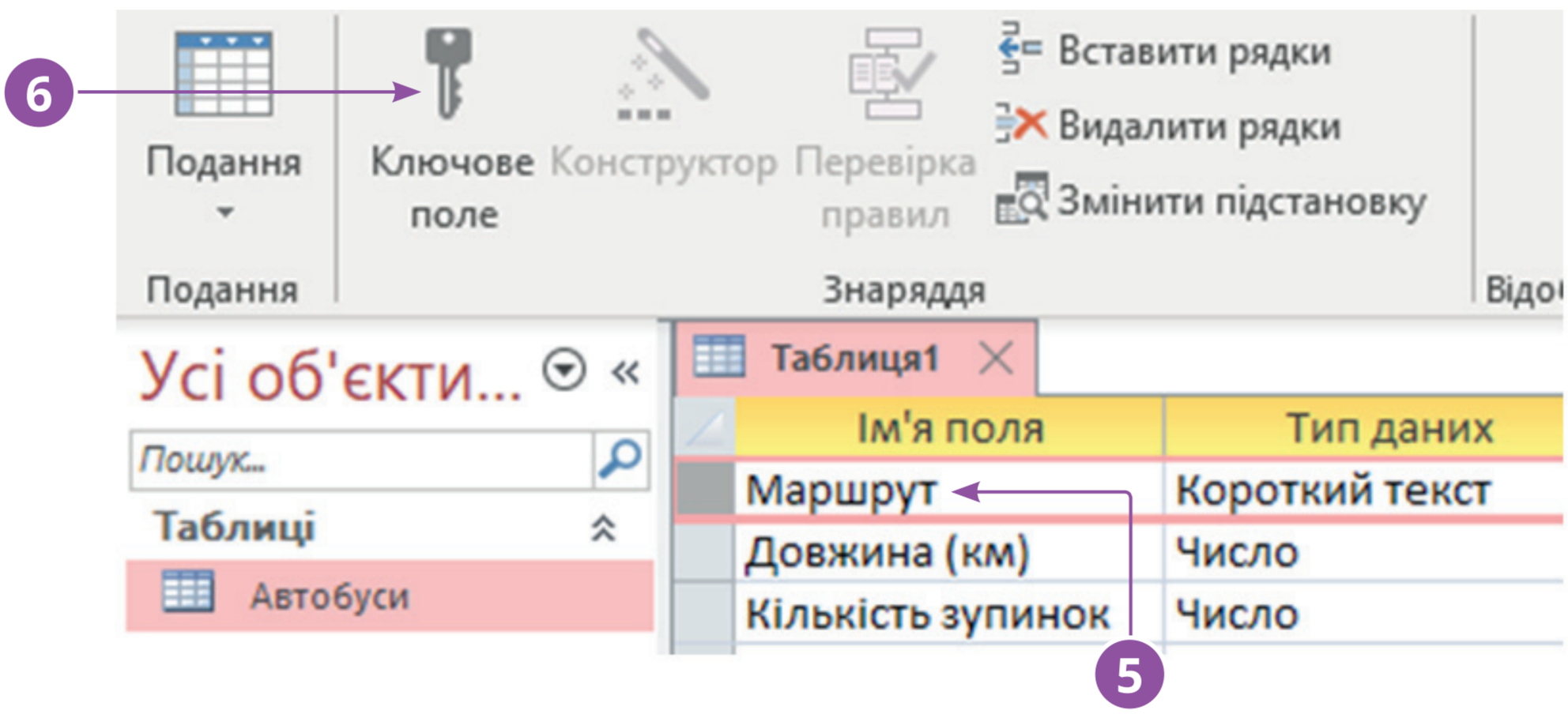
1. Керуючись зображеною послідовністю дій, створи таблицю «Маршрути».



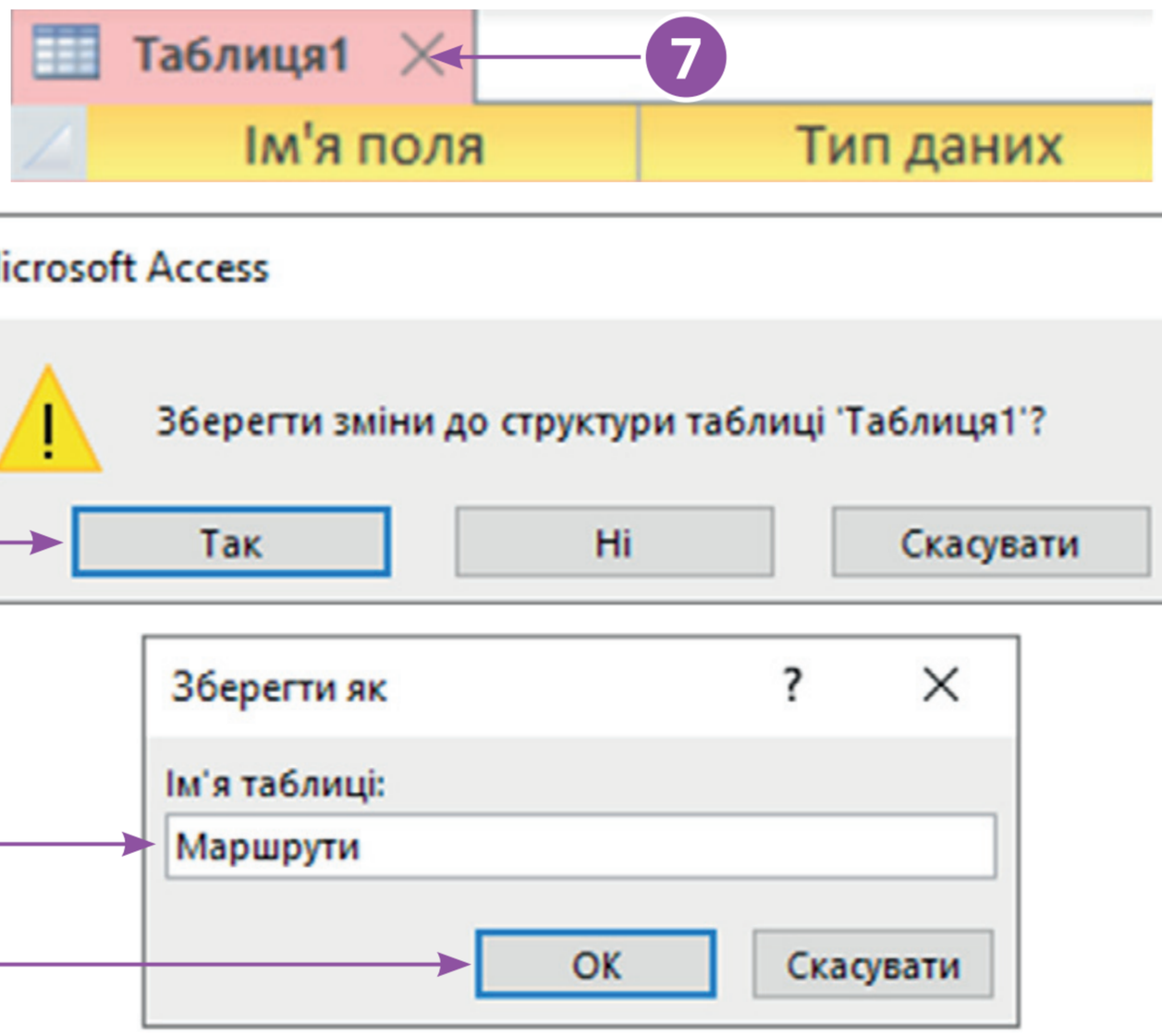
Щоб можна було вводити нецілі значення довжин маршрутів, для поля «Довжина» вибери тип даних «Число» і внизу на вкладці «Властивості поля» для властивості «Розмір поля» вибери «Одинарне значення».



2. Задай для таблиці «Маршрути» ключ. Для цього у вікні конструктора виділи поле «Маршрут» і натисни кнопку «Ключове поле».



3. Закрий конструктор таблиці «Маршрути» і введи її ім’я.



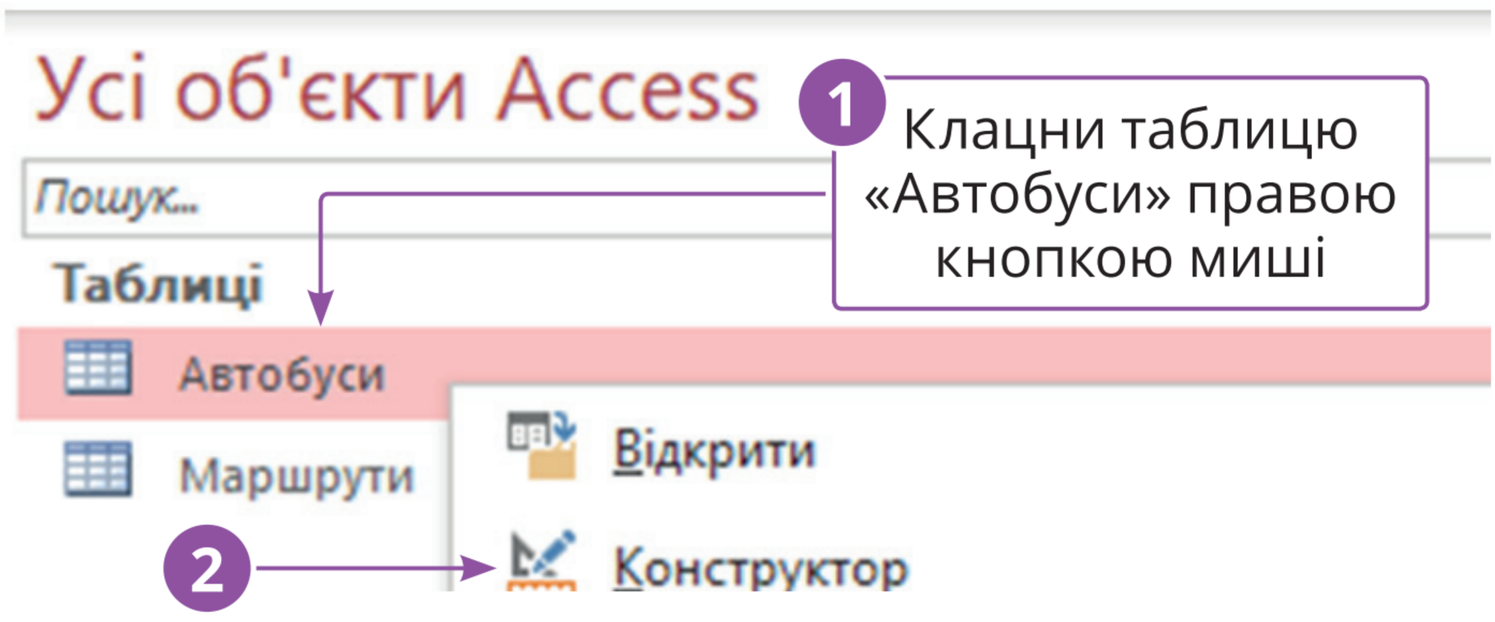
4. Заповни таблицю «Маршрути» даними про три маршрути.

Маршрути

Маршрут	Довжина (км)	Кількість зупинок
11	8	18
12	5,4	14
9	7,3	18

Завдання № 5

Керуючись зображеною послідовністю дій, перейди в режим конструктора таблиці «Автобуси» і додай до цієї таблиці стовпець «Номер маршруту».



Автобуси

Ім'я поля	Тип даних
Номерний знак	Короткий текст
Модель	Короткий текст
Рік випуску	Число
Маршрут	Число

3

Завдання № 6

Введи дані про те, які автобуси якими маршрутами курсують:

- автобуси AA3129KH та AA1684PT — маршрутом № 9;
- автобуси AI4124KK та AA3312TB — маршрутом № 11;
- автобуси AI2917KT та AA4409KH — маршрутом № 12.

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке система керування базами даних?

Система керування базами даних (СКБД) — це програмне забезпечення, яке дає користувачам можливість створювати і змінювати бази даних, вводити в них і отримувати з них інформацію.

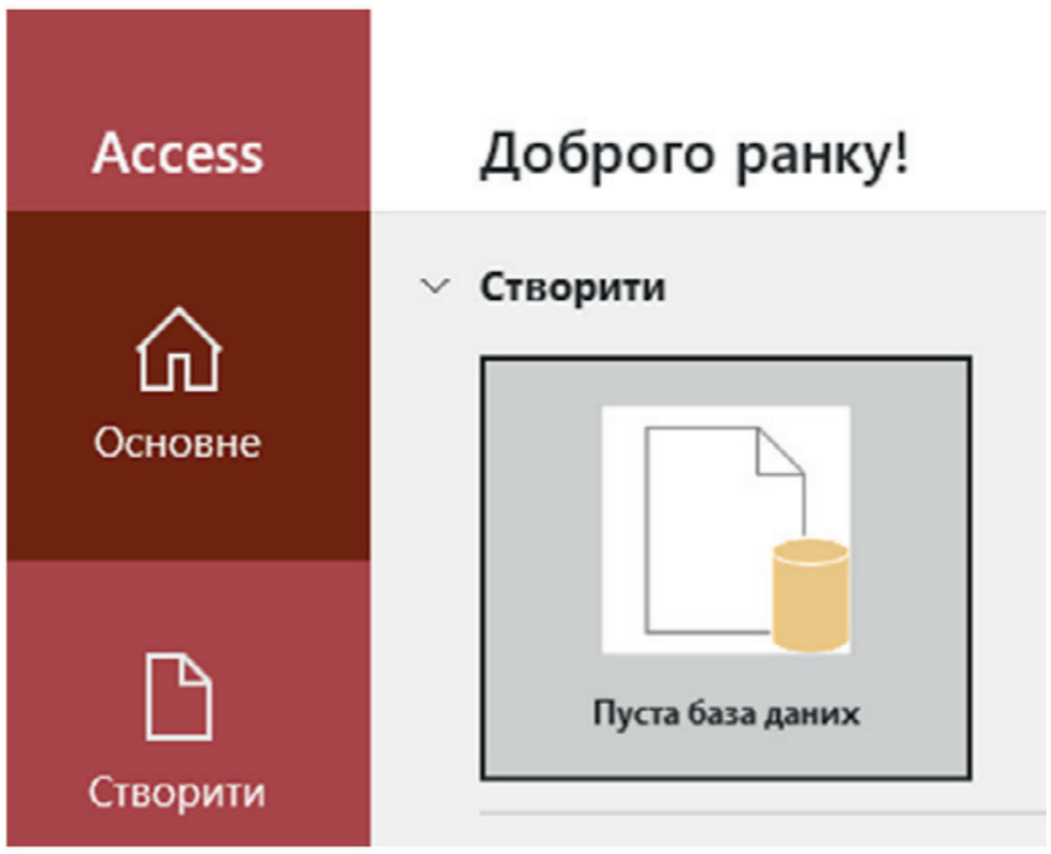
Які є популярні СКБД?

Є десятки широко відомих СКБД. Серед найпопулярніших реляційних СКБД варто виділити такі.

- **Microsoft Access** — одна з найпопулярніших у світі навчальних СКБД. Крім засобів створення таблиць та маніпулювання даними, які є в усіх СКБД, містить також засоби розробки інтерфейсу користувача для роботи з базою даних. Таким чином, у MS Access можна створити не лише базу даних, а й повноцінний програмний проєкт для роботи з нею.
- **Microsoft SQL Server** — промислова СКБД від корпорації Microsoft. Установлюється на сервері і надає доступ до даних програмам, що можуть виконуватися як на тому ж сервері, так і на віддалених комп'ютерах. Забезпечує значно вищу ефективність у роботі з великими базами даних, ніж Microsoft Access.
- **SQLite** — «полегшена» СКБД, що здобула популярність останніми роками завдяки своїй простоті. Важливою перевагою цієї системи є те, що її не потрібно окремо встановлювати на комп'ютері, навіть якщо з нею працюють прикладні програми. Адже ядро цієї СКБД може вбудовуватися безпосередньо в програму, що її використовує.
- **MySQL** — вільнопоширювана СКБД, створена для прискорення операцій із великими базами даних. Набула найбільшого застосування у вебпроєктах, оскільки є дуже зручною у взаємодії з мовою генерування вебсторінок PHP.
- **Oracle** — найпотужніша з альтернатив Microsoft SQL Server, насамперед для інших операційних систем, ніж Windows.

Зауваж, що в назвах багатьох СКБД трапляється абревіатура SQL, що розшифровується як Structured Query Language. Так називається найпоширеніша скриптова мова для роботи з реляційними базами даних.

Як створити та зберегти базу даних в Microsoft Access?

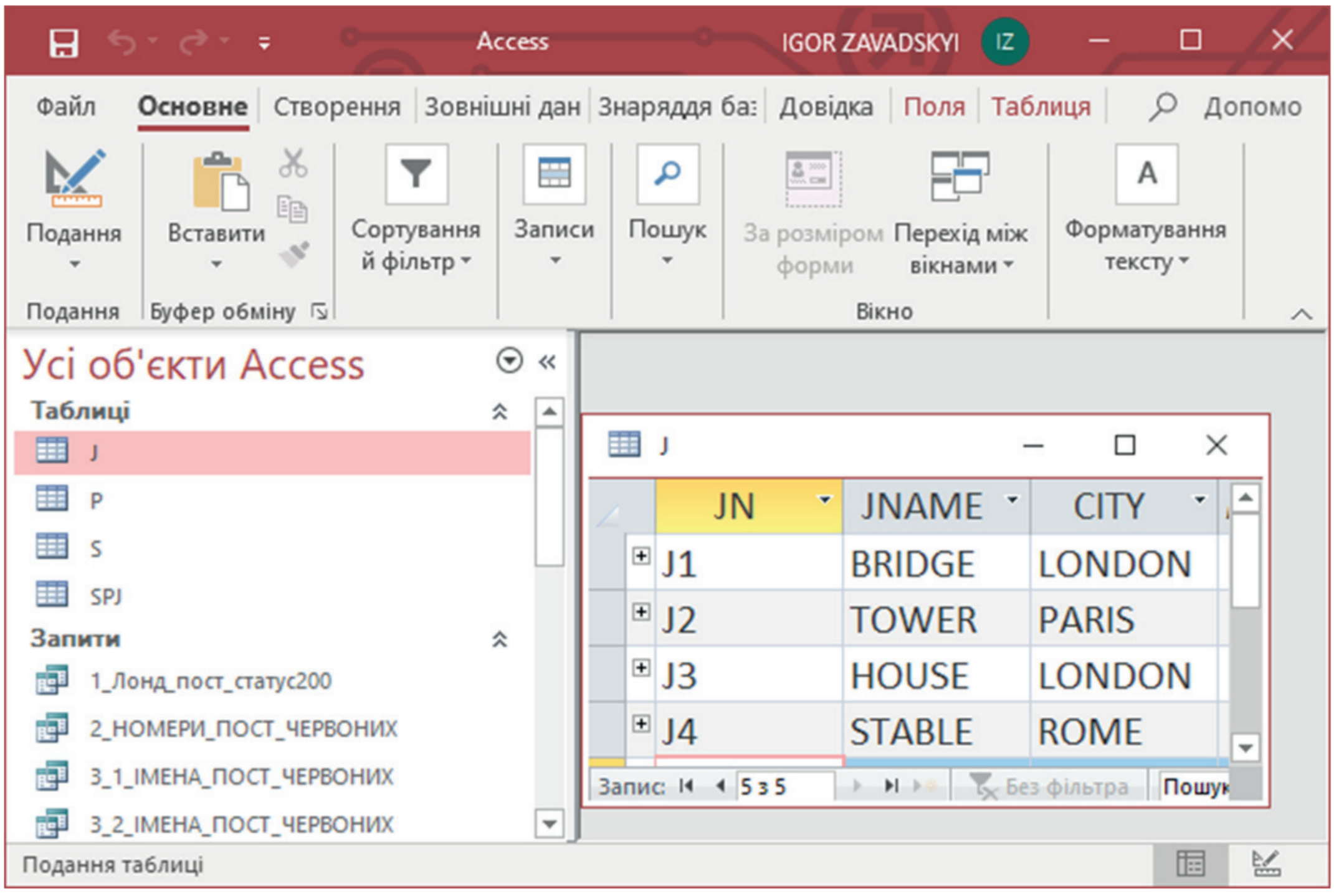


Щоб створити нову базу відразу після запуску MS Access, потрібно на головній сторінці вибрати пункт «Пуста база даних». А щоб створити нову базу даних під час роботи з іншою базою, пункт «Пуста база даних» треба вибрати в меню «Файл».

Щоб зберегти базу даних, використовують команду «Файл\ Зберегти», однак у разі закриття будь-якого об'єкта бази, наприклад таблиці, збереження виконується автоматично.

Якою є структура вікна Microsoft Access?

У вікні MS Access є дві області. Ліва область називається «Усі об'єкти Access». У ній відображається список усіх об'єктів, які є у відкритій базі даних, а в правій — вміст або структура вибраного в лівій області об'єкта. Наприклад, на рисунку в лівій області вибрано таблицю J, а в правій — відображено дані, що містяться в цій таблиці.



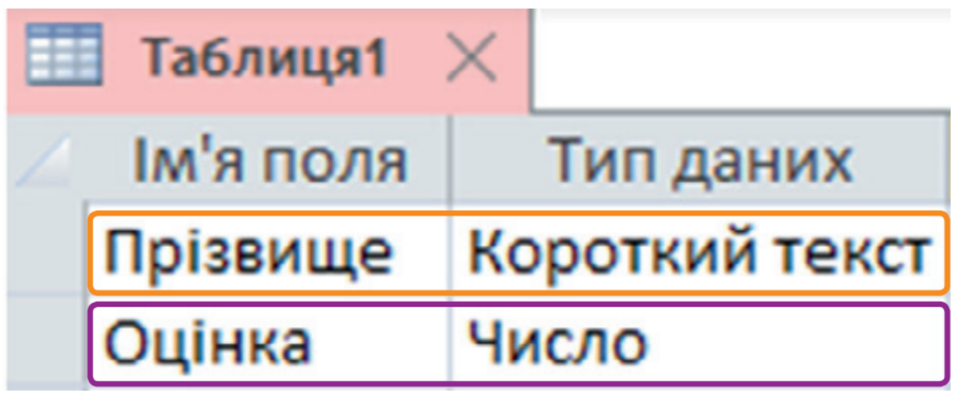
Що таке поля та записи?

Стовпці таблиць у реляційних СКБД називаються *полями*, а рядки — *записами*.

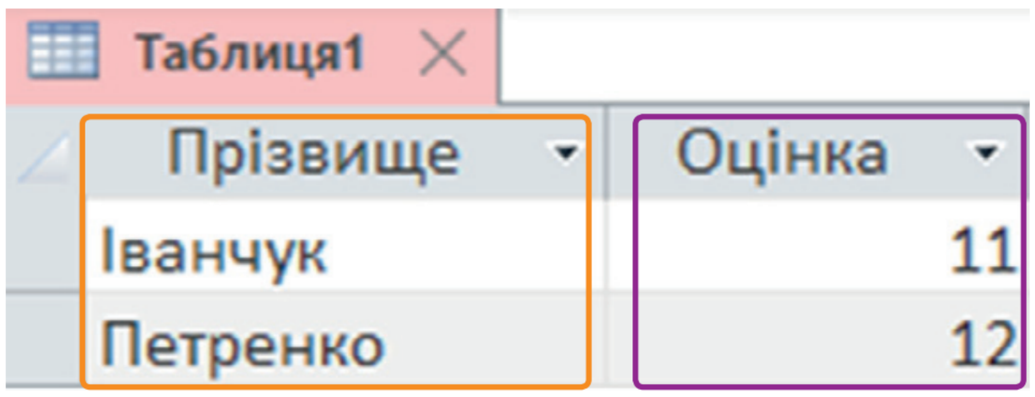
Як створити таблицю в Microsoft Access?

Потрібно в меню «Створення» вибрати пункт «Конструктор таблиць». Тоді в правій області відобразиться вікно «Таблиця1», у якому можна визначити структуру новоствореної таблиці. Його також називають вікном *конструктора*.

Не плутай вікно конструктора з вікном даних таблиці! У кожному рядку вікна конструктора відображаються не самі дані, а лише інформація про певний стовпець таблиці. У кожному рядку вікна даних міститься інформація про певний об'єкт, що зберігається в базі.



Вікно конструктора



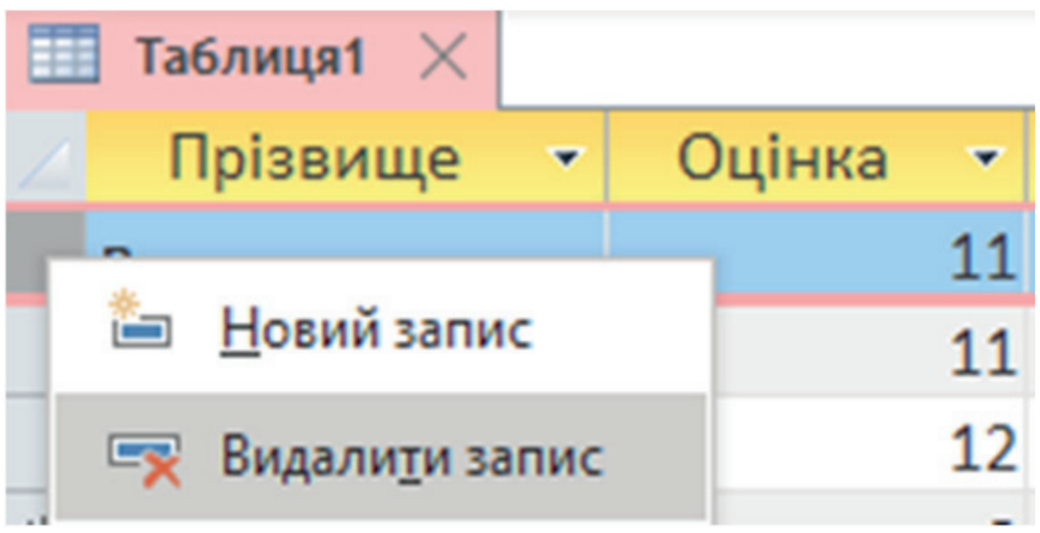
Вікно даних

Як ввести чи змінити дані в таблиці Access?

Для цього потрібно в області «Усі об'єкти Access» двічі клацнути значок таблиці. Тоді в правій області ця таблиця відкриється в режимі редагування даних. Дані можна змінювати в будь-якій клітинці. Для переходу в наступну клітинку натисни клавішу Enter. Також у сусідні клітинки можна переходити за допомогою клавіш керування курсором.

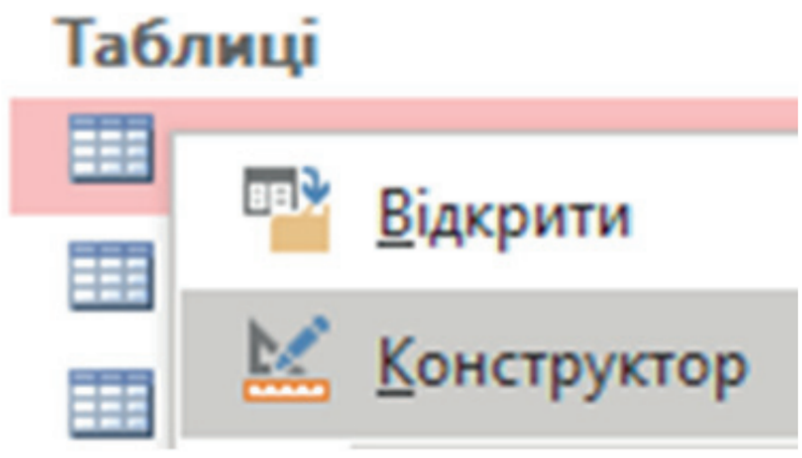
Як видалити дані в таблиці Access?

Щоб видалити значення в клітинці або частину цього значення, потрібно у вікні даних клацнути цю клітинку й виконати необхідну операцію за допомогою звичайних дій редагування, як у табличному процесорі. Однак видалити цілий запис (рядок) так не вийде. Для цього потрібно клацнути правою кнопкою миші заголовок рядка і з контекстного меню вибрати пункт «Видалити запис».



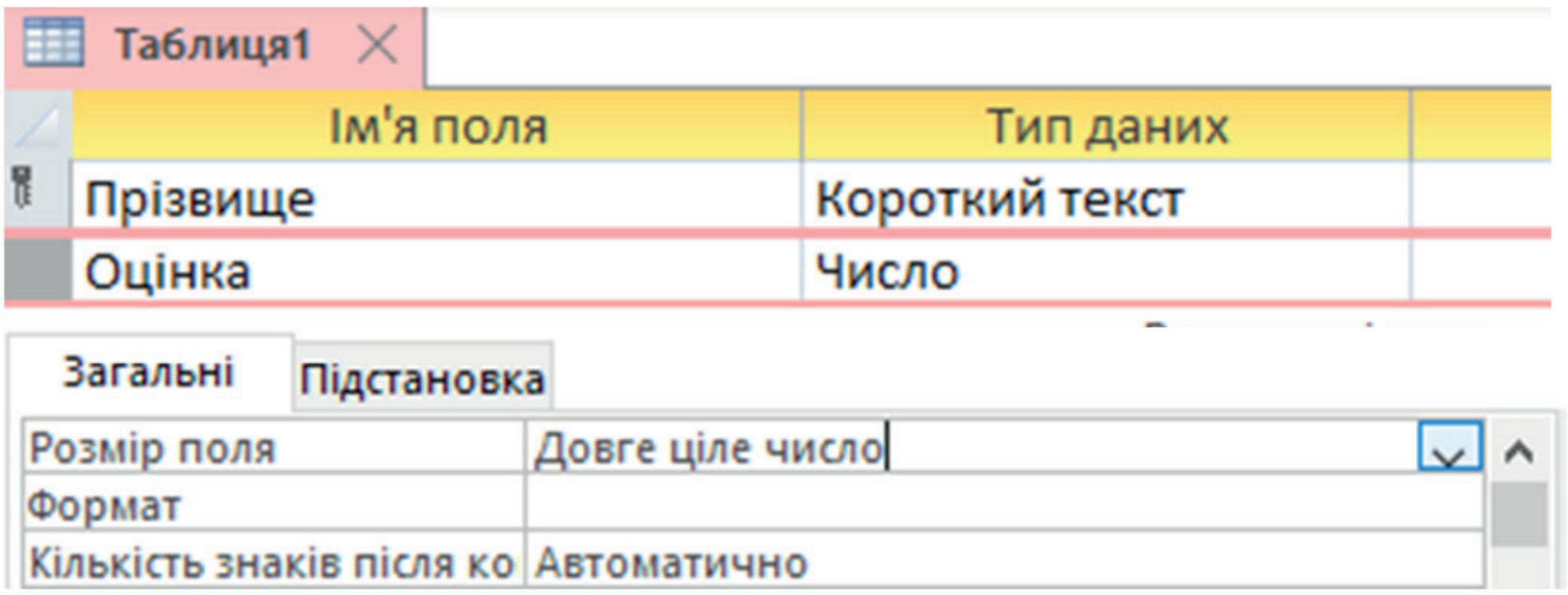
Як змінити структуру наявної таблиці в базі даних Access?

Потрібно в лівій області клацнути значок таблиці правою кнопкою миші й вибрати пункт контекстного меню «Конструктор». Тоді на правій панелі відкриється вікно конструктора таблиці, де можна додати до неї поля, видалити їх або змінити їхні властивості.



Як змінити значення властивостей поля таблиці?

Потрібно виділити це поле у вікні конструктора таблиці і внизу цього вікна задати значення тих чи інших властивостей.



Наприклад, якщо поле числове, то для властивості «Розмір поля» можна вибрати такі значення:

- Байт — ціле значення від 0 до 255.
- Ціле число — ціле значення від –32768 до 32767.
- Довге ціле число — ціле значення від –2147483648 до 2147483647.
- Десятковий дріб в діапазоні від $-3{,}4 \cdot 10^{38}$ до $3{,}4 \cdot 10^{38}$.

А для поля типу «Короткий текст» властивість «Розмір поля» визначає максимальну довжину рядка в діапазоні від 1 до 255 символів.

Якщо спробувати ввести в поле значення, яке не відповідає заданим властивостям, цю операцію буде заблоковано.

Як зробити поле ключовим?

Потрібно виділити це поле у вікні конструктора таблиці й натиснути кнопку «Ключове поле».

Зауваж, що значення в ключовому полі не можуть повторюватися, а також не можуть бути порожніми. Якщо дані в полі, яке ти намагаєшся зробити ключовим, порушують ці вимоги, то буде відображено повідомлення про помилку.

ПЕРЕВІР СЕБЕ

ЗНАННЯ – РОЗУМІННЯ – АНАЛІЗ – СТВОРЕННЯ – ВЛАСНІ ДУМКИ

1. Опиши алгоритми виконання в СКБД таких дій:
   - створити нову таблицю;
   - додати до таблиці запис;
   - додати до таблиці поле;
   - видалити з таблиці запис;
   - видалити з таблиці поле;
   - змінити ключ таблиці.

2. Переглянь відповідь на запитання «Як змінити значення властивостей поля таблиці?» в рубриці «Запитання-відповіді» і спробуй пояснити, чому обмеження на значення типу «байт», «ціле число» та «довге ціле число» є саме такими. За якими формулами визначаються ці числа? Скільки байтів пам'яті може займати кожен із вказаних різновидів значень?

3. У вікні конструктора таблиця має такий самий вигляд, як на рисунку нижче.

	Ім'я поля	Тип даних
	x	Короткий текст
	y	Число

Якими даними може бути заповнена ця таблиця?

А)

x	y
Іванчук	11
Петренко	12

Б)

x	y
Іванчук	11
Петренко	11

В)

x	y
Іванчук	11
Іванчук	12

Г)

x	y
11	11
12	11

4. Створи базу даних, що міститиме відомості про учнів та класи твоєї школи. У ній мають зберігатися відомості про назви і, можливо, профіль класів, а також про прізвища, імена, дати народження і стать учнів. Введи інформацію про 2–3 класи та щонайменше про шість учнів різних класів. Необхідно вказати, у якому класі вчиться кожен з учнів.

5. Чому клітинки в ключовому полі не можна залишати порожніми?

§ 8. Обмеження цілісності

До яких наслідків може призвести порушення обмежень цілісності в базі даних?

На минулому уроці ми реалізували проєкт бази даних автобусів і маршрутів у СКБД Microsoft Access. Однак створену базу даних не можна назвати готовою. Адже в ній можна вводити очевидно хибні дані, наприклад 2100 рік випуску автобуса. Щоб гарантувати відповідність бази реальним даним, у ній створюють спеціальні обмеження цілісності, які ми вивчатимемо сьогодні.

Завдання № 1

Які твердження можуть бути обмеженнями цілісності в базі даних автобусів і маршрутів, що містить дві таблиці зображеної структури?

Автобуси

Ім'я поля	Тип даних
Номерний знак	Короткий текст
Модель	Короткий текст
Рік випуску	Число
Маршрут	Число

Маршрути

Ім'я поля	Тип даних
Номер	Число
Довжина	Число
К-ть зупинок	Число

- Рік випуску автобуса належить діапазону від 1900 до поточного року.
- Кількість зупинок повинна бути додатною.
- Кожне значення з поля «Маршрут» таблиці «Автобуси» має також бути в полі «Номер» таблиці «Маршрути».
- Кожне значення поля «Номер» з таблиці «Маршрути» має також бути в полі «Маршрут» таблиці «Автобуси».

Завдання № 2

Укажи в таблицях зображеної бази даних чотири клітинки, дані в яких порушують обмеження цілісності, визначені в попередньому завданні.

Автобуси

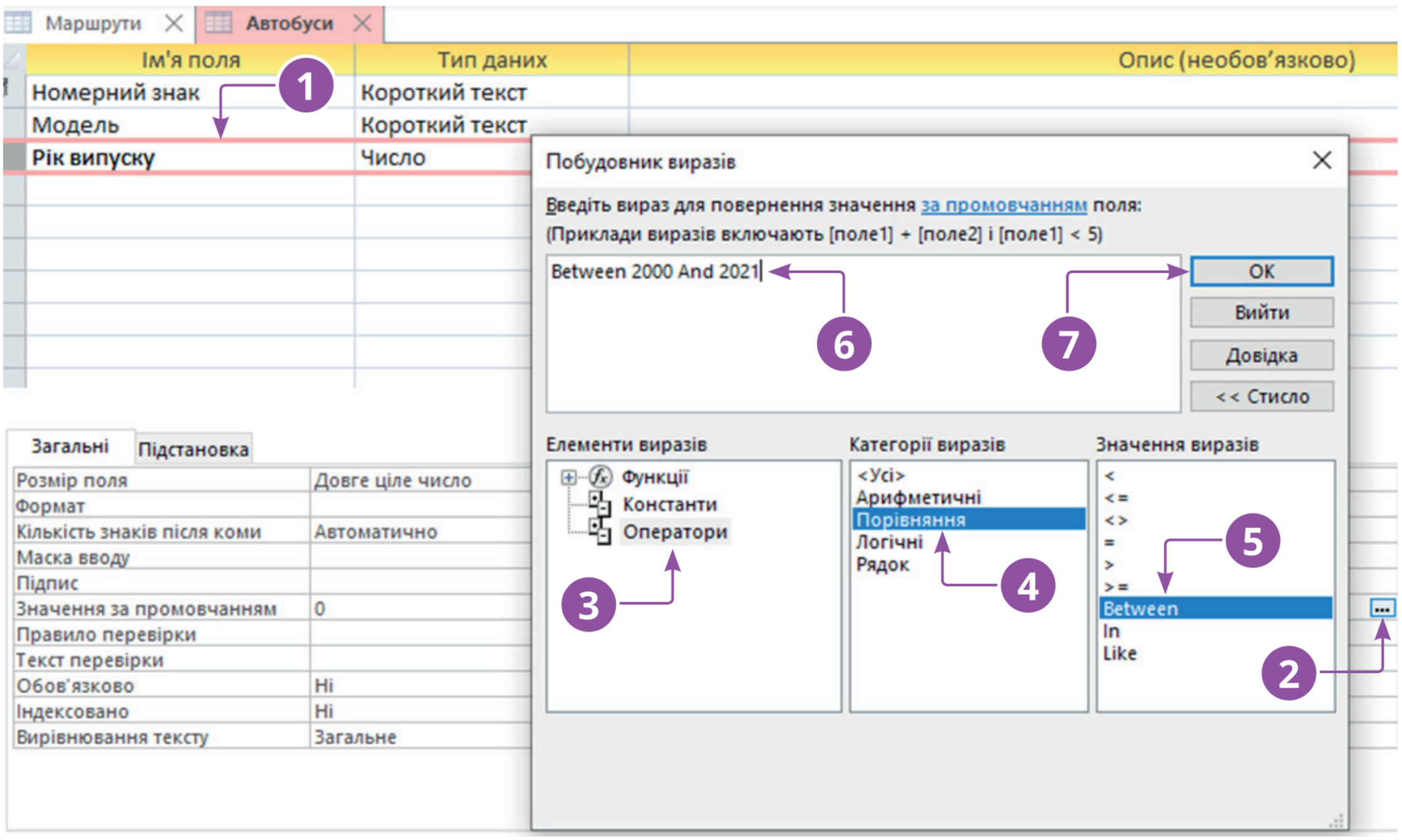
Номерний знак	Модель	Рік випуску	Маршрут
AA1684PT	Богдан A144	2017	9
AA3129KH	CityLAZ-12	2011	12
AA3312TB	МАЗ 103	2041	9
AA4409KH	CityLAZ-12	2021	7
AI2917KT	Богдан A144	1815	12
AI4124KK	CityLAZ-12	2012	12

Маршрути

Номер	Довжина	К-ть зупинок
8	9,1	12
9	7,3	0
12	5,4	18

Завдання № 3

Запусти СКБД Microsoft Access. Користуючись зображеною послідовністю дій та рубрикою «Запитання-відповіді», задай обмеження для значень поля «Рік випуску» в таблиці «Автобуси»: *рік випуску автобуса має перебувати в межах від 2000 року до поточного року включно.*



Спробуй ввести рік випуску якогось із автобусів, що не задовольняє створеному обмеженню. СКБД має відобразити повідомлення про помилку, проаналізуй його. Щоб скасувати введення неправильного значення, натисни клавішу Esc.

Завдання № 4

Задай такі обмеження в таблиці маршрутів:

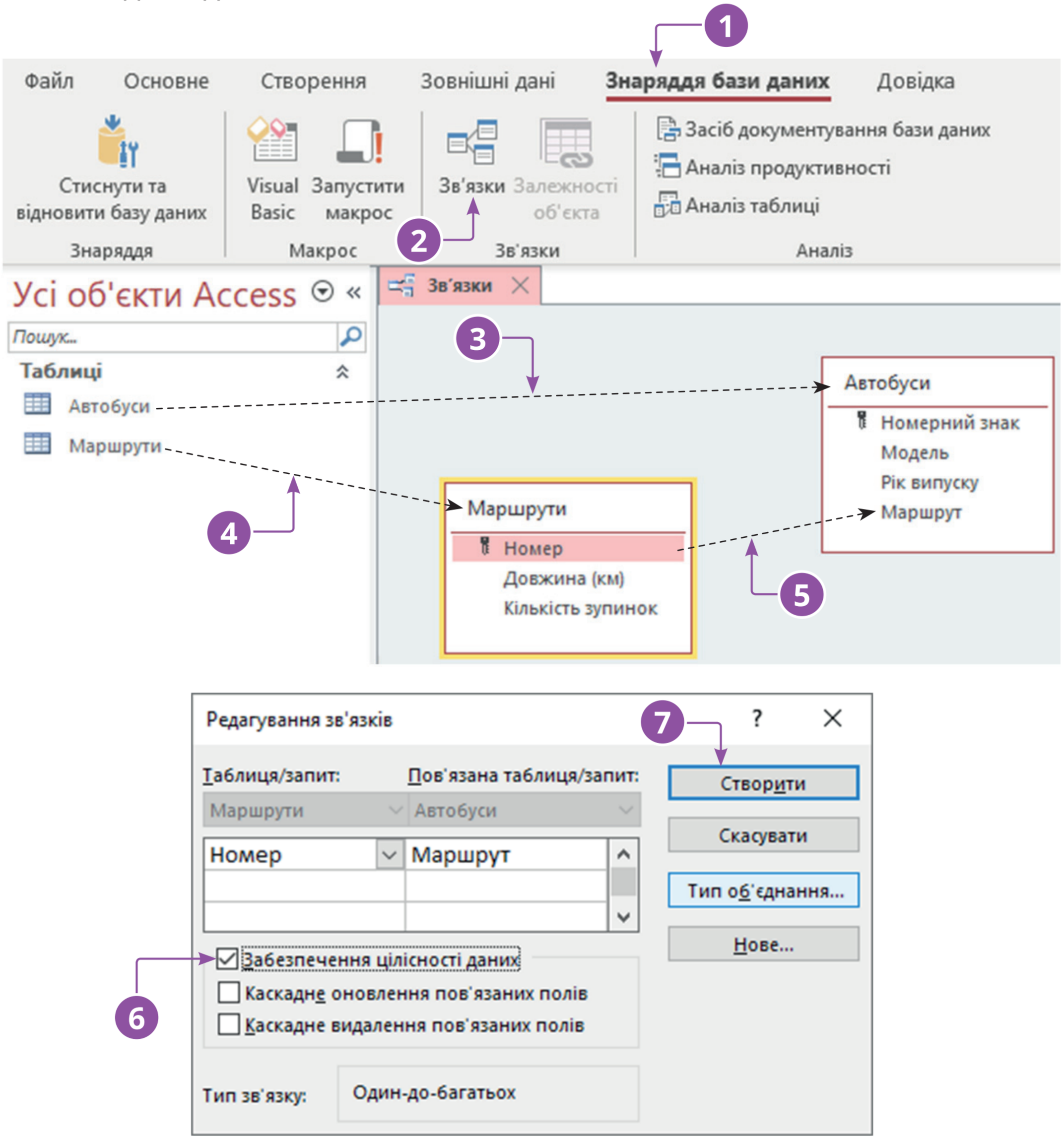
- номер маршруту — від 1 до 500;
- довжина маршруту — більше 0;
- кількість зупинок на маршруті — більше 0.

Спробуй ввести дані, що не задовольняють цим обмеженням, і переконайся, що СКБД відображає повідомлення про помилку.

Завдання № 5

Досі ми створювали обмеження цілісності, що стосувалися однієї таблиці. Тепер створи обмеження, що пов'язує дані двох таблиць: *кожне значення з поля «Маршрут» таблиці «Автобуси» має також бути в полі «Номер» таблиці «Маршрути».*

Скористайся зображенням послідовності дій, а також рубрикою «Запитання-відповіді».



Спробуйте ввести в поле «Маршрут» таблиці «Автобуси» значення, якого немає в полі «Номер» таблиці «Маршрути», переконайтеся, що СКБД видасть повідомлення про помилку та проаналізуйте його.

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке обмеження цілісності?

Обмеження цілісності — це правило, якому мають задовольняти дані в базі, щоб не суперечити одне одному та предметній області, тобто бути цілісними.

Як забезпечують підтримання обмежень цілісності?

Обмеження цілісності створюють через зазначення певних властивостей полів таблиць, а їх підтримання забезпечує СКДБ. Якщо вводити або змінювати дані так, що обмеження цілісності порушуються, цю операцію СКДБ заблокує.

Які є різновиди обмежень цілісності?

Варто розрізняти обмеження двох типів, адже засоби їх створення різні:

- обмеження, що стосуються полів однієї таблиці;
- обмеження, що пов’язують певною умовою поля кількох таблиць.

Серед обмежень першого типу ми вивчатимемо ті, що стосуються значень якогось одного поля, а серед обмежень другого типу — ті, що накладаються зв’язком «один-до-багатьох».

Як створити обмеження на значення в певному полі?

Потрібно виділити це поле у вікні конструктора й внизу в області «Властивості поля» у рядку «Правило перевірки» записати правило, якому мають задовольняти дані.
Наприклад, на рисунку зображено, як задати обмеження «Кількість зупинок має бути додатною» на значення поля «Кількість зупинок» у таблиці «Маршрути».

Маршрути

Ім'я поля	Тип даних
Номер	Короткий текст
Довжина (км)	Число
Кількість зупинок	Число

Властивості поля

Загальні | Підстановка

Розмір поля	Довге ціле число
Формат	
Кількість знаків після коми	Автоматично
Маска вводу	
Підпис	
Значення за промовчанням	0
Правило перевірки	> 0
Текст перевірки	

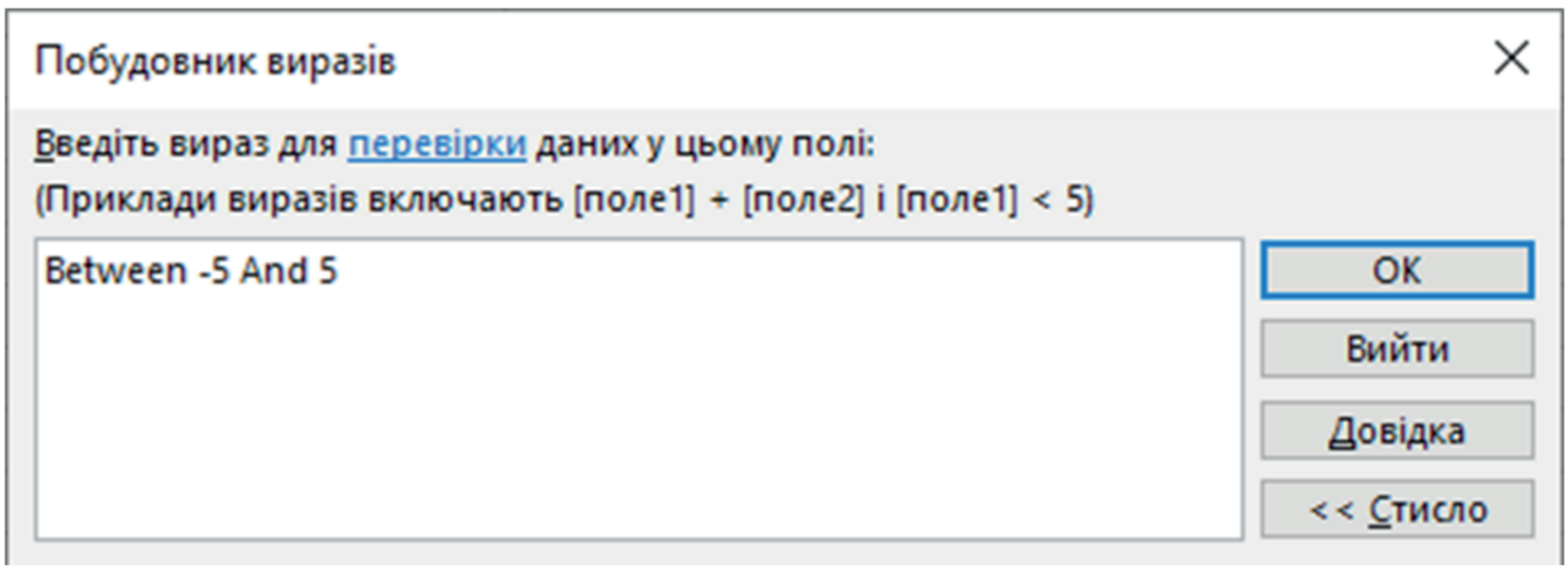
Якщо потрібно ввести більш складне правило, наприклад вказати, що значення поля має перебувати в діапазоні від одного числа до іншого, можна скористатися Побудовником виразів — спеціальним вікном, що відкривається кнопкою ..., розташованою в рядку «Правило перевірки» справа.

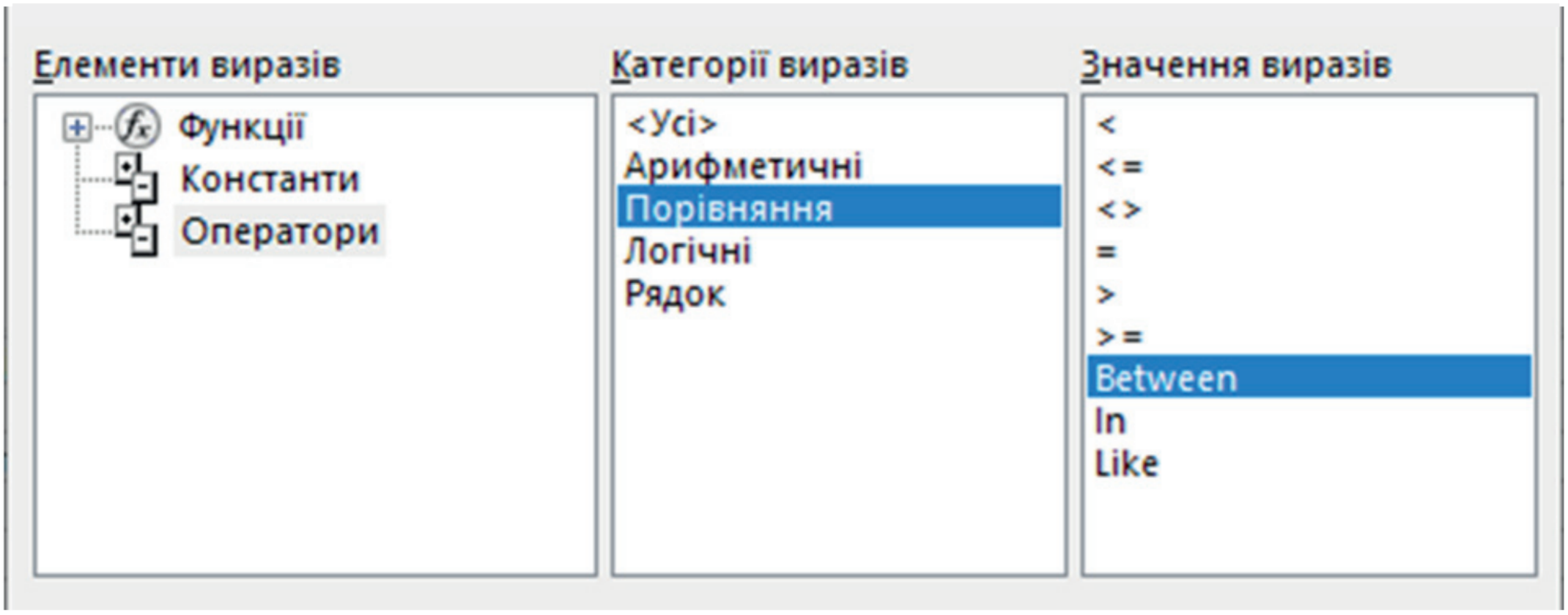
Так, у зображеному нижче вікні сконструйовано обмеження, яке означає, що значення певного поля має перебувати в діапазоні від –5 до 5 включно. Для його побудови виконано такі дії:

- в області «Елементи виразів» вибрати пункт «Оператори»;
- в області «Категорії виразів» вибрати пункт «Порівняння»;
- в області «Значення виразів» вибрати оператор «Between»;
- доповнити вираз так, як показано у верхньому полі.

Як створити обмеження на значення в певному полі?

Потрібно виділити це поле у вікні конструктора й внизу в області «Властивості поля» у рядку «Правило перевірки» записати правило, якому мають задовольняти дані.
Наприклад, на рисунку зображено, як задати обмеження «Кількість зупинок має бути додатною» на значення поля «Кількість зупинок» у таблиці «Маршрути».





Яке обмеження накладається зв’язком «один-до-багатьох» і як його створити?

Нагадаємо, що для реалізації зв’язку «один-до-багатьох» в таблиці, яка розташована з боку «багато», створюють стовпець, значення в якому повторюють деякі зі значень ключового стовпця таблиці, яка розташована з боку «один». Наприклад, у таблиці «Люди» створюють стовпець «Квартира», де записують номери квартир (номер — це ключ таблиці «Квартири»). Обмеження полягає в тому, що не можна у стовпці «Квартира» таблиці «Люди» записувати номери неіснуючих квартир, тобто тих, яких немає в таблиці «Квартири». Інакше кажучи, у додатковий стовпець таблиці, що розташована з боку «багато», не можна вводити значень, яких немає в ключовому стовпці таблиці, що розташована з боку «один».

Прізвище	Ім’я	Стать	Номер квартири
Огороднік	Юлія	ж	41
Огороднік	Євген	ч	41
Огороднік	Микола	ч	41
Чубенко	Іван	ч	87
Чубенко	Інна	ж	87

Номер квартири	Площа квартири (кв. м)	Поверх
41	39	1
87	77	9

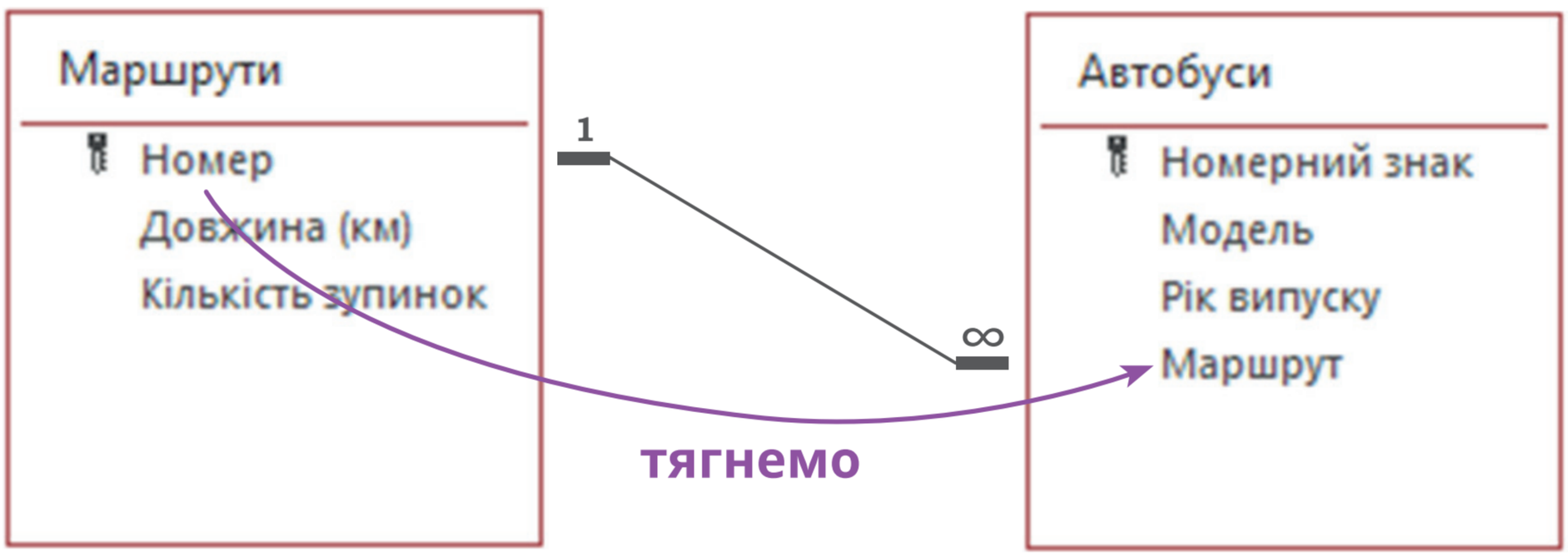
Щоб створити таке обмеження цілісності, потрібно:

- вибрати стрічку «Знаряддя бази даних» і натиснути кнопку «Зв’язки» — з’явиться вікно «Зв’язки»;
- з області «Усі об’єкти Access» у вікно «Зв’язки» перетягнути таблиці, що з’єднані зв’язком «один-до-багатьох»;

- у вікні «Зв’язки» ключ таблиці, що розташована з боку «один», потрібно перетягнути на додатковий стовпець таблиці, розташованої з боку «багато», — буде відображено вікно «Редагування зв’язків»;
- у вікні «Редагування зв’язків» потрібно встановити прапорець «Забезпечення цілісності даних» і натиснути кнопку «ОК».

У результаті зв’язок «один-до-багатьох» відобразиться графічно.

Зауважте, якщо не встановлювати прапорця «Забезпечення цілісності даних», то символи «1» і «∞» біля лінії зв’язку не відображатимуться й обмеження цілісності не підтримуватиметься.



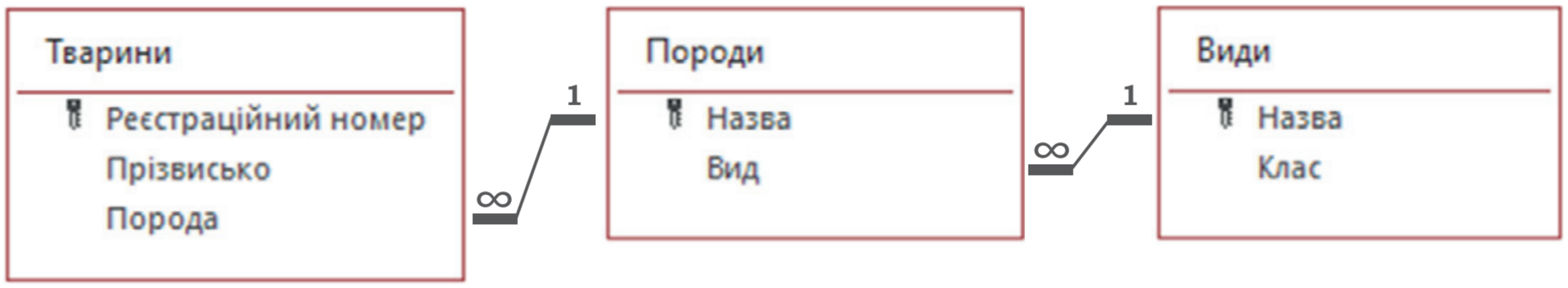
ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Опиши алгоритми створення в СКБД таких обмежень цілісності:
   - значення числового поля має бути більшим за певне число;
   - значення числового поля має належати певному діапазону;
   - значення у стовпці x таблиці A мають також належати ключовому стовпцю в таблиці B.

2. Схема бази даних у вікні «Зв'язки» має вигляд, як на рисунку нижче.

Укажи клітинки, значення в яких порушують обмеження цілісності.



Тварини

Реєстрацій	Прізвиськс	Порода
KH2903	Грей	Лабрадор
TP4111	Кетті	Сіам
X3517	Альфа	Шнауцер

Породи

Назва	Вид
Ангорський	Кролик
Сіам	Кіт
Шнауцер	Собака

Види

Назва	Клас
Кіт	Ссавці
Собака	Ссавці

3. До створеної на минулому уроці бази даних учнів та класів додай такі обмеження цілісності:
   - дати народження учнів мають бути такими, щоб їхній вік становив від 6 до 11 років;
   - не можна вказувати, що учень вчиться в неіснуючому класі, тобто такому, якого немає в таблиці «Класи».

4. Коли ми створювали обмеження цілісності, що накладається зв'язком «один-до-багатьох», то СКБД Microsoft Access на схемі зв'язків поставила символ «один» біля таблиці «Маршрути», а символ «∞» (багато) — біля таблиці «Автобуси». Як ти вважаєш, звідки СКБД «дізналася», що ці символи ставити треба саме в такому порядку, а не навпаки?

   Спробуй перетягнути поля у зворотному напрямі, тобто поле «Маршрут» з таблиці «Автобуси» на поле «Номер» з таблиці «Маршрути». Проаналізуй результат. Чи підтверджує він твою здогадку? Якщо ні, то спробуй пояснити принцип, за яким СКБД визначає множинність зв'язку, ще раз.

§ 9. Вибір даних

Чи можна запит, що стосується багатьох таблиць, замінити кількома розширеними фільтрами, застосованими до кожної із цих таблиць?

Ми вже вміємо створювати бази даних, вводити в них дані й певною мірою забезпечувати їх несуперечливість. Однак від баз даних не буде ніякої користі, якщо дані, які в них зберігаються, ніхто не отримуватиме. Щоб отримати дані з маленьких баз, таких як ми створювали, достатньо просто переглянути таблиці. Але бази, що використовуються в реальному житті, часто містять десятки таблиць і тисячі записів у кожній. Щоб знайти в такій базі потрібні дані, їх потрібно відібрати за певними критеріями. Є кілька засобів такого відбору: сортування та фільтрація подібні до однойменних засобів табличного процесора і застосовуються до однієї таблиці, а запити дають змогу відібрати дані за умовами, що стосуються кількох таблиць.

Завдання № 1

Укажи, за якими стовпцями і в якому порядку (за зростанням чи спаданням) могло бути відсортовано зображену таблицю.

Автобуси

Номерний знак	Модель	Рік випуску	Маршрут
AA1684PT	Богдан A144	2017	9
AA3129KH	CityLAZ-12	2010	9
AA3312TB	МАЗ 103	2001	11
AA4409KH	CityLAZ-12	2012	11
AI2917KT	Богдан A144	2017	12
AI4124KK	CityLAZ-12	2012	12

Завдання № 2

Дізнайся з рубрики «Запитання-відповіді» про фільтри в Microsoft Access (вони дуже подібні до фільтрів у табличному процесорі), а також про оператора Like (Схоже). Укажи, яким фільтрам відповідає наповнення зображеної таблиці.

А)

Поле:	Номерний знак	Модель
Сортування:		
Критерії:	Схоже "*АА"	
Або:		Схоже "City*"

Б)

Поле:	Номерний знак	Модель
Сортування:		
Критерії:	Схоже "АА*"	
Або:		Схоже "City*"

В)

Поле:	Номерний знак	Модель
Сортування:		
Критерії:	Схоже "*АА"	Схоже "City*"
Або:		

Г)

Поле:	Маршрут	Маршрут
Сортування:		
Критерії:	9	
Або:		>10

Завдання № 3

1. Застосувавши розширений фільтр, вибери всі автобуси, які або випущені після 2015 року, або курсують маршрутом із номером >10.
2. За допомогою розширеного фільтра вибери всі автобуси 9-го маршруту, номерний знак яких містить англійську літеру «К».

Завдання № 4

Опиши словами, що знаходять зображені запити.

А)

Поле:	Рік випуску	Номерний знак	Модель	Маршрути.*
Таблиця:	Автобуси	Автобуси	Автобуси	Маршрути
Сортування:				
Відображення:	☐	☐	☐	☑
Критерії:	>2015			
Або:		"АА3129КН"		

Б)

Поле:	Номерний знак	Модель	Номер	Довжина (км)
Таблиця:	Автобуси	Автобуси	Маршрути	Маршрути
Сортування:				
Відображення:	☑	☑	☐	☐
Критерії:			11	
Або:				>7

В)

Поле:	Рік випуску	Модель	Номер
Таблиця:	Автобуси	Автобуси	Маршрути
Сортування:			
Відображення:	☑	☐	☑
Критерії:		Схоже "*Богдан"	10
Або:			

Г)

Поле:	Рік випуску	Модель	Номер
Таблиця:	Автобуси	Автобуси	Маршрути
Сортування:			
Відображення:	☑	☐	☑
Критерії:		Схоже "*Богдан*"	
Або:			10

Завдання № 5

Уважно ознайомся з матеріалом рубрики «Запитання-відповіді», що стосується конструювання запитів. Проаналізуй наповнення таблиць A і B:

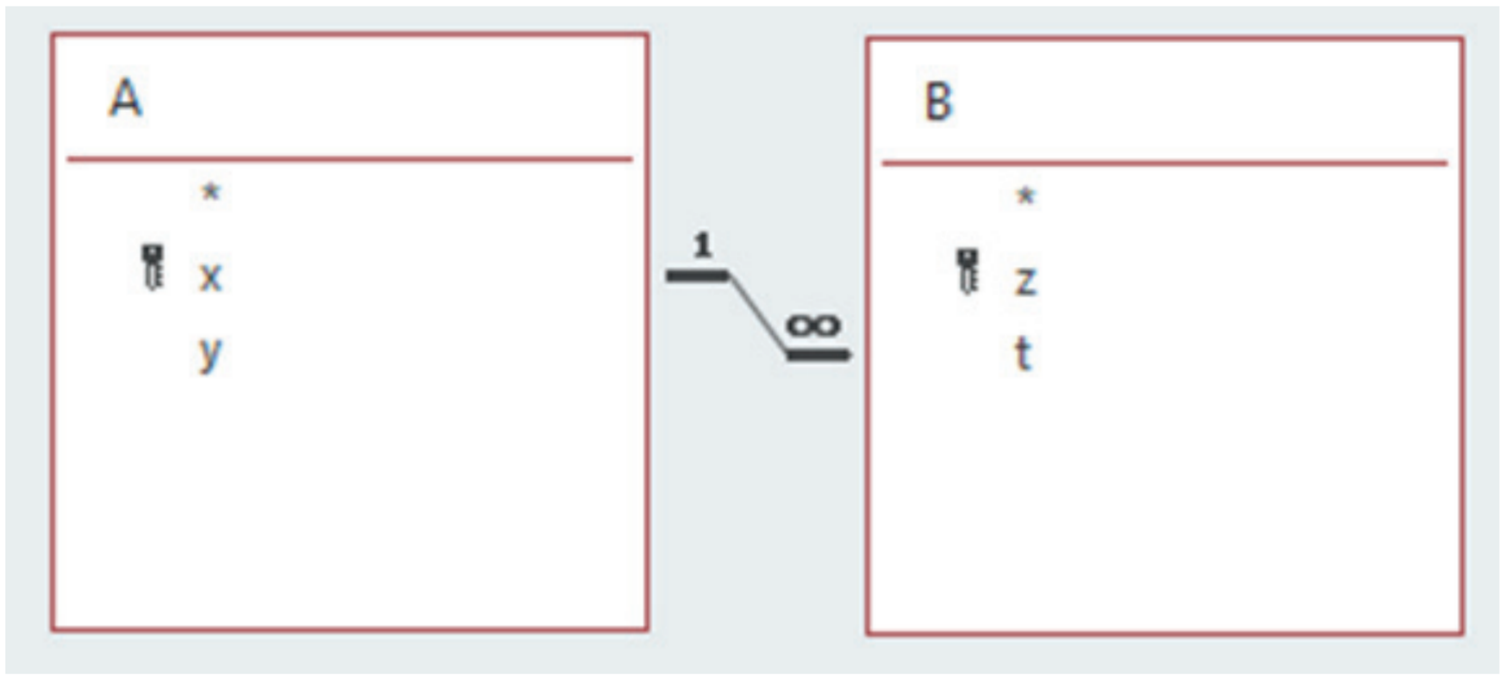
A

x	y
1	2
3	4

B

z	t
5	3
6	1

Запит:



Поле:	A.*	B.*	
Таблиця:	A	B	
Сортування:			
Відображення:	☑	☑	☐
Критерії:			
Або:			

І вкажи, які з наведених записів відображатимуться в результаті виконання цього запиту.

	x	y	z	t
	1	2	5	3
	3	4	5	3
	1	2	6	1
	3	4	6	1

Завдання № 6

Якими даними могли бути заповнені таблиці автобусів та маршрутів, щоб виконання зображеного запиту дало зображений результат?

Поле:	Рік випуску	Модель	Номер
Таблиця:	Автобуси	Автобуси	Маршрути
Сортування:			
Відображення:	☑	☑	☑
Критерії:		Схоже "*Богдан*"	
Або:			10

Рік випуску	Номер
2017	9
2017	12

А)

Маршрути

Номер	Довжина (км)	Кількість зупинок
9	7,3	18
11	8	18
12	5,4	14

Автобуси

Номерний знак	Модель	Рік випуску	Маршрут
AA1684PT	Богдан A144	2017	9
AA3129KH	CityLAZ-12	2010	9
AA3312TB	МАЗ 103	2001	11
AA4409KH	CityLAZ-12	2012	11
AI2917KT	Богдан A144	2017	12
AI4124KK	CityLAZ-12	2012	12

Б)

Маршрути

Номер	Довжина (км)	Кількість зупинок
9	7,3	18
10	8	18
12	5,4	14

Автобуси

Номерний знак	Модель	Рік випуску	Маршрут
AA1684PT	Богдан A144	2017	9
AA3129KH	CityLAZ-12	2010	9
AA3312TB	МАЗ 103	2001	10
AA4409KH	CityLAZ-12	2012	10
AI2917KT	Богдан A144	2017	12
AI4124KK	CityLAZ-12	2012	12

В)

Маршрути

Номер	Довжина (км)	Кількість зупинок
9	8	18
10	7,3	18
12	5,4	14

Автобуси

Номерний знак	Модель	Рік випуску	Маршрут
AA1684PT	Богдан A144	2017	10
AA3312TB	МАЗ 103	2001	9
AA4409KH	CityLAZ-12	2012	9
AI2917KT	Богдан A144	2017	12
AI4124KK	CityLAZ-12	2012	12

Завдання № 7

Дізнайся з рубрики «Запитання-відповіді» про конструювання запитів, зокрема параметризованих. Сконструюй та виконай такі запити.

1. Визначити номерні знаки та моделі автобусів, на маршрутах яких більше 15 зупинок.
2. Визначити всю інформацію про автобуси, що курсують маршрутами, кількість зупинок на яких перевищує введену користувачем.
3. Визначити всю інформацію про маршрути довжиною більше 10 км, на яких курсують автобуси введеної користувачем моделі.

ЗАПИТАННЯ-ВІДПОВІДІ

Як виконують і в яких задачах використовують сортування?

Сортування, як ти вже знаєш, — це впорядкування об'єктів у порядку зростання або спадання значень певної властивості. У СКБД Microsoft Access сортування виконують майже так само, як і в табличному процесорі:

- відкрий таблицю в режимі редагування даних;
- виділи стовпець або стовпці, за значеннями яких потрібно відсортувати таблицю;
- натисни кнопку За зростанням або За спаданням.

Номерний знак	Модель	Рік випуску	Маршрут
AA3129KH	CityLAZ-12	2010	9
AA4409KH	CityLAZ-12	2012	11
AI4124KK	CityLAZ-12	2012	12
AI2917KT	Богдан А144	2017	12
AA1684PT	Богдан А144	2017	9
AA3312TB	МАЗ 103	2001	11
		0	

! *Увага!*

На відміну від табличного процесора, у таблиці бази даних під час сортування переставляються цілі рядки, навіть якщо виділено не всю таблицю, а лише деякі стовпці.

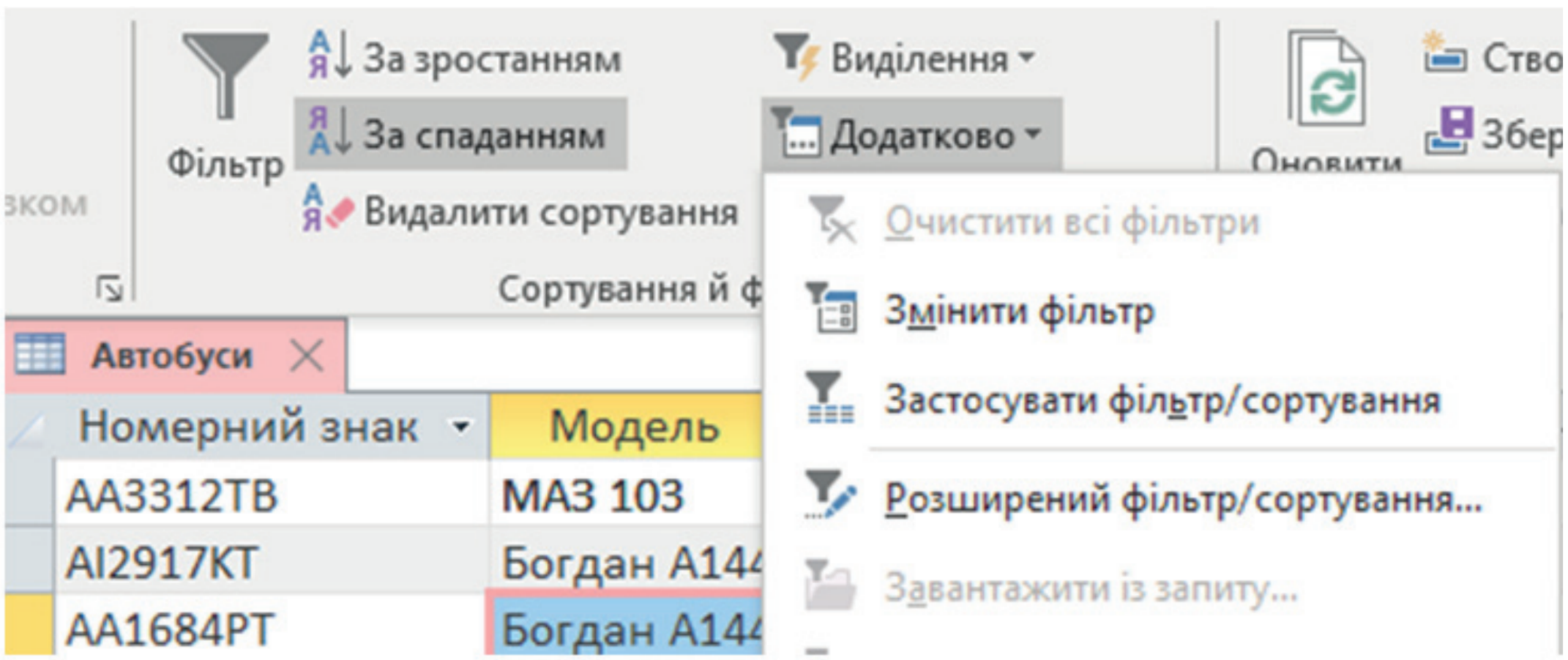
Про те, що таблицю відсортовано за певним стовпцем, свідчить позначка ▾↑ *(за зростанням) чи* ▾↓ *(за спаданням) у заголовку стовпця. Щоб скасувати сортування, натисни кнопку* Видалити сортування .

Якщо під час сортування виділено кілька стовпців, то таблиця спочатку сортується за найлівішим із виділених стовпців, а потім кожна група записів з однаковим значенням у найлівішому виділеному стовпці сортується за значеннями наступного стовпця тощо.

Найпоширеніша задача, для розв'язання якої стане в пригоді сортування, — це пошук записів, що містять у певному полі найменше або найбільше значення серед усіх записів таблиці. Після сортування всі такі записи відображаються вгорі таблиці.

Для чого призначені та яку структуру мають фільтри в Microsoft Access?

Призначення фільтрів — відбір записів однієї таблиці за певною умовою.

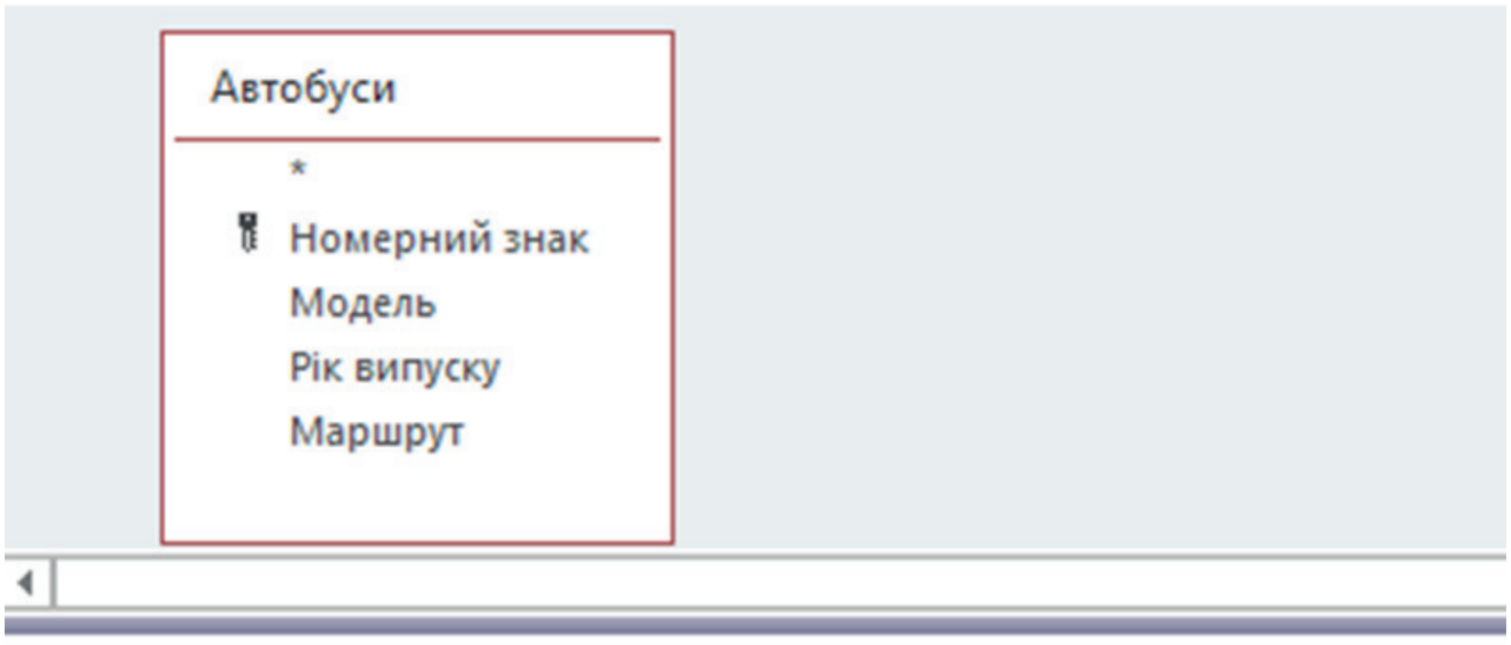


Як і в табличному процесорі, у MS Access є звичайні фільтри, що накладають умови на значення одного стовпця, і розширені фільтри, у яких умови відбору можуть формуватися за значеннями кількох стовпців.

- Звичайні фільтри створюють кнопками Фільтр і Виділення .
- Розширений фільтр створюють за допомогою команди «Додатково \ Розширений фільтр/сортування».

Як і в табличному процесорі, у розширеному фільтрі умови, що розташовані в одному рядку, з'єднуються сполучником «І», а в різних

рядках — сполучником «АБО». Крім того, у цьому фільтрі можна задавати сортування за певними стовпцями. Наприклад, на малюнку показано фільтр для відбору автобусів, які або курсують маршрутом із номером більше 10, або випущені після 2015 року. Також результати фільтрації сортуються за зростанням року випуску.



Поле:	Маршрут	Рік випуску
Сортування:		За зростанням
Критерії:	>10	
Або:		>2015

До фільтрації:

Номерний знак	Модель	Рік випуску	Маршрут
AA1684PT	Богдан A144	2017	9
AA3129KH	CityLAZ-12	2010	9
AA3312TB	МАЗ 103	2001	11
AA4409KH	CityLAZ-12	2012	11
AI2917KT	Богдан A144	2017	12
AI4124KK	CityLAZ-12	2012	12

Після фільтрації:

Номерний знак	Модель	Рік випуску	Маршрут
AA3312TB	МАЗ 103	2001	11
AA4409KH	CityLAZ-12	2012	11
AI4124KK	CityLAZ-12	2012	12
AI2917KT	Богдан A144	2017	12
AA1684PT	Богдан A144	2017	9

Для чого призначений оператор Like (Схоже)?

Цей оператор використовують, якщо умовою пошуку або фільтрації даних є входження певного рядка в більший рядок тексту. Після слова Like записують вираз із символами підстановки, у якому «*» означає довільну послідовність символів, а «?» — один довільний символ. Наприклад, якщо потрібно відібрати автобуси, у назві моделей яких є слово «LAZ», можна застосувати такий розширений фільтр:

Поле:	Модель
Сортування:	
Критерії:	Like *LAZ*
Або:	

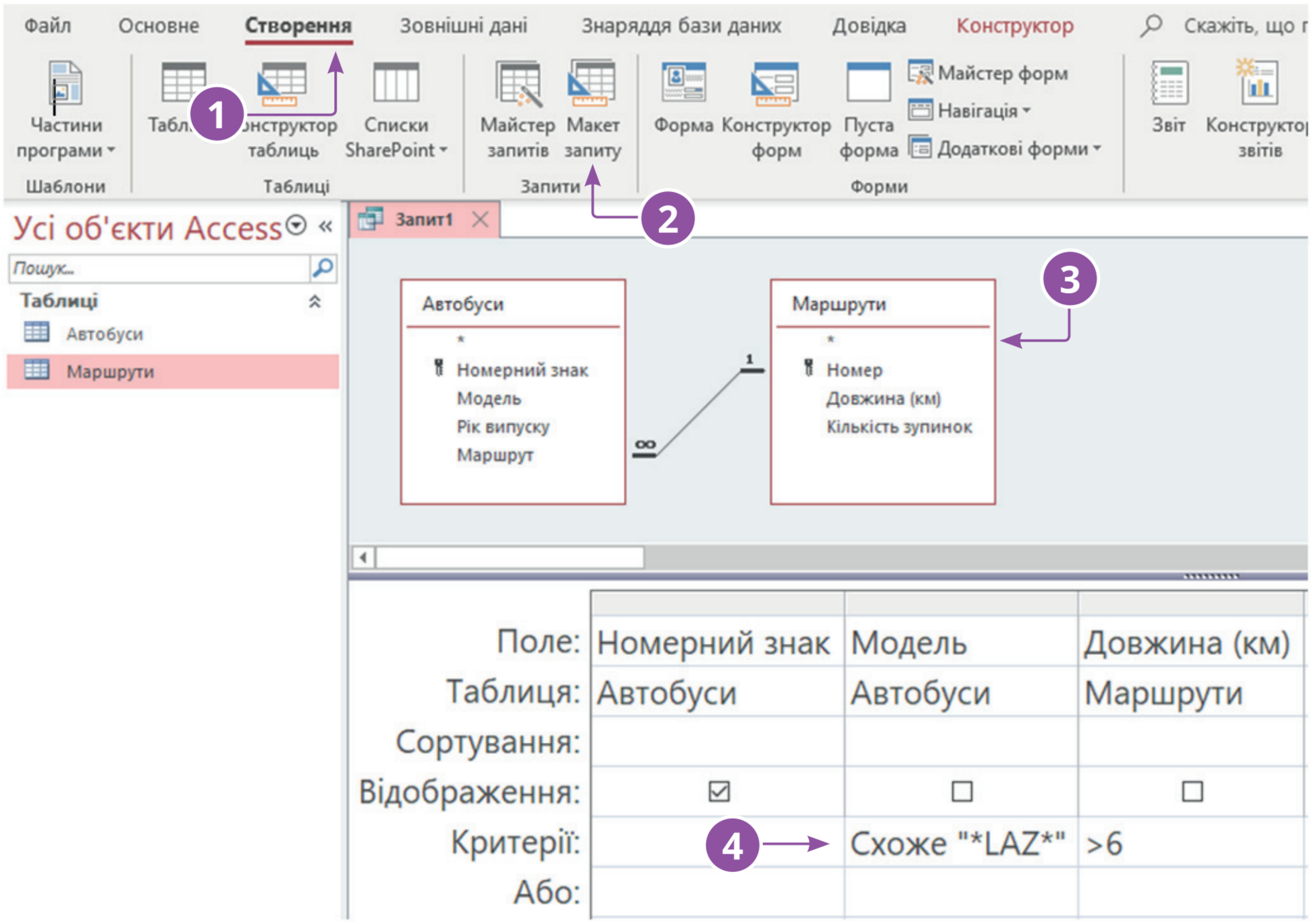
В українській версії MS Access замість слова Like в конструкторі фільтра можна записувати його переклад Схоже.

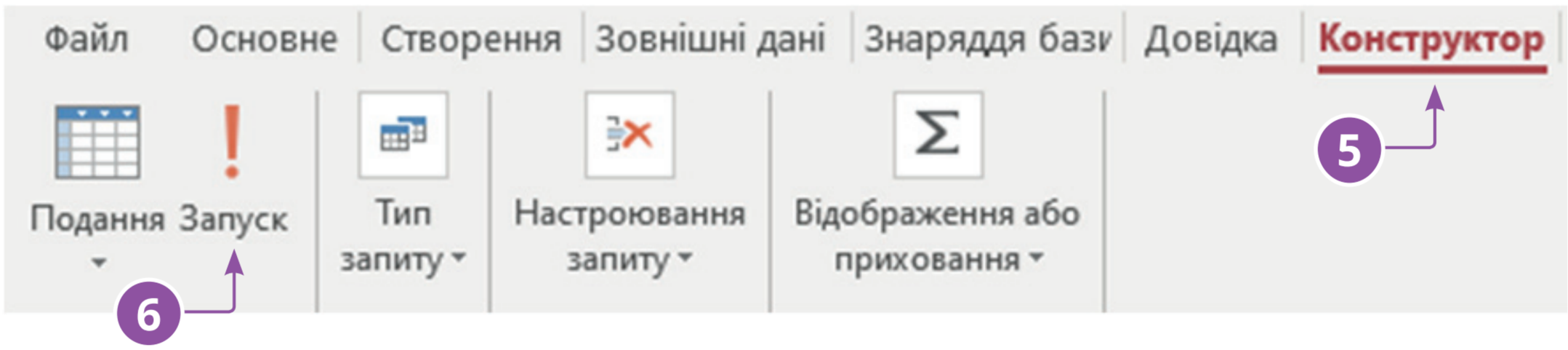
Що таке запит і як його сконструювати?

Запит подібний до фільтра, але може бути застосований до кількох таблиць відразу. Щоб створити запит, потрібно:

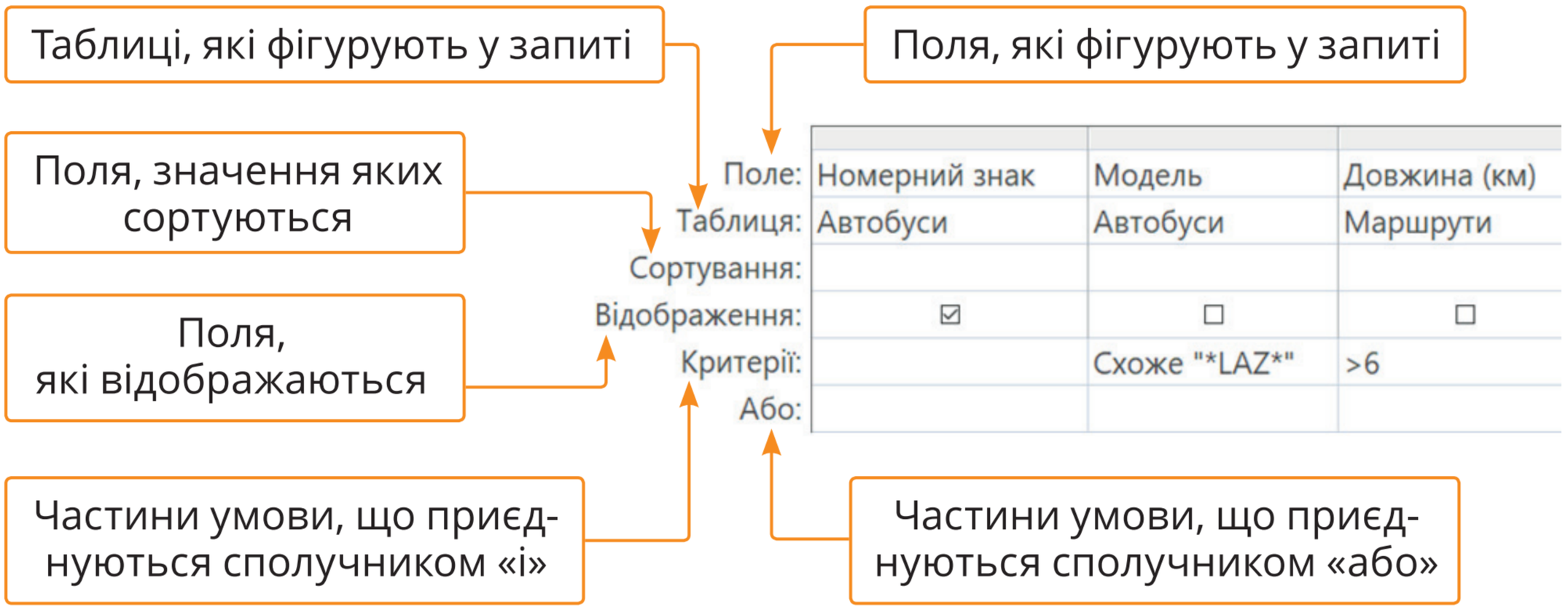
- вибрати команду «Створення\Макет запиту», у результаті виконання якої відкриється вікно конструктора запиту;
- у верхню частину вікна конструктора з області «Усі об’єкти Access» перетягнути задіяні в запиті таблиці;
- у нижній частині вікна конструктора вказати, за якою умовою відбиратимуться записи та які поля відображатимуться.

Після створення запит можна запустити, вибравши на стрічці «Конструктор» команду «Запуск».





Рядки в нижній частині вікна конструктора мають таке призначення:



Наприклад, на рисунку вище зображено запит *Визначити номерні знаки автобусів, у назві моделі яких є слово LAZ і які курсують маршрутами довжиною більше 6 км.*

Увага!

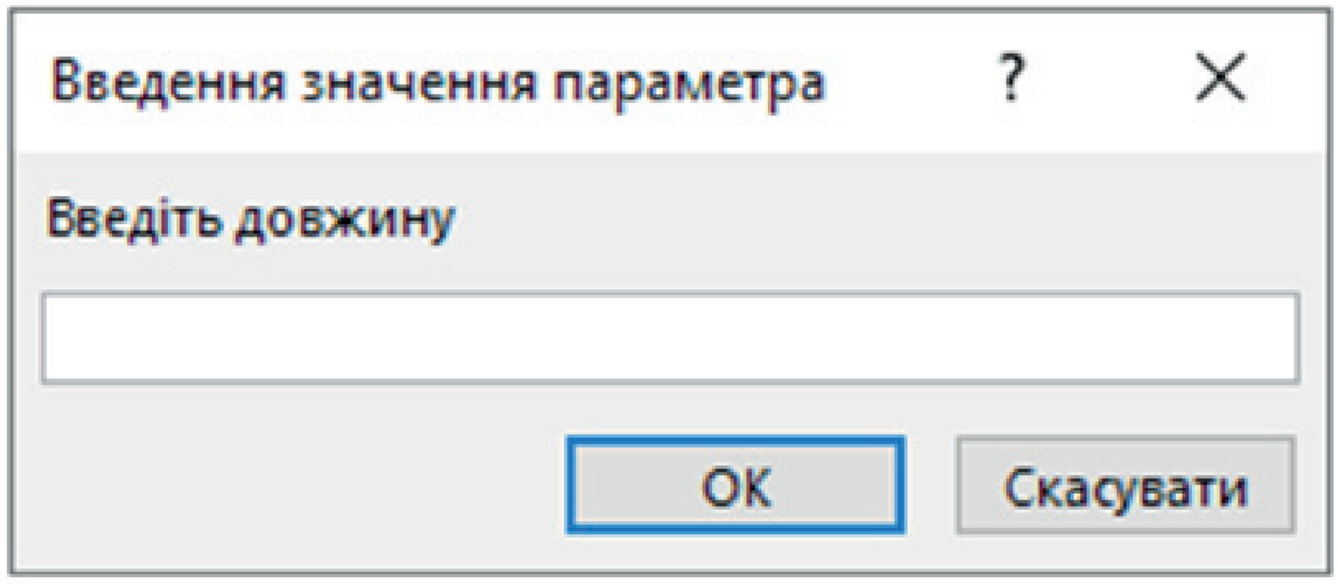
Якщо в запиті фігурують кілька таблиць, вони мають бути з'єднані між собою у вікні «Зв'язки» (ці сполучення відображаються також у верхній частині вікна конструктора запиту). Тоді під час виконання запиту з'єднуватимуться записи з рівними значеннями зв'язаних полів, наприклад автобуси приєднуватимуться саме до своїх маршрутів. Якщо таблиці, що фігурують у запиті, не з'єднані, то всі записи однієї таблиці з'єднаються з усіма записами іншої і в результатах запиту буде мало логіки.

Що таке параметризований запит і як його створити?

У параметризованому запиті значення полів порівнюються не з константами (тобто фіксованими числами чи рядками), а з іншими значеннями, які вводить користувач. Замість такого значення слід записати в квадратних дужках запит на введення. Наприклад, на малюнку зображено запит *Визначити номери маршрутів, довжина яких перевищує введену користувачем, а також моделі автобусів, що курсують цими маршрутами.*

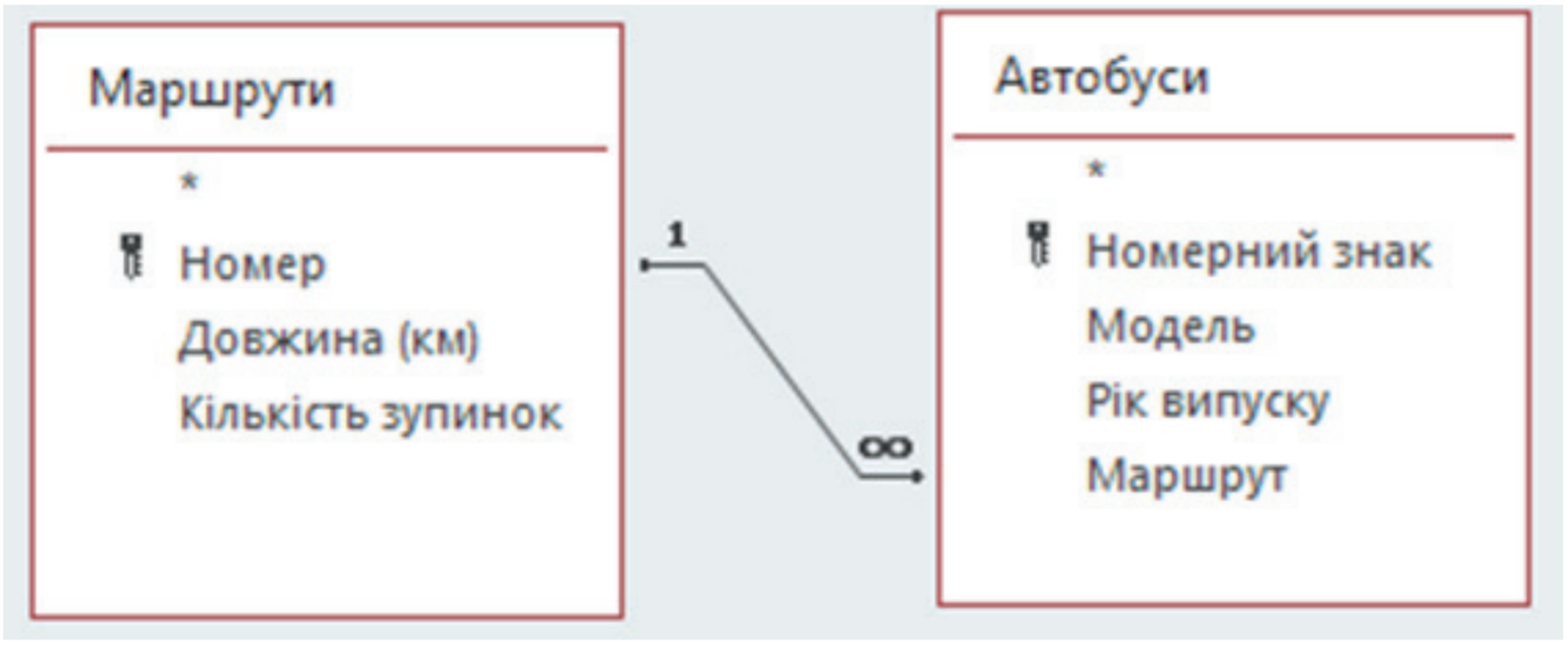
Поле:	Довжина (км)	Номер	Модель
Таблиця:	Маршрути	Маршрути	Автобуси
Сортування:			
Відображення:	☐	☑	☑
Критерії:	>[Введіть довжину]		

Під час виконання запиту буде відображено вікно, у яке користувач має ввести мінімальну довжину шуканого маршруту:



Як у запиті відобразити всі поля певної таблиці?

Потрібно з таблиці у верхній частині вікна запиту перетягнути в рядок «Поле» нижньої частини не назву якогось поля, а символ «*». Наприклад, нижче зображено запит *Визначити всю інформацію про маршрути, якими курсують автобуси 2010 року випуску.*

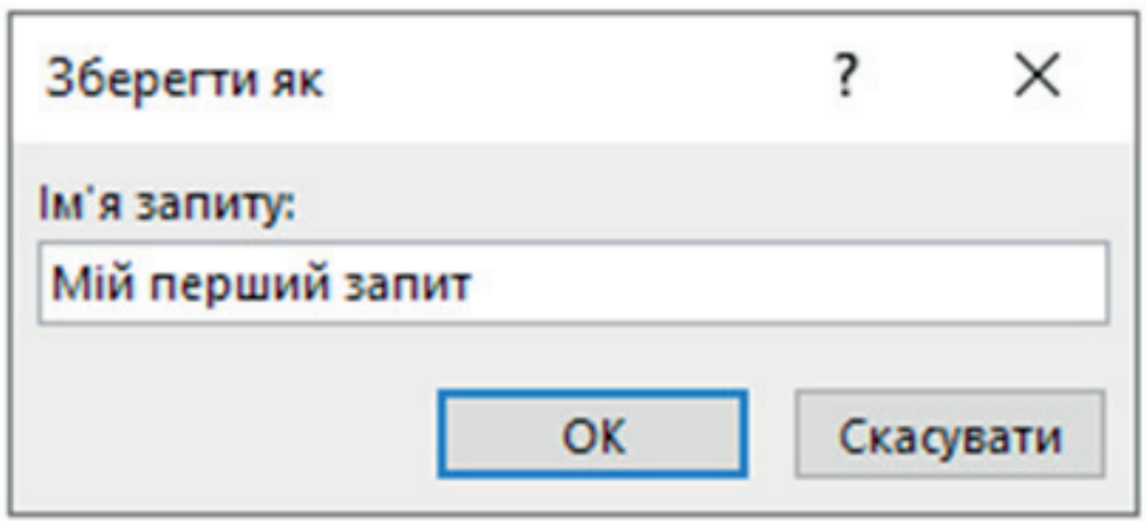


Поле:	Маршрути.*	Рік випуску
Таблиця:	Маршрути	Автобуси
Сортування:		
Відображення:	☑	☐
Критерії:		2010

Як зберегти запит і для чого це робити?

Запити, особливо параметризовані, призначені для багаторазового виконання. Користувач може забажати виконати запит знову з іншими значеннями параметрів або після того, як дані в базі змінилися. Щоб зберегти запит, потрібно:

- закрити вікно його конструктора;
- підтвердити збереження змін;
- у вікні «Зберегти як» ввести назву запиту і натиснути кнопку «ОК».



ПЕРЕВІР СЕБЕ

ЗНАННЯ – РОЗУМІННЯ – АНАЛІЗ – СТВОРЕННЯ – ВЛАСНІ ДУМКИ

1. Поясни відмінності між:
   - звичайними та розширеними фільтрами;
   - розширеними фільтрами та запитами;
   - звичайними та параметризованими запитами.

2. Опиши словами, що знаходять зображені запити для бази даних учнів та вчителів.

А)

Поле:	Прізвище	Назва
Таблиця:	Учні	Класи
Сортування:		
Відображення:	☑	☐
Критерії:		"9А"
Або:		

Б)

Поле:	Класи.*	Ім'я
Таблиця:	Класи	Учні
Сортування:		
Відображення:	☑	☐
Критерії:		"Дмитро"
Або:		"Петро"

В)

Поле:	Класи.*	Учні.*
Таблиця:	Класи	Учні
Сортування:		
Відображення:	☑	☑
Критерії:		
Або:		

Г)

Поле:	Класи.*	Ім'я
Таблиця:	Класи	Учні
Сортування:		
Відображення:	☑	☑
Критерії:		"Дмитро"
Або:	Схоже "9*"	

3. Які із зображених фільтрів завжди відображають усі записи з таблиці автобусів незалежно від її наповнення? Укажи всі правильні відповіді. Їх може бути одна або більше.

А)

Поле:	Маршрут	Рік випуску
Сортування:		
Критерії:	>=10	>2010
Або:	<10	<2010

Б)

Поле:	Маршрут	Рік випуску
Сортування:		
Критерії:	>=10	
Або:	<10	<2010

В)

Поле:	Маршрут	Рік випуску
Сортування:		
Критерії:	>10	
Або:	<10	

Г)

Поле:	Маршрут	Рік випуску
Сортування:		
Критерії:		>2010
Або:		<=2010

4. У базі даних учнів та класів створи такі запити:
   - визначити всю інформацію про класи, де вчиться учень із введеним користувачем прізвищем;
   - визначити прізвища учнів, які або народилися після 01.01.2010, або вчаться в будь-якому з 9-х класів.

5. Уявіть, що ви працюєте над інформаційною системою для шкільної служби підтримки. У базі даних створіть такі запити:
   «Цифрова доступність»: Визначити прізвища та контакти учнів та учениць, які мають особливі освітні потреби (стан здоров'я) або потребують асистентських технологій для дистанційного навчання.
   «Міжнародний обмін»: Визначити всю інформацію про класи, учнів та учениць, які мають досвід проживання в інших країнах або є представниками різних етнічних груп, для організації фестивалю культур (люди різного походження).

6. Чи можна запит, що стосується багатьох таблиць, замінити кількома розширеними фільтрами, застосованими до кожної із цих таблиць? Аргументуй відповідь.

РОЗДІЛ III

АЛГОРИТМИ ТА ПРОГРАМИ

§ 10. Поняття списку

§ 11. Операції з елементами списку

§ 12. Введення елементів списку

§ 13. Елемент керування ListBox

§ 14. Операції з рядками. Шифр Цезаря

§ 15. Упорядкування елементів списку

§ 10. Поняття списку

Для чого у програмуванні потрібні списки?

У табличному процесорі ми працювали з наборами елементів даних. Кожен елемент містився в окремій клітинці. У Python є подібна структура, що називається списком і дає можливість розглядати велику кількість різних значень як одне ціле.

Списки — цей той тип даних, який використовують у Python найчастіше. Адже важко уявити серйозну програму, яка не оперує жодним набором елементів.

У 7 класі ми використовували список кольорів, коли малювали веселку чи кольорового равлика, а у 8 класі в невеличкій програмі «Привітання» з графічним інтерфейсом можна було зі списку вибрати потрібне свято.

Завдання № 1

Дано список слів:

```
w=["пізно.", "Вчитися", "ніколи", "не"]
```

З рубрики «Запитання-відповіді» дізнайся, як отримати доступ до кожного елемента списку. Доповни код, щоб вивести на екран відоме народне прислів'я про навчання

```
print(w[__],w[__],w[__],w[__])
```

Завдання № 2

Реалізуйте код із завдання №1 у програмному середовищі. Зміни елементи списку на слова з інших відомих тобі крилатих висловів, подані у випадковому порядку. Перешли оголошення списку однокласнику чи однокласниці. За допомогою команди **print** виведи вислів зі списку, отриманого від однокласників / однокласниць.

Завдання № 3

1) Створи список материків:

```
continent=["Австралія","Азія","Антарктида","Африка","Європа",
"Південна Америка","Північна Америка"]
```

Виведи цей список за допомогою однієї команди **print**.

2) На наступній сторінці наведено інформацію про площі материків. Створи другий список **area** цих площ.

3) З рубрики «Запитання-відповіді» дізнайся, як вивести список, по одному елементу в кожному рядку, та доповни відповідний код:

```
for i in range(_____ ):
    print(continent[____ ],"–",area[____ ],"тис. кв. км")
```

Цей код має виводити інформацію про материки та їхні площі в такому форматі:

Австралія — 7687 тис. кв. км
Азія — 43 820 тис. кв. км
Антарктида — 13 970 тис. кв. км
Африка — 30 300 тис. кв. км
Європа — 10 180 тис. кв. км
Південна Америка — 18 280 тис. кв. км
Північна Америка — 24 250 тис. кв. км

4) Реалізуй у програмному середовищі отриманий код.

Увага!

Щоб не рахувати кількості елементів, можна скористатися функцією визначення довжини списку: `len(ім'я списку)`

Завдання № 4

Дізнайся з рубрики «Запитання-відповіді», як обчислити суму елементів числового списку, а також максимальне, мінімальне та середнє значення. Доповни код із попереднього завдання обчисленням мінімальної, максимальної, сумарної та середньої площі континентів.

Завдання № 5

Створи список із будь-яких десяти чисел. Згадай із матеріалу 7 класу, знайди в рубриці «Запитання-відповіді» або в інтернеті інформацію щодо команди **range** з двома або трьома аргументами. Запрограмуй виведення:

1) елементів списку з другого по восьмий;
2) елементів списку з парними індексами;
3) елементів списку з непарними індексами;
4) елементів списку у зворотному порядку.

Завдання № 6

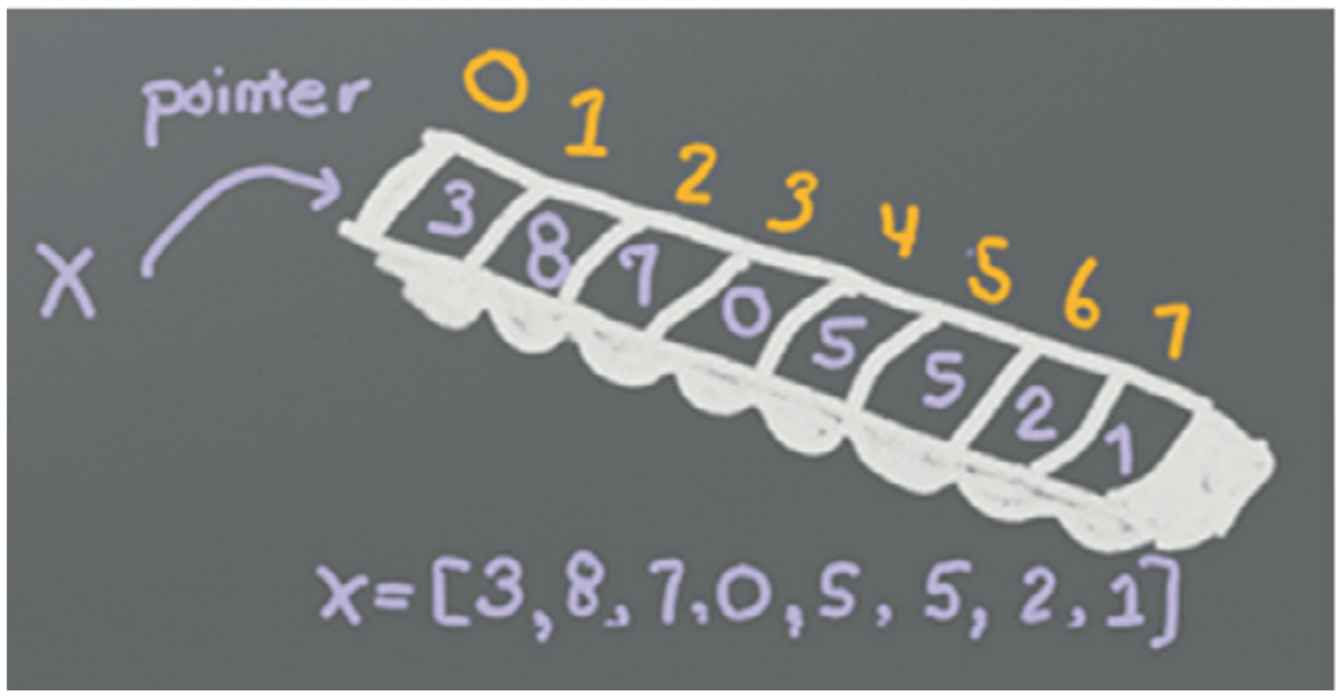
У списку можуть зберігатися різнотипні елементи. З огляду на це спробуй виконати завдання № 3, використавши лише один список, у якому зберігатимуться і назви материків, і їхні площі. Можеш зберігати назви материків в елементах з парними індексами, а площі — з непарними. Або назви в першій половині списку, а площі — у другій. Подумай, у якому діапазоні та з яким кроком має змінюватися лічильник циклу, що перебирає елементи списку.

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке список у Python?

Список *— це вбудований тип даних, який дає можливість зберігати впорядкований набір значень, що називаються* ***елементами*** *списку. Кожний елемент списку має свій порядковий номер, який називається індексом.*

У Python, як і в багатьох інших мовах програмування, нумерація елементів списку починається з 0.



Якщо провести аналогію з рядком чи стовпцем даних у табличному процесорі, то значення в клітинках — це елементи списку, а адреси цих клітинок — індекси.

	A
1	25
2	30
3	45
4	50
5	17

Як створити список?

У Python список записують як набір перелічених через кому елементів у квадратних дужках **[]**. Його можна присвоювати змінній. Наприклад,

```
n=[5,23,7,-8] або spysok=[1,-2,"hello",0]
```

Також для створення списку можна використовувати конструктор **list()**. Обидва наступні рядки коду створюють порожній список.

```
Spysok1=[]
Spysok2=list()
```

Ініціалізація звичайної змінної	Ініціалізація списку
A= 5	A=[5,6,7,-8]
B= "hello"	B=["one","two","three"]

Як вивести весь список?

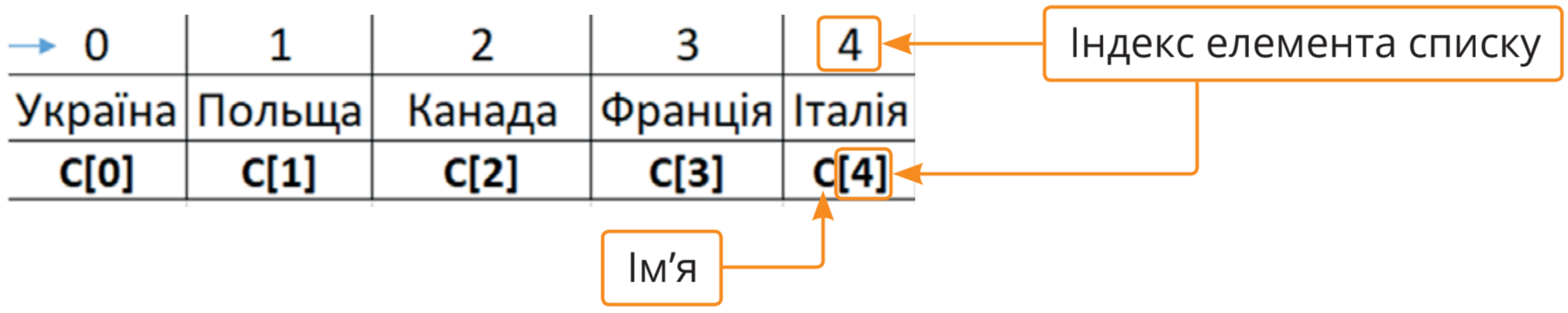
Відразу весь список можна вивести, вказавши в команді print його ім’я.

```
print(ім’я списку)
```

Як отримати доступ до елемента в списку та вивести його на екран?

Потрібно записати ім’я списку, а після нього — індекс елемента в квадратних дужках.

Індекс першого елемента списку дорівнює 0.



Наприклад, у результаті виконання наведених нижче команд буде виведено число 23:

```
A=[5,23,7,-8]
print(A[1])
```

Як вивести кожен елемент списку окремо з нового рядка?

Якщо в команді print просто вказати ім’я списку, його елементи будуть виведені в одному рядку. А щоб вивести елементи списку «у стовпчик», скористайся циклом for:

```
>>>
====
5
23
7
23
```

File Edit Format Run Options Wi

```
A=[5,23,7,-8]
for i in range(3):
    print(A[i])
```

Як працює команда range?

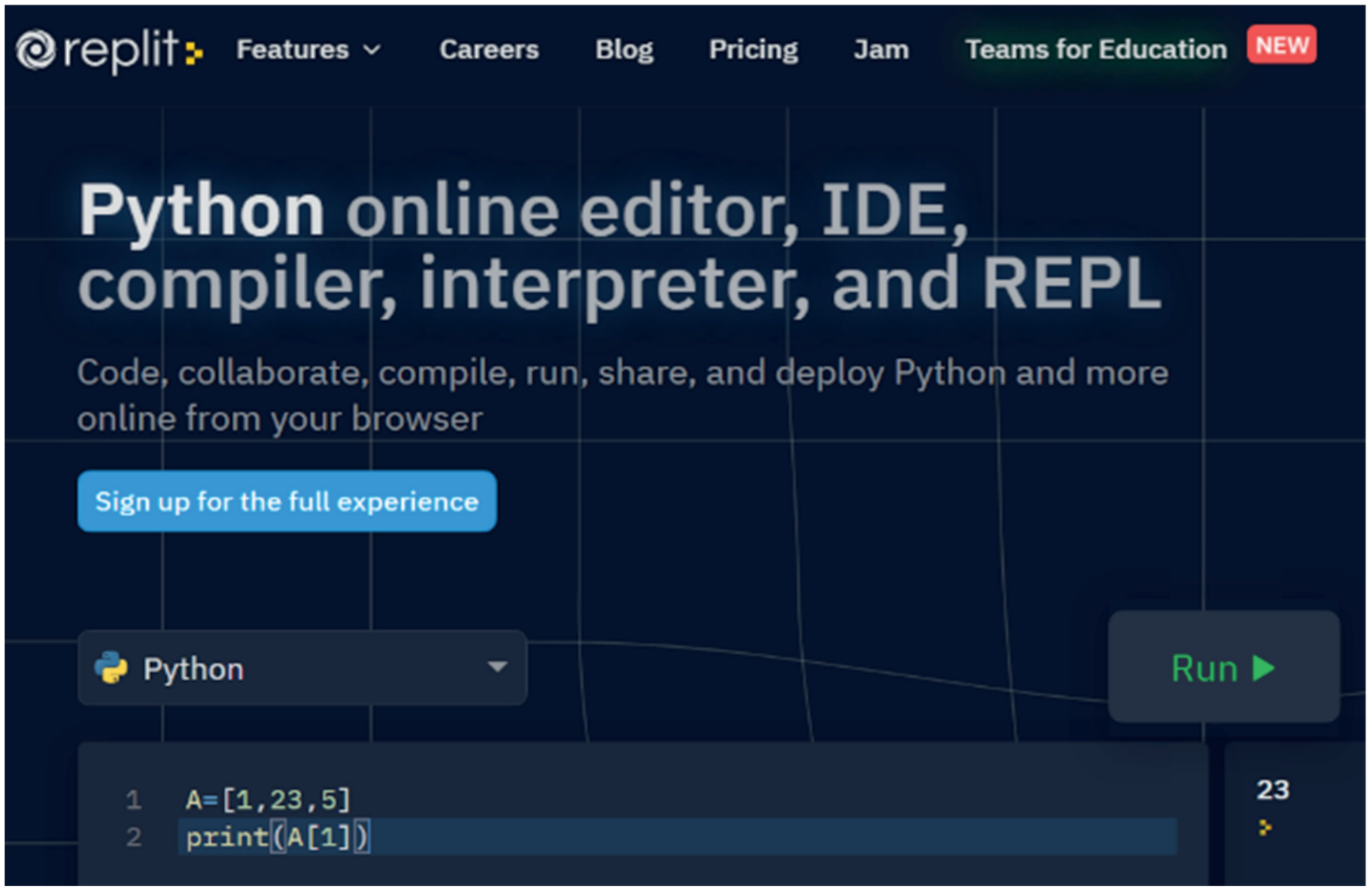
Щоб сформувати діапазон значень, які набуватиме змінна, у мові Python використовується спеціальна функція **range**.

```
range(k,n,m)
```

- **k** — початкове значення діапазону;
- **n** — наступне число після кінцевого значення діапазону, тобто діапазон буде завершено значенням n–1;
- **m** — крок, на який збільшуватиметься змінна.

Рекомендуємо!

Python IDLE — найпростіше середовище розробки програм, що встановлюється разом із Python. Однак не найзручніше! Рекомендуємо онлайновий редактор **replit**! На лівій панелі можеш набрати код, натиснути кнопку **Run** — і результат його виконання одразу відобразиться на правій панелі.



https://repl.it/languages/python3

Як знайти максимальне, мінімальне значення та суму елементів числового списку?

Знаходження **суми** — sum(ім'я списку)

Знаходження **максимального** значення — max(ім'я списку)

Знаходження **мінімального** значення — min(ім'я списку)

Як знайти середнє значення елементів списку?

Щоб знайти середнє значення елементів списку, можна їхню суму поділити на кількість елементів списку, яка визначається за допомогою вбудованої функції **len(ім'я списку)**.

ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Яка команда виведе перший елемент списку? `x = [1, 2, 3, 4, 5]`

а) `print( [1] )`

б) `print( x[0] )`

в) `print( x[1] )`

г) `print( x )`

д) `print( x(0) )`

2. Що виведе зображений нижче фрагмент коду?

```
my_array = [2, 4, 6]
for i in range (3) :
    print(my_array)
```

а)
```
[2, 4, 6]
```

б)
```
0
1
2
```

в)
```
1
2
3
```

г)
```
[2, 4, 6]
[2, 4, 6]
[2, 4, 6]
```

д)
```
246
```

е)
```
2
4
6
```

3. Як створити числовий список?
4. Назви три функції для роботи зі списком та поясни їхнє призначення.
5. Укажи фрагмент коду, за допомогою якого список **seasons** буде виведено в зображеному вигляді.

```
seasons=["зима", "весна", "літо", "осінь"]
```

```
зима
весна
літо
осінь
```

А)
```
for i in range (4):
    print(seasons[i])
```

Б)
```
print(seasons)
```

В)
```
for i in range (1,4):
    print(seasons[i])
```

6. У результаті використання програмного коду змінні **x**, **y** та **z** одержали такі значення:

```
x=-8, y=5, z=10
```

Проаналізуй елементи списку та вкажи функції, завдяки яким змінні могли одержати такі значення.

```
list=[2, 5, -8, -4, 10]
x= ☐ (list)
y= ☐ (list)
z= ☐ (list)
```

7. Тобі добре відома гра «Камінь-ножиці-папір». У її комп'ютерному варіанті програма вибирає один предмет, користувач — інший, після чого програма має порівнювати вибір та виводити результат. У 8 класі ми вже програмували цю гру, а зараз зробимо це за допомогою списків. Ти побачиш, що код вийде компактнішим!

1) Створи список рядків "камінь", "ножиці", "папір".
2) За допомогою модуля random реалізуй випадковий вибір індексу елемента цього списку, за яким отримай значення елемента — це і буде вибір з боку ПК.
3) За допомогою команди input реалізуй введення вибору користувача.
4) Порівняй вибір користувача та комп'ютера за допомогою операторів **if** та виведи на екран повідомлення про виграш, програш або нічию:
   - якщо вибрано камінь та ножиці, виграє камінь;
   - якщо вибрано камінь та папір, виграє папір;
   - якщо вибрано ножиці та папір, виграють ножиці;
   - якщо вибрано два однакових предмети, оголошується нічия.

8. Які переваги мають списки порівняно із числовим і текстовим типами даних?

§ 11. Операції з елементами списку

Як зручніше опрацьовувати списки — в електронних таблицях чи за допомогою мови програмування?

Минулого уроку ми навчилися створювати списки та виконувати деякі операції відразу над усіма їхніми елементами: знаходити суму елементів, а також їхнє максимальне, мінімальне та середнє значення. Однак часто ці та інші операції потрібно виконувати не з усіма елементами списку, а лише з тими, які задовольняють певну умову. Наприклад, уяви, що ти шукаєш найшвидший автобусний рейс із Житомира до Києва. Ти відкриваєш список рейсів з автовокзалу м. Житомир. Тоді тобі треба знайти рейс із мінімальним значенням параметра «час у дорозі», але не серед усіх рейсів, а лише серед тих, у яких пунктом призначення є Київ. Подібні задачі ми розглянемо сьогодні.

Завдання № 1

Дано список курсу біткоїна до долара США за місяцями 2020 року.

```
rate=[7339.5,
      9328.25,
      8525.8,
      6872.0,
      8982.85,
      9666.80,
      9058.05,
      11071.65,
      10148.15,
      10551.35,
      19178.00,
      28895.25]
```

Заповни пропуски в коді визначення кількості місяців, у яких курс перевищував 9000 доларів, та прокоментуй кожен рядок коду. Можеш скористатися рубрикою «Запитання-відповіді».

```
k= ____
for element in ____:
    if element>____:
        k= ____
print(____)
```

Завдання № 2

Реалізуй програмний код із завдання № 1 у середовищі розробки програм. Перевір результат.

Завдання № 3

Заповни пропуски в коді, що виводить назви та кількість місяців, коли курс біткоїна перевищував 9000 доларів. Зауваж, що потрібно перевірити значення кожного елемента списку **rate** для того, щоб вивести назву місяця зі списку **month**.

```
month=["січень",
       "лютий",
       "березень",
       "квітень",
       "травень",
       "червень",
       "липень",
       "серпень",
       "вересень",
       "жовтень",
       "листопад",
       "грудень"]
```

```
k= [ ]
for i in [ ]:
    if [ ]>[ ]:
        k= [ ]
        print(month[ ], "-", kurs [ ], "$")
print([ ], "місяців")
```

Завдання № 4

Реалізуй у програмному середовищі доповнений тобою код із завдання № 3. Результат виконання програми має бути таким:

```
лютий – 9328.25$
червень – 9666.8$
липень – 9058.05$
серпень – 11071.65$
вересень – 10148.15$
жовтень – 10551.35$
листопад – 19178.0$
грудень – 28895.25$
Кількість місяців: 8
```

Завдання № 5

Тепер дізнаймося, яким був середній курс біткоїна в тих місяцях, коли він перевищував 9000 доларів. Для цього потрібно спочатку визначити суму курсів у місяцях, що задовольняють умову. Кількість таких місяців було визначено в завданні № 2. Доповни самостійно код командами визначення суми та середнього курсу за ці місяці й перевір результат. Якщо ти правильно ввів початкові елементи списку, то середній курс має дорівнювати 13487,19 доларів.

Завдання № 6

Зробимо нашу програму більш універсальною: тепер вона має рахувати середній курс не протягом року, а за введений користувачем період, наприклад з 4-го по 9-й місяць. Крім того, межа, яку має перевищувати курс, тепер буде не фіксованою (9000 дол.), а її вводитиме користувач.

1) Заповни пропуски в коді, який реалізує це завдання, та прокоментуй кожен доповнений тобою рядок.

```
np=int(input("Введіть номер початкового місяця"))
nk=int(input("Введіть номер кінцевого місяця"))
z=float(input("Введіть курс, більше якого проводити обчислення"))
k=
suma=
for i in range ():
    if >z:
        k=k+1
        suma=
        print(month[], "-", kurs [], "$")
if k!=0:
    print("Середній курс за", , "м.-", , "$")
else:
    print('''Відсутні дані у списку для обчислення середнього
        курсу з врахуванням введеного курсу >''', , "$")
```

2) Подумай, чому в коді є перевірка, чи кількість місяців **k** не дорівнює 0.

Завдання № 7

Перевір на практиці роботу коду, доповненого в завданні № 6.

Якщо ти введеш такі дані:

Введіть номер початкового місяця 1
Введіть номер кінцевого місяця 3
Введіть курс, більше якого проводити обчислення 8500

то маєш отримати результат:

лютий — 9328.25 $
березень — 8525.8 $
Середній курс за 2 м. — 8927.02 $

Завдання № 8

Зміни код із попереднього завдання так, щоб обчислювалася кількість місяців у 2020 році, у які курс зростав більше ніж на **5 000 доларів,** та кількість місяців, у яких відбувалося зростання більше ніж на введений користувачем відсоток зростання **percent_growth**. Заповни пропуски в коді.

```
percent_growth=int(input("Введіть відсоток росту"))
k1=0
k2=0
print('''Перелік місяців, у яких курс зростав більше ніж на 5000–'"')
for i in range (      ):
    if                  >5000:
        k1=k1+1
        print(month[i+1], "-", kurs [i+1], "$")

print('''Всього — '", k1, "м.")
print('''Перелік місяців, у яких курс зростав більше ніж на–'",
      percent_growth, "%")
for i in range (      ):
    if                  :
        k2=k2+1
        print(month[i+1], "-", kurs [i+1], "$")
print('''Всього — '", k2, "м.")
```

Реалізуй цю програму на комп’ютері та перевір її роботу.

ЗАПИТАННЯ-ВІДПОВІДІ

Який загальний вигляд коду для обчислення суми елементів списку, що задовольняють умову?

```
s=0
for лічильник in range (len(ім’я списку)):
    if умова:
        s=s+ім’я списку[лічильник]
```

Шукана сума накопичується у змінній s.

Як знайти кількість елементів списку, що задовольняють умову?

```
k=0
for лічильник in range (len(ім’я списку)):
    if умова:
        k=k+1
```

Шукана кількість накопичується у змінній k.

Як повідомлення, що виводиться командою print, записати в кількох рядках?

Для цього потрібно початок та кінець повідомлення взяти у тричі записані одинарні лапки. Наприклад,

```
print('''перше слово
        дорожче за друге''')
```

Як створити список з багатьма однаковими елементами?

Потрібно скористатися операцією *****. Наприклад, за допомогою команди **A=1000*[5]** створюється список **A**, усі елементи якого мають значення 5, а їх кількість дорівнює 1000.

Що таке біткоїн?

Біткоїн — це найперша і найпопулярніша у світі криптовалюта, тобто електронний аналог грошей, які не випускаються жодною державою або якоюсь приватною особою. Ці гроші обертаються виключно в інтернеті і можуть передаватися між електронними «гаманцями» — спеціальними вільнопоширюваними програмами чи файлами. Кожен гаманець захищений паролем, що відомий лише його власнику.

ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Доповни код, що визначає середнє арифметичне елементів списку **a**, які кратні 3:

```
a=[9, 3, 5, -12, 10, 25]
s=0
k=0
for i in range(len(a)) :
    if [          ] :
        s=[    ] #обчислення суми
        k=[    ] #обчислення кількості
s_avg=[    ] #обчислення середнього арифметичного
print(s_avg)
```

2. Які дії виконує наведена програма?

```
list1=[2, 13, 4, -5, 6, 17]
k=0
s=0
for i in range(len(list)):
    if list[i]%2==0:
        k=k+1
    if list[i]>0:
        s=s+list[i]
```

3. Доповни програму так, щоб у результаті було виведено 26 і 23.

```
list1=[1, 5, 2, 10, 8]
list2=[2, 4, 9, 7, 1, 5, 1, 4]
s1=0
s2=0
for i in range(5) :
    s1=    list1[i]
    s2=    list2[i]
print(s1)
print(s2)
```

4. Проаналізуй наведені нижче фрагменти коду та вкажи той, у якому визначається загальна кількість додатних елементів у списках **a** і **b**:

```
k=0
for i in range(len(a)):
    if a[i]>0:
        k=k+1
    if b[i]>0:
        k=k+1
```

```
k=0
for i in range(len(a)):
    if a[i]>0:
        k=k+1
if b[i]>0:
    k=k+1
```

```
k=0
for i in range(len(a)):
    if a[i]>0 or b[i]>0:
        k=k+1
```

```
k=0
for i in range(len(a)):
    if a[i]>0 and b[i]>0:
        k=k+1
```

5. Обери списки, у результаті опрацювання яких цією програмою буде отримано значення змінних **k=3**, **s=–11**.

```
k=0
s=0
for i in range(len(a)-1):
    if a[i]>a[i+1]:
        k=k+1
    if a[i]<0:
        s=s+a[i]
print(k,s)
```

- ☐ a=[2, 4, 6, 1, 3, 5, -2, -4, -6]
- ☐ a=[1, 3, 5, -4, -6, -1, 0]
- ☐ a=[-1, 8, -4, -3, -2, 3, 2]
- ☐ a=[4, 11, -10, -1, 7, 2, -1]
- ☐ a=[1, -8, 4, 3, 12, -3, -2]

6. Які з наведених значень можуть бути виведені цією програмою (користувач може вводити будь-яке значення **t**)?

```
k=1000*[0]
s=0
t=int(input())
for i in range(t):
    a[i]=i
for i in range(t):
    if a[i]%5==0:
        s=s+a[i]
print(s)
```

- ☐ 4
- ☐ 5
- ☐ 10
- ☐ 15
- ☐ 25

7. Створи список, елементи якого містять кількість населення десяти найбільших міст України. Визнач кількість міст, у яких проживає понад 1 мільйон осіб, а також загальну кількість населення цих міст.

8. Створи програму, яка:
   1) створює два списки з від'ємними та додатними елементами, по 8 елементів у кожному;
   2) визначає, у якому списку більше додатних елементів;
   3) визначає, сума додатних елементів якого списку більша.

9. Спробуй реалізувати в електронній таблиці розглянуті сьогодні операції з елементами списків. Які переваги в опрацюванні списків має мова програмування порівняно з електронними таблицями?

§ 12. Введення елементів списку

Як у Python перетворити список рядків на список чисел за допомогою одного рядка коду?

На минулих уроках ми навчилися виконувати дії зі списками, створеними в програмному коді. А якщо ми хочемо, щоб значення елемента списку вводив користувач?

Для цього є кілька способів. Наприклад, ці значення можна ввести і зчитати з консолі як звичайний текстовий рядок, а в графічному інтерфейсі елементи списку можна ввести та зчитати з текстового поля.

Завдання № 1

Наведений нижче код дає користувачу можливість ввести послідовність чисел через кому, а потім перетворює отриманий рядок на числовий список.

1) Дізнайся з рубрики «Запитання-відповіді», як перетворити текстовий рядок на список, і заповни пропуски в коді:

```
ryadok=input('''Введіть елемент списку
                в одному рядку через кому\n''')
print('Ви ввели –', ryadok)
spysok = ryadok, [    ] ('[    ]')
print('Створено список, елементи якого є типу str: \n', spysok)
spysok_number = [int(i) for i in spysok]
print('Числовий список: \n', spysok_number)
```

2) Спробуй описати послідовність дій, які виконуються в рядку

   spysok_number = [int(i) for i in spysok]

Для цього дай відповіді на запитання:

- Що буде міститися у змінній **spysok**, якщо користувач ввів рядок 2,3,5,6?
- Яких значень набуває змінна «i», якщо користувач ввів рядок 2,3,5,6?
- Для чого в коді використовується функція int()?
- Яких значень набуватиме вираз int(i), якщо користувач ввів рядок 7,8,9,10?
- Що буде міститися у змінній **spysok_number**, якщо користувач ввів рядок 7,8,9,10?

Завдання № 2

1) Реалізуй на практиці код із завдання № 1. Запусти його, введи текст 1,2,3,4,5 і перевір, чи відображається такий результат:

```
Введіть елементи списку в одному рядку через кому
1, 2, 3, 4, 5
Ви ввели — 1, 2, 3, 4, 5
Створено список, елементи якого є типу str:
['1', '2', '3', '4', '5']
Числовий список:
[1, 2, 3, 4, 5]
```

2) Ти вже вмієш опрацьовувати числовий список, використовуючи функції **max**, **min** та **sum**. Доповни код, щоб у результаті його реалізації було виведено максимальне, мінімальне та середнє значення числового списку.

Завдання № 3

Користувачу простіше буде вводити елементи списку не через кому, а через пропуск. Як ти думаєш, що потрібно змінити в коді з попереднього завдання?

Реалізуй у програмному середовищі змінений тобою код.

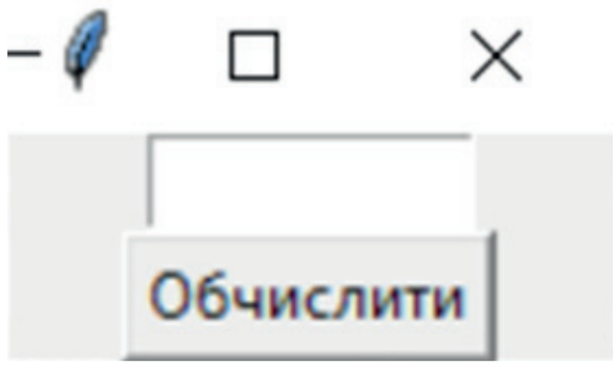
Завдання № 4

Зміни код із попереднього завдання так, щоб користувач сам визначав, який символ буде розділювачем у рядку для елементів списку.

Завдання № 5

Повторимо основні команди створення графічного інтерфейсу програми з 8 класу. Проаналізуй наведений нижче фрагмент програми, а також результат його виконання, і поясни призначення кожного рядка коду:

```
from tkinter import *
window=Tk()
text_pole=Entry(window, width=10)
text_pole.pack()
knopka=Button(text='Обчислити')
knopka.pack()
```



Якщо виникають складнощі, скористайся рубрикою «Запитання-відповіді».

Завдання № 6

Реалізуй у Python створення зображеного вище вікна програми з текстовим полем і кнопкою.

Завдання № 7

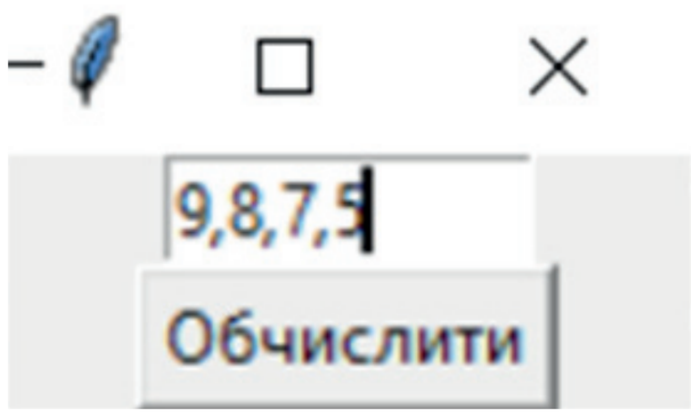
1) Заповни пропуски в коді введення списку з текстового поля та виведення його у вікні консолі:

```
from tkinter import *
window=Tk()
def button_click():
    #Зчитування рядка з текстового поля
    string=text_box____()
    #Перетворення зчитаного рядка на список
    string_list=string____(' , ')
    #Перетворення створеного списку на числовий
    number_list=[____ (i) for ____ in ____]
    print(number_list)
text_box=Entry(window, width=10)
text_box.pack()
btn1=Button(text='Обчислити', command=button_click)
btn1.pack()
```

2) Поясни, які рядки коду виконуються в разі натискання кнопки 'Обчислити'.

Завдання № 8

Реалізуй код із завдання № 7 на практиці. Перевір результат роботи програми:




```
====RESTART: F:/
>>> [9, 8, 7, 5]
```

ЗАПИТАННЯ-ВІДПОВІДІ

Як перетворити рядок на список?

Щоб перетворити рядок на список, потрібно до цього рядка застосувати метод **split**. Його аргументом є символ-роздільник, за яким рядок поділяється на підрядки. Ці підрядки стають елементами списку.

Наприклад, ось як за допомогою методу **split** перетворити на список рядок '24/09/2021':

```
data = '24/09/2021'
spysok=data.split('/')
```

Якщо дописати до вищенаведеного коду print(spysok), то отримаємо такий результат:

```
====RESTART: F:/
['21', '09', '2021']
>>>
```

Як реалізувати введення списку користувачем?

Можна ввести рядок у якусь змінну з консолі, а потім перетворити значення цієї змінної на список:

```
data=input()
spysok=data.split('/')
```

Якщо не потрібно зберігати введений текст, можна обійтися лише одним рядком коду:

```
data=input()
spysok=data.split('/')
```

→

```
spysok=input().split('/')
```

Якщо в методі **split** не вказувати ніякого аргументу, то роздільником вважатиметься символ пропуску. Це корисно, наприклад, тоді, коли потрібно перетворити речення на список слів:

```
s='Я вивчаю програмування'
words=s.split()
```

Як перетворити список рядків на числовий список?

Для перетворення елементів списку в цілочисловий тип даних можна скористатися класичним перебором елементів списку, який ми розглядали в попередньому параграфі за допомогою циклу for та функції перетворення на цілочисловий тип int():

```
for i in range (len(string_list)):
    number_list[i]=int(string_list[i])
```

У Python існує і більш компактний спосіб такого перетворення:

```
spysok_number = [int(element) for element in spysok]
```

- spysok_number — Ім'я нового числового списку
- int(element) — Перетворення елемента
- for element in spysok — Для кожного елемента в списку

ЗАПИТАННЯ НА ПОВТОРЕННЯ МАТЕРІАЛУ 8 КЛАСУ

Як створюється вікно програми?

1. Підключається модуль tkinter.
2. За допомогою конструктора Tk створюється вікно.

```
from tkinter import *
window = Tk()
```

Як створити текстове поле у вікні?

За допомогою конструктора Entry та методу pack(), що розташує створене поле у вікні:

```
text_pole=Entry(window,width=10)
text_pole.pack()
```

Як створити кнопку у вікні?

За допомогою конструктора **Button** та методу **pack()**.

```
btn1=Button(text='Текст на кнопці')
btn1.pack()
```

Як опрацьовувати подію натискання кнопки?

1. Потрібно створити *обробник події* — функцію, яка виконуватиметься в разі натискання кнопки.

```
def button_click():
    #Команди обробника
```

2. У конструкторі кнопки потрібно вказати параметр **command**, якому присвоїти ім'я функції — обробника події натискання кнопки.

```
btn1=Button(text='Текст на кнопці'), command=button_click
```

Як зчитати з текстового поля дані, введені користувачем?

За допомогою методу **get()**:

```
ryadok=text_pole.get()
```

ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Запиши одним рядком код перетворення списку чисел **a** на список **b** рядків, що подають ці числа в текстовому вигляді.

```
a=[1, 2, 3]
```

```
b=['1', '2', '3']
```

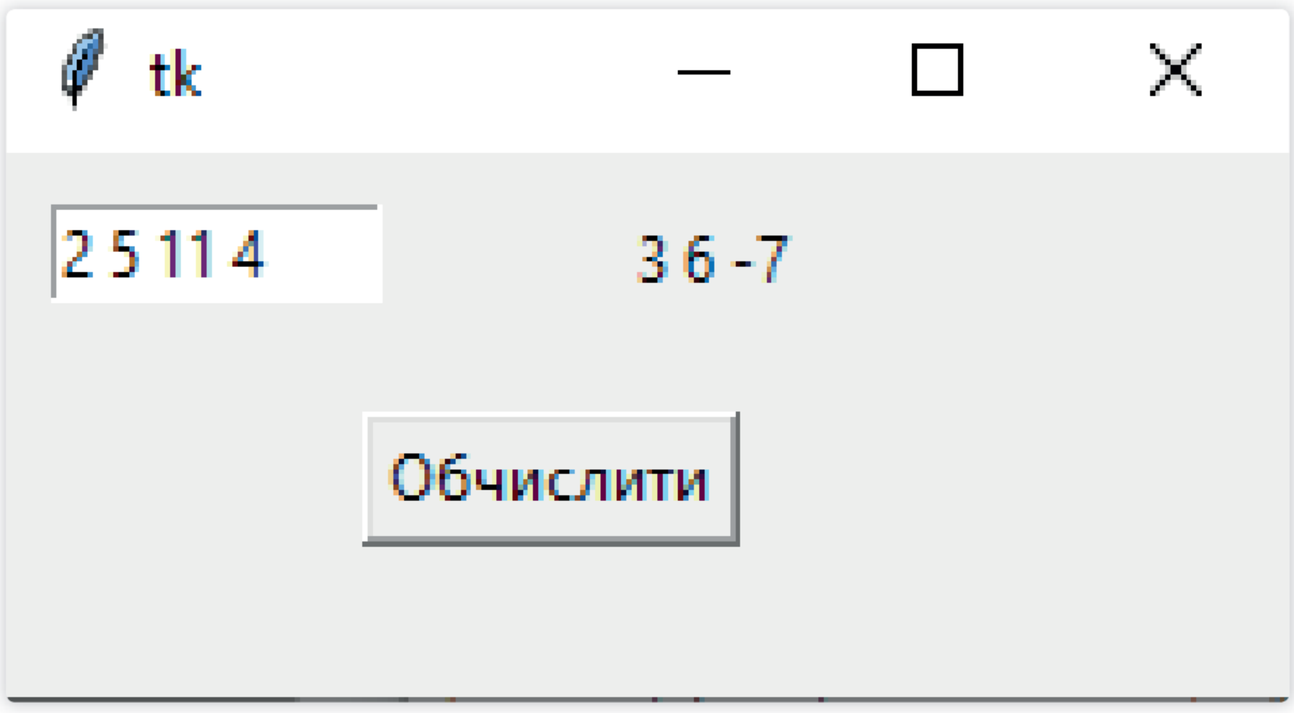
2. Проаналізуй кожен рядок коду та визнач, за допомогою якого з них можна створити список дійсних чисел, записаних у текстовому полі **ent** через пропуск.

```
a=[int(i) for i in ent.get().split()]
a=[float(i) for i in ent.get().split(" , ")]
a=[float(i) for i in ent.get().split()]
a=[i for i in ent.get().split()]
```

3. Минулого уроку ми створювали список середнього курсу біткоїна за місяцями 2020 року безпосередньо в коді. Зроби цю програму більш універсальною, а саме:

- забезпеч введення списку курсу валюти з текстового поля, щоб користувач мав можливість вводити послідовність чисел;
- доповни код для виведення середнього значення введених чисел у написі;
- доповни код для того, щоб вивести в написі кількість місяців, у яких відбувалося підвищення курсу порівняно з попереднім місяцем. Якщо не пам'ятаєш, як виводити в написі текст, скористайся рубрикою «Запитання-відповіді» з наступного розділу.

4. Створи програму із зображеним на малюнку інтерфейсом. У ній користувач вводить у текстове поле послідовність розділених пробілами чисел, натискає кнопку «Обчислити» — і програма виводить у написі праворуч послідовність різниць між сусідніми елементами введеного списку.



5. Порівняй введення списку за допомогою текстового поля в графічному інтерфейсі та введення списку за допомогою рядка в консольному режимі. Назви переваги та недоліки.
6. Порівняй два способи перетворення списку рядків на список чисел, наведені в рубриці «Запитання-відповіді». Який спосіб кращий? Аргументуй свою думку.

§ 13. Елемент керування ListBox

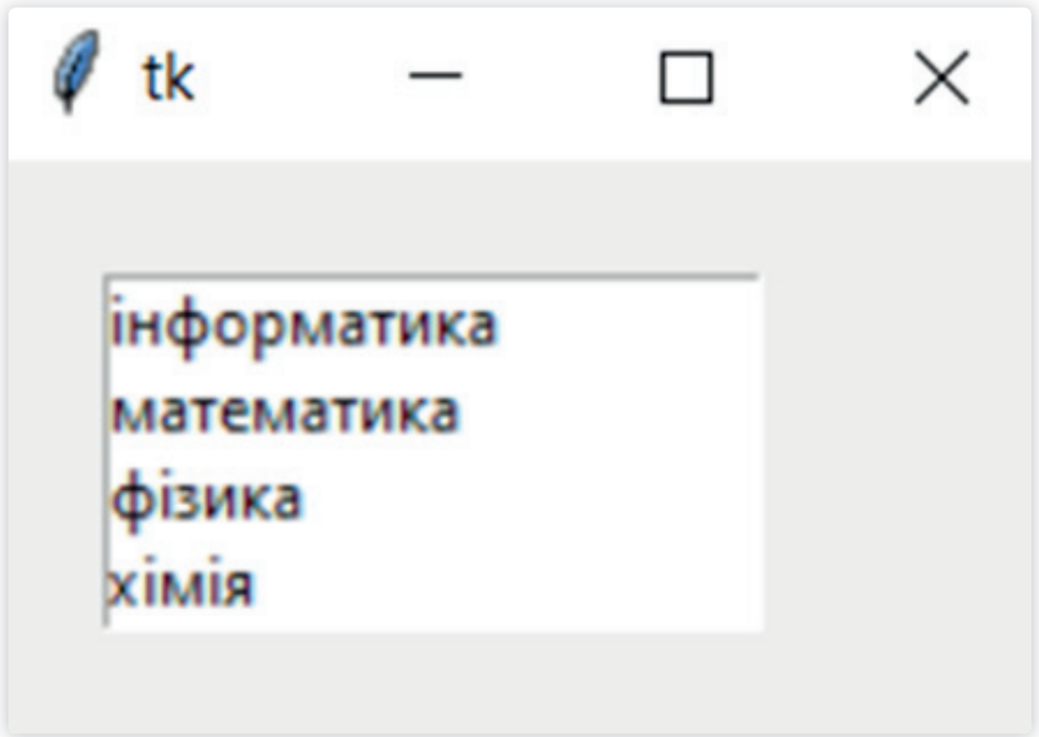
Текстове поле — єдиний елемент керування для роботи зі списками?

На минулому уроці ми навчилися вводити список за допомогою графічного інтерфейсу, однак виводили його все ще у вікні консолі. Проте є спеціальний елемент керування, що забезпечує виведення списків та вибір користувачем їх елементів. Його конструктор називається Listbox, і ви ознайомилися з ним ще у 8 класі. Власне, «Listbox» перекладається як «поле списку», і сьогодні ми навчимося виводити списки за допомогою цього елемента керування.

Завдання № 1

Користуючись рубрикою «Запитання-відповіді», заповни пропуски в коді створення поля списку, що зображене на малюнку.

```
from tkinter import *
window=Tk()
lbox=Listbox(height=4)
lbox.place(x=20, y=20)
lbox.insert(END, [    ])
lbox.insert(END, [    ])
lbox.insert(END, [    ])
lbox.insert(END, [    ])
```

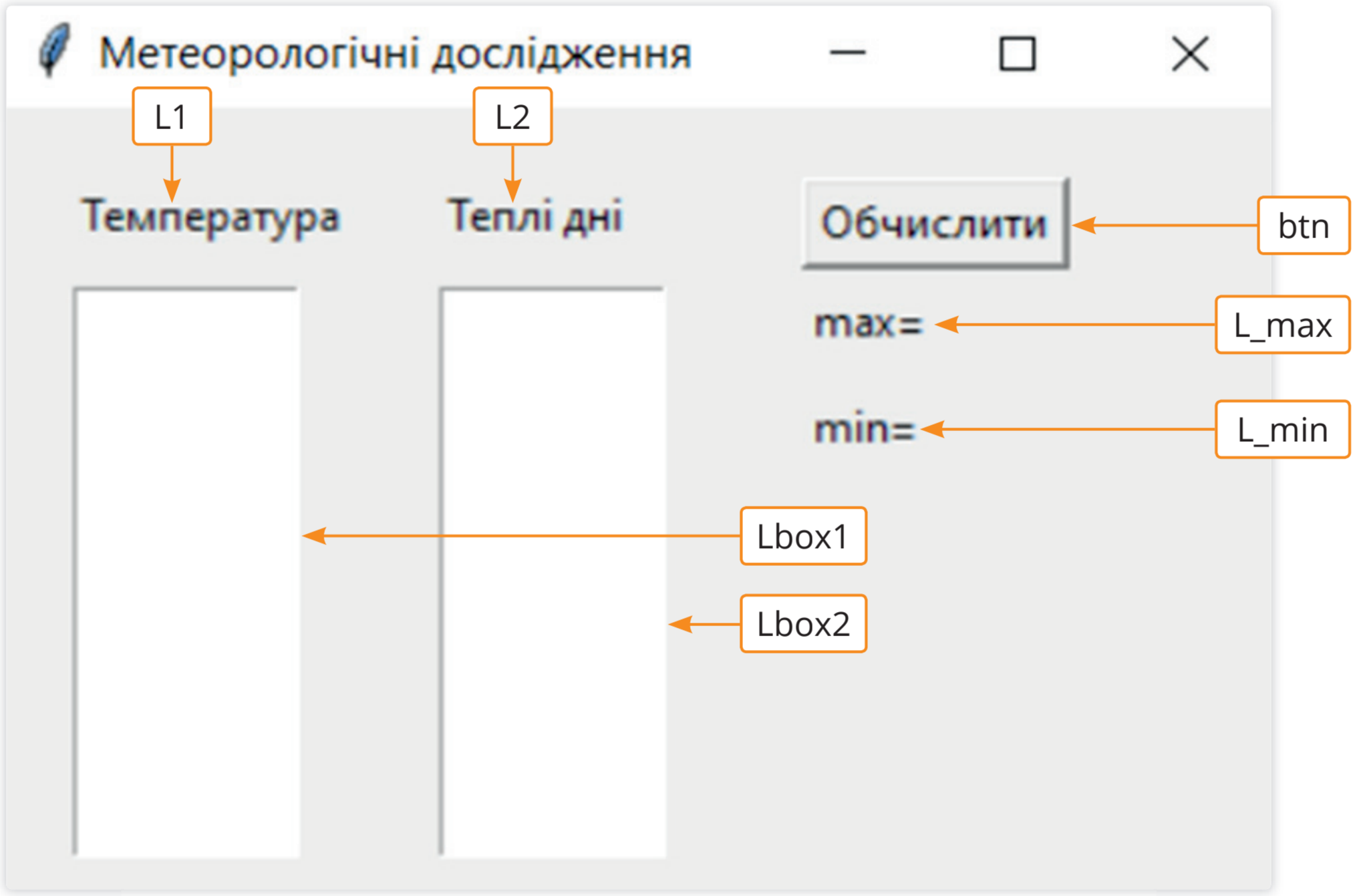


Завдання № 2

У наступних кількох завданнях створимо програму, у графічному вікні якої буде виводитися:

- список температур протягом семи днів тижня;
- максимальна та мінімальна температура;
- список усіх теплих днів, коли температура перевищувала 0 градусів.

Користуючись рубрикою «Запитання-відповіді», створи графічний інтерфейс програми «Метеорологічні дослідження». Задай розмір вікна програми 350x230 пікселів.



Завдання № 3

Щоб після кожного запуску програми не вводити значення температур вручну, заповнимо список температур випадковими числами у проміжку від –5 до 8. Для додавання елемента до списку використовують метод **append**. Користуючись рубрикою «Запитання-відповіді», дізнайся, як згенерувати випадкові числа на вказаному проміжку, та заповни пропуск у коді створення списку із семи випадкових температур:

створення порожнього списку

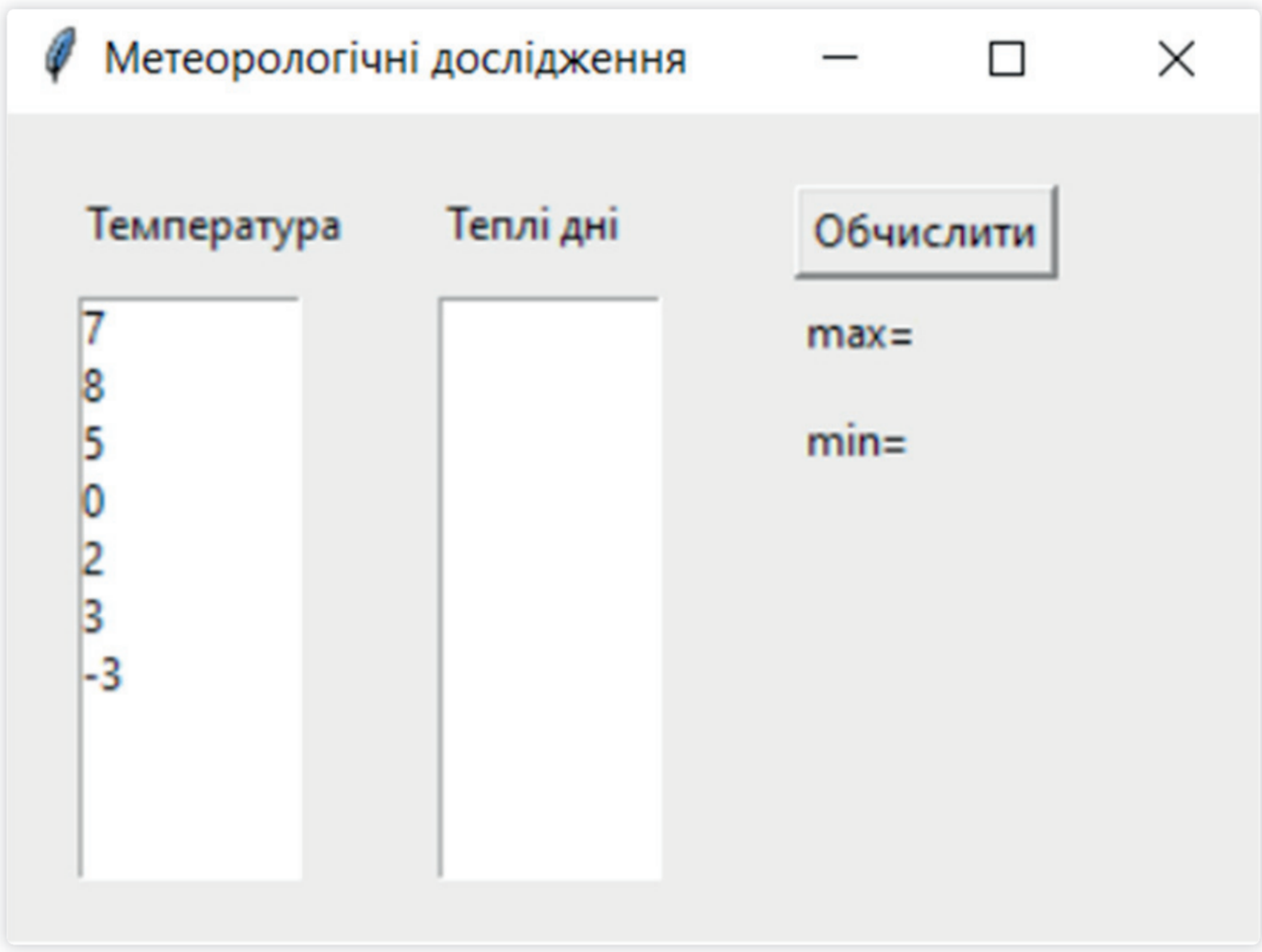
```
a=[]
for i in range(7):
    a.append(      )
```

додавання випадкового значення до списку

Завдання № 4

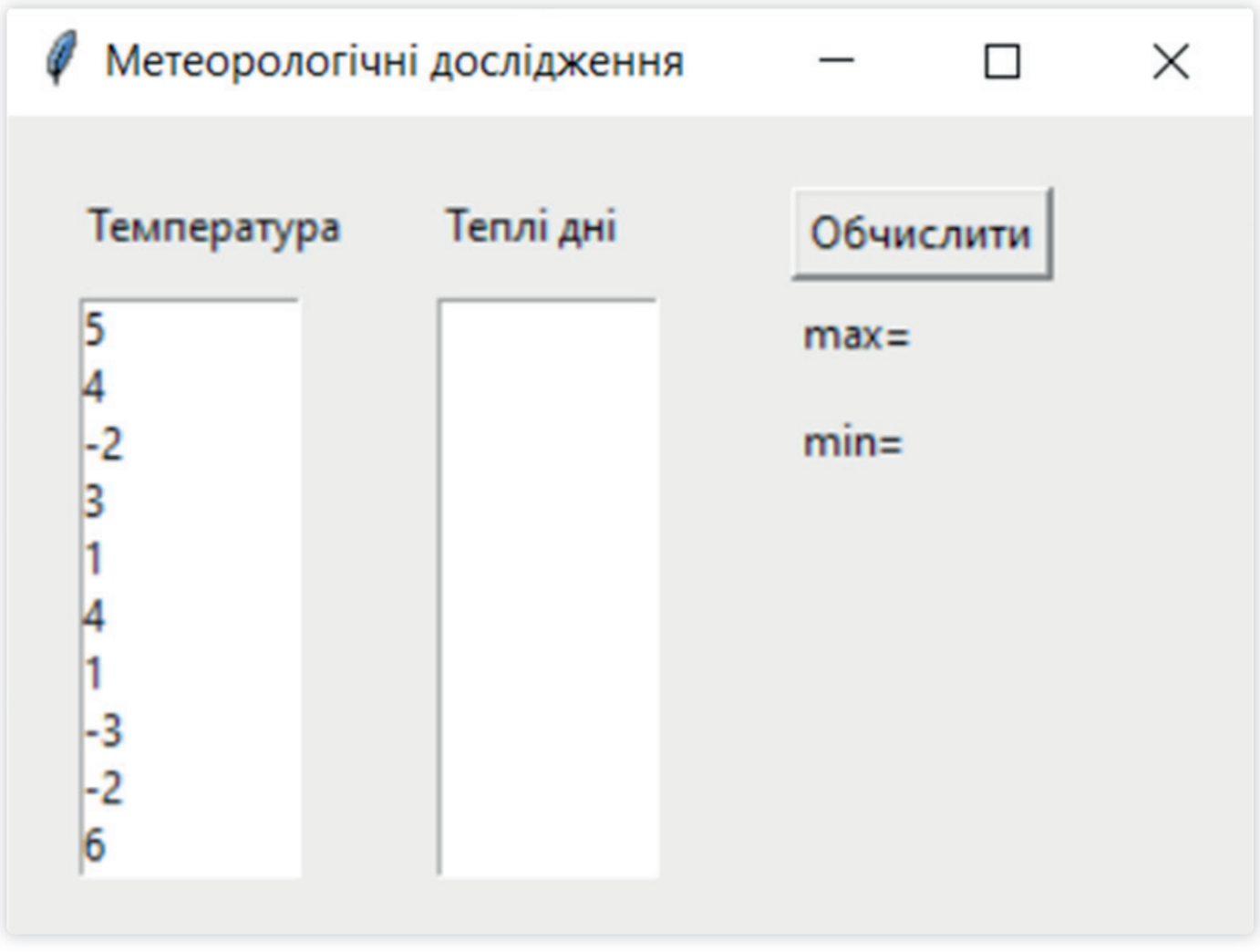
Створи обробник події натискання кнопки «Обчислити», після натискання якої має бути створено список **a** із семи випадкових значень температур та відображено його в елементі керування **Lbox1** за наведеним зразком. Користуючись рубрикою «Запитання-відповіді», додай до коду обробника події команду відображення списку в елементі керування **Lbox1**.

```
from random import *
def btn_click():
    a=[]
    for i in range(7):
        a.append(randint(-5,8))
        
btn=Button(text="Обчислити", command=btn_click)
btn.place(x="220", y="20")
```



Завдання № 5

У програмі, створеній у попередньому завданні, натисни кнопку двічі. Зауваж, що після повторного натискання кнопки новостворений список відображається під попередніми значеннями, хоча мав би замінювати їх.



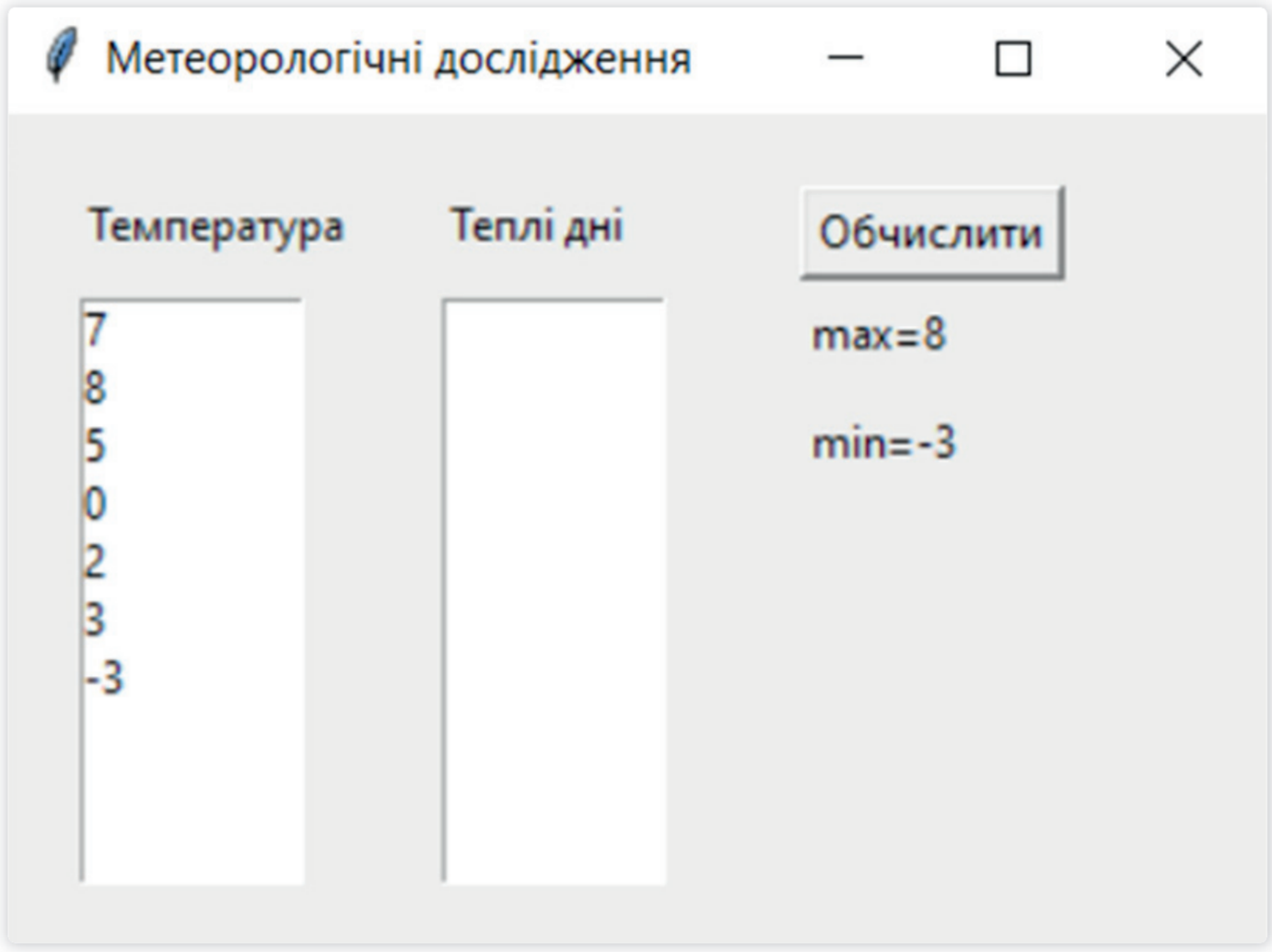
Щоб нові дані відображалися замість попередніх, попередні дані необхідно видалити. Тому щоразу після натискання кнопки поле **Lbox1** має очищуватися. Дізнайся з рубрики «Запитання-відповіді», як це зробити, подумай, куди саме потрібно вставити відповідну команду в коді. Свою відповідь перевір на практиці.

Завдання № 6

Тепер обчислимо мінімальне та максимальне значення температури та відобразимо його у відповідних написах. Нижче наведено код, що відображає максимальну температуру. Реалізуй його та створи за аналогією код, що відображає в написі мінімальну температуру.

```
L_max['text']="max="+str(max(a))
```

Додай код для визначення та виведення максимальної та мінімальної температури до обробника події натискання кнопки.

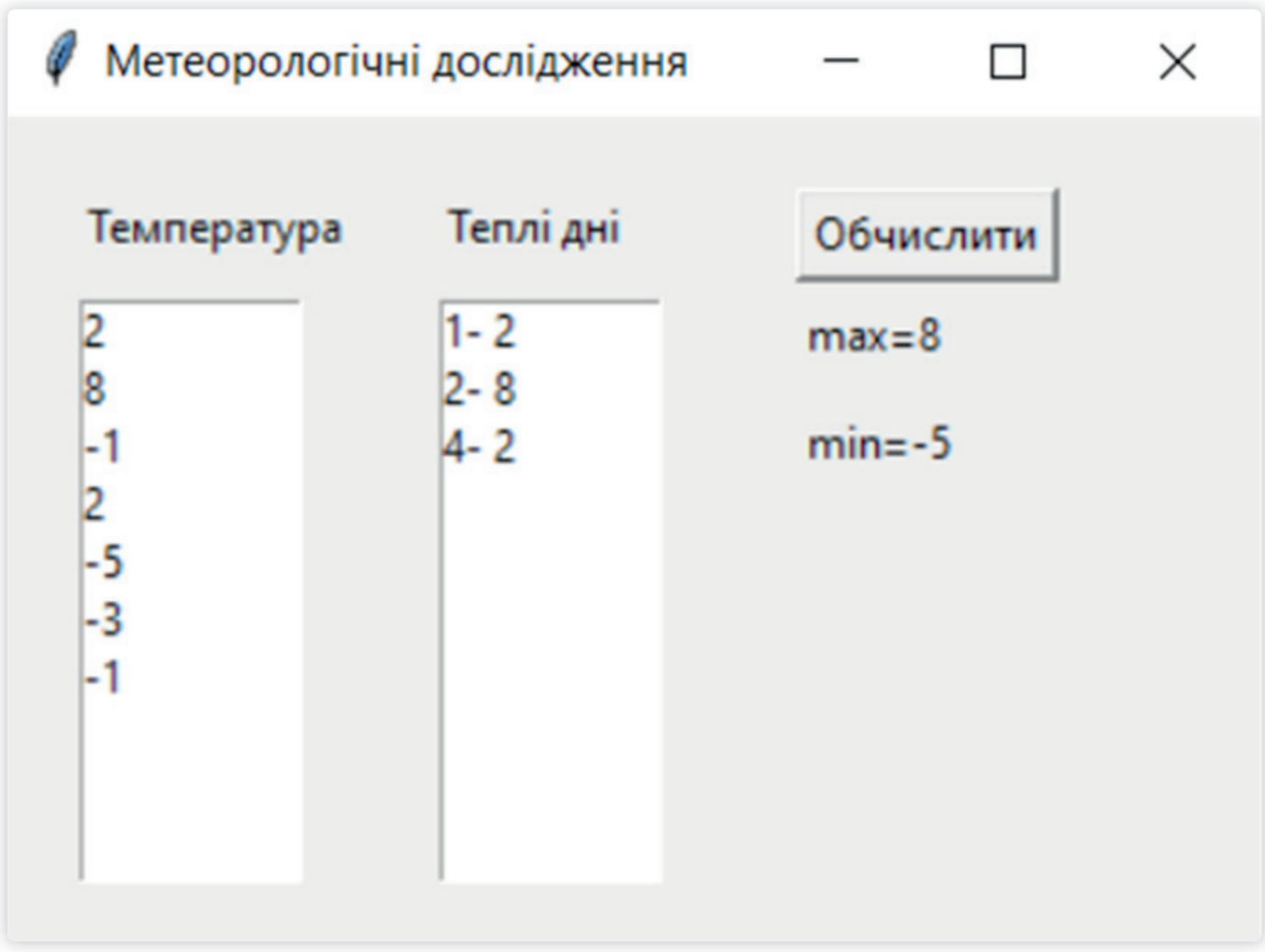


Завдання № 7

Виведемо в полі списку **Lbox2** номери днів, коли температура була більше 0 градусів, а також самі значення температури в ці дні. Заповни пропуск у коді, який доповнює цикл в обробнику події натискання кнопки, та реалізуй його в Python:

```
if a[i]>0:
    Lbox2.insert(END, )
```

Щоб дані з поля списку **Lbox2** після повторного натискання кнопки перезаписувалися, а не виводилися під попередніми, необхідно викликати метод **delete(0,END)** для поля **Lbox2**. Подумай, де саме в обробнику події натискання кнопки можна додати виклик цього методу.



ЗАПИТАННЯ-ВІДПОВІДІ

Як згенерувати ціле випадкове число в числовому проміжку від a до b?

Потрібно викликати функцію **randint(a,b)**, аргументами якої будуть межі числового проміжку **a, b** (випадкові числа генеруватимуться в проміжку від **a** до **b** включно). Ця функція належить модулю **random**. Тому на початку програми необхідно імпортувати модуль **random**. Наприклад,

```
from random import *
print(randint(50,100))
```

```
78
>>>
```

Як створити елемент керування «поле списку»?

Поле списку створюється за допомогою конструктора **Listbox**. Наприклад, **lbox=Listbox(height=0)**. Тут

- **lbox** — ім’я поля списку;
- **Listbox** — конструктор поля списку;
- **height** — кількість рядків у полі списку.

Як додати новий елемент до списку?

За допомогою методу **append()**, аргументом якого є новий елемент. Наприклад,

```
a=[1, 2, 3, 4]
a.append(7)
print(a)
```

```
====
[1, 2, 3, 4, 7]
>>>
```

Як відобразити список у полі списку?

Для виведення списку в елементі керування **Listbox** використовують такий код:

```
for i in range(len(a)):
    lbox.insert(END, a[i])
```

- **a** — ім'я списку;
- **lbox** — ім'я елемента керування, у якому треба вивести список;
- **insert** — метод додавання значення до списку;
- **END** — параметр, який вказує на те, що елементи додаються в кінець списку.

Як очистити поле списку?

За допомогою методу **delete(first, last)** можна видалити елементи поля списку з індексами від **first** до **last** включно. Якщо замість **first** вказати **0**, а замість **last** — **END**, то поле списку очиститься повністю. Наприклад,

Lb.delete(0,END) — команда очищення поля списку **Lb**.

ЗАПИТАННЯ НА ПОВТОРЕННЯ МАТЕРІАЛУ 8 КЛАСУ

Як створити текстовий напис?

Напис створюють за допомогою конструктора **Label**, а розміщують у вікні програми — за допомогою методу **place**.

```
napys=Label(text="Кількість днів")
napys.place(x="20", y="20")
```

Як змінити текст напису під час роботи програми?

Для зміни тексту в елементі керування використовують таку команду:

```
ім'я об'єкта['text']=значення властивості
```

Наприклад,

```
Lbl['text']="Hello!"
```

Як задати назву заголовка вікна програми?

За допомогою методу **title**. Наприклад,

```
W1.title("інформатика")
```

Тут **W1** — ім'я вікна програми.

Як вивести в одному рядку текст та числове значення?

Потрібно числове значення або назву змінної числового типу, або числове значення виразу записати як аргумент функції перетворення на рядковий тип даних **str** і за допомогою оператора **+** з'єднати відповідні рядки.

Наприклад,

```
a=[1, 2, 3, 4]
print('Сума='+str(sum(a)))
```

```
Сума=10
>>>
```

ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ОЦІНЮВАННЯ

1. Обери фрагмент коду, за допомогою якого в полі списку **Lbox1** можна відобразити список **a**.

А)
```
for i in range(len(a)):
    Lbox1.insert(END,a[i])
```

Б)
```
for i in range(len(a)):
Lbox1.insert(END,a[i])
```

В)
```
for i in range(a):
    Lbox1.insert(a[i])
```

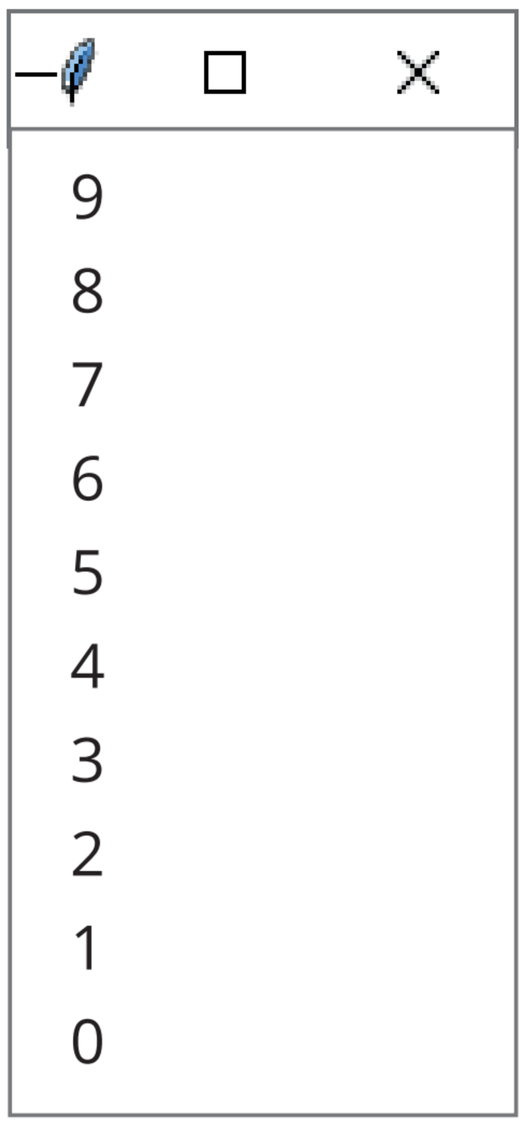
Г)
```
for i in range(len(a)):
Lbox1.insert(a[i],END)
```

2. Проаналізуй фрагмент коду, що виводить дані в полі списку **lb**, як зображено на малюнках. Заповни пропуск у цьому коді для кожного з випадків на малюнках а) — в).

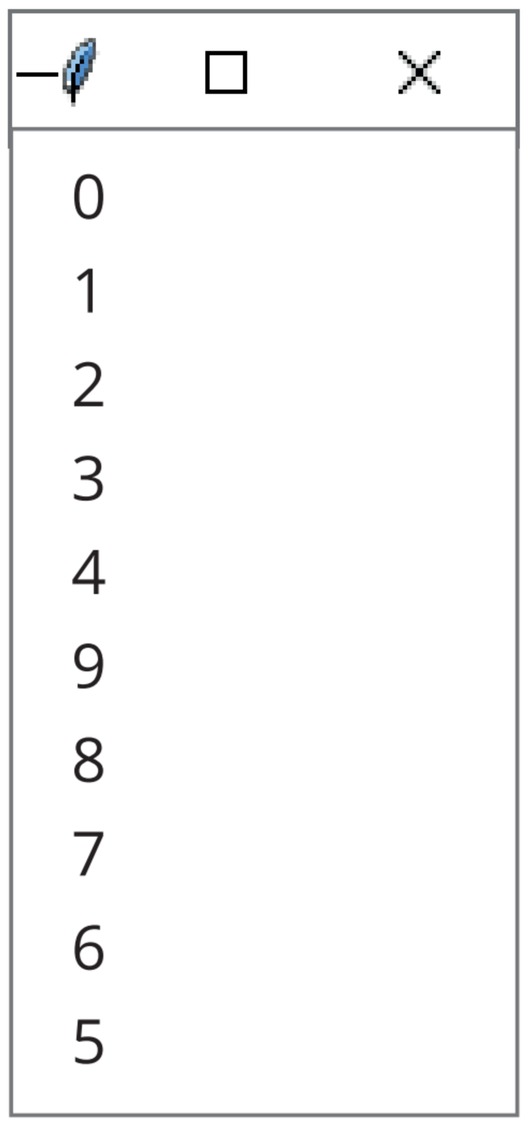
```
for i in range(18):
    lb.insert(____, i)
```

Нагадаємо, що метод **lb.insert(n,x)** вставляє у поле списку **lb** елемент **x** перед елементом з номером **n** або в кінець списку, якщо кількість елементів у списку менша за **n**. Відповіді перевір на практиці.

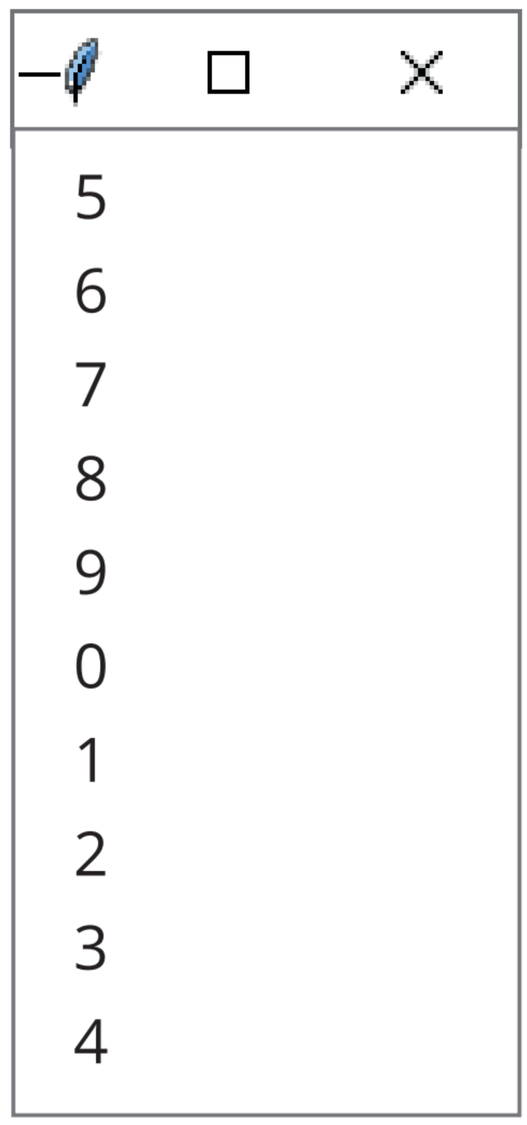
А)



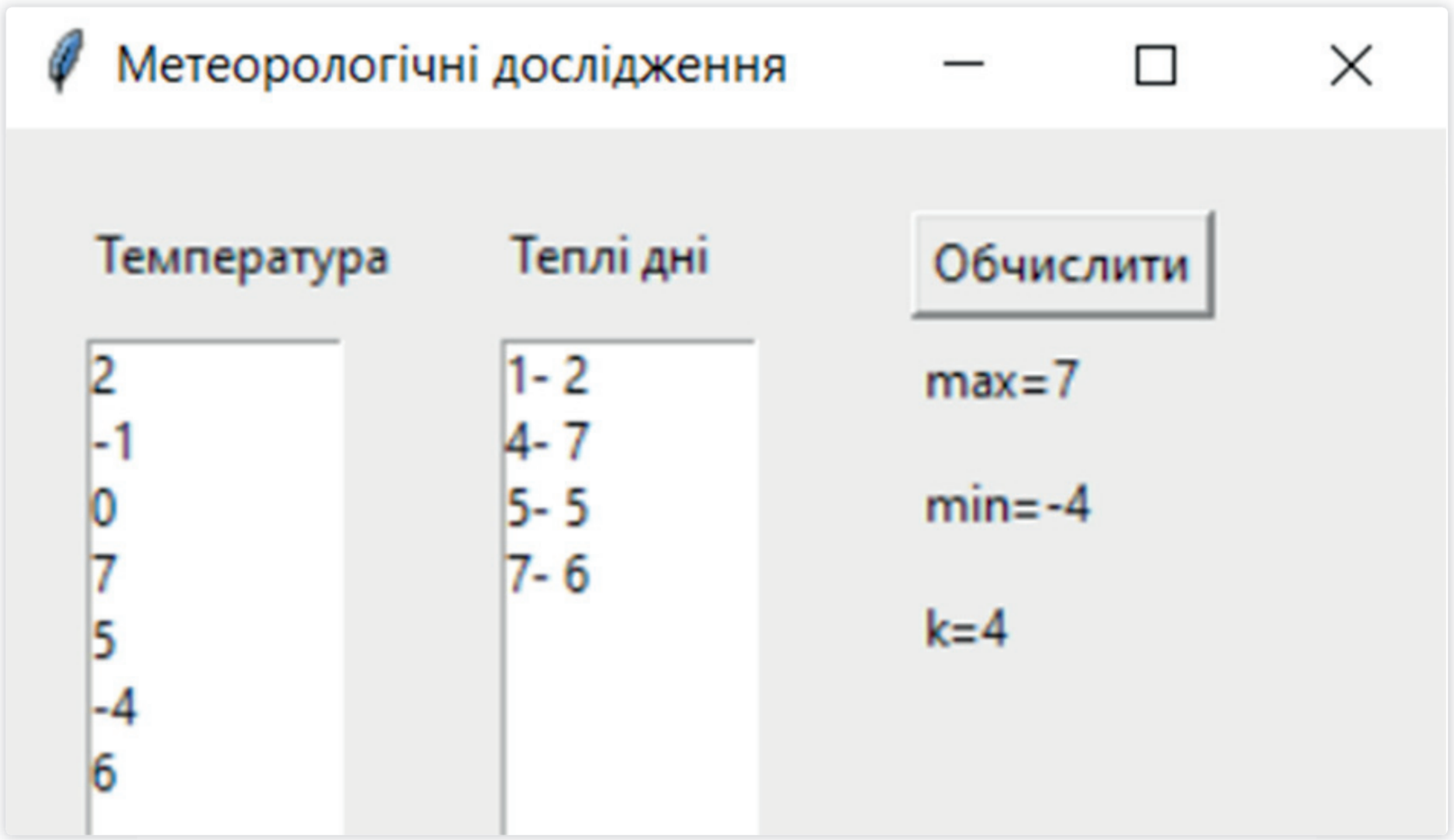
Б)



В)



3. Доповни створену на уроці програму так, щоб ще рахувалася кількість теплих днів **k** (коли температура була **більше 0 градусів**), та виведи це значення в написі **«k=»**.



4. Порівняй два способи виведення елементів списку: у текстовому полі та в полі списку. Наведи приклади задач, коли доцільно використовувати кожен спосіб.

§ 14. Операції з рядками. Шифр Цезаря

Як розшифрувати повідомлення, зашифроване шифром Цезаря, не знаючи ключа?

На минулих уроках ми виконували різноманітні дії зі списками, що містять числа. Сьогодні ти будеш працювати з текстовими даними. Перетворившись на справжнього секретного агента, ти створиш програму для шифрування та розшифрування текстових повідомлень.

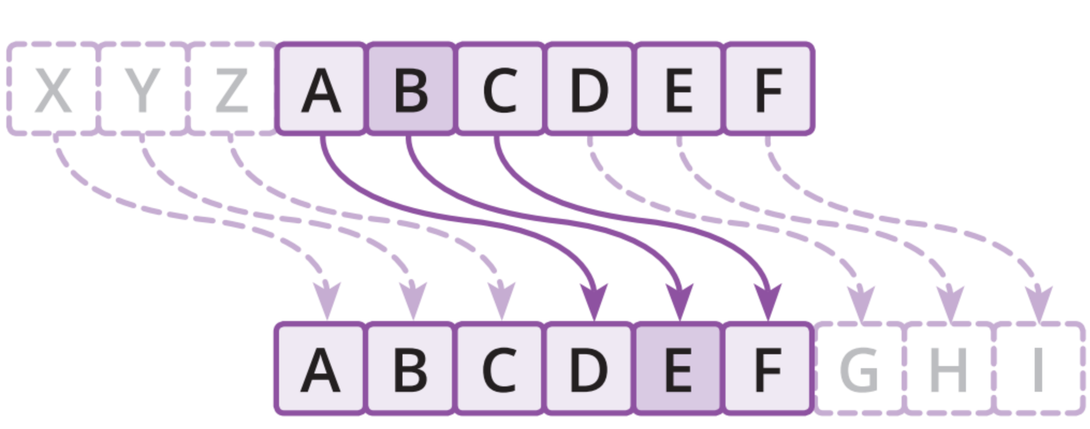
Зараз одним із найпростіших вважається шифр Цезаря, однак за часів Давнього Риму цей шифр не міг розшифрувати ніхто. Цим шифром користувався Юлій Цезар для приватного листування.

Зашифрувати повідомлення — це все одно, як замкнути його в скрині: прочитати його зможе лише той, хто має ключ.

i *В ІТ **ключем** називають невеликий обсяг інформації, за допомогою якого шифрують та/або розшифровують повідомлення.*

Завдання № 1

Принцип шифрування за допомогою шифру Цезаря: кожну літеру тексту замінити на ту, яка в алфавіті (або в кодовій таблиці символів) іде від неї на **k** літер далі. **k** — це ключ. Перехопивши повідомлення типу «тук шзфе хцфяїсвуїе», малограмотні вороги навіть не намагалися його прочитати, оскільки думали, що це повідомлення написане невідомою мовою.



1) Зашифруй шифром Цезаря слово ***шифрування***. Ключ шифру — 2.

літера	а	б	в	г	ґ	д	е	є	ж	з
шифр	в	г	ґ	д	е	є	ж	з	и	і
літера	и	і	ї	й	к	л	м	н	о	п
шифр	ї	й	к	л	м	н	о	п	р	с
літера	р	с	т	у	ф	х	ц	ч	ш	щ
шифр	т	у	ф	х	ц	ч	ш	щ	ь	ю
літера	ь	ю	я							
шифр	я	а	б							

← **на 2**

> ! *За часів Юлія Цезаря не існувало кодових таблиць символів, тому для шифрування / дешифрування повідомлень можна було послуговуватися лише латинським алфавітом. Але якщо до номерів останніх літер алфавіту додати ключ, то отримаємо числа, які перевищують кількість символів в алфавіті. Тому застосовувався принцип циклічності: вважалося, що літери алфавіту записані по колу й після Z знову йде A.*

2) Зашифруй слово ***цезар***, якщо ключ — 1.

літера	а	б	в	г	ґ	д	е	є	ж	з
код	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081
літера	и	і	ї	й	к	л	м	н	о	п
код	1082	1083	1084	1085	1086	1087	1088	1089	1090	1091
літера	р	с	т	у	ф	х	ц	ч	ш	щ
код	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101
літера	ь	ю	я							
код	1102	1103	1104							

Завдання № 2

Розшифруй слово ***ргцнг***, якщо ключ — 3.

А	Б	В	Г	Ґ	Д	Е
Є	Ж	З	И	І	Ї	Й
К	Л	М	Н	О	П	Р
С	Т	У	Ф	Х	Ц	Ч
	Ш	Щ	Ь	Ю	Я	

Завдання № 3

Ознайомся в рубриці «Запитання-відповіді» з принципом зберігання текстових даних і визнач, що виведе така програма:

```
a="йцуикенгшщзфівапролджєї"
print(a[8]+a[3]+a[11]+a[16])
```

Завдання № 4

Дізнайся з рубрики «Запитання-відповіді», для чого призначено функцію **chr()**, і спробуй визначити, що виведе такий код, не запускаючи його:

```
print(chr(1082)+chr(1086)+chr(1076))
```

Завдання № 5

Дізнайся з рубрики «Запитання-відповіді», для чого призначено функцію **ord()**, і спробуй визначити, що виведе такий код, не запускаючи його:

```
print(ord('м')+ord('и')+ord('р'))
```

Завдання № 6

Дізнайся з рубрики «Запитання-відповіді» про перебір символів рядка. Доповни код програми, у якій користувач вводить із клавіатури ключ і рядок **s**, а програма його шифрує і виводить закодований рядок **sz**.

```
key=int(input('Введіть ключ'))
s=input('Введіть рядок')
sz=''
for ____ in ____:
    kod=____(i)+key
    sz=sz+chr(____)
print(____)
```

Завдання № 7

Реалізуй задачу із завдання № 6 у програмному середовищі. Перевір результат на прикладі:

```
Введіть ключ 6
Введіть рядок програмування
хцфйцжтщижуys
```

Завдання № 8

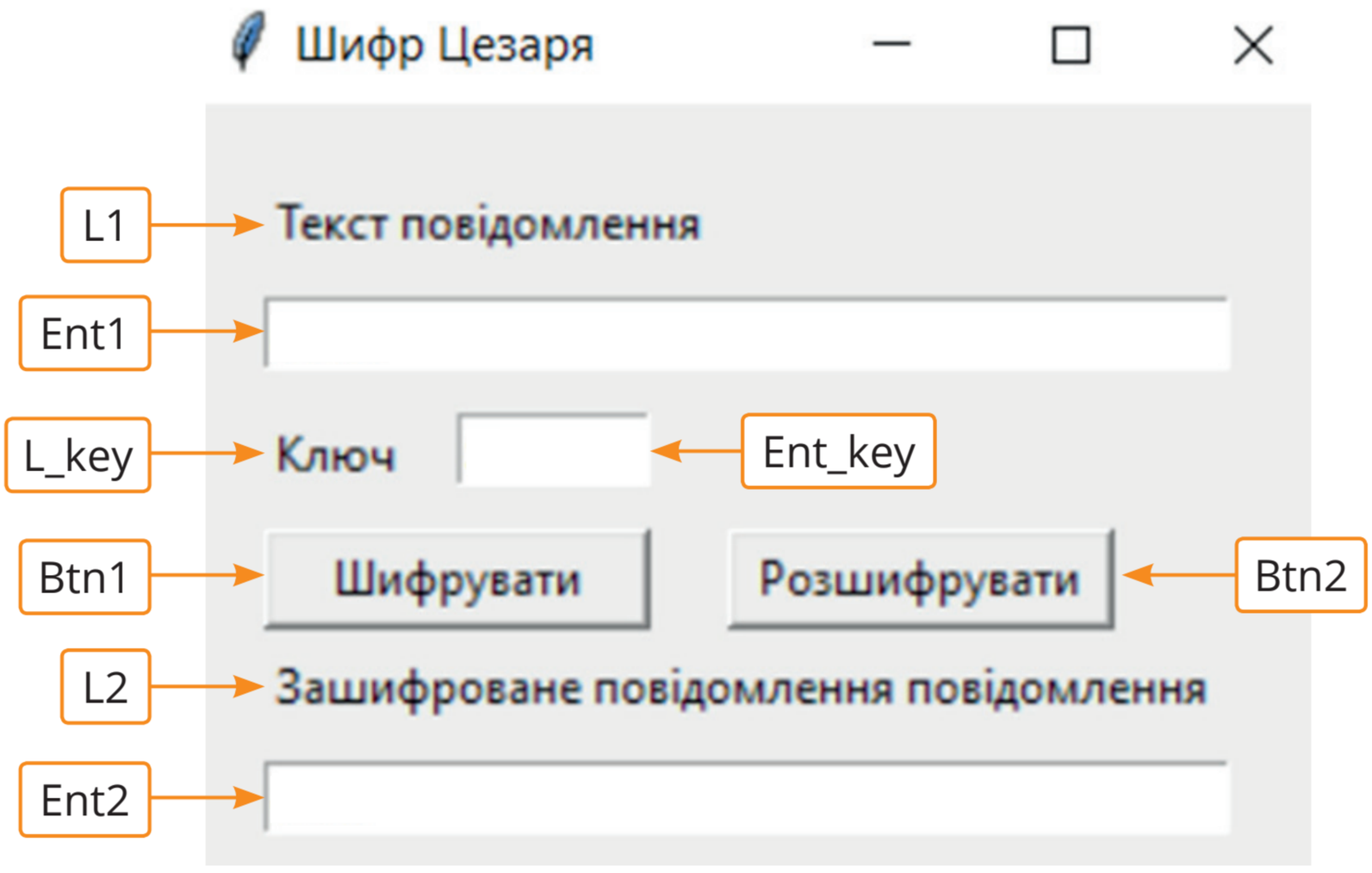
На основі попередньої програми створи ще одну, яка виконуватиме протилежні дії: вона має розшифровувати закодований текст, що, як і ключ, вводиться з клавіатури.

Перевір результат на прикладі:

```
Введіть ключ 5
Введіть закодований рядок пуйшзеттє
кодування
```

Завдання № 9

Створи відповідно до зразка графічний інтерфейс для програми шифрування / розшифрування. Надай елементам керування імена, що вказані у виносках.



Завдання № 10

Доповни код, у якому текстовий рядок і ключ вводяться не з клавіатури, а зчитуються з текстових полів **Ent_key** (у ньому введено ключ **key**) та **Ent1** (у ньому введено рядок **s**).

```
key=int(Ent_key [ ]())
s=Ent1 [ ] ()
sz=' '
```

Можеш скористатися рубрикою «Запитання-відповіді» з розділу «Введення елементів списку».

Завдання № 11

Дізнайся з рубрики «Запитання-відповіді», як відобразити текстовий рядок в елементі керування «Текстове поле», та заповни пропуск **y**, що відображає рядок **s2** в текстовому полі **Ent1**.

```
Ent1.____(END, s2)
```

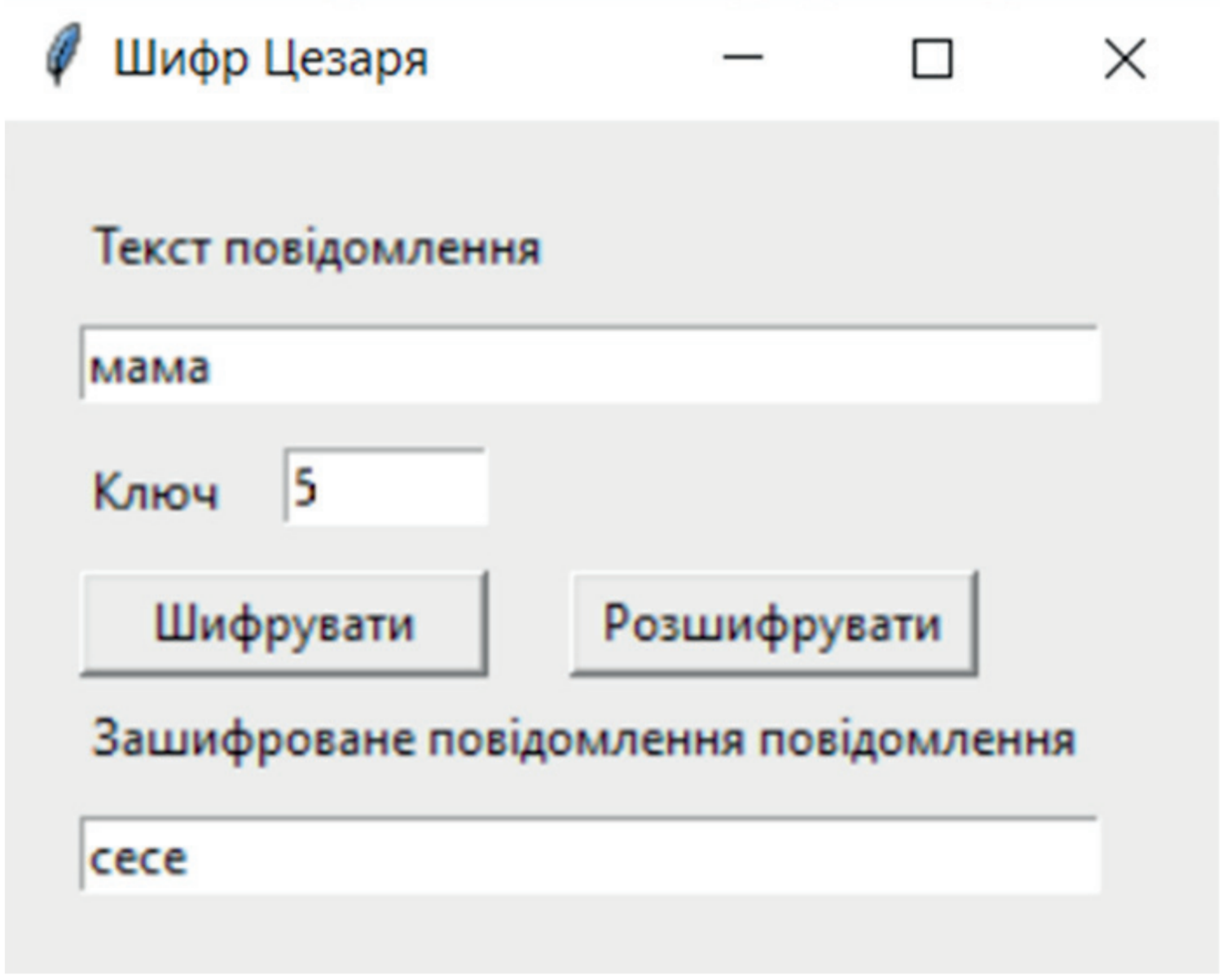
Завдання № 12

Користуючись кодом з попереднього завдання та із завдань № 7 і № 8, створи функції—обробники подій натискання кнопок «Шифрувати» та «Розшифрувати», які створені в інтерфейсі програми в завданні № 9.

Щоб дані в текстовому полі перезаписувалися, а не дописувалися до попередніх значень, потрібно додати до обробника події команду очищення текстового поля під час кожного натискання кнопок:

```
Ent2.delete(0,END)
```

Перевір результат виконання програми:



ЗАПИТАННЯ-ВІДПОВІДІ

Чи можна отримати доступ до окремого символу в рядку?

Текстовий рядок — це список символів. Щоб отримати один символ рядка, необхідно вказати ім'я змінної, що містить рядок, а в квадратних дужках — індекс (порядковий номер) символу.

```
>>> a="моя програма"
>>> s=a[4]
>>> print(s)
п
```

Символи рядка нумеруються з нуля. Ось так, наприклад, у пам'яті комп'ютера буде записано рядок, який містить змінна **a**:

```
a="моя програма"
```

i	0	1	2	3	4	5	6	7	8	9	10	11
a	м	о	я		п	р	о	г	р	а	м	а

Тут **i** — це порядковий номер символу рядка **a**.

Як отримати доступ до певного фрагмента (підрядка) заданого рядка?

Отримати певний фрагмент рядка, який ще називають зрізом або підрядком, можна за допомогою такого оператора:

```
S=text[I:J:K]
```

- **S** — змінна для зберігання підрядка;
- **text** — початковий рядок;
- **I** — індекс символу, з якого починається підрядок;
- **J** — індекс символу закінчення підрядка, що вже не включається до зрізу;
- **K** — крок, з яким вибираються символи.

Стандартне значення для параметра **K** — 1, тобто якщо третій параметр не вказано, то символи вибираються послідовно. Якщо параметр **I** не задано, то зріз формується від 0-го до **J-1**-го символу включно. Якщо параметр **J** не задано, то зріз формується від **I**-го символу до кінця рядка.

Наприклад,

```
a='Інформатика'
print(a[:3])
print(a[4:])
print(a[3:5])
```

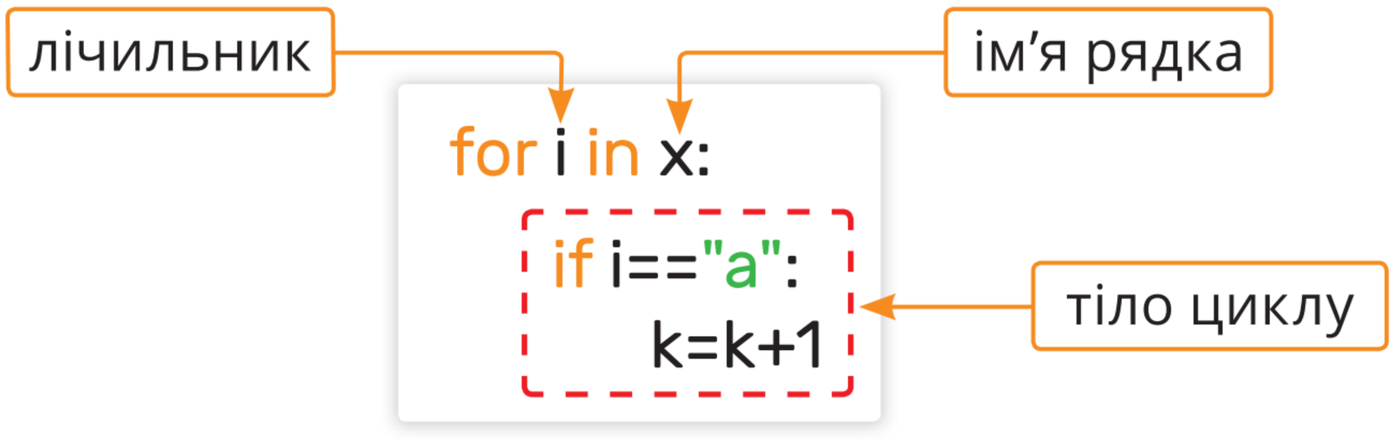
```
======
Інф
матика
ор
>>>
```

Як послідовно перебрати всі символи рядка?

Для перебору всіх символів рядка використовують цикл, що має такий вигляд:

```
for лічильник in ім'я рядка:
    тіло циклу
```

Із символами, що перебираються, можна виконувати різноманітні дії. Наприклад, такий вигляд матиме цикл, який визначає, скільки разів у рядку **x** трапляється літера "a":



Увага!

У наведеному циклі для доступу до символу рядка використовується лічильник **i**, *а не вираз* **s[i]**.

Як ви вже знаєте з попередніх тем, кожен символ має числовий код. Коди ставляться у відповідність символам за допомогою кодової таблиці.

У кириличних текстах Python-програм використовується таблиця Unicode. Ось фрагмент кодової таблиці Unicode, що містить символи українського алфавіту.

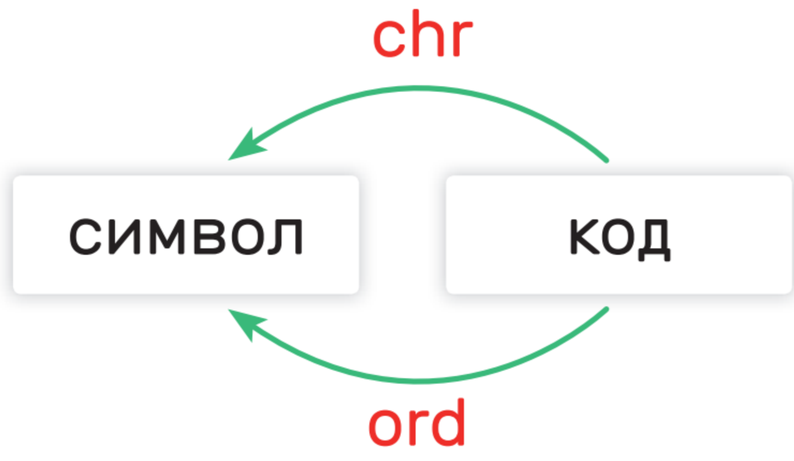
літера	а	б	в	г	д	е	ж	з	и	й
код	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081
літера	к	л	м	н	о	п	р	с	т	у
код	1082	1083	1084	1085	1086	1087	1088	1089	1090	1091
літера	ф	х	ц	ч	ш	щ	ъ	ы	ь	э
код	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101
літера	ю	я	è	ë	ђ	ѓ	є	ѕ	і	ї
код	1102	1103	1104	1105	1106	1107	1108	1109	1110	1111
літера	ј	љ	њ	ћ	ќ	ѝ	ў	џ	Ѡ	ѡ
код	1112	1113	1114	1115	1116	1117	1118	1119	1120	1121
літера	Ѣ	ѣ	Ѥ	ѥ	Ѧ	ѧ	Ѩ	ѩ	Ѫ	ѫ
код	1122	1123	1124	1125	1126	1127	1128	1129	1130	1131
літера	Ѭ	ѭ	Ѯ	ѯ	Ѱ	ѱ	Ѳ	ѳ	Ѵ	ѵ
код	1132	1133	1134	1135	1136	1137	1138	1139	1140	1141

Для визначення символу Unicode за його кодом **y** використовують функцію **chr(y)**, а для визначення коду символу **x** — функцію **ord(x)**.

Наприклад,

```
print(ord('в'))
print(chr(1074))
```

```
======
1074
в
>>>
```



Як відобразити рядок у текстовому полі?

Метод **ent.insert(K,string)** вставляє текст **string** у позицію **K** у текстовому полі **ent**.

Якщо як параметр **K** вказати константу **END**, то рядок **string** буде дописано до тексту, що вже є в полі, справа.

Наприклад, якщо поле **ent** було порожнім, то в результаті виконання коду **ent.insert(END,s)** у ньому відобразиться текстовий рядок **s**.

ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Укажи значення індексів у команді формування рядка b, який у результаті одержав значення "паста".

```
s="програміст"
b=s[ ]+s[ ]+s[ ]+s[ ]+s[ ]
```

2. Дізнайся з рубрики «Запитання-відповіді» про оператор формування фрагмента (зрізу) рядка. Дано рядки:

```
a='структура'
b='організований'
c='алоритмізація'
d='перерва'
```

Створи програму, що формує та виводить слово "реструктуризація", використовуючи лише зрізи заданих рядків.

3. Проаналізуй наведену програму-ребус і визнач, яке слово вона виведе на екран, не створюючи коду в Python.

```
s1='боб'
s2='чиж'
s3=s3='слоненя'
print(s1[1:]+s2[:2]+s3[:2]+s3[4:6]+s3[5:])
```

Перевір свою відповідь, ввівши програму в Python.

Створи іншу програму-ребус самостійно і запропонуй розгадати його однокласнику/однокласниці.

4. Як ми вже зазначали, шифр Цезаря дуже легко «зламати», тобто розшифрувати, не знаючи ключа. Для цього достатньо перебрати всі можливі значення ключа, яких може бути не більше, ніж символів в алфавіті.

   Створи програму, у якій користувач вводитиме зашифрований текст, а в циклі перебиратимуться значення ключа в діапазоні від 1 до 30 включно. Програма має виводити ці значення ключа, а також розшифроване повідомлення для кожного зі значень. Правильний ключ — той, для якого відображене повідомлення містить осмислений текст.

   Попроси когось із однокласників / однокласниць надіслати тобі зашифрований текст у месенджері, а ключ не повідомляти. Розшифруй цей текст.

   Придумай історію про програму-шифрувальник для «сімейного секретного чату». Поміркуй, як інформаційні технології допомагають долати відстані та підтримувати зв'язок у різних типах сучасних родин.

   Презентуй свою історію та створену програму.

ЦЕ ЦІКАВО

Наука про шифрування повідомлень називається *криптографією*, а про розшифрування повідомлень без інформації про ключ — *криптоаналізом*. У щойно створеній програмі було виконано криптоаналіз шифру Цезаря. Як і в цій програмі, під час криптоаналізу завжди вважають, що алгоритм шифрування і зашифрований текст відомі, а ключ — ні.

5. Чому, на твою думку, за часів античності ніхто не міг зламати шифр Цезаря?

§ 15. Упорядкування елементів списку

У яких життєвих завданнях потрібно сортувати набори об’єктів?

На минулих уроках ми виконували різні дії зі списками. Досить часто в програмуванні працювати зі списками ефективніше, коли вони відсортовані, тобто їхні елементи впорядковані за зростанням чи спаданням. Для сортування списків у більшості мов програмування, зокрема в Python, є стандартні (бібліотечні) методи, з якими ми ознайомимося на цьому уроці. Використовувати такі методи просто, однак не дуже цікаво. Значно цікавіше розібратися «зсередини», за якими алгоритмами ці методи можуть працювати! І цим ми теж займемося сьогодні, розглянувши один із найвідоміших і найпростіших алгоритмів сортування списків.

Завдання № 1

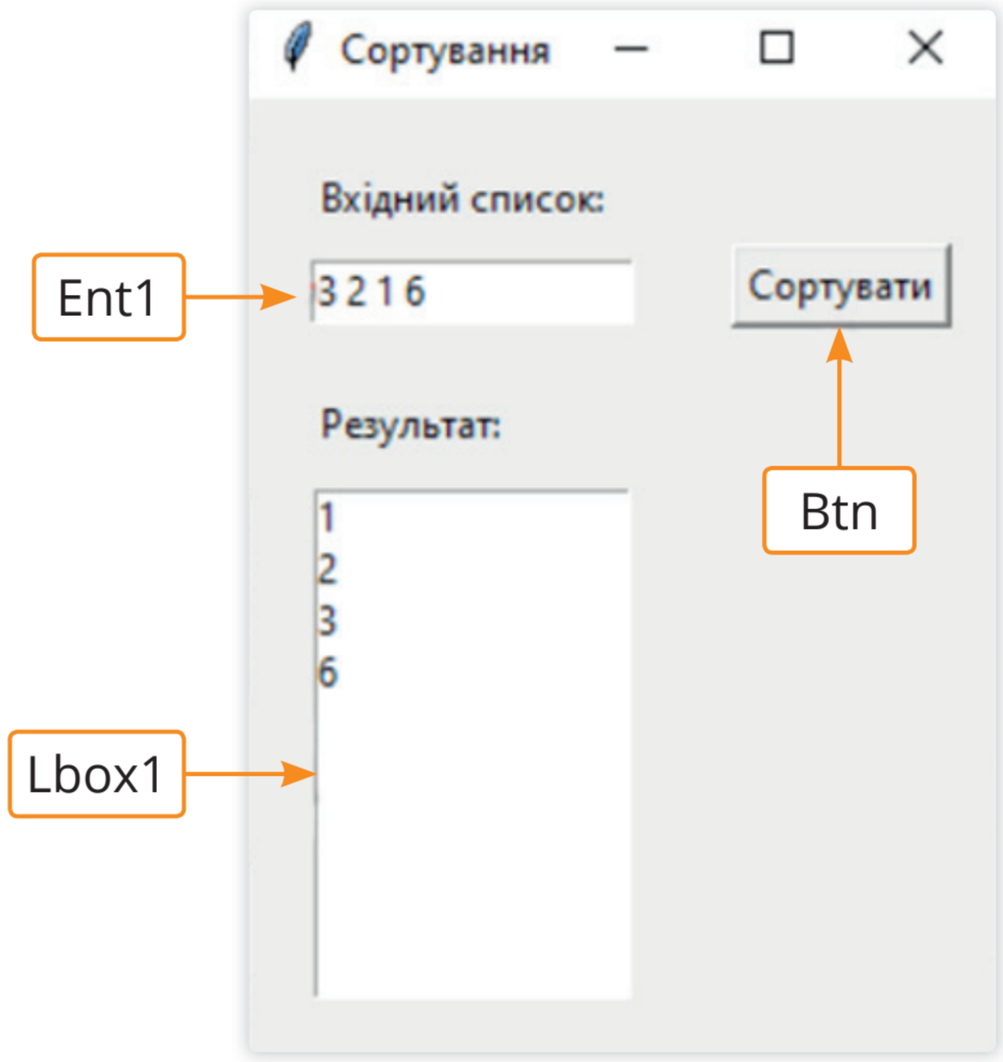
Дізнайся з рубрики «Запитання-відповіді» про метод **sort()**.
Що буде виведено в результаті виконання команди **print(a.sort())**, якщо спочатку цей список був таким: 4, –3, 12, 0, 5?
Як будуть розташовані елементи списку **a** після виконання команди **a.sort(reverse=True)**, якщо спочатку цей список був таким: 4, –3, 12, 0, 5?

Завдання № 2

Створи за зразком інтерфейс для програми сортування списку.

У програмі мають бути:

- текстове поле **Ent1**;
- кнопка **Btn** з написом «Сортувати»;
- поле списку **Lbox1**;
- два написи — «Вхідний список:» і «Результат:».



Завдання № 3

Доповни код, за допомогою якого можна створити список дійсних чисел **a** з розташованих у текстовому полі **Ent** даних, записаних через пробіл.

```
a=[      for i in Ent.      .      ]
```

Завдання № 4

Програма має працювати так: користувач вводить у текстове поле список чисел через пробіл, натискає кнопку — і цей список має відобразитися відсортованим у порядку зростання в полі списку.

Заповни пропуск у коді обробника події натискання кнопки «Сортувати» та реалізуй його в програмному середовищі. Не забудь у конструкторі кнопки «Сортувати» присвоїти параметру **command** назву методу **btn_click**.

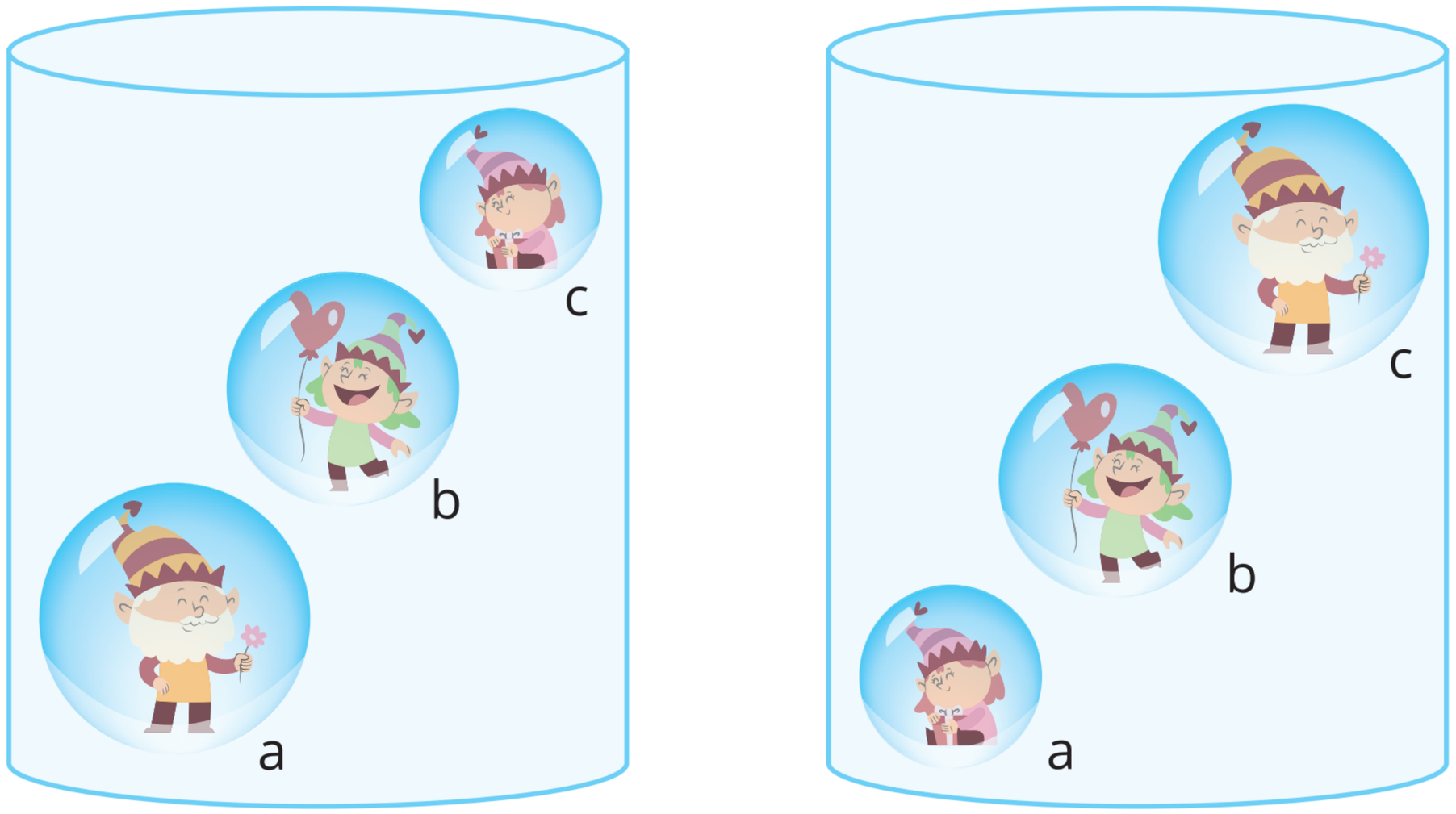
```
def btn_click():
    Lbox1.delet(0,END)
    a=Ent1.get().split()
    a=[float(i) for i in a]
    #Сортування списку a в порядку зростання
    for i in range (len(a)):
        Lbox1.insert(END,a[i])
```

Завдання № 5

Ми ознайомилися зі стандартним методом **sort()**, а тепер розберемся, як він працює «зсередини», який алгоритм у ньому може бути реалізовано. Існують десятки алгоритмів сортування, більш чи менш швидких. Одним із найпростіших є метод «бульбашок».

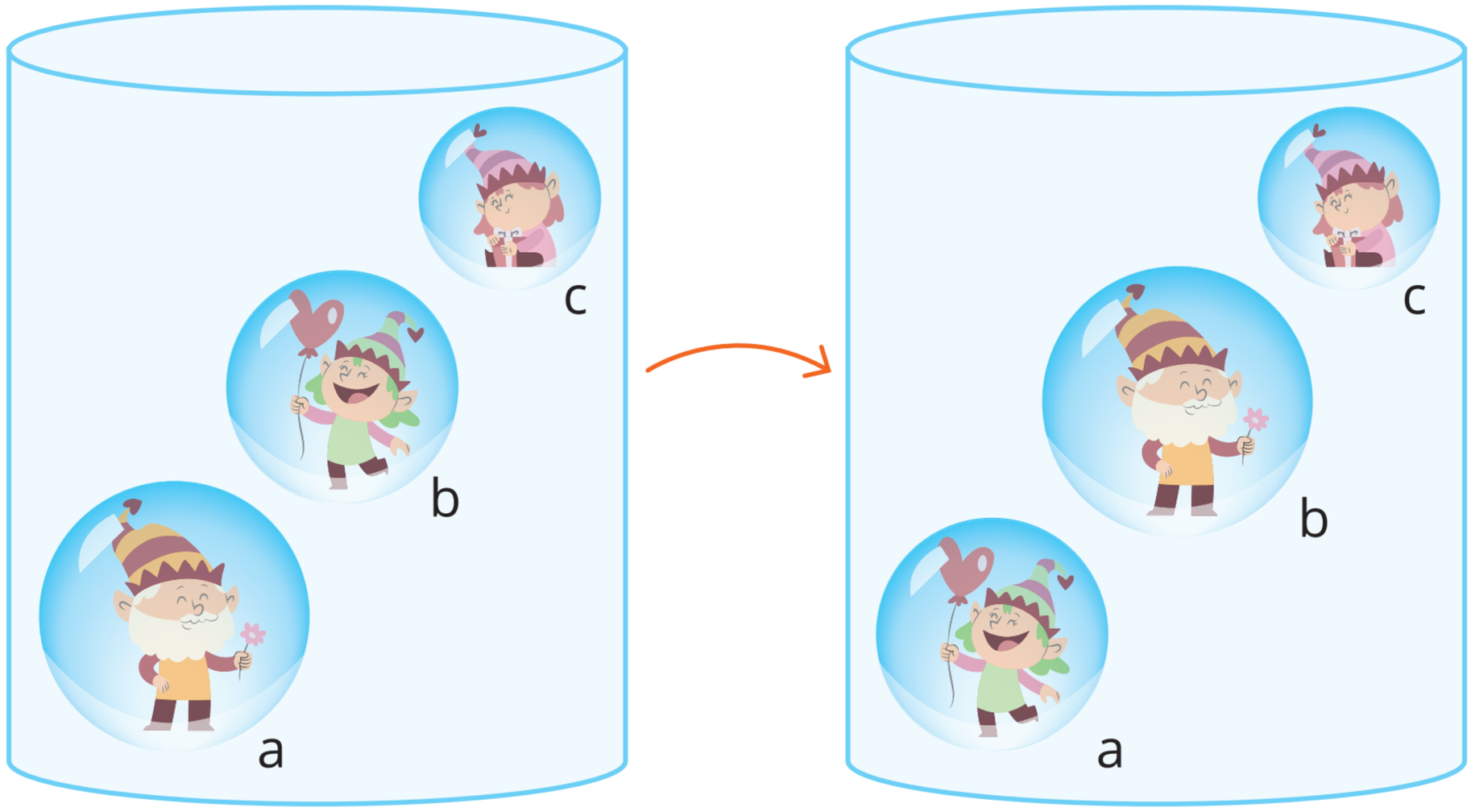
Принцип цього методу полягає в попарному порівнянні елементів та обміні їх місцями, доки більші елементи не перестануть «спливати» до кінця списку, а дрібніші залишатимуться «знизу».

Нехай у нас є три позиції (змінні) **a, b, c**. Нам потрібно розмістити на цих позиціях три бульбашки в порядку зростання їхнього розміру. Припустимо, що спочатку вони розташовувалися в протилежному порядку:



було — має стати

Крок 1 Порівняємо дві перші бульбашки, **a** і **b**. Нам потрібно, щоб менша бульбашка була лівіше. Оскільки **a>b**, міняємо бульбашки місцями.

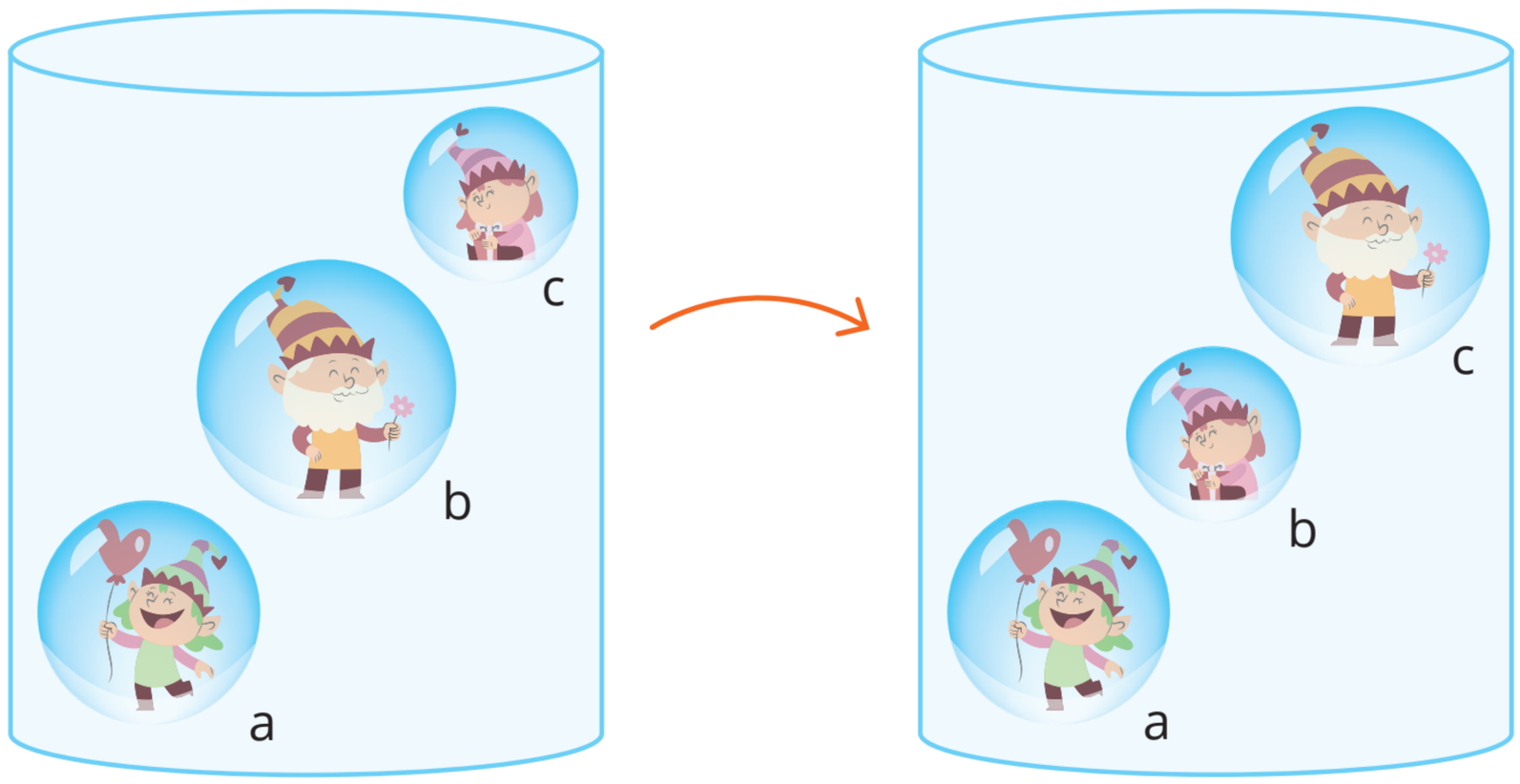


Головний принцип: менша бульбашка має бути лівіше.

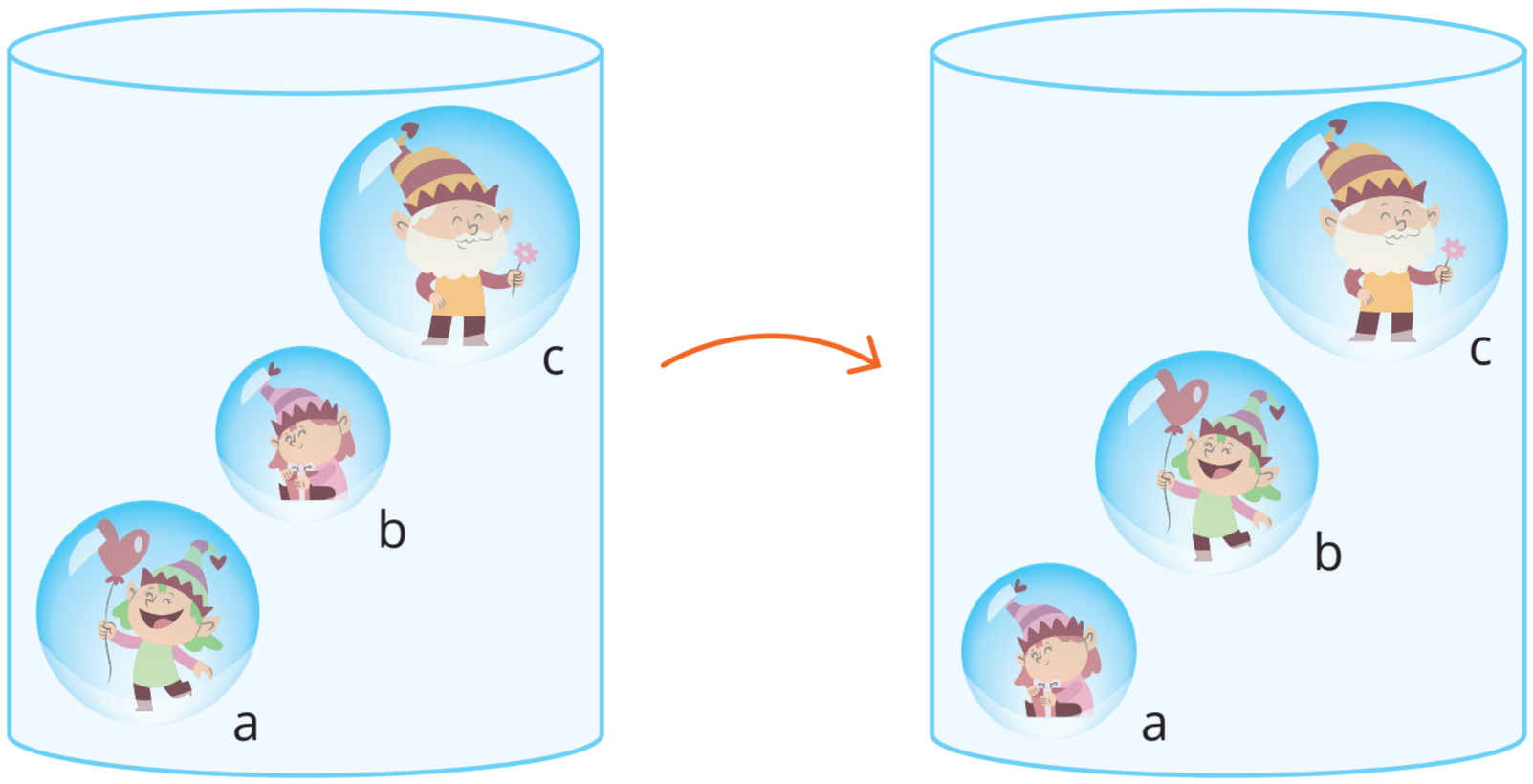
У «бульбашковому» методі спочатку порівнюються 1-й і 2-й елементи, потім 2-й і 3-й, потім — 3-й і 4-й тощо. У результаті найбільший елемент опиниться справа.

Доповни словесний алгоритм наступних кроків.

Крок 2 Порівнюємо бульбашки __ і __. Якщо _____, то_____.



Крок 3 Повторюємо алгоритм спочатку і порівнюємо бульбашки __ і __. Якщо _____, то _____.



Ми отримали бажаний результат!

Завдання № 6

Ознайомся із застосуванням «бульбашкового» методу до списку з десяти елементів, переглянувши відео.

Проаналізуй відео (е-додаток (15.1)) та опиши процес «бульбашкового» сортування за зростанням.

Увага!

Коли найбільша бульбашка спливла справа, повторюємо весь процес, але цю найбільшу бульбашку вже не чіпаємо.

Під час перегляду відео став його на паузу і відповідай на запитання письмово в зошиті:

- Яким був початковий список?
- Які елементи порівнюються? (Занотуй кожне порівняння.)
- Доки повторюється процес попарного порівняння елементів?
- Що трапляється, коли всі пари елементів порівняно?

Завдання № 7

Заповни пропуски в тексті, що описує алгоритм сортування списку за зростанням методом «бульбашок».

1. Порівнюється черговий елемент списку з _____.
2. Якщо _____ елемент _____, то _____.
3. Після порівняння всіх пар _____ елемент переміщується до _____ списку.
4. Процес повторюється від початку списку до _____ _____ елемента включно.

Завдання № 8

Виконай покроково алгоритм сортування списку **A=[9,5,4,8,3]** методом «бульбашок» і занотуй послідовність змін вмісту списку за прикладом:

Індекси елементів списку	0	1	2	3	4
Заданий список А	6	4	3	7	1
Результат 1-ї ітерації	4	3	6	1	7
Результат 2-ї ітерації	3	4	1	6	7
Результат 3-ї ітерації	3	1	4	6	7
Результат 4-ї ітерації	1	3	4	6	7

Індекси елементів списку	0	1	2	3	4
Заданий список А	9	5	4	8	3
Результат 1-ї ітерації					
Результат 2-ї ітерації					
Результат 3-ї ітерації					
Результат 4-ї ітерації					

Завдання № 9

Користуючись рубрикою «Запитання-відповіді», вибери команду, за допомогою якої перший та останній елементи списку **[2,6,7,4,3]** поміняються місцями:

a[0],a[4]=a[4],a[0]

a[2],a[3]=a[3],a[2]

a[1],a[5]=a[5],a[1]

a[0],a[5]=a[5],a[0]

Завдання № 10

1) Заповни пропуски в коді, що сортує список **a** методом «бульбашок». **i**-й елемент списку має порівнюватися з **i+1**-м, і якщо **i**-й елемент більший, то елементи мінятимуться місцями.

2) Можна зауважити, що в коді лічильники **j** (зовнішнього) та **i** (внутрішнього) циклів змінюються в діапазонах, що залежать від довжини заданого списку. У цьому випадку список містить п’ять елементів. Які зміни в коді необхідно зробити для того, щоб програма діяла для списків із будь-якою кількістю елементів?

3) Чому в команді **range** у внутрішньому циклі записано формулу **5-j**? Поясни словами, що вона означає.

```
for j in range(1,5):
    for i in range(5-j):
    #порівняння елементів
    if [ ]:
       #обмін елементів
```

Завдання № 11

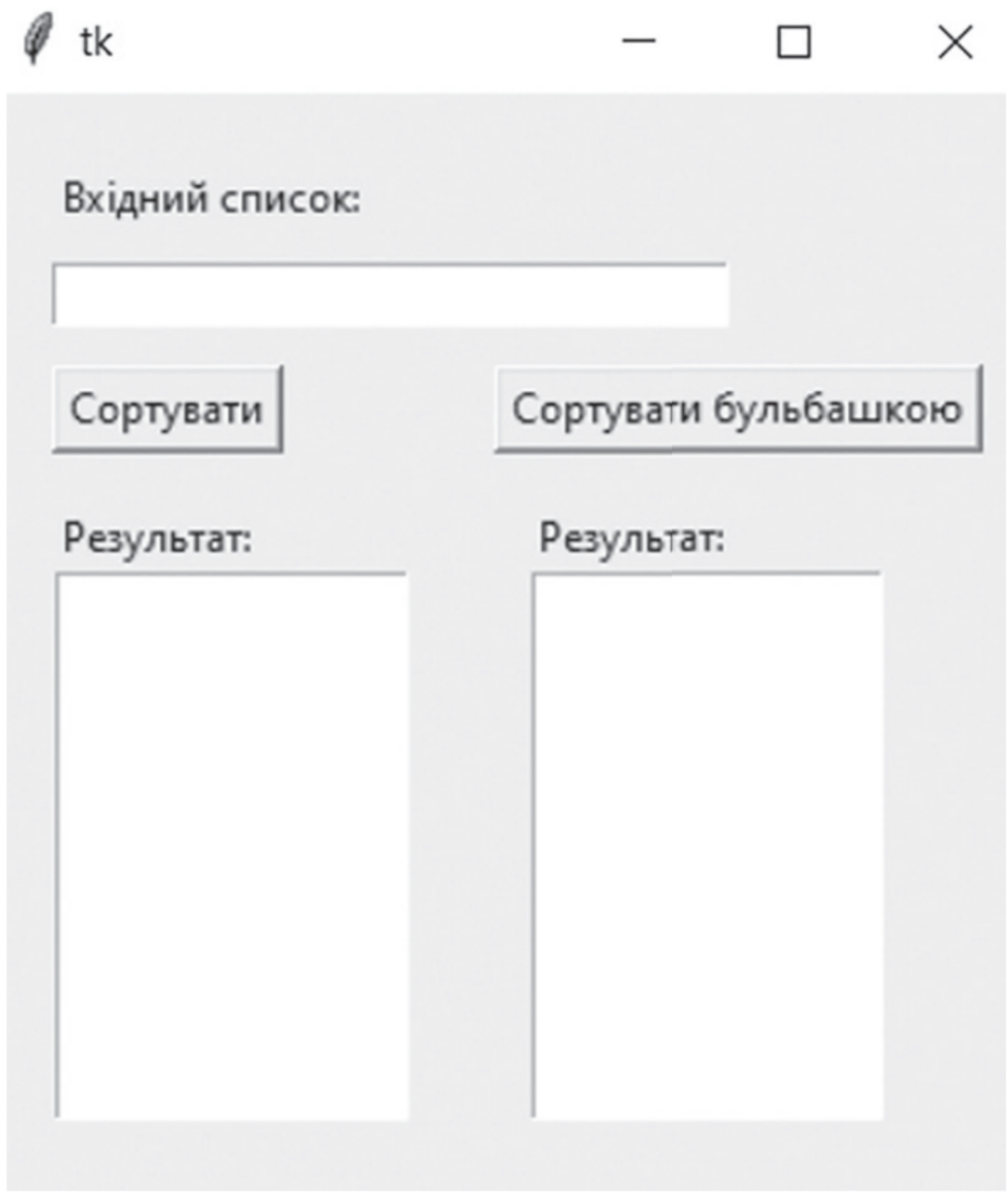
Реалізуй сортування списку методом «бульбашок» у Python, визначивши вміст списку за допомогою оператора присвоювання.

Виводь список командою **print** після кожної ітерації зовнішнього циклу, який відповідає за один прохід по всьому списку.

Завдання № 12

1) Додай до інтерфейсу програми із завдання № 4 ще одну кнопку для реалізації сортування за зростанням методом «бульбашок» і ще одне поле списку для виведення результату й порівняння.

2) Скопіюй код обробника події натискання кнопки «Сортувати» та заміни його кодом із завдання №10. Модифікуй цей код так, щоб він сортував методом «бульбашок» стільки елементів, скільки буде у списку. Забезпеч виконання цього методу після натискання кнопки «Сортувати бульбашкою». Порівняй сортування двома способами на кількох прикладах — результати мають збігатися.



ЗАПИТАННЯ-ВІДПОВІДІ

Що таке сортування списку?

Сортування списку — це розміщення його елементів у порядку їх зростання (від меншого до більшого) чи спадання (від більшого до меншого).

Даний список	Список впорядковано **за зростанням**	Список впорядковано **за спаданням**
3	1	**7**
1	2	**5**
7	3	**3**
5	5	**2**
2	7	1

Сортування та *впорядкування* — це одне те саме.

Для чого призначений метод sort()?

Метод **sort** застосовується до списку (тобто його назву записують через крапку після імені списку). Якщо цей метод викликано без параметрів, він упорядковує елементи списку в порядку зростання. Для сортування списку в порядку спадання використовують параметр **reverse** (англ. «зворотний») зі значенням **True**. Наприклад, у результаті виконання фрагмента програми

```
a=[5,7,2,3,8]
a.sort(reverse=True)
print(a)
```

буде одержано результат

```
[8,7,5,3,2]
```

Як поміняти місцями значення змінних a і b?

Значення змінних **a** і **b** можна поміняти місцями, використовуючи третю змінну, наприклад **c**:

```
c=a
a=b
b=c
```

Цей спосіб ви мали вивчати в 7 класі. Але в Python існує значно простіший спосіб обміну значеннями змінних: **a, b = b, a.**
Обидва способи є рівносильними, тобто завжди дають однаковий результат.

У чому полягає «бульбашковий» метод сортування списку?

Метод передбачає кілька проходів списком. А саме: кількість проходів дорівнює довжині списку, зменшеній на 1. На кожному проході послідовно порівнюють усі пари сусідніх елементів у ще не відсортованій частині списку. Якщо список сортують за зростанням і лівіший елемент більший за правіший, то ці елементи міняють місцями.

У результаті першого проходу більший елемент «спливе» в кінці списку справа. На другому проході його вже не треба чіпати, а на передостанній справа позиції «спливе» другий за величиною елемент списку. Отже, відсортована частина списку складатиметься з двох елементів і буде розташована з його правого краю. З кожним проходом ця відсортована частина збільшуватиметься на один елемент, аж поки не охопить увесь список.

Проаналізуємо метод «бульбашок» на прикладі сортування списку **a=[6,5,7,3,1]** за зростанням. Червоним овалом позначено порівняння сусідніх елементів, і якщо лівіший є більшим, то їх обмін місцями.

1 прохід

6 5 7 3 1
5 6 7 3 1
5 6 7 3 1
5 6 3 7 1
5 6 3 1 7

2 прохід

5 6 3 1 7
5 6 3 1 7
5 3 6 1 7
5 3 1 6 7

3 прохід

5 3 1 6 7
3 5 1 6 7
3 1 5 6 7

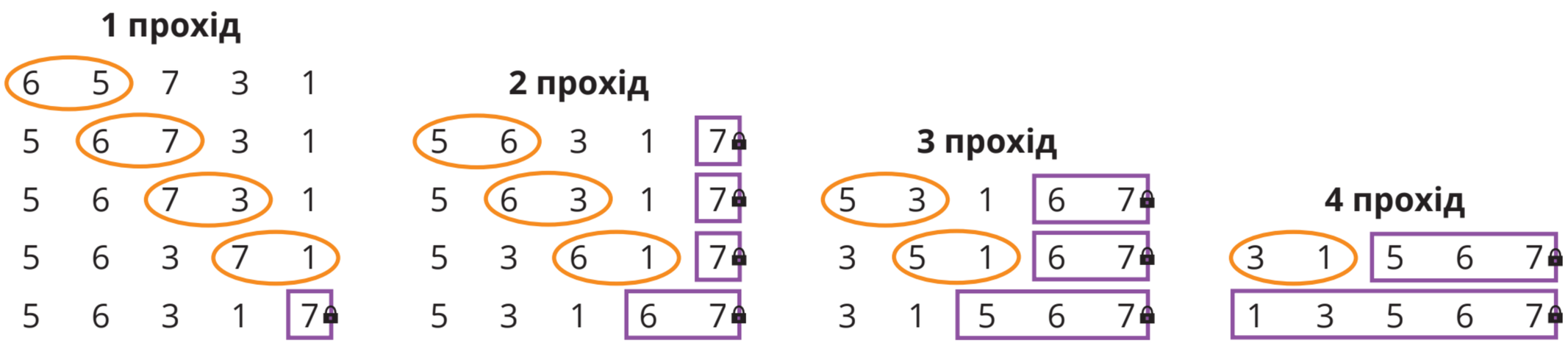
4 прохід

3 1 5 6 7
1 3 5 6 7

Розглянутий алгоритм можна реалізувати за допомогою вкладених циклів:

- у зовнішньому циклі рахуватиметься кількість проходів;
- у внутрішньому циклі порівнюватимуться сусідні елементи.

Призначення лічильників зовнішнього та внутрішнього циклу буде таким: номер проходу — зовнішній цикл; індекс елемента, що порівнюється з наступним, — внутрішній цикл.



Діапазон зміни лічильника **j** зовнішнього циклу — від **1** до **len(a)-1** включно, а внутрішнього циклу — від **0** до **len(a)-j**.

ПЕРЕВІР СЕБЕ

РОЗУМІННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Скільки порівнянь елементів буде виконано під час сортування списку **[4,1,2,3]** за зростанням «бульбашковим» методом? А скільки обмінів елементів місцями?

2. Є певний список **x**. Проаналізуй наведений нижче фрагмент програми. Укажи, яким буде індекс елемента, що матиме значення **y** після виконання зображеного коду.

```
for j in range(len(x)):
    for i in range(len(x)-j-1):
        if x[i]>x[i+1]:
            x[i], x[i+1]=x[i+1], x[i]
y=max(x)
```

3. Скільки порівнянь елементів потрібно зробити під час сортування методом «бульбашок» *n*-елементного списку?

4. Доповни програму, створену в завданні № 12, ще двома полями списку. У першому має відображатися введений користувачем список, відсортований за спаданням бібліотечним методом, а в другому — за спаданням методом «бульбашок».

5. Метод сортування списку «бульбашками» є одним із найпростіших. Як ти думаєш, чи є він найшвидшим? Чи можеш запропонувати інший метод, який працюватиме швидше?

6. Знайди в інтернеті інформацію про різноманітні алгоритми сортування і порівняй їх із запропонованим тобою методом.

РОЗДІЛ IV

СТВОРЕННЯ ТА ПУБЛІКАЦІЯ ВЕБРЕСУРСІВ

§ 16. Структура вебсторінки. Мова HTML

§ 17. Як змінити вигляд вебсторінки? Основи CSS

§ 18. Вебсторінка як єдина система

§ 19. Автоматизовані засоби для створення та публікації вебресурсів

§ 16. Структура вебсторінки. Мова HTML

Чи є HTML мовою програмування?

Щодня ви відкриваєте десятки вебсторінок: читаєте новини, дивитеся відео, шукаєте інформацію, спілкуєтесь у соцмережах.

Але чи замислювалися ви, що насправді отримує браузер, коли ви вводите адресу сайту? Чи бачить він одразу красиву сторінку з кольорами, зображеннями й кнопками? Чи, можливо, спочатку отримує щось зовсім інше?

Уявіть, що замість звичайної сторінки ви відкрили її «зсередини» — і побачили набір незрозумілих слів у кутових дужках, дивні конструкції, службові записи.

То що це таке? І як із цього набору символів виникає звична для нас вебсторінка? Спробуйте з'ясувати це самостійно.

Завдання № 1

1. Розглянь приклад вебсторінки та її вихідного коду.

День програміста

День програміста відзначають у **256-й** день року.

Число 256 тісно пов'язане з комп'ютерами. Спробуй пояснити чому.

Дізнатися більше про це свято

© 2026

code

```
<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <title>День програміста</title>
    <style>
        body {
            background-color: #f0f6ff;
            font-family: Arial;
        }
        h1 {
            color: #1e3a8a;
        }
        .highlight {
            color: #d97706;
            font-weight: bold;
        }
        a {
            color: #2563eb;
        }
    </style>
</head>
<body>
    <h1>День програміста</h1>
    <p>День програміста відзначають у <span class="highlight">256-й</span>
      день року.</p>
    <img src="code.png" alt="Ілюстрація">
    <p><a href="https://uk.wikipedia.org/wiki/День програміста">
    Дізнатися більше</a></p>
</body>
</html>
```

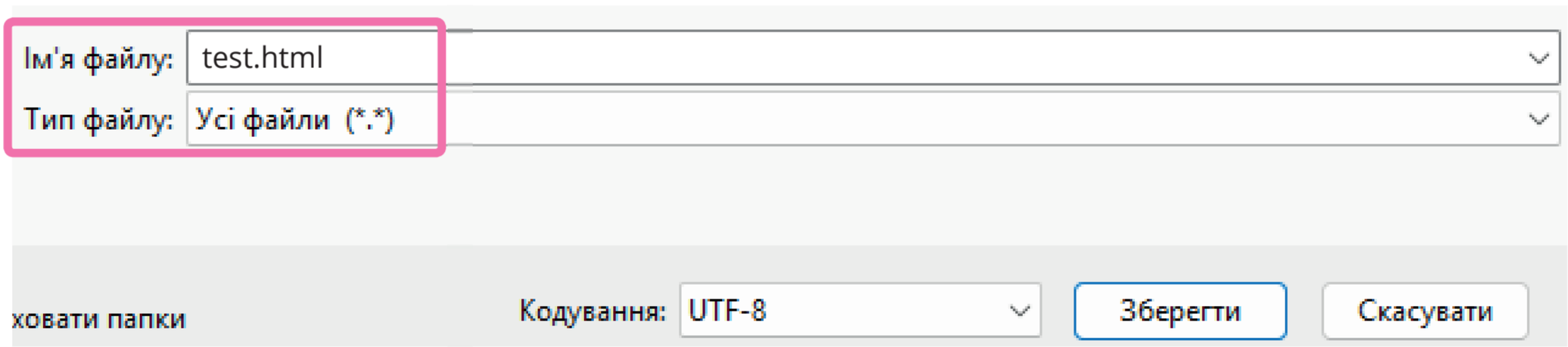
2. Порівняй те, що бачить користувач, із тим, що «бачить» браузер.
3. Знайди в коді записи, взяті у кутові дужки. Спробуй пояснити, чи відображаються вони на сторінці. Поясни, для чого, на твою думку, вони потрібні?
4. Чи є цей код програмою?

Якщо виникнуть труднощі — звернися до рубрики «Запитання–відповіді», але спочатку спробуй зробити власні висновки.

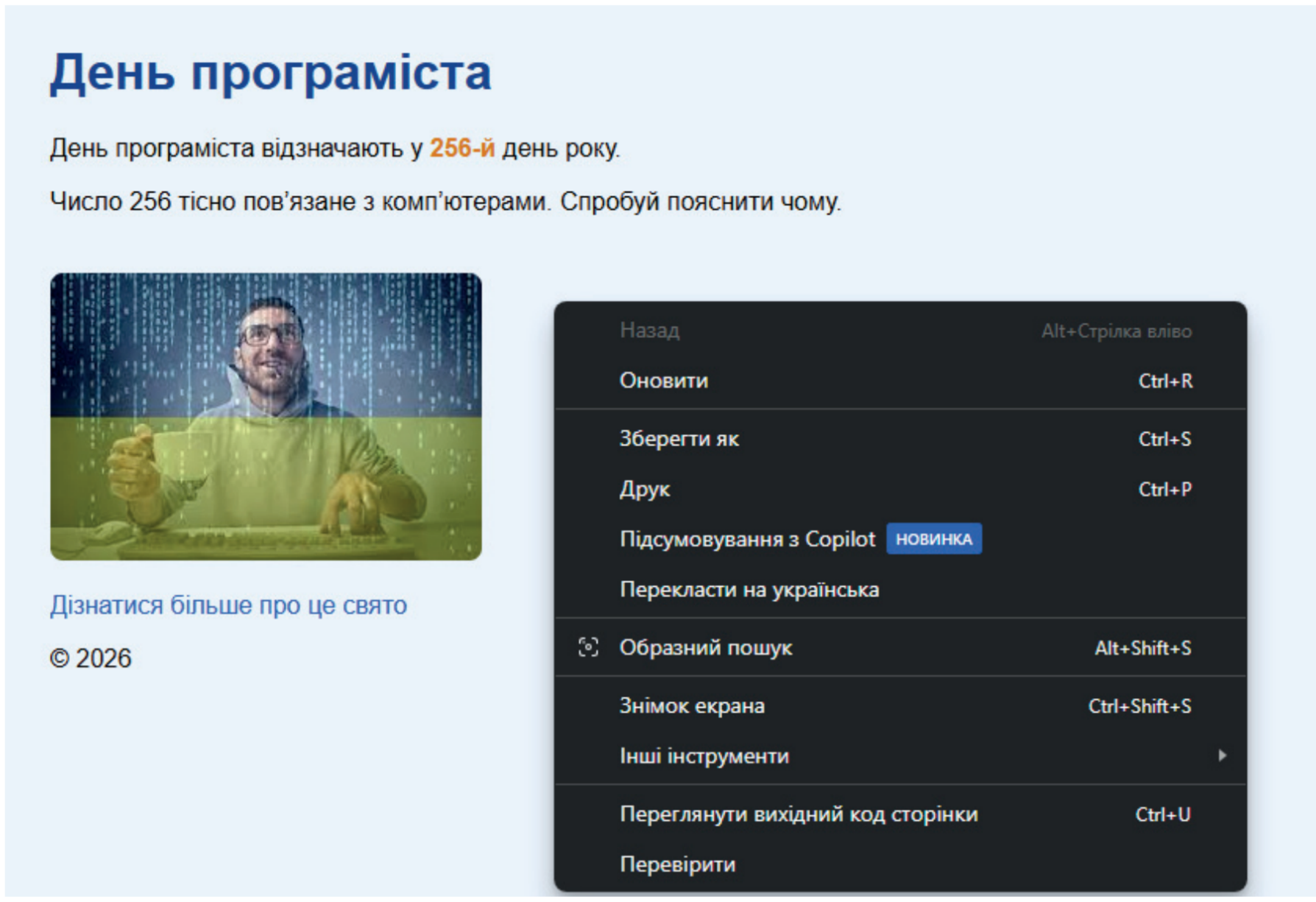
Завдання № 2

1. Відкрий е-додаток (16.1), завантаж файл з малюнком code.png у папку для твоїх робіт.
2. Відкрий Notepad (Блокнот) та скопіюй до нього на сторінці текст коду з цієї ж сторінки е-додатку.

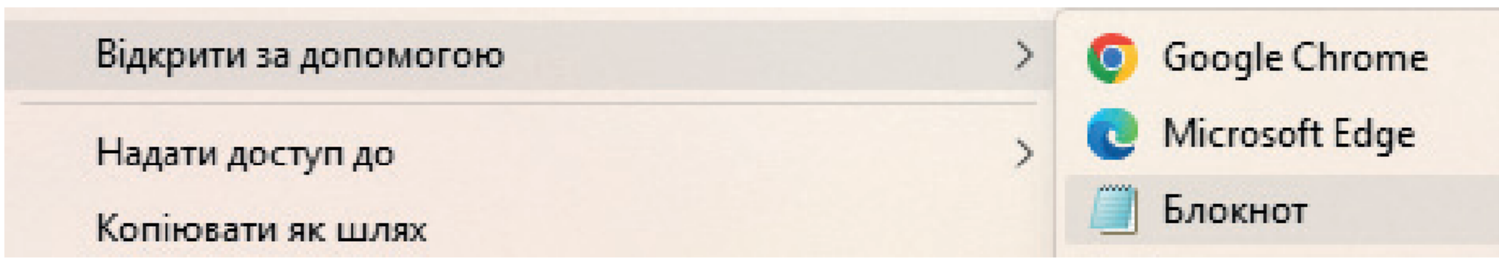
3. Збережи файл у папці для твоїх робіт:



4. Знайди у папці для твоїх робіт файл ***test.html*** та відкрий його у будь-якому браузері.
5. За допомогою контекстного меню або клавіш **Ctrl+U** переглянь код.



6. Відкрий файл ***test.html*** у програмі Notepad (Блокнот):



Видали тільки текст «День програміста» та збережи файл із розширенням .html. (дивись п.3). Відкрий файл у браузері та поясни що змінилось.

7. Віднови текст але прибери текст у <>:

<h1>День програміста</h1>

Збережи файл із розширенням .html. (дивись п.3). Відкрий файл у браузері та поясни що змінилось. Віднови початковий код сторінки.

8. Порівняй, чим відрізняються тексти:
   - «День програміста»
   - «Дізнатися більше про це свято»

   Знайди в коді відповідні фрагменти та визнач, які теги впливають на їх вигляд або поведінку.

Завдання № 3

Розглянь вихідний HTML-код вебсторінки із завдання № 2. У ньому вебсторінку описано за допомогою тегів — спеціальних позначок у кутових дужках, які допомагають браузеру зрозуміти структуру сторінки та призначення її елементів.

Підказка:

Тег — це елемент мови HTML, який визначає тип елемента та його призначення у структурі вебсторінки заголовок, абзац, зображення, посилання).

Спробуй відповісти на запитання самостійно. За потреби скористайся рубрикою «Запитання–відповіді» або знайди інформацію в інтернеті.

- Який вигляд має тег в HTML-коді? Якими символами він починається і закінчується?
- Які теги є в наведеному HTML-коді? Яку роль на сторінці виконує кожен із них?
- Що таке контейнерні (парні) теги та порожні теги? Які з тегів у наведеному коді належать до кожного виду?
- Чим відрізняється дія тегів <p> і у цьому прикладі?
- Знайди в коді атрибути (наприклад, `href`, `src`, `alt`, `class`). Для чого призначений кожен із них?
- Назви деяких тегів і атрибутів пов'язані з англійськими словами. Спробуй здогадатися, що означають: `header`, `main`, `footer`, `img`, `href`, `src`, `title`.

Поміркуй:
Чи всі теги впливають на зовнішній вигляд сторінки? Які з них визначають структуру, а які — оформлення?

Завдання № 4

Відкрий файл ***test.html***, створений у попередньому завданні, у програмі Notepad (Блокнот).

Виконай запропоновані зміни та після кожної з них зберігай файл і відкривай його в браузері:

- зміни заголовок;
- зміни текст під заголовком;
- зміни посилання.

Завдання № 4

Створи вебсторінку з інформацією про себе. Скористайся рубрикою «Запитання-відповіді».

На сторінці має бути:

- заголовок сторінки (тег <title>)
- головний заголовок на сторінці (тег <h1>);
- не менше двох абзаців тексту (тег <p>);
- виділення важливого слова або фрагмента тексту за допомогою тега ;
- зображення (тег <img> з атрибутами src і alt);
- посилання на інший вебресурс (тег <a> з атрибутом href).

Поясни після виконання:

- Які теги відповідають за структуру сторінки?
- Які елементи відображаються на сторінці, а які — ні?
- Які атрибути ти використав/використала і для чого вони потрібні?

Увага!

Щоб зображення відобразилося:

- ✓ *розмісти файл зображення в тій самій папці, що й HTML-файл (або вкажи правильний шлях до нього);*
- ✓ *у значенні атрибута src потрібно зазначити повне ім'я файлу разом із розширенням (наприклад: photo.jpg або image.png);*
- ✓ *атрибут alt повинен містити короткий опис зображення*

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке HTML?

HTML (англ. HyperText Markup Language — мова розмітки гіпертекстових документів) — це мова, за допомогою якої описують структуру вебсторінки: які об'єкти на ній розміщені та яку роль вони виконують (заголовок, абзац, зображення, посилання тощо).

HTML-документ опрацьовується браузером і відображається на екрані у зручному для людини вигляді.

Як виникла мова розмітки гіпертексту?

Мову HTML створив британський учений Тім Бернерс-Лі на початку 1990-х років для впорядкування документів у Європейській організації з ядерних досліджень (CERN). Більше про це у е-додатку (16.2).

Разом із мовою HTML він розробив перший веббраузер і запропонував систему гіпертекстових посилань. У той час точилися дискусії

про те, якою має бути система документів у мережі: текстовою чи з підтримкою зображень і кольорів. Зрештою поширення набув гіпертекст із можливістю додавання графіки та інших об'єктів.

Що таке теги та атрибути?

Тег — це основний елемент мови HTML. За допомогою тегів позначають частини вебсторінки та їх призначення.

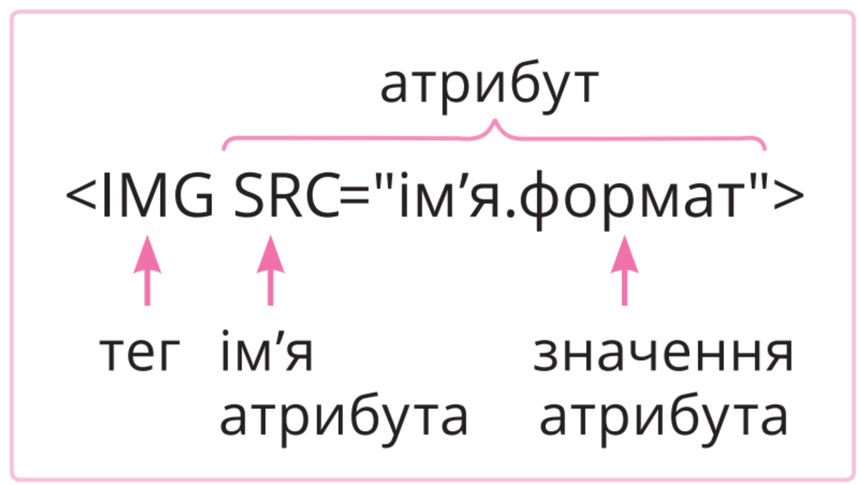
Тег записують у кутових дужках < >. Більшість тегів є парними і складаються з відкривного та закривного тегів, наприклад:

Текст абзацу

Закривний тег містить символ /.

Деякі теги не мають закривної частини (наприклад, <img>).

Атрибут — це додаткова характеристика тега, яка уточнює його властивості або поведінку.



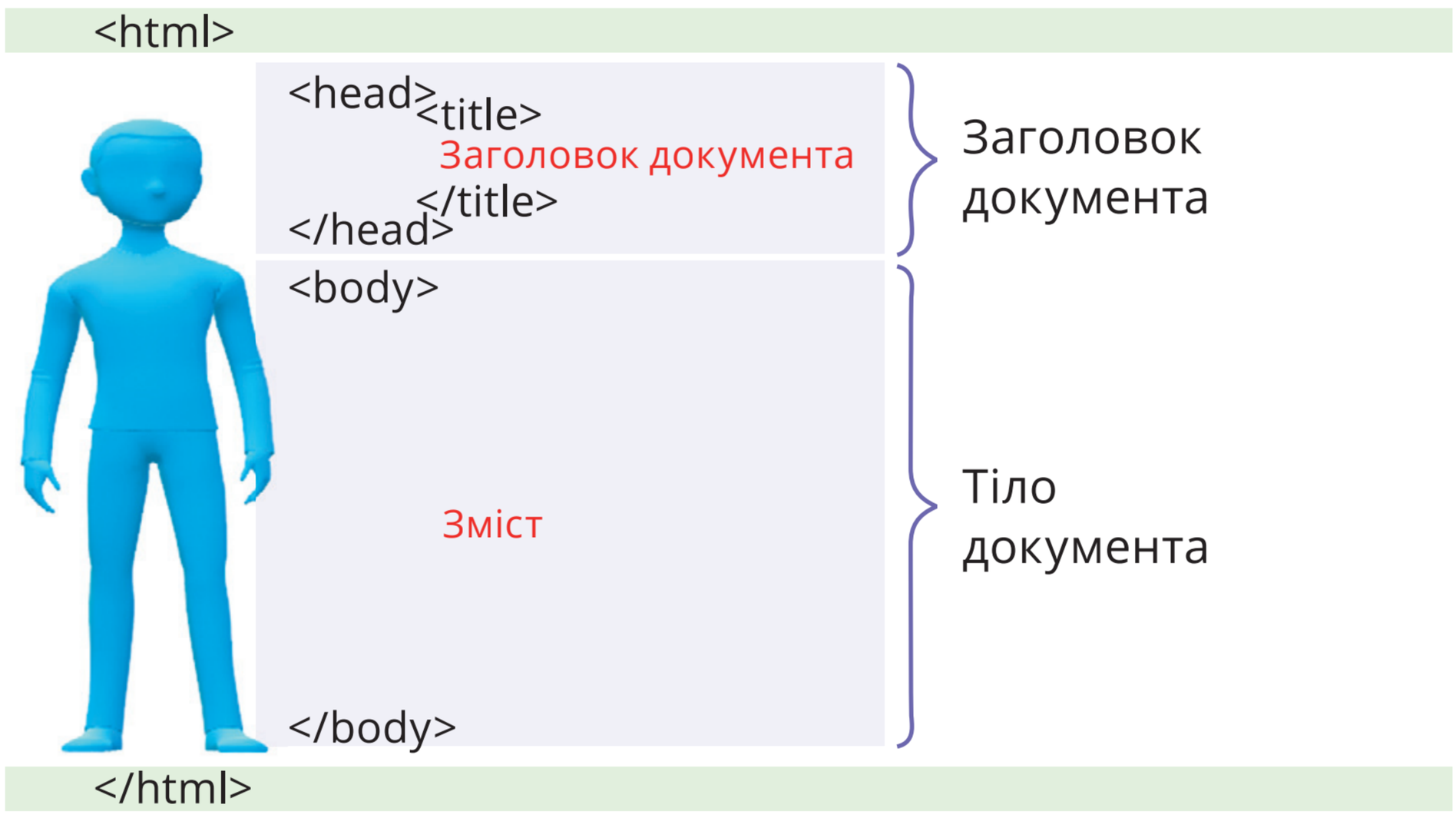
Атрибут записують усередині відкривного тега після його імені.

Якою є повна структура HTML-документа?

Документ починається з оголошення типу документа: <!DOCTYPE html>. Воно повідомляє браузеру, що документ створено за стандартом HTML5.

Увесь вміст сторінки розміщується між тегами <html> і </html> та складається з двох основних частин:

- заголовка документа (між тегами <head> і </head>);
- тіла документа (між тегами <body> і </body>).



Що містить заголовок документа?

У розділі <head> розміщують службову інформацію про сторінку. Зокрема:

- тег <title>, який визначає назву сторінки у вкладці браузера;
- метадані (наприклад, кодування документа);
- за потреби — опис стилів сторінки (<style>).

code

```
<head>
    <meta charset="UTF-8">
    <title>Назва сторінки</title>
</head>
```

Ця інформація зазвичай не відображається безпосередньо на сторінці.

Що містить тіло документа?

У розділі <body> розміщують основний вміст сторінки:

- заголовки,
- абзаци тексту,
- зображення,
- посилання та інші об'єкти.

Саме цей вміст бачить користувач у браузері.

Які основні теги та атрибути HTML використано в цьому параграфі?

Тег	Призначення
<html>...</html>	Містить увесь HTML-документ
<head>...</head>	Службова частина сторінки (назва, метадані)
<title>...</title>	Назва сторінки, що відображається у вкладці браузера
<body>...</body>	Основний вміст сторінки
<header>...</header>	Верхня частина сторінки або окремого розділу. Зазвичай містить заголовок, логотип, вступний текст або меню.
<main>...</main>	Основний зміст сторінки — найважливіша інформація, яку читає користувач.
<section>...</section>	Тематичний блок сторінки, що об'єднує пов'язані між собою матеріали.
<footer>...</footer>	Нижня частина сторінки або розділу. Зазвичай містить додаткову інформацію: контакти, авторство, дату створення, посилання

Тег	Призначення
...	Заголовок першого рівня
...	Абзац тексту
...	Виділення частини тексту всередині рядка
<a>...	Створює гіпертекстове посилання — текст або зображення, при натисканні на яке відкривається інша сторінка або ресурс. Використовують для переходу на інші вебсторінки, файли або розділи цієї ж сторінки. Обов'язково містить атрибут href, у якому вказують адресу переходу
<img>	Додавання зображення
<meta>	Службова інформація про документ
<style>...</style>	Опис стилів сторінки (CSS)

Довідник тег можна переглянути у е-додатку (16.3).

Атрибут	Призначення
lang	Мова документа
charset	Кодування символів
name	Назва службового параметра
content	Значення службового параметра
src	Шлях до файлу (наприклад, зображення)
alt	Текстовий опис зображення
href	Адреса ресурсу, на який веде посилання
class	Належність елемента до певного класу

Приклад використання:

1
```code
<p>Текст <span>важливий</span></p>
```

Тег створює абзац тексту.

Усередині цього абзацу розміщено тег , який виділяє лише частину тексту — слово «важливий».

Тег не створює нового абзацу і не переносить текст на новий рядок. Він використовується для виділення окремого фрагмента тексту всередині рядка, наприклад для подальшого оформлення або надання особливого значення.

2

```code
<img src="photo.jpg" alt="Фото">
```

Тег <img> додає зображення на вебсторінку.

У цього тега є атрибути:

- src — вказує шлях до файлу зображення. У цьому прикладі браузер шукає файл photo.jpg у тій самій папці, що й HTML-файл.
- alt — текстовий опис зображення. Він відображається, якщо зображення не завантажилось, а також використовується програмами для людей із порушеннями зору.

Важливо:

- Тег <img> не має закривної частини, він непарний тег.
- Без атрибута src зображення не відобразиться.
- Атрибут alt є важливим для доступності та коректного відображення сторінки.

3

```code
<a href="https://example.com">Посилання</a>
```

Тег <a> створює гіпертекстове посилання.

Текст між відкривним і закривним тегами (Посилання) — це те, що бачить користувач і на що можна натиснути.

Атрибут href містить адресу ресурсу, на який відбудеться перехід після натискання.

У цьому прикладі при натисканні відкриється сторінка за адресою https://example.com.

Важливо:

- <a> — контейнерний (парний) тег.
- Без атрибута href текст не буде активним посиланням.
- Саме завдяки тегу <a> вебсторінки поєднуються між собою у Всесвітню павутину.

Як створити вебсторінку?

Для створення вебсторінки можна використати як спеціальні редактори, так і звичайний текстовий редактор. Наприклад, Блокнот (Notepad) або безплатну програму Notepad++. Перевага Notepad++ у тому, що він підсвічує теги та атрибути різними кольорами, тому код легше читати.

Порада. Зберігай файл у кодуванні UTF-8 і додавай у код рядок <meta charset="UTF-8">, щоб українські літери відображалися правильно.

Серед сучасних редакторів популярним є **Visual Studio Code (VS Code)**. Цей редактор полегшує роботу, оскільки має:

- підсвічування синтаксису;
- автоматичне форматування коду;
- перегляд результату в браузері в реальному часі (за допомогою розширень);
- зручну навігацію по коду.

Завантажити редактор можна з офіційного сайту: https://code.visualstudio.com

Існують також онлайн-редактори, наприклад **CodePen** (https://codepen.io), які дозволяють створювати й переглядати вебсторінки без встановлення програм на комп'ютер.

Як зберегти HTML-файл?

У Блокноті (Notepad)

1. Введи HTML-код вебсторінки.
2. Обери **Файл** → **Зберегти як...**
3. У полі ***Тип файлу*** вибери **Усі файли**.
4. У полі Ім'я файлу введи назву з розширенням **.html** (наприклад, index.html).
5. Якщо є вибір кодування — обери UTF-8.
6. Натисни **Зберегти**.

У Notepad++

1. Створи новий файл.
2. Введи HTML-код вебсторінки.
3. Обери **Файл** → **Зберегти як...**
4. Вкажи ім'я з розширенням **.html** (наприклад, index.html) і натисни **Зберегти**.
5. (За потреби) Перевір кодування: **Кодування** → **UTF-8**.

Як відкрити вебсторінку в браузері

Після створення файлу відкрий його в браузері:

- двічі клацни файл index.html, або
- натисни правою кнопкою миші → **Відкрити за допомогою** → обери браузер.

Якщо ти вносиш зміни в HTML-код, збережи файл, а в браузері натисни **F5** (або кнопку оновлення сторінки).

ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Що таке HTML? Чим мова розмітки відрізняється від мови програмування?
2. Що таке тег? Якими символами він записується?
3. Чим відрізняються парні та непарні теги? Наведи приклади.
4. Що таке атрибут? Де він записується?
5. Яку роль виконують теги: `<html>`, `<head>`, `<body>`, `<title>`?
6. Для чого використовують атрибути `href`, `src`, `alt`, `lang`, `charset`?

7. Що станеться, якщо:
   - прибрати закривний тег?
   - не вказати розширення .html під час збереження файлу?
   - не додати атрибут src у тег `<img>`?
8. Чому файл потрібно зберігати в кодуванні UTF-8?
9. Поясни, чому тег <a> є основою існування Всесвітньої павутини.

10. Створи фрагмент коду, який містить:
    1) заголовок першого рівня;
    2) абзац тексту;
    3) виділене слово за допомогою ;
    4) зображення;
    5) посилання.
11. Зміни у створеній раніше сторінці:
    1) заголовок;
    2) текст;
    3) адресу посилання;
    4) назву вкладки браузера.
12. Створи тематичний розділ сторінки за допомогою тегів `<header>`, `<main>`, `<section>`, `<footer>`.

13. Чому, на твою думку, HTML називають мовою розмітки, а не мовою програмування?
14. Чому важливо правильно структурувати сторінку?
15. Чи можна створити вебсторінку без спеціальних програм? Чому?
16. Які труднощі виникли під час роботи з HTML-кодом?
17. Яку роль відіграє гіпертекст у сучасному інформаційному світі?
17. Чи легко «читати» HTML-код людині? Чому браузеру це простіше?

§ 17. Як змінити вигляд вебсторінки? Основи CSS

Як змінити вигляд вебсторінки, не змінюючи її структуру?

Ти вже вмієш створювати структуру вебсторінки за допомогою HTML: додавати заголовки, текст, зображення, посилання. Але якщо відкрити створену сторінку в браузері, вона виглядає досить просто: стандартний шрифт, білий фон, елементи розташовані один під одним.

Чому так відбувається?

HTML повідомляє браузеру, що є заголовком, що є абзацом, а що — зображенням але він майже не впливає на те, як саме це виглядатиме. Тоді виникає запитання:

- як змінити колір тексту?
- як зробити фон іншим?
- як розташувати елементи не один під одним, а поруч?

Для цього використовують інший інструмент — CSS.

Завдання № 1

1. Створи два файли з поданим кодом або завантаж текстові файли з е-додатку (17.1).

Файл 1 — basic.html

code

```
<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <title>Приклад 1</title>
</head>
<body>

<h1>Світ вебтехнологій</h1>
<p>HTML визначає структуру сторінки.</p>
<p>CSS відповідає за її вигляд.</p>
```

```
<ul>
    <li>Заголовки</li>
    <li>Абзаци</li>
    <li>Посилання</li>
</ul>
<a href="#">Дізнатися більше</a>
</body>
</html>
```

Файл 2 — styled.html

code

```
<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <title>Приклад 2</title>

    <style>
        body {
            background-color: #f0f6ff;
            font-family: Arial, sans-serif;
        }
        h1 {
            color: #1e3a8a;
        }
        p {
            color: #333333;
        }
        a {
            color: #d97706;
        }
    </style>

</head>
<body>

<h1>Світ вебтехнологій</h1>
<p>HTML визначає структуру сторінки.</p>
<p>CSS відповідає за її вигляд.</p>
<ul>
    <li>Заголовки</li>
    <li>Абзаци</li>
    <li>Посилання</li>
</ul>
<a href="#">Дізнатися більше</a>
</body>
</html>
```

2. Відкрий обидва файли в браузері та порівняй. Дай відповіді на запитання:
   - Чи відрізняється фон сторінок?
   - Чи змінився зміст сторінки?
   - Чи змінилася структура HTML-коду?
   - Що саме вплинуло на вигляд другої сторінки?

3. Досліди синтаксис CSS.
   Відкрий файл styled.html і уважно розглянь блок:

code
```
<style>
    body {
        background-color: #f0f6ff;
        font-family: Arial, sans-serif;
    }
    h1 {
        color: #1e3a8a;
    }
    p {
        color: #333333;
    }
    a {
        color: #d97706;
    }
</style>
```

4. Спробуй дати визначення власними словами
   У записі:

code
```
h1 {
    color: #1e3a8a;
}
```

h1 — це _____
color — це _____
#1e3a8a — це _____

5. Проведи експеримент, зміни значення властивостей:
   1) заміни background-color на інший колір. Знайти шістнадцятковий код кольору можна у е-додатку (17.2).
   2) зміни колір заголовка;
   3) зміни шрифт у font-family;
   4) зроби текст посилання іншим кольором.

   Після кожної зміни:
   - Збережи файл.
   - Онови сторінку в браузері.
   - Опиши, що змінилося.

6. Спробуй зламати правило:
   1) Видали крапку з комою ;
   2) Видали одну фігурну дужку { або }
   3) Зміни назву властивості на випадкове слово

   Поясни:
   - Що відбувається?
   - Чи працює правило?
   - Чому?

Завдання № 2

Використай одну й ту саму HTML-структуру (код basic.html) і створи дві версії оформлення за допомогою CSS:
1) Світла тема «Денний режим»
2) Темна тема «Нічний режим»

Світло навколо нас впливає не лише на настрій, а й на те, як ми сприймаємо інформацію. Під час роботи з екраном очі постійно адаптуються до рівня освітлення, і надмірний контраст може викликати втому. Саме тому в денному режимі використовують світлий фон і темний текст — так сторінка залишається читабельною при яскравому освітленні, а очі менше напружуються. У нічному режимі навпаки застосовують темний фон і світлий текст. Це допомагає зменшити навантаження на зір у темряві, знижує дискомфорт і робить роботу з екраном безпечнішою для здоров'я.
Турбота про оформлення сторінки — це не лише питання дизайну, а й питання збереження зору користувача.

Скористайся інструкцією:

1. Зроби копію файлу basic.html двічі та перейменуй створені копії:
   - light.html
   - dark.html
2. У кожному файлі додай у `<head>` блок `<style>...</style>` та зміни щонайменше:
   - колір фону сторінки (`background-color`);
   - колір заголовка (`h1 { color: ... }`);
   - колір тексту абзаців (`p { color: ... }`);
   - колір посилання (`a { color: ... }`);
   - шрифт сторінки (`font-family`).

Порівняй результати

1. Чи змінилася HTML-структура сторінки в обох файлах?
2. Які CSS-властивості найбільше вплинули на відчуття «денного» і «нічного» режимів?
3. Яка тема зручніша для читання? Чому?

Завдання № 3

Ознайомся із повідомленням.

Сучасний вебдизайн дедалі частіше відмовляється від зайвого декору на користь чистоти та простоти. Одним із провідних напрямів є мінімалістичний шрифтовий дизайн.

Наприклад:

Адреса сайту: https://www.calebraney.com/

У такому підході головну роль відіграє не кількість ефектів, а якість типографіки — правильний вибір шрифту, його розміру, товщини та відступів. Зазвичай використовують лише 1–2 шрифти, але продумано вибудовують ієрархію: великий виразний заголовок, помірний підзаголовок і зручний для читання основний текст.

Коли ми створюємо вебсторінку, то розраховуємо лише на ті шрифти, які встановлені на комп'ютер. А як же зробити так, щоб сторінка використовувала сучасні дизайнерські шрифти?

Для цього існує сервіс **Google Fonts** (https://fonts.google.com).

Google Fonts — це безкоштовна бібліотека з понад 1500 шрифтів з відкритим вихідним кодом, що надається компанією Google, а також інтерактивний каталог для їх перегляду. Вона призначена для полегшення інтеграції типографіки в вебсайти, мобільні додатки, документи та інші цифрові проекти за допомогою CSS або безпосереднього завантаження.

Об'єднайтесь у пари та створіть титульну вебсторінку сайту, основним оформленням якої буде текст. Скористайтесь інструкцією:

Перейди на сайт https://fonts.google.com та обери будь-який шрифт (наприклад, Roboto, Montserrat або Open Sans). Після вибору натисни **Select** або **Get font**. Сервіс запропонує код для підключення шрифту. Скопіюй рядок виду:

code

```
<link href="https://fonts.googleapis.com/css2?family=Roboto&
display=swap" rel="stylesheet">
```

Потім у блоці <style> задай вибраний шрифт для сторінки, наприклад:

```code
body {
    font-family: 'Roboto', sans-serif;
}
```

Проведи експеримент

1. Зміни шрифт заголовка (h1) на інший.
2. Використай один шрифт для тексту абзаців, інший — для заголовка.
3. Порівняй:
   - Чи змінився настрій сторінки?
   - Який шрифт виглядає сучасніше?
   - Який зручніший для читання?

Додаткове дослідження

Зверни увагу, що деякі шрифти мають різні варіанти товщини:
font-weight: 300;
font-weight: 700;

Спробуй змінити товщину шрифту заголовка.

Поясни

- Чому потрібно підключати шрифт через <link>?
- Чи змінилася HTML-структура сторінки?
- Що саме вплинуло на зовнішній вигляд?

ЗАПИТАННЯ-ВІДПОВІДІ

Що таке CSS?

CSS (Cascading Style Sheets) — це мова стилів, яка визначає зовнішній вигляд вебсторінки: кольори, шрифти, відступи, розміри, розташування елементів. Якщо HTML описує *що є на сторінці* (заголовок, абзац, зображення), то CSS описує *як це має виглядати.*

HTML (структура):

```
<h1>Світ вебтехнологій</h1>
<p>CSS робить сторінку охайною та читабельною.</p>
<a href="#">Дізнатися більше</a>
```

CSS (змінює вигляд):

```
body { font-family: Arial; }
h1 { color: #1e3a8a; }
a  { color: #d97706; }
```

Як підключають CSS до сторінки?

На початку можна записувати стилі прямо в HTML у розділі <head>:

code
```
<head>
  <style>
    h1 { color: blue; }
  </style>
</head>
```

Більш професійно стилі зберігають в окремому файлі .css і підключають до сторінки.

Синтаксис CSS: з чого складається правило?

CSS записують у вигляді правил. Кожне правило має таку структуру:

```
селектор {
    властивість: значення;
}
```

1) Селектор — до кого *застосувати стиль*
Наприклад, p означає «застосувати до всіх абзаців <p>»:

```
p { color: #333; }
```

2) Властивість — *що саме змінюємо*
Наприклад, `color` — колір тексту, `background-color` — колір фону, `font-family` — шрифт.

3) Значення — *яким буде результат*
Наприклад, `#333`, `lightblue`, `Arial`.

Важливі символи в CSS:

- Фігурні дужки { } — відокремлюють правила для одного селектора.
- Двокрапка : — відділяє властивість від значення.
- Крапка з комою ; — завершує запис однієї властивості.

Приклад із кількома властивостями:

code
```
body {
    background-color: #f0f6ff;
    font-family: Arial;
    margin: 30px;
}
```

Які властивості стильових правил використовують найчастіше?

Властивість	Призначення	Приклади значень
font-family	Назва шрифту	Arial, "Times New Roman", sans-serif
font-style	Накреслення шрифту	normal, italic
font-weight	Жирність шрифту	normal (400), bold (700), 100–900
font-size	Розмір шрифту	16px, 1.2em, 120%
color	Колір тексту	#FF0000, red, rgb(255,0,0)
background-color	Колір фону	yellow, #f0f6ff
text-align	Вирівнювання тексту	left, right, center, justify
line-height	Висота рядка	1.5, 150%

ПЕРЕВІР СЕБЕ

ЗНАННЯ — РОЗУМІННЯ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Що таке CSS?
2. Із чого складається CSS-правило?
3. Що таке селектор?
4. Які властивості відповідають за:
   - колір тексту;
   - колір фону;
   - розмір шрифту;
   - відступи між елементами?
5. Які одиниці вимірювання використовують у CSS?
6. Чим відрізняється роль HTML і CSS на вебсторінці?
7. Чому потрібно підключати шрифт з сайту https://fonts.google.com через `<link>`?
8. Напиши CSS-правило, яке:
   - змінить колір заголовка;
   - збільшить розмір тексту;
   - додасть відступ під абзацом.
9. Чому важливо відокремлювати структуру сторінки (HTML) від її оформлення (CSS)?

§ 18. Вебсторінка як єдина система

Як HTML і CSS разом створюють сучасну вебсторінку?

Ти вже вмієш створювати структуру сторінки за допомогою HTML і змінювати її вигляд за допомогою CSS. Але як саме браузер «поєднує» ці дві мови?

Чому одні елементи вебсторінки розташовуються один під одним, а інші — в одному рядку? Чому іноді стиль «не працює»? І що означає, що CSS — каскадна мова?

Щоб створювати вебсторінки впевнено, потрібно зрозуміти, як усе це працює разом.

Завдання № 1

1. Створи HTML-файл із таким кодом:

code

```
<!DOCTYPE html>
<html lang="uk">
<head>
<meta charset="UTF-8">
<title>Каскадність</title>

<style>
    p {
        color: blue;
    }
    p {
        color: red;
    }
</style>

</head>
<body>

<p>Перевір роботу каскаду</p>

</body>
</html>
```

2. Досліди
   - Якого кольору текст?
   - Скільки правил задано для абзацу?
   - Чому спрацювало лише одне?
3. Поміняй порядок правил:

```
p {
    color: red;
}

p {
    color: blue;
}
```

4. Збережи файл і перевір результат. Зроби висновки, як CSS застосовує правила.

   Поясни, що означає вислів «CSS — каскадна мова».

Завдання № 2

Не створюючи сторінку проаналізуй код і дай відповіді на запитання.

code

```
<!DOCTYPE html>
<html lang="uk">
<head>
<meta charset="UTF-8">
<title>Аналіз каскаду</title>

<style>
    body {
        color: green;
        font-size: 18px;
    }

    p {
        color: blue;
    }

    p {
        font-size: 24px;
    }

    p {
        color: red;
    }
</style>

</head>
<body>

<p>Перевір роботу каскаду</p>

</body>
</html>
```

1. Якого кольору буде текст?
2. Який розмір шрифту матиме абзац?
3. Які правила застосуються повністю, а які частково?
4. Чому стиль, заданий для body, не зник повністю?

Після відповіді перевір себе, відкривши файл у браузері.

Завдання № 3

Розглянь три варіанти коду та дай відповіді на запитання:

а

code

```
<h1>Заголовок</h1><p>Текст</p>
```

б

code

```
<h1>Заголовок <span>Текст</span></h1>
```

в

code

```
<span>Текст</span><span>Ще текст</span>
```

- У яких випадках елементи будуть в одному рядку?
- У яких — в різних? Чому?

Дізнайся більше про рядкові і блочні теги із рубрики «Запитання-відповіді».

Завдання № 4

1. Створи вебсторінку із короткою «новиною». Сторінка повинна містити:
   - заголовок;
   - основний текст;
   - одне слово в тексті має виглядати як кнопка (але залишатися в рядку);
   - під текстом — блок із датою публікації.

Для створення кнопки скористайся кодом:

html:

```
<button class="btn">Натисни мене</button>
```

css:

```
<style>
    .btn {
        background-color: blue;
        color: white;
        padding: 8px 15px;
        border: none;
        border-radius: 5px;
    }
</style>
```

2. За допомогою рубрики «Запитання-відповіді» дай відповідь на питання, що таке класи і для чого їх використовують у сучасному веб-дизайні.
3. Збільш значення padding у кнопки. Поясни, що змінилося і чому кнопка стала більшою.
4. Додай до кнопки:

```
margin-top: 15px;
```

   Проаналізуй що змінилось.
5. Додай відступ знизу до заголовка:

```
margin-bottom: 20px;
```

   Як це вплинуло на розташування елементів?
6. За допомогою рубрики «Запитання-відповіді» зроби висновки:
   Padding — це ...
   Margin — це ...

Завдання № 5

1. Створи вебсторінку скориставшись кодом:

```
<!DOCTYPE html>
<html lang="uk">
<head>
<meta charset="UTF-8">
<title>Адаптивний текст</title>

<style>
    body {
        font-family: Arial, sans-serif;
    }
```

```
    p {
        width: 800px;
    }
</style>

</head>
<body>

<h1>Штучний інтелект став частиною шкільного життя</h1>

<p>
    Усе більше школярів використовують інструменти штучного інтелекту для навчання. Вони допомагають пояснити складні теми, знайти помилки в коді або підготуватися до контрольної. Водночас важливо пам’ятати: технології мають доповнювати власне мислення, а не замінювати його.
</p>

</body>
</html>
```

2. Досліди:

   Змінюй розміри вікна браузера і спостерігай що відбувається із текстом. Відкрий декілька інших сайтів, які ти використовуєш та повтори дії. Чи є різниця? Як ти думаєш чому?

3. Заміни у CSS:

   ```
   width: 800px;
   ```

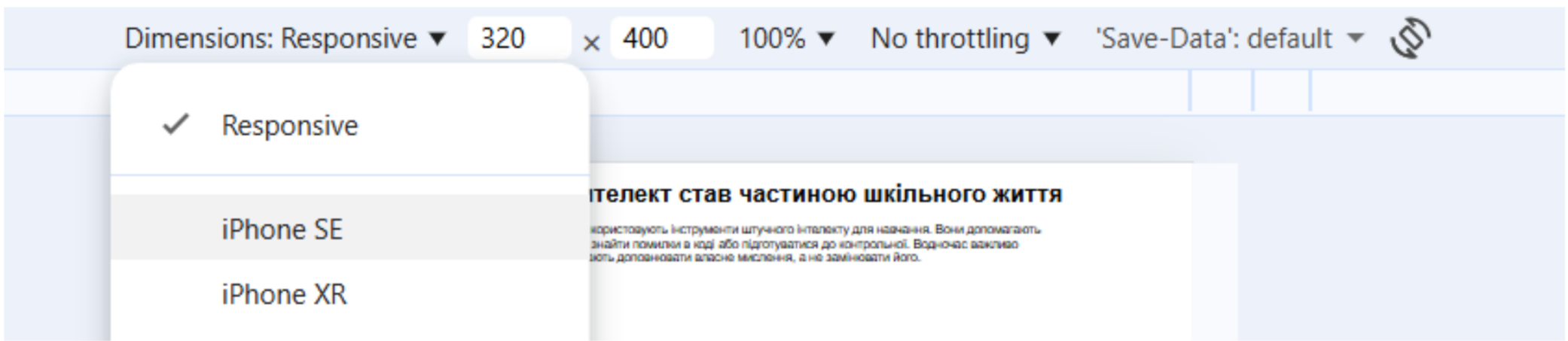
   на

   ```
   max-width: 800px;
   width: 100%;
   ```

   Перевір, що змінилося? Пясни, який «механізм» змін.

4. Натисни клавішу **F12**, у правій частині вікна знайди кнопку і натисни її.

   Досліди, який вигляд матиме створена вебсторінка для різних мобільних пристроїв.



5. За допомогою рубрики «Запитання-відповіді» поясни, що означає «сторінка є адаптивною».

ЗАПИТАННЯ-ВІДПОВІДІ

Як HTML і CSS працюють разом?

HTML створює структуру сторінки: визначає заголовки, абзаци, посилання, зображення. CSS відповідає за зовнішній вигляд: кольори, шрифти, розміри, відступи.

Браузер спочатку читає HTML-документ і будує структуру сторінки, а потім застосовує до елементів правила CSS.

У якій послідовності браузер обробляє сторінку?

1. Зчитує HTML.
2. Формує структуру документа.
3. Застосовує правила CSS.
4. Відображає результат на екрані.

Що означає, що CSS — каскадна мова?

Каскадність означає, що до одного елемента може застосовуватися кілька правил CSS. Якщо одна й та сама властивість задана кілька разів, браузер визначає, яке значення використати. У простому випадку діє правило:

Якщо селектори однакові, спрацьовує те правило, яке записане пізніше.

Наприклад:

```
h1 {
    font-size: 20px;
}

h1 {
    font-size: 40px;
}
```

Заголовок матиме розмір 40px, тому що друге правило записане нижче.

Водночас інші властивості з попередніх правил не зникають — браузер поєднує їх.

Також важливо знати, що деякі селектори можуть мати більшу «силу» (наприклад, стиль, записаний безпосередньо в тегу через `style=""`). У такому випадку вони мають пріоритет над загальними правилами.

Чому іноді стиль «не працює»?

Можливі причини:

- правило записане вище і перезаписується пізніше;
- селектор не відповідає потрібному елементу;
- інша властивість має більший пріоритет.

Чим відрізняються блокові та рядкові елементи?

Тег — це запис у кутових дужках (<p>,). **Елемент** — це повна конструкція разом із вмістом:

```
<p>Текст</p>
```

Блокові елементи починаються з нового рядка і займають усю доступну ширину. Наприклад, елемент:

```
<p>текст, текст, текст </p>
<p>текст, текст, текст </p>
```

завжди буде розташований окремо. На сторінці у браузері це виглядає так:

[Абзац]
[Абзац]

Рядкові елементи розміщуються в одному рядку з текстом і займають лише необхідне місце. Наприклад:

```
Текст <span>виділений текст</span> текст.
```

Тег не переносить текст на новий рядок.

На сторінці у браузері це виглядає так:

Текст [текст] текст.

Тип відображення можна змінити за допомогою CSS:

```
display: block;
display: inline;
```

Блокові теги
<h1>–<h6>
<p>
<section>
<header>
<footer>
<button>

Рядкові теги
<a>
<img>
<input>

Що таке клас у CSS?

Клас — це спосіб задати стиль окремій групі елементів. В HTML він записується як атрибут `class`, а в CSS — через крапку:

```
.btn {
    color: white;
}
```

Назву класу придумує розробник (але бажано давати змістовні назви).

Класи дозволяють повторно використовувати стилі в різних частинах сторінки.

- Як задати клас в HTML?

Клас задають за допомогою атрибута `class` у відкривальному тегу:

```
<p class="important">Це важливий текст.</p>
```

Тут:

`class` — це атрибут;
`"important"` — назва класу.

Назву класу ви вигадуєте самостійно (але без пробілів і спеціальних символів).

- Як використати клас у CSS?

У CSS клас позначають крапкою `.` перед його назвою:

```
.important {
    color: red;
    font-weight: bold;
}
```

Це означає:
Застосувати ці стилі до всіх елементів, які мають `class="important"`.

Наприклад:

1 html:

```
<p>Число <span class="highlight">256</span>
пов’язане з комп’ютерами.</p>
```

code

css:

```
.highlight {
    color: orange;
    font-weight: bold;
}
```

code

Стиль застосовується лише до числа 256, а не до всього абзацу.

2 html:

```
<p>Це звичайний абзац.</p>
<p class="note">Це примітка.</p>
<p>Це ще один звичайний абзац.</p>
```

code

css:

```code
.note {
    background-color: #fff3cd;
    padding: 10px;
}
```

Один абзац зі стилями, а інші ні.

3 html:

```code
<p class="note important">Увага! Це важлива примітка.</p>
```

css:

```code
.note { background-color: #fff3cd; }
.important { font-weight: bold; }
```

Один елемент може мати кілька класів.

Як покращити читабельність тексту?

Читабельність залежить не лише від змісту тексту, а й від того, як він розміщений на сторінці. Якщо текст «притиснутий» до країв або блоки стоять занадто близько один до одного, читати його важче.

Щоб зробити сторінку зручнішою для сприйняття, використовують відступи.

Є два основні типи відступів:

`margin` — зовнішній відступ

Це відстань між різними елементами сторінки. Він додає «простір» між абзацами, заголовками та іншими блоками.

Приклад:

```
p {
    margin-bottom: 15px;
}
```

Такий запис додає відстань під кожним абзацом, і текст стає менш «злиплим».

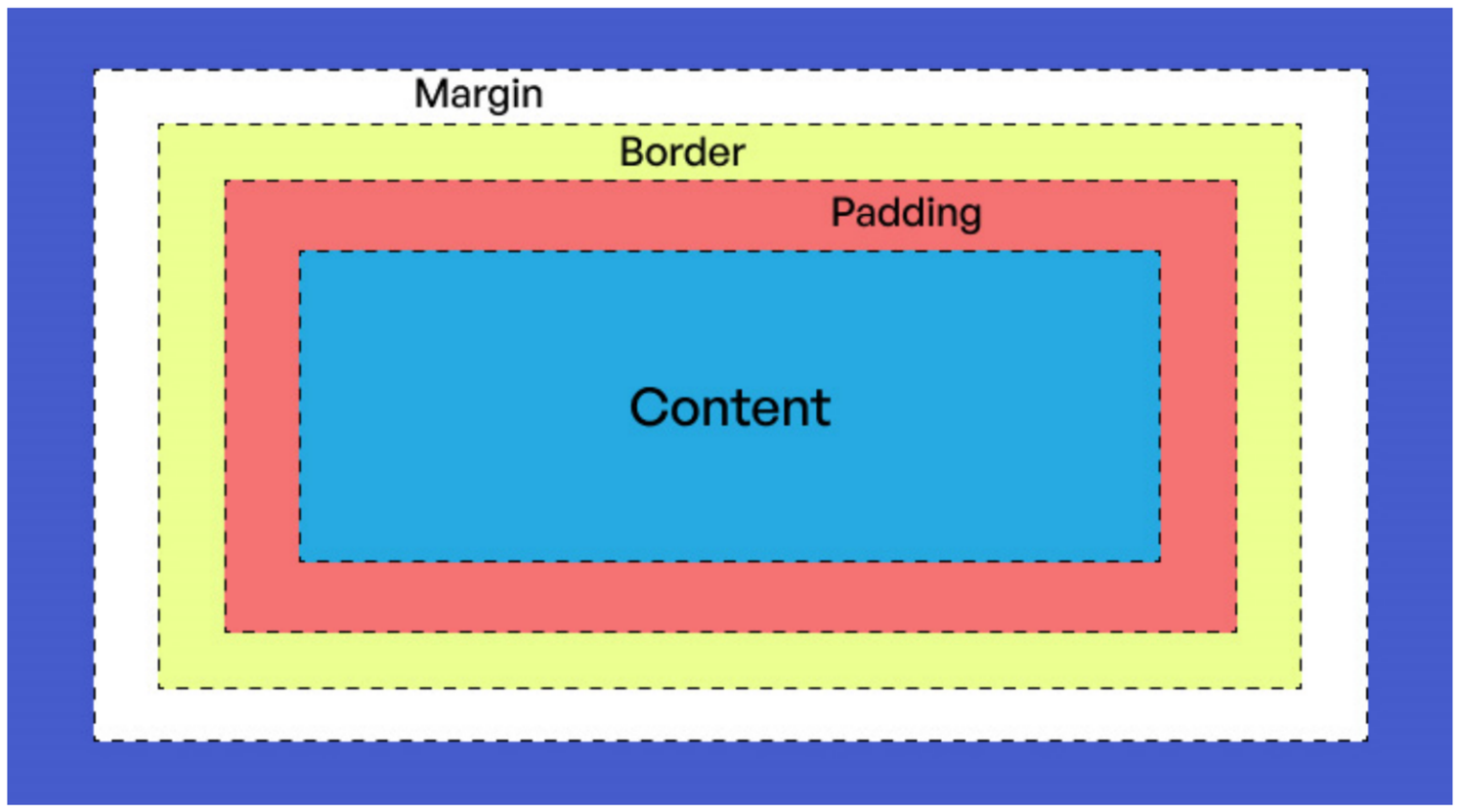
`padding` — внутрішній відступ

Це відстань між текстом і межами самого елемента.

Приклад:

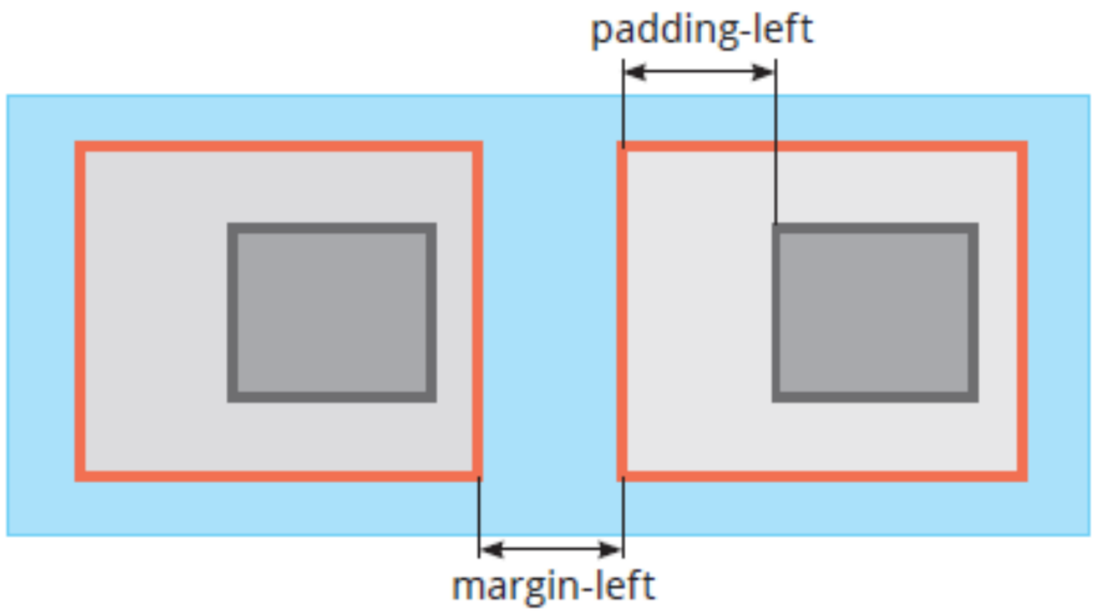
```
p {
    padding: 10px;
}
```

Текст не буде притиснутий до краю блоку.



У CSS відступи можна задавати не тільки для всього елемента одразу, а й для окремих сторін. Кожен елемент на сторінці має чотири сторони:

- **top** — верх
- **bottom** — низ
- **left** — ліва сторона
- **right** — права сторона



Наприклад:

```code
p {
    margin-top: 10px;
    margin-bottom: 20px;
}
```

`margin-top` — відступ над абзацом (10 пікселів).

`margin-bottom` — відступ під абзацом (20 пікселів).

Отже, абзац матиме більше простору знизу, ніж зверху.

```code
p {
    padding-left: 10px;
}
```

Текст у абзаці буде зміщений вправо на 15 пікселів від лівого краю блоку

```
top
left | текст | right
bottom
```

Що означає, що сторінка є адаптивною і чому це важливо?

Адаптивна сторінка — це вебсторінка, яка автоматично підлаштовується під розмір екрана пристрою. Її елементи змінюють ширину, розташування або масштаб так, щоб текст залишався

читабельним, а зображення й кнопки — зручними для використання як на комп'ютері, так і на смартфоні.

Особливо важливо, щоб сторінка була зручною для мобільних пристроїв, адже більшість користувачів переглядають сайти саме з телефона. Якщо текст занадто дрібний або потрібно прокручувати сторінку горизонтально, така сторінка не є адаптивною.

Перевірити адаптивність можна за допомогою інструментів розробника у браузері (клавіша **F12**), перемикаючись між різними розмірами екранів (кнопка). Це дозволяє побачити, як сторінка виглядатиме на різних пристроях.

Адаптивність важлива не лише для зручності користувачів (UX/UI). Пошукова система Google надає перевагу адаптивним сайтам (підхід *Mobile-first indexing*), що може підвищити їх у результатах пошуку.

Як сторінка підлаштовується під ширину екрана?

Сторінка підлаштовується під ширину екрана завдяки поєднанню HTML і CSS. HTML визначає структуру елементів, а CSS задає їхні розміри та розташування.

Щоб сторінка могла змінюватися разом із шириною екрана, замість фіксованих значень у пікселях використовують відносні величини, наприклад відсотки (%) або властивість max-width. У такому разі елемент не має жорстко заданої ширини, а може розширюватися або звужуватися залежно від розміру вікна браузера.

Також важливо правильно задати метатег масштабу в розділі <head>:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Він повідомляє браузеру, що сторінка повинна відображатися відповідно до ширини пристрою.

У майбутньому для створення більш складної адаптивної верстки використовують спеціальні механізми CSS — наприклад, гнучкі сітки (Flexbox, Grid) та медіазапити. Вони дозволяють змінювати розташування елементів залежно від розміру екрана.

Що означає width: 100%?

Властивість width: 100% означає, що елемент займає всю доступну ширину тієї частини сторінки, у якій він знаходиться.

Наприклад:

```
img {
    width: 100%;
}
```

Це означає, що зображення розтягнеться на всю ширину області, у якій воно розміщене. Якщо ширина сторінки зміниться, зображення також змінить свій розмір.

Як у CSS задаються розміри елементів, шрифтів і відступів?

У CSS розміри елементів, шрифтів і відступів задають за допомогою одиниць вимірювання. Вони бувають абсолютні та відносні.

Абсолютні одиниці.

Одиниця	Що означає	Приклад
px	пікселі (точки екрана)	font-size: 16px;

Відносні одниниці.

Одиниця	Відносно чого обчислюється	Приклад
%	від розміру батьківського елемента*	width: 50%;
em	від розміру шрифту батьківського елемента	margin: 2em;
rem	від розміру шрифту кореневого елемента	font-size: 1.2rem;

**HTML-структуру можна уявити як дерево:*
Кожен рівень має свого «батька» і може мати «дітей».
Елемент, який містить інший елемент усередині, називається батьківським елементом і відповідно елемент, що знаходиться всередині, називається дочірнім елементом.

```
html  ← кореневий
 └── body
      └── div
           └── p
```

Наприклад:

```
<section>
   <p>Текст <span>важливий</span></p>
</section>
```

У цьому прикладі:
<section> — батьківський елемент для <p>;
<p> — дочірній для <section> і водночас батьківський для ;
* — дочірній елемент для <p>.*

ПЕРЕВІР СЕБЕ

ЗНАННЯ — РОЗУМІННЯ — АНАЛІЗ — СТВОРЕННЯ

1. Яку роль виконує HTML у створенні вебсторінки?
2. За що відповідає CSS?
3. У якій послідовності браузер обробляє вебсторінку?
4. Що означає, що CSS — каскадна мова?
5. Чим відрізняються блокові та рядкові елементи?

6. Чим відрізняються margin і padding?
7. Що означає `width: 100%`?
8. Що означає, що сторінка є адаптивною?

9. Чому друге правило CSS може перезаписати перше?
10. Чому деякі властивості застосовуються, навіть якщо інші були змінені?
11. Чому текст може виглядати по-різному при зміні ширини вікна браузера?
12. Чому для мобільних пристроїв краще використовувати відносні одиниці вимірювання?

13. Проаналізуй фрагмент коду та поясни, які стилі будуть застосовані і чому:

```
p { font-size: 18px; }
p { color: blue; }
p { font-size: 24px; }
```

14. Поясни, чому елемент може змінити розмір після заміни `width: 800px;` на `max-width: 800px; width: 100%;`.
15. Визнач, які елементи в коді є батьківськими, а які — дочірніми:

```
<section>
    <p>Текст <span>важливий</span></p>
</section>
```

16. Поясни, чому кнопка стає більшою після збільшення padding.

17. Створи вебсторінку із текстом:

```
Новина дня
Сьогодні відбулося відкриття нового IT-хабу.
Дізнатися більше
```

Перетворивміст сторінки сучасний міні-блок новини:

- заголовок;
- правильні відступи;
- текст;
- адаптивна ширина.
- слово «Дізнатися більше» оформлене як кнопка;

18. Міні-проєкт: «Моя цифрова візитівка».
Створи невелику сторінку про себе, яка містить:
    - заголовок;
    - короткий опис;
    - список інтересів;
    - кнопку або посилання;
    - адаптивну ширину тексту;
    - продумані відступи.

    Завдання — зробити сторінку не просто правильною, а зручною для читання.

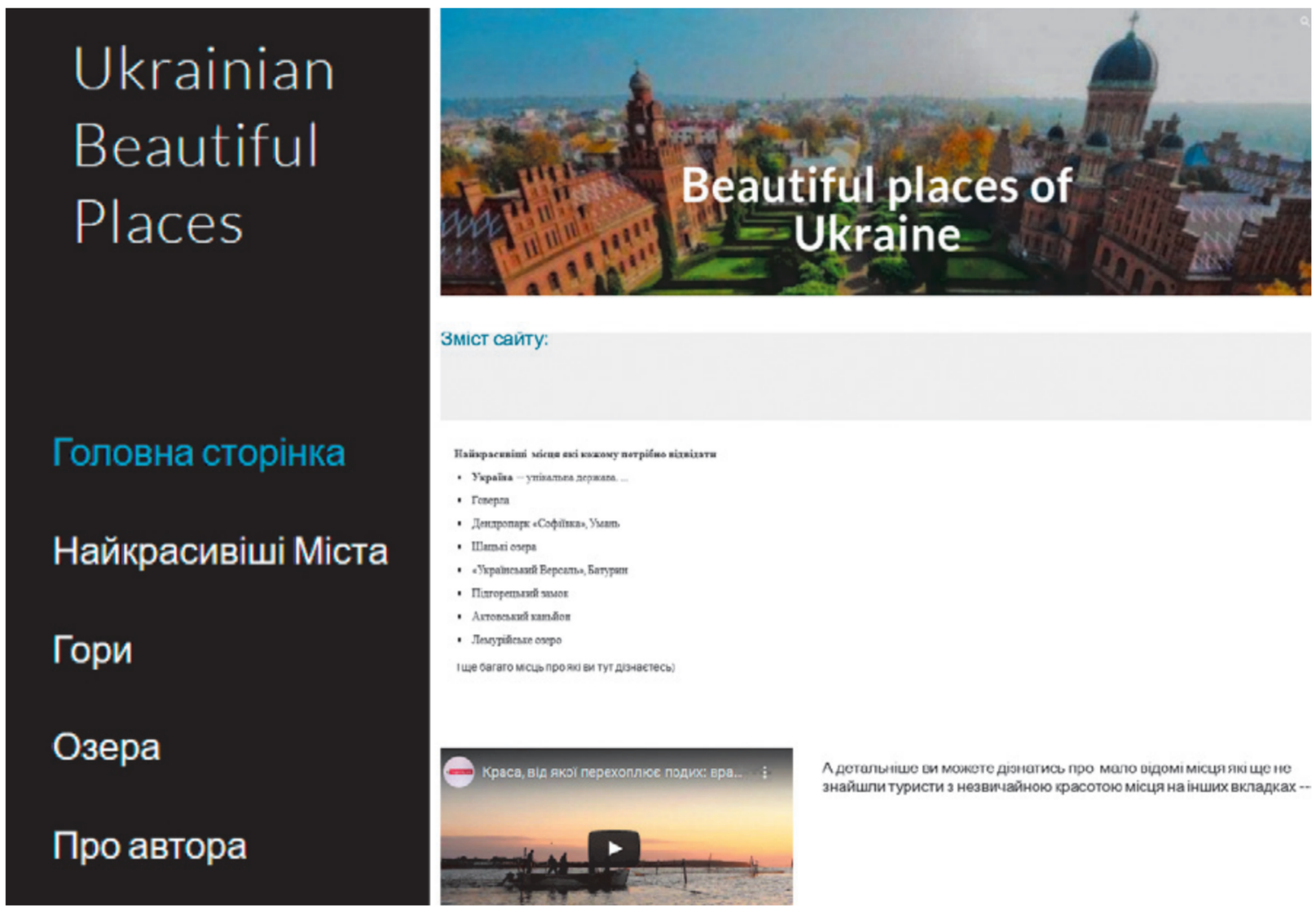
§ 19. Автоматизовані засоби для створення та публікації вебресурсів

Що мав на увазі Білл Гейтс?
«Якщо ваш бізнес не в інтернеті, то він скоро припинить своє існування» — Bill Gates

Ти вже вмієш створювати прості сайти за допомогою таких технологій, як HTML та CSS. Однак для чого потрібен сайт, який ніхто, крім автора, не побачить? А щоб на сайті з'явилися відвідувачі, його потрібно розмістити в інтернеті. Для цього призначено *служби хостингу*. Часто такі служби поєднано з автоматизованими засобами розробки сайтів — тоді їх називають ***конструкторами сайтів***. Серед популярних конструкторів сайтів є Google Sites — онлайн-сервіс від Google, який дозволяє швидко створювати і публікувати вебсторінки без програмування. Існують також інші сервіси для створення сайтів, наприклад Wix, Tilda або Webflow.

Завдання № 1

Проаналізуй зображений нижче сайт. З яких елементів він складається?



У наступних завданнях пропонуємо тобі розробити в службі Google Сайти свій персональний вебсайт. Можеш відобразити на ньому інформацію, яку ти вже надавав / надавала на сайтах, створених на попередніх уроках.

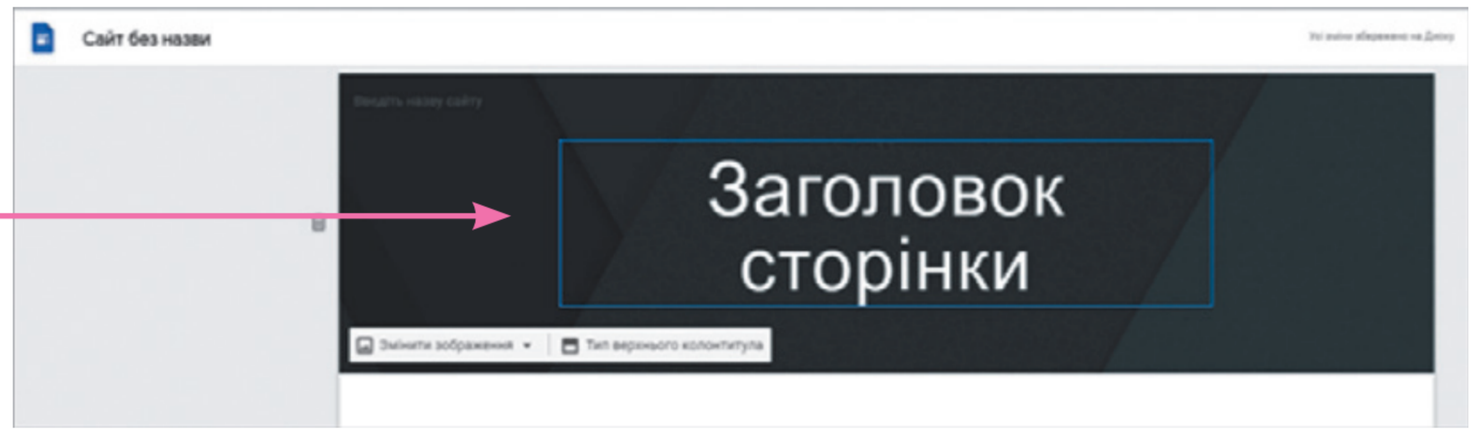
Завдання № 2

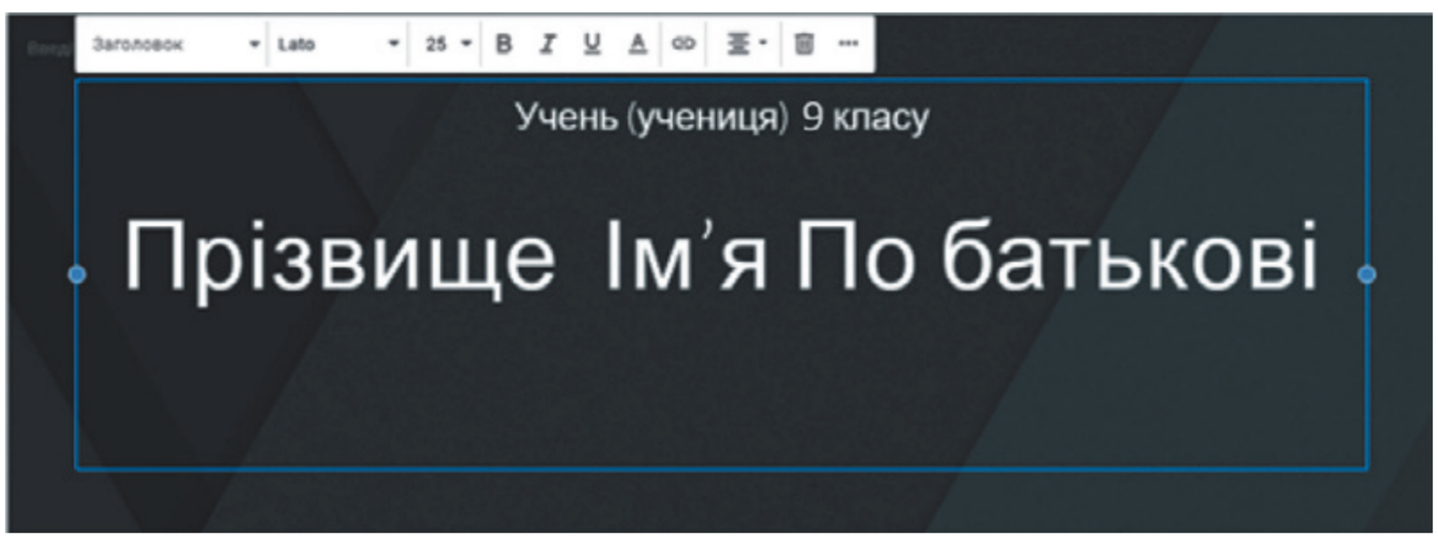
1. Користуючись рубрикою «Запитання-відповіді», довідкою Google та сервісом sites.google.com, створи сайт, у заголовку домашньої сторінки якого введи такі дані:

Учень (учениця)
9 класу

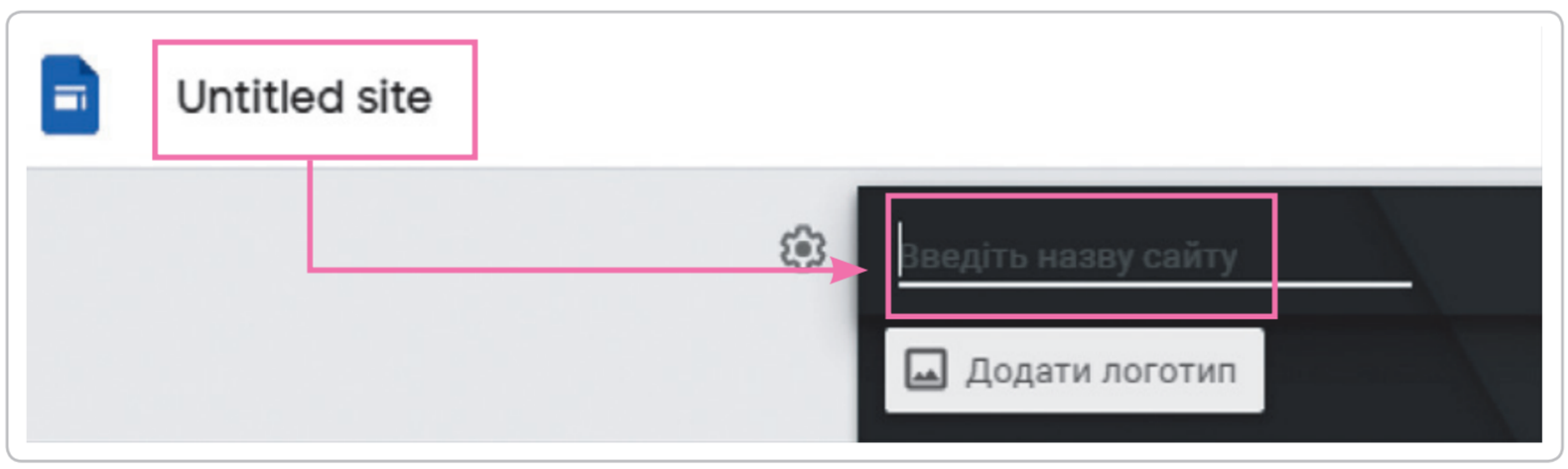
Прізвище Ім'я
По батькові

Можеш вибрати-
тематику сайту
на власний розсуд

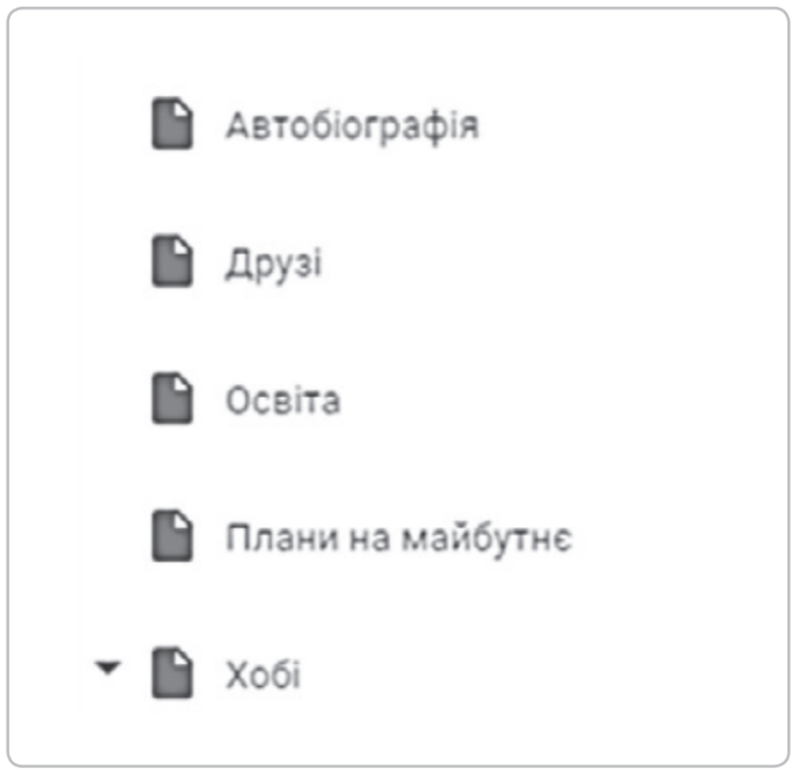




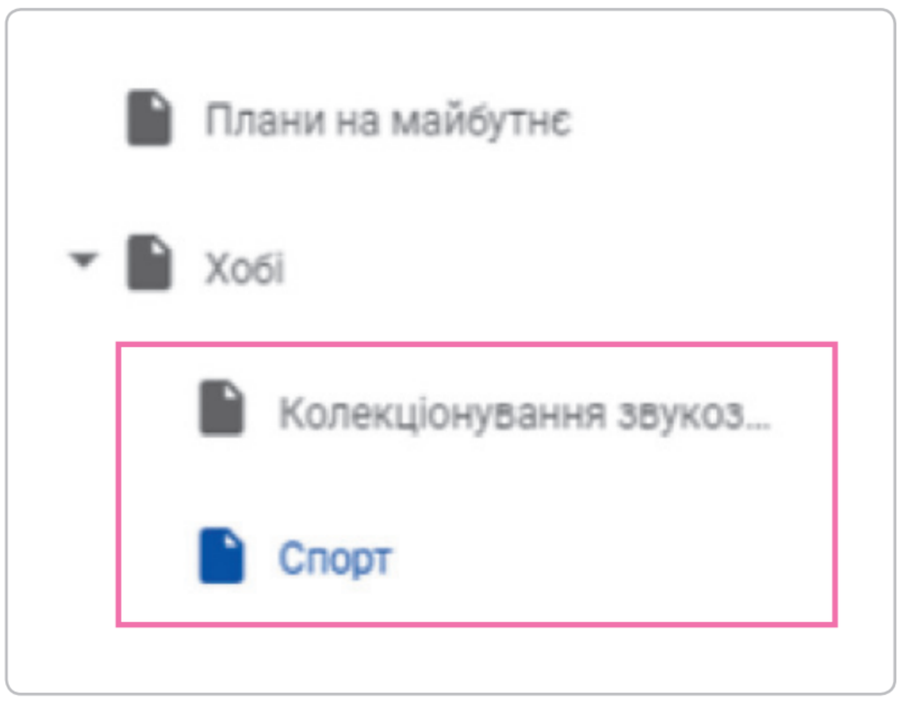
Самостійно введи назву сайту та його логотип:



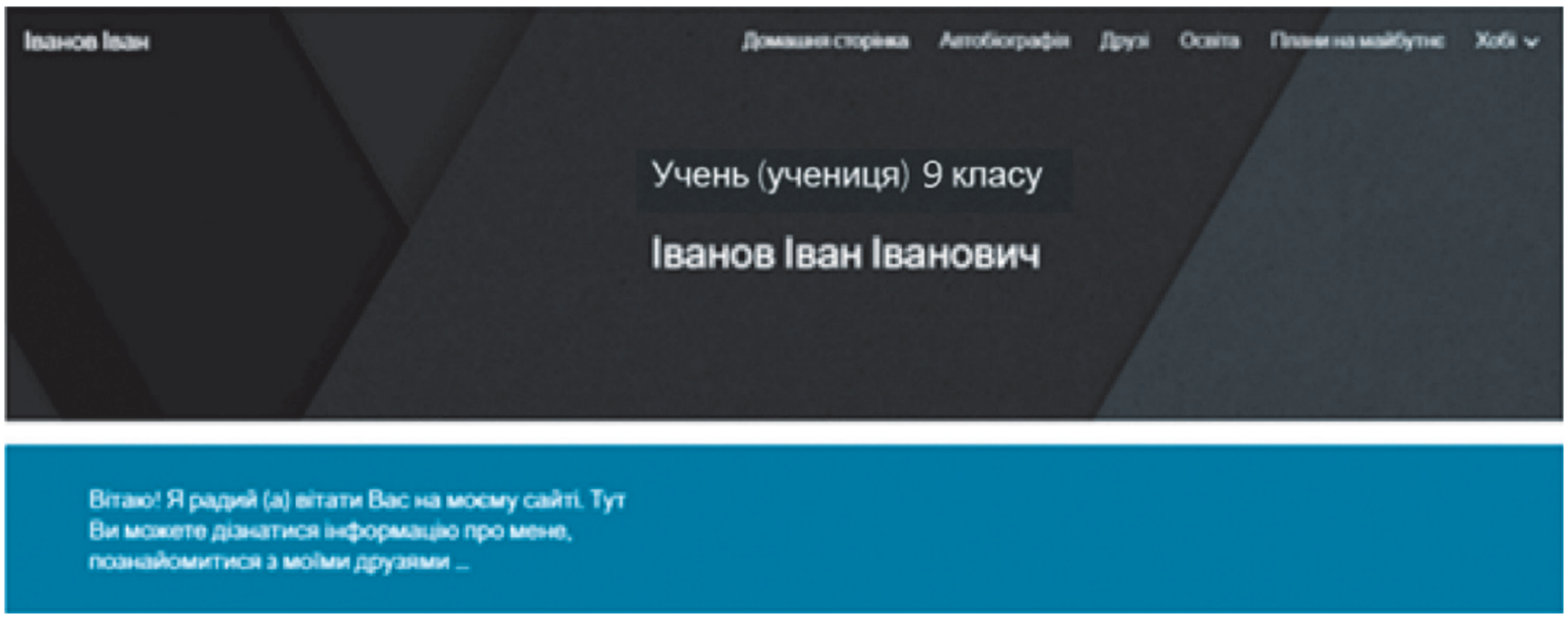
2. Додай до сайту такі сторінки:



3. До сторінки «Хобі» додай дві підсторінки, імена яких придумай самостійно:



4. Додай до домашньої сторінки текст привітання та відформатуй його.



5. Додай до домашньої сторінки ще одне текстове поле із заголовком «Що мене цікавить?». Створи у ньому нумерований список. До кожного елемента списку додай опис за взірцем.

Що мене цікавить?

1. Улюблений фільм

«Пірати Карибського моря» (англ. Pirates of the Caribbean) — пригодницька франшиза.

2. Улюблена книга

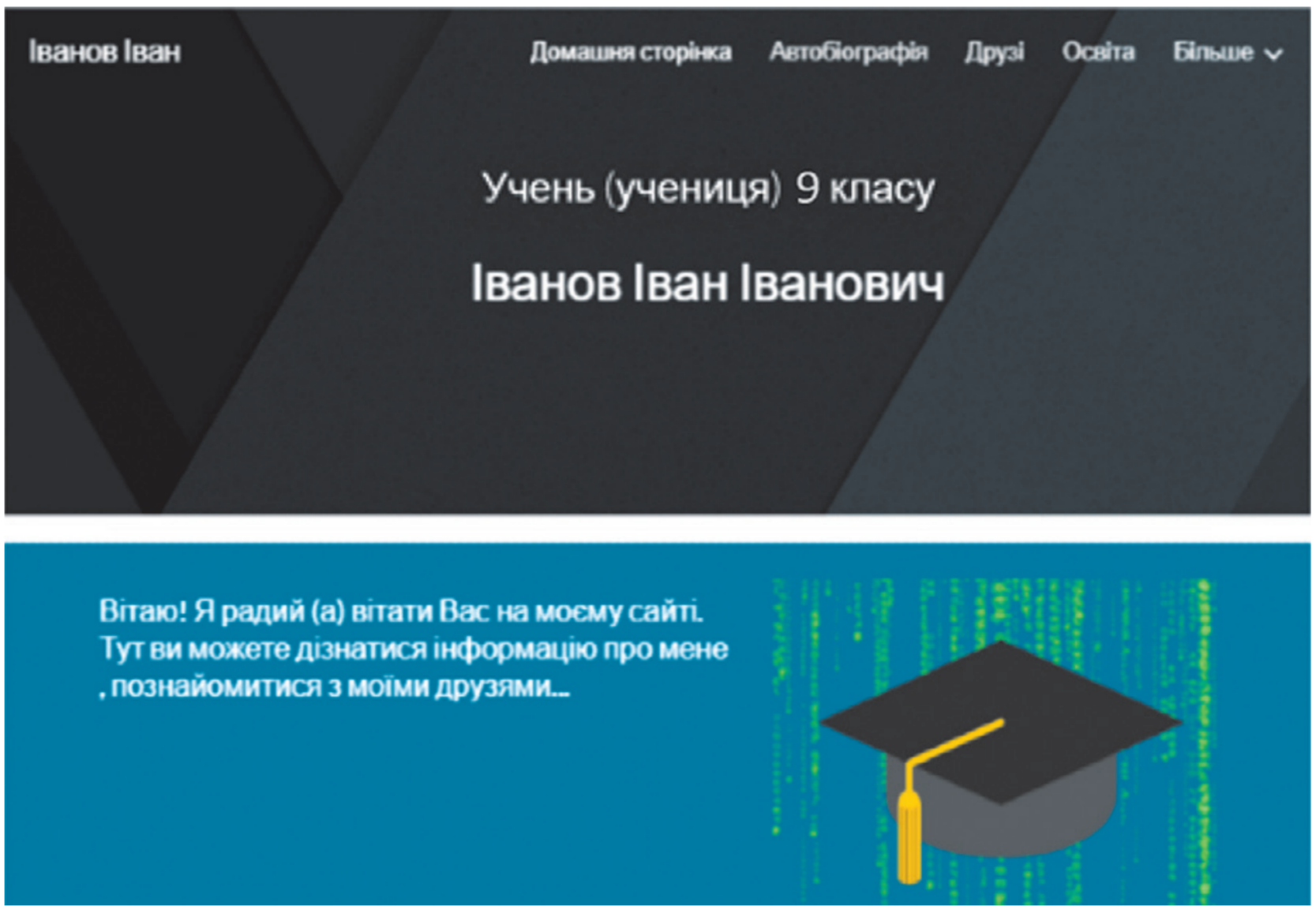
«Гаррі Поттер» (англ. Harry Potter) — серія з семи фантастичних романів, гепталогія («семикнижжя») англійської письменниці Джоан Роулінг.

3. Улюблена музика

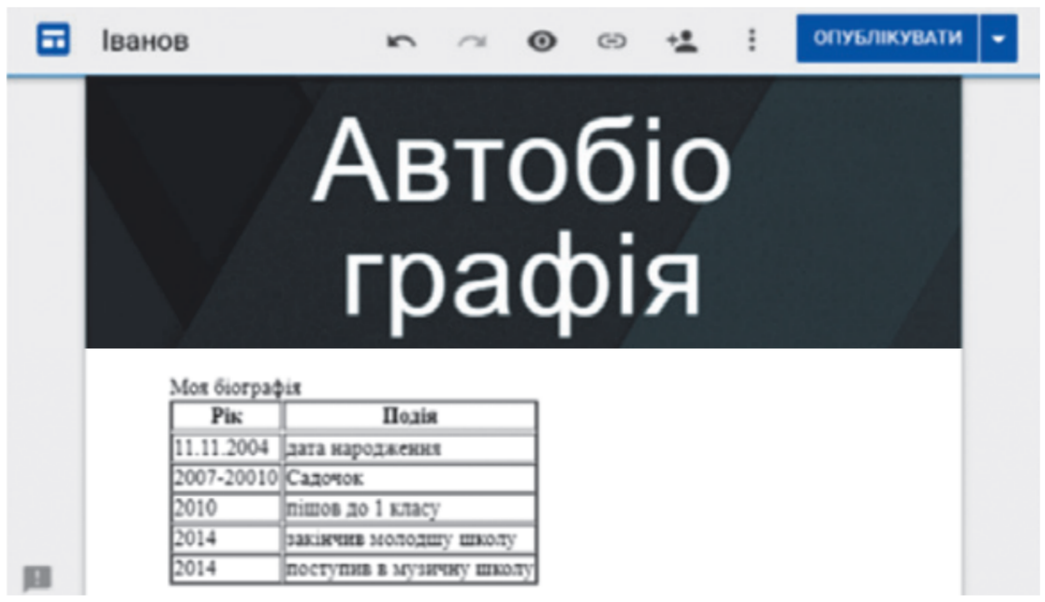
Джаз (англ. jazz) — вид музичного мистецтва, що виник на межі XIX—XX століття в США

6. Створи нижній колонтитул, у якому вкажи свою адресу електронної пошти та/або інші контактні дані.

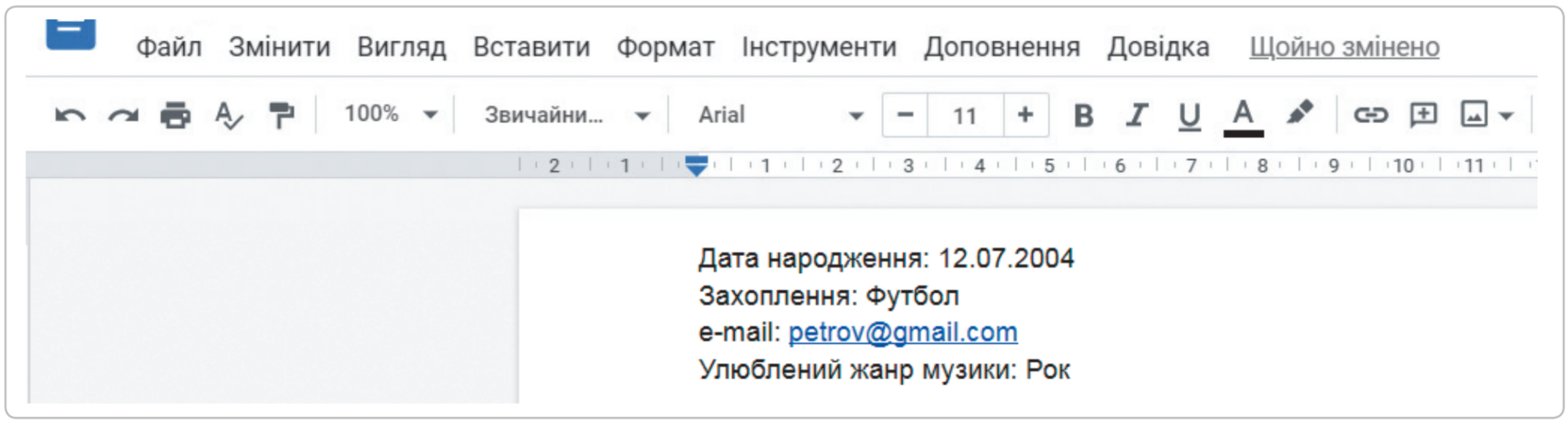
7. Додай до домашньої сторінки зображення відповідної тематики.



8. Додай до однієї зі своїх сторінок відомості, оформлені в таблиці. Це може бути автобіографія або інші впорядковані дані якоїсь однієї тематики.



9. Створи Google-документ з інформацією, що відповідає тематиці сайту, наприклад із відомостями про одного зі своїх друзів. Встав створений документ на відповідну сторінку.



10. На одній зі сторінок розмісти зображення з посиланням на іншу сторінку в інтернеті. Це може бути Google Карта з указаним на ній місцем, що стосується твого сайту, наприклад місцем твого народження чи навчання.

Завдання № 3

Опублікуй свій сайт в інтернеті, а його адресу вкажи у створеному вчителем або вчителькою спільному документі. Відвідай сайти, адреси яких у цьому документі вказали твої однокласники та однокласниці.

Завдання № 4

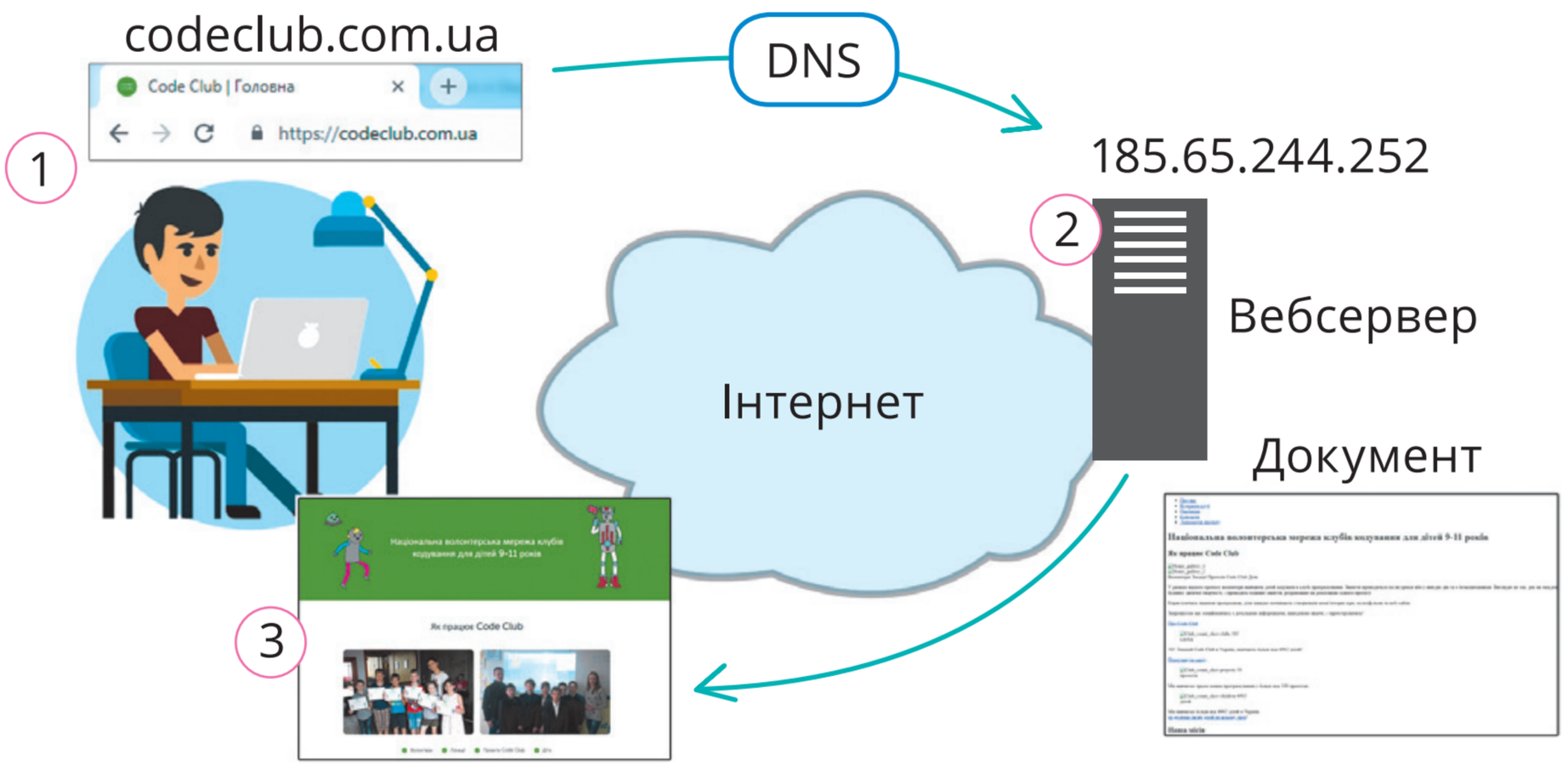
Знайди в інтернеті інші конструктори сайтів, наприклад:
Webnode, Wix, Weebly, Ucraft, Webflow, SITE123, GoDaddy, Jimdo, Strikingly

Вибери два з них та спробуй створити за їх допомогою односторінковий сайт. Який із трьох опрацьованих на уроці конструкторів сайтів тобі сподобався найбільше? Чому?

ЗАПИТАННЯ-ВІДПОВІДІ

Як взаємодіють браузер і вебсервер?

Користувач вводить адресу вебсторінки в рядок браузера (1), а спеціальна служба інтернету DNS (Domain Name System) перетворює адресу зі зручної для користувача форми на цифрову, що називається *IP-адресою* сайту (від англ. Internet Protocol). Потім запит потрапляє на відповідний вебсервер (2), який розпізнає IP-адресу і видає потрібний HTML-документ. Браузер користувача відображає документ у зручному для сприйняття вигляді (3).



Які вебсайти називають статичними, а які — динамічними?

Вміст статичного вебсайту створюється розробником і не змінюється під час перегляду сторінки. Кожна сторінка містить фіксований HTML-код.

Вміст динамічного вебсайту може змінюватися залежно від дій користувача або даних, що надходять із сервера. Наприклад, після введення даних у форму, натискання кнопки або завантаження нової інформації.

Існує два основні способи створення динамічних вебсторінок:

- За допомогою JavaScript, який виконується в браузері користувача. Такий підхід дозволяє змінювати вміст сторінки без її перезавантаження.
- За допомогою серверних програм, які формують сторінку перед надсиланням її в браузер. Для цього використовують спеціальні мови програмування, наприклад PHP, Python або JavaScript (Node.js). У сучасних сайтах обмін даними з сервером часто відбувається без повного перезавантаження сторінки.

Що таке хостинг?

Хостинг — послуга з розміщення даних на серверах. Будь-які сервери, де розміщуються вебсторінки, поштові скриньки тощо, обслуговуються хостинг-провайдерами.

Що таке служба Google Sites?

Google Sites (Сайти Google) — це безкоштовний онлайн-конструктор сайтів із вбудованим хостингом. Він дозволяє створювати прості вебсайти без знання HTML і CSS.

Створений сайт автоматично розміщується на серверах Google та отримує адресу виду:

https://sites.google.com/view/назва-сайту

де «назва-сайту» — це ім'я, яке задає користувач під час створення.

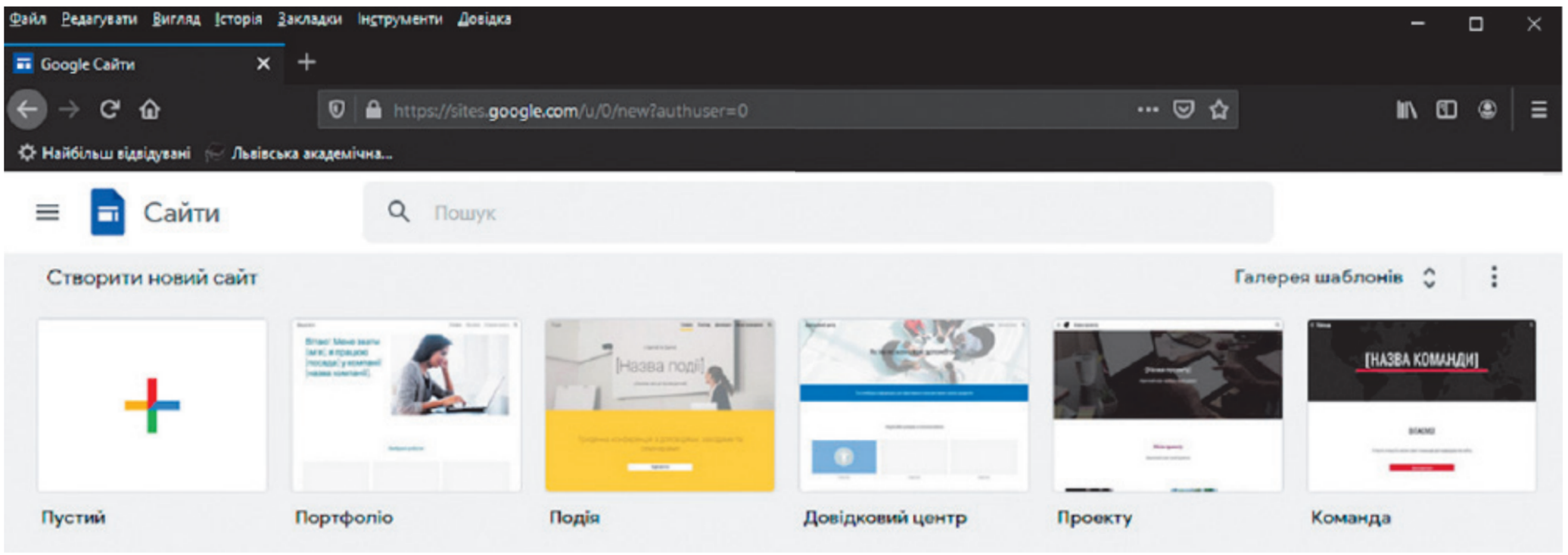
Google Sites інтегрується з іншими сервісами Google: на сторінку можна легко додавати файли з Google Диска, документи, презентації, таблиці, відео з YouTube, форми для опитування тощо. Це робить сервіс зручним для навчальних проєктів і командної роботи.

Які основні можливості Сайтів Google?

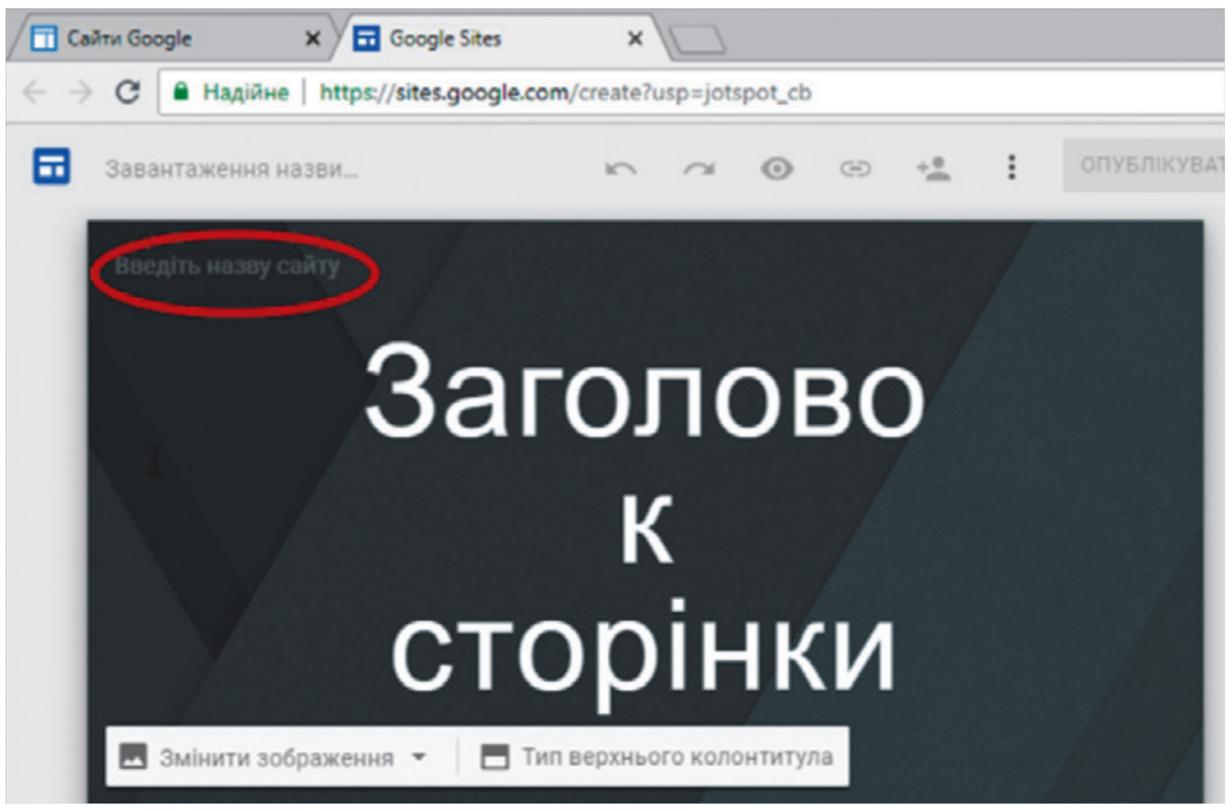
- Створення сайту без знання мов програмування та розмітки;
- додавання на сторінку інформації з інших сервісів Google;
- прикріплення до сторінок файлів, які можна завантажувати;
- розмежування прав відвідувачів на зміну або читання інформації на сайті;
- наявність колекції стилізованих шаблонів;
- публікація сайту в інтернеті без сторонньої реклами.

Як створити сайт?

Увійшовши у свій обліковий запис Google та відкривши службу Сайти Google, у розділі «Створити новий сайт» потрібно клацнути один із запропонованих тематичних шаблонів. Відкриється заготовка нового сайту, що нагадує слайд у Презентаціях Google.

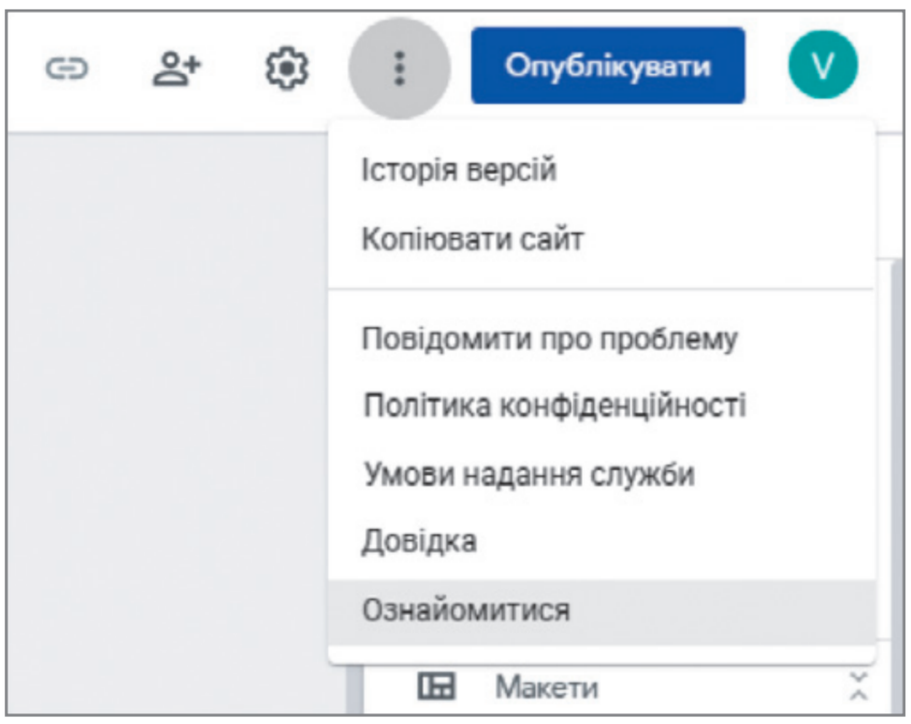


У лівому верхньому куті потрібно ввести назву сайту.

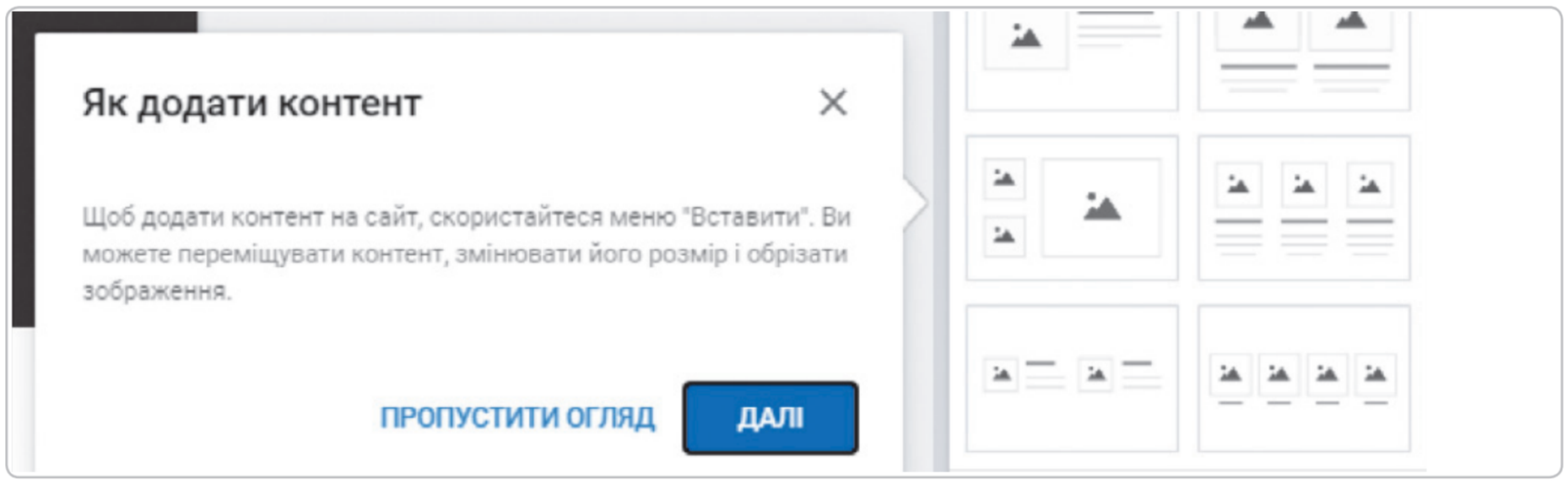


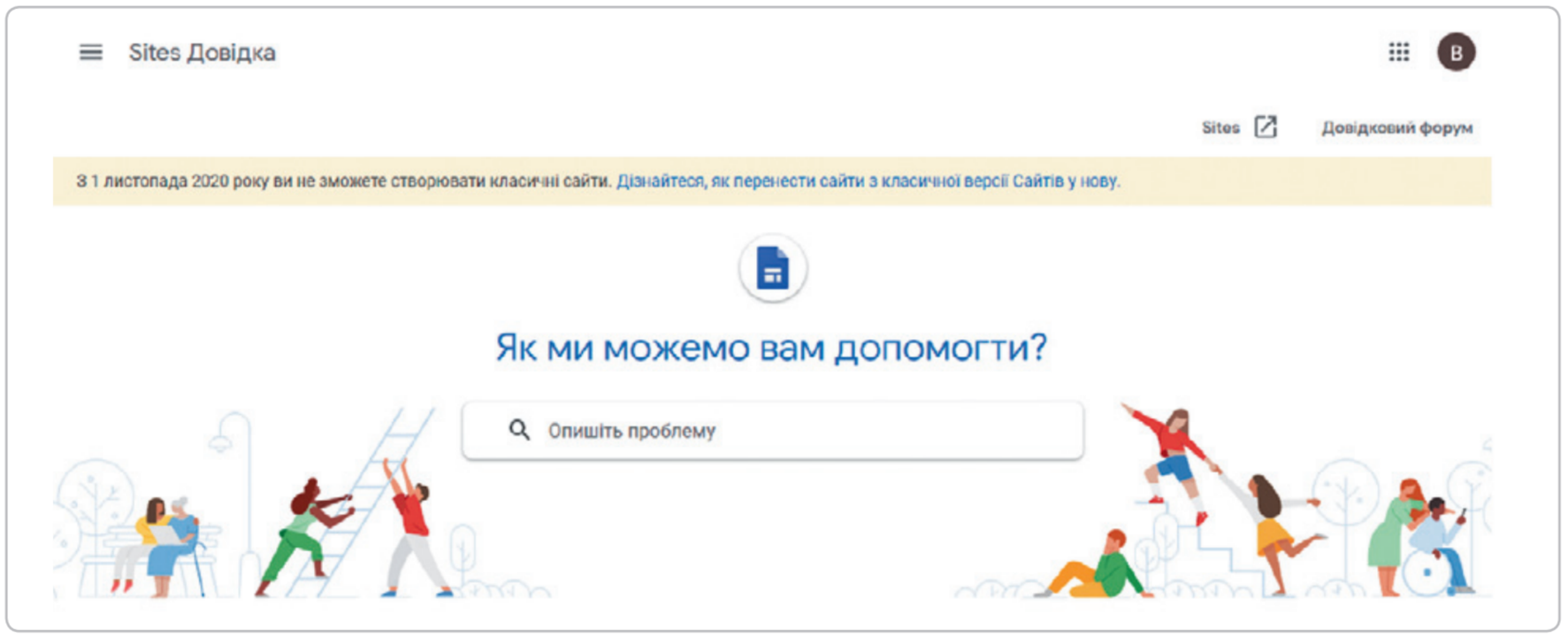
Як відкрити довідку з користування Сайтами Google?

У вікні конструктора сайту потрібно натиснути кнопку ⋮ та обрати «Довідка» або «Ознайомитися».



Після вибору пункту «Ознайомитися» відкриється інтерактивний огляд, а після вибору пункту «Довідка» — інтерактивна довідка, у якій ти можеш вибрати тематичний розділ або ввести пошуковий запит.

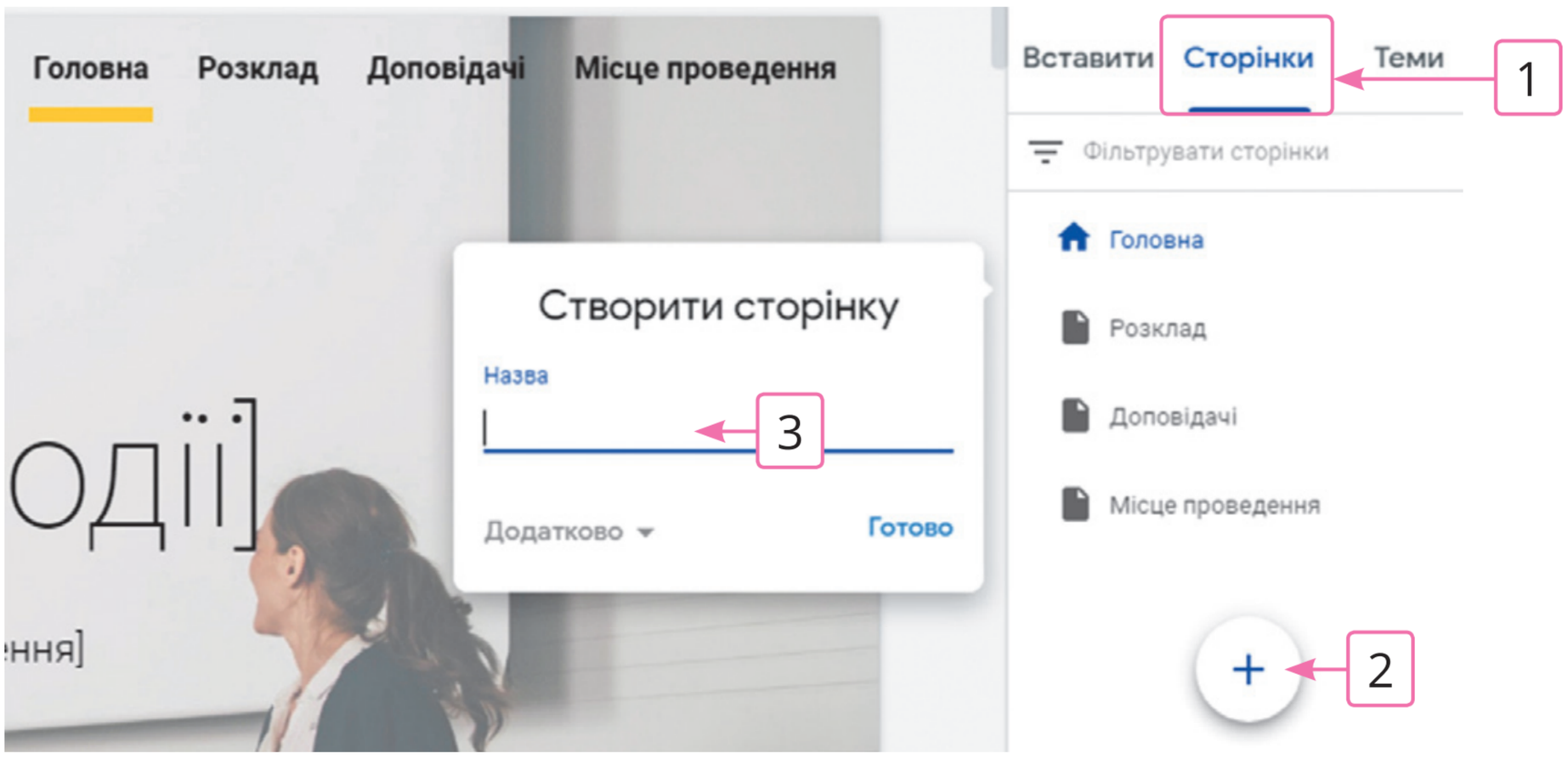




Як додати сторінку та підсторінку?

Щоб додати сторінку, виконай такі дії.

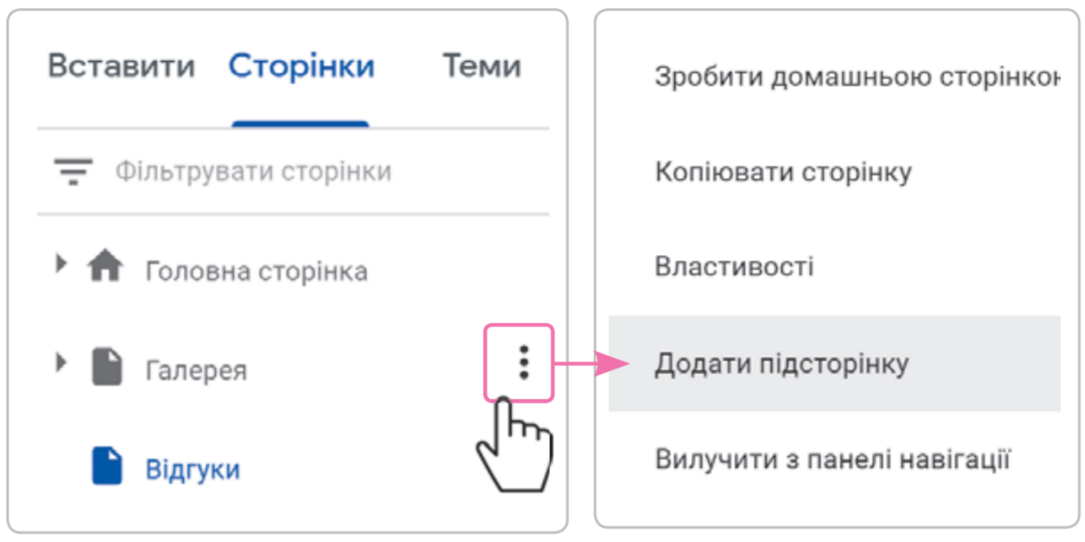
1) Відкрий вкладку «Сторінки».
2) Внизу сторінки клацни **+** (Створити нову сторінку).
3) У вікні «Нова сторінка» введи назву сторінки.

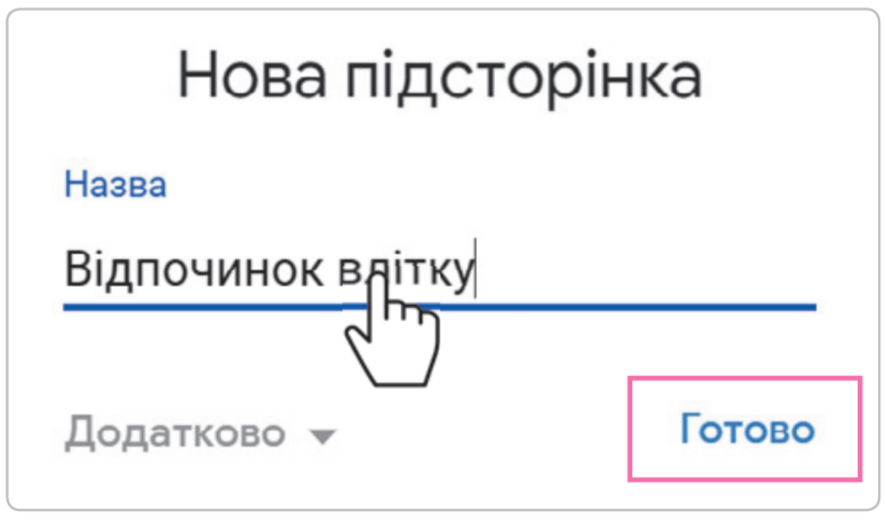


Щоб додати підсторінку, потрібно:

1) У списку «Сторінки» вибрати потрібну сторінку, справа від її назви натиснути кнопку та вибрати пункт «Додати підсторінку».

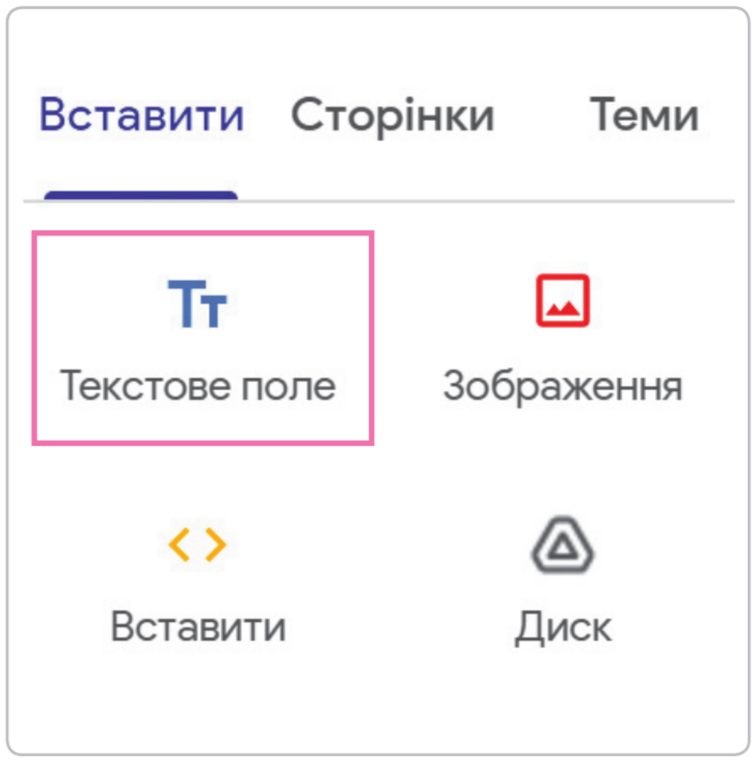
2) У вікні «Нова підсторінка» потрібно ввести назву підсторінки та натиснути кнопку «Готово».





Як додати до сторінки текстове поле?

Щоб додати текст на сторінку, потрібно відкрити меню «Вставити» та вибрати пункт «Текстове поле».



Як змінити стиль сайту?

У межах кожної теми в службі Сайти Google є декілька стилів, що визначають колір тла сторінок, колір та розмір шрифтів основного тексту та заголовків тощо.

Щоб вибрати стиль, перейди на вкладку «Теми» — і побачиш стилізовані шаблони, як у редакторі презентацій. Вибери бажаний — і оформлення всіх сторінок сайту буде змінено. За допомогою круглих різнокольорових кнопок можеш вибрати колір тла серед кількох стандартних. Контрастний колір тексту буде встановлено автоматично.

За допомогою останньої круглої кнопки можна вибрати будь-який колір, а не лише запропоновані в темі.

Увага!

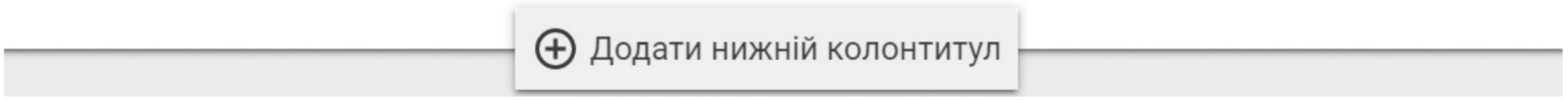
За допомогою мови HTML можна легко змінити колір будь-якого фрагмента тексту.

```
<font color = "red">ivanov@gmail.com</font>
```

ivanov@gmail.com

Як додати колонтитул?

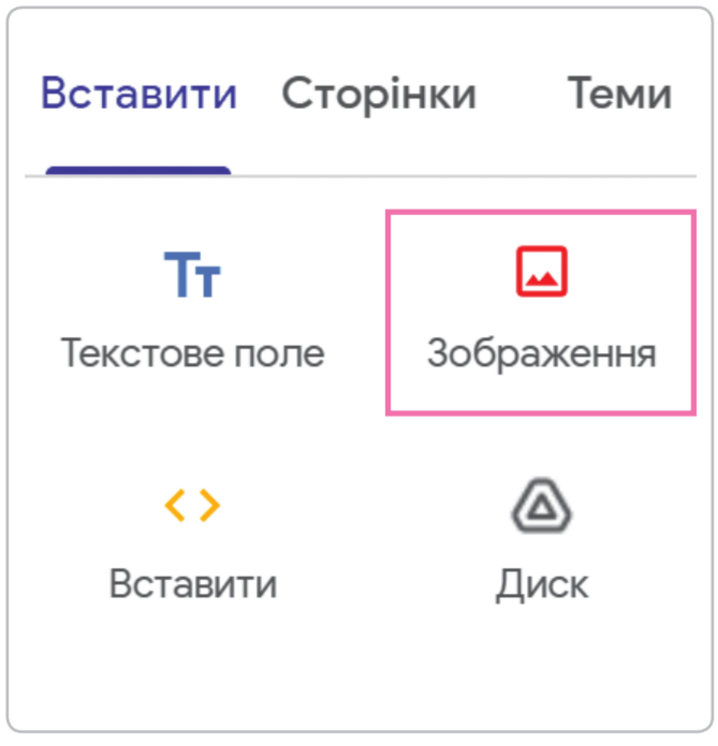
Як і в текстовому процесорі, колонтитул — це поле, що відображається на всіх сторінках. Нижній колонтитул створюється за допомогою кнопки «Додати нижній колонтитул», яка відображається, якщо підвести курсор до нижнього краю сторінки.



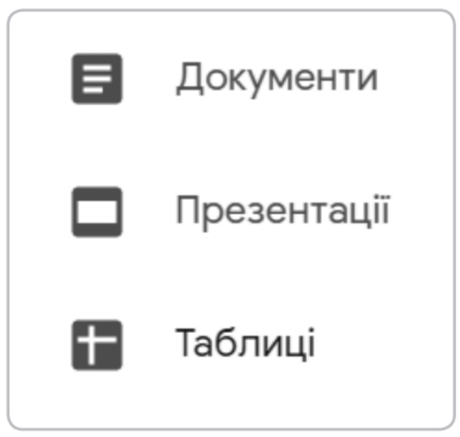
Як додати зображення?

Щоб додати на сторінку малюнок, потрібно в меню «Вставити» вибрати пункт «Зображення» або «Додати».

Кнопка «Зображення» призначена для вставлення зображення з інтернету, а кнопка «Додати» — для зображень, які збережені на ПК. Поряд зі вставленим зображенням з'являються кнопки для його форматування (обтинання, введення підпису, заміна).



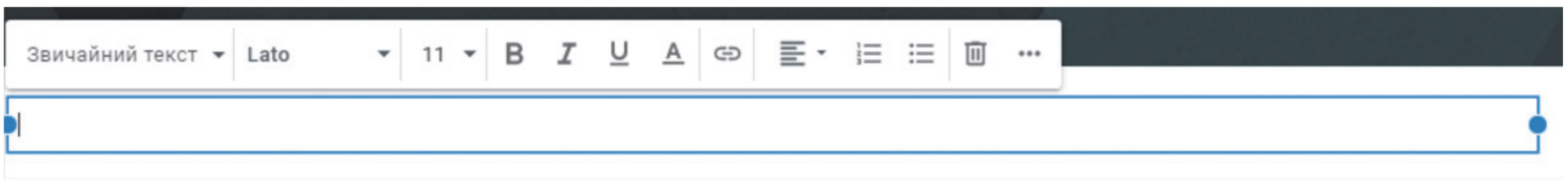
Як відобразити на сторінці документ чи таблицю Google?



Прокрути вкладку «Вставити» вниз, вибери потрібний тип документа, потім обери сам документ і натисни кнопку «Вставити».

Як на своєму сайті розмістити посилання на інший сайт?

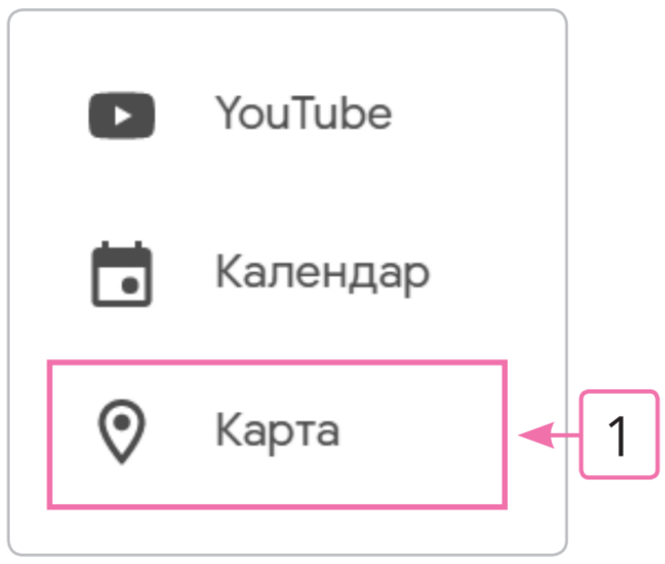
Щоб вставити посилання, у текстовому полі натисни кнопку «Вставити посилання» і введи саме посилання в поле, що з'явиться.



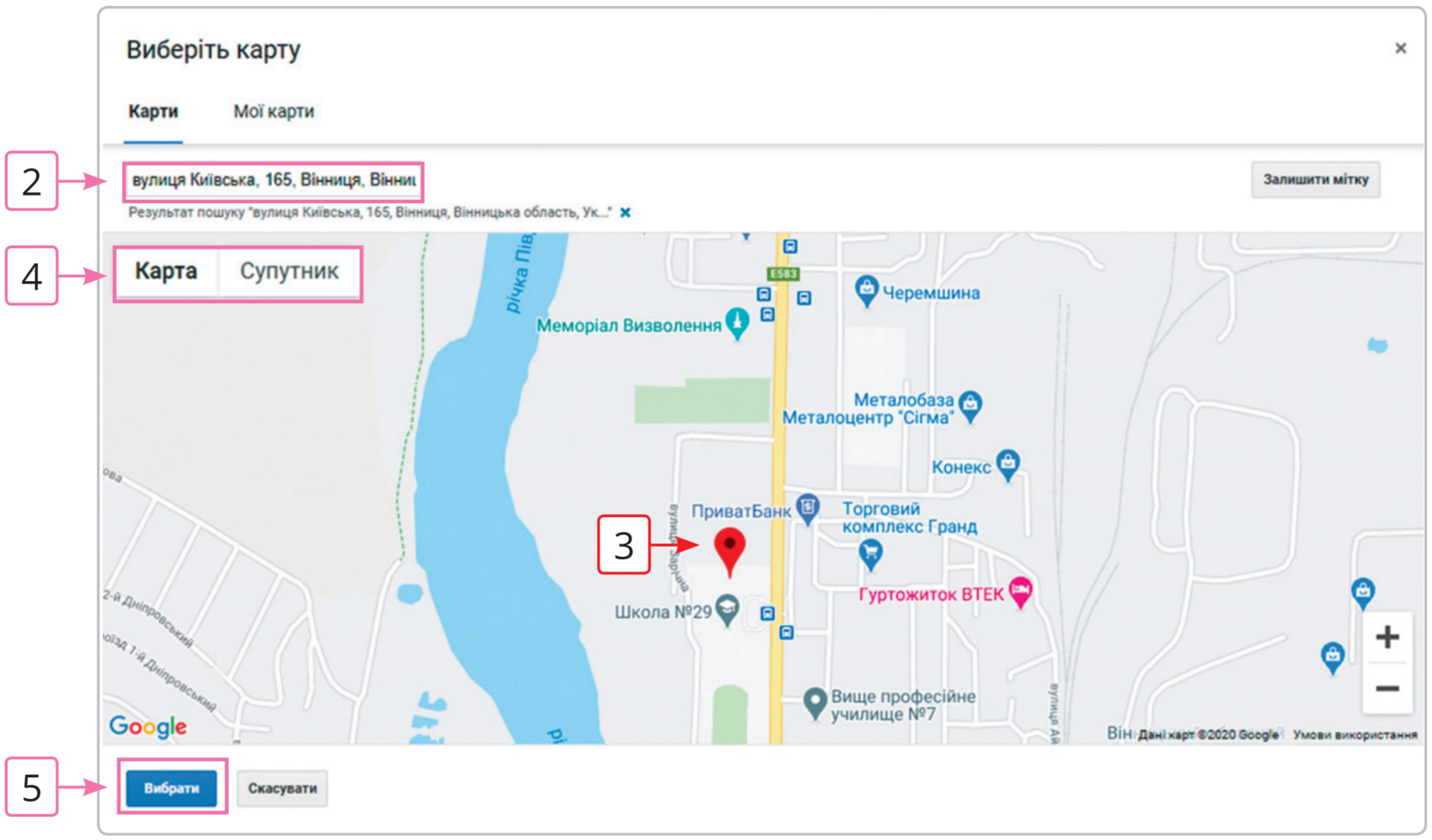
Як додати на сторінку карту?

Карту, як правило, додають, щоб користувачі сайту могли дізнатися точне місце розташування певного об'єкта.

1) На вкладці «Вставити» вибери інструмент «Карта».



2) У вікні «Виберіть карту» у відповідному полі введи назву місця чи об'єкта.
3) Відкоригуй точне місце розташування, перемістивши позначку.
4) Вибери тип зображення на карті — фотографія із супутника чи власне карта.
5) Натисни кнопку «Вибрати».

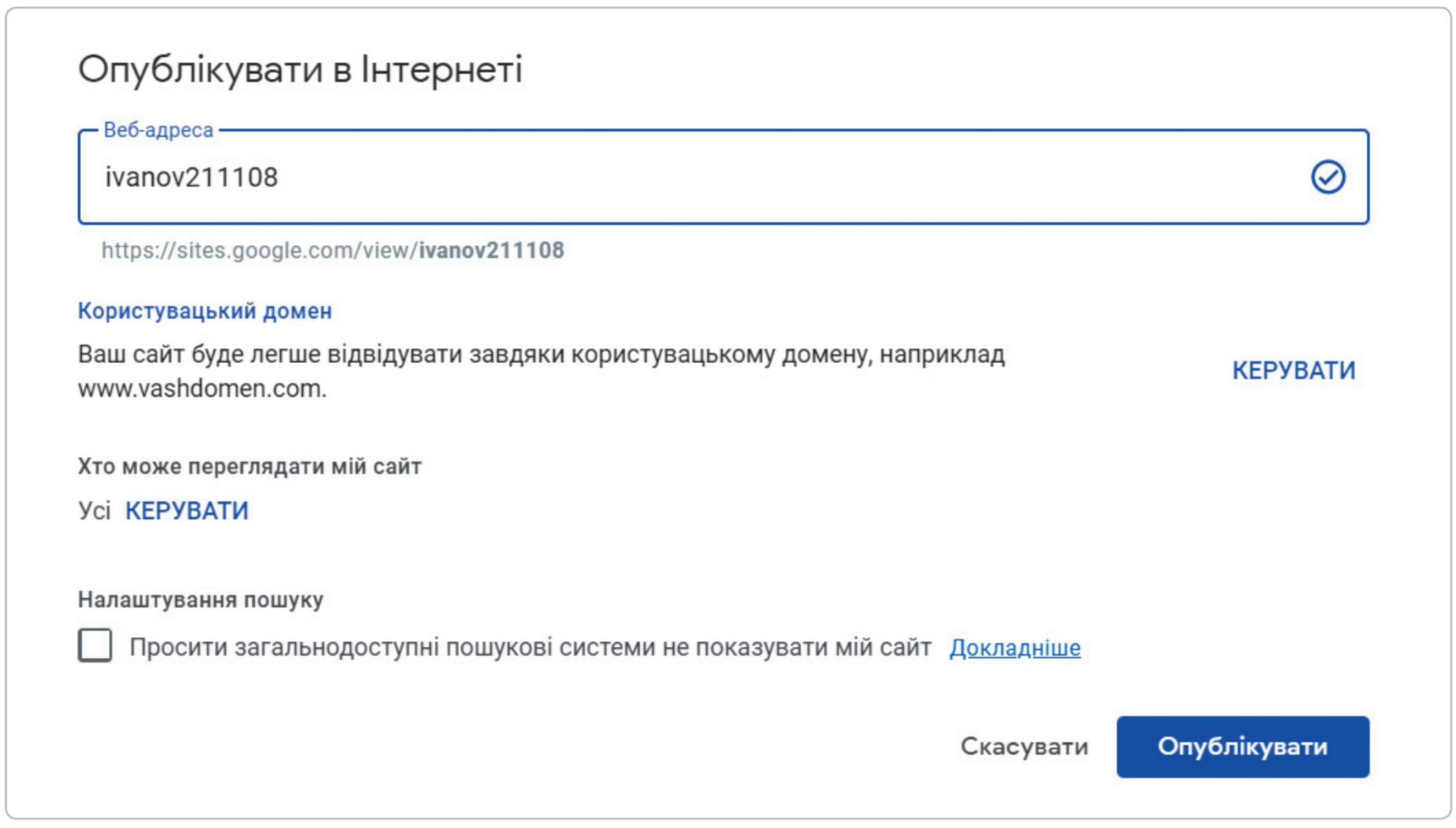


Як опублікувати сайт?

Сайтам, створеним у службі Сайти Google, надаються імена, що мають такий вигляд: **sites.google.com/view/ім'я** сайту, де ***ім'я сайту*** користувач вигадує самостійно.

Щоб зробити сайт доступним з інтернету, виконай такі дії:

1) Натисни кнопку Опублікувати.
2) У полі введи ім'я свого сайту. Це може бути будь-яке слово, що містить малі латинські літери та цифри.
3) Вкажи, хто може переглядати сайт.
4) Натисни ще одну кнопку «Опублікувати».



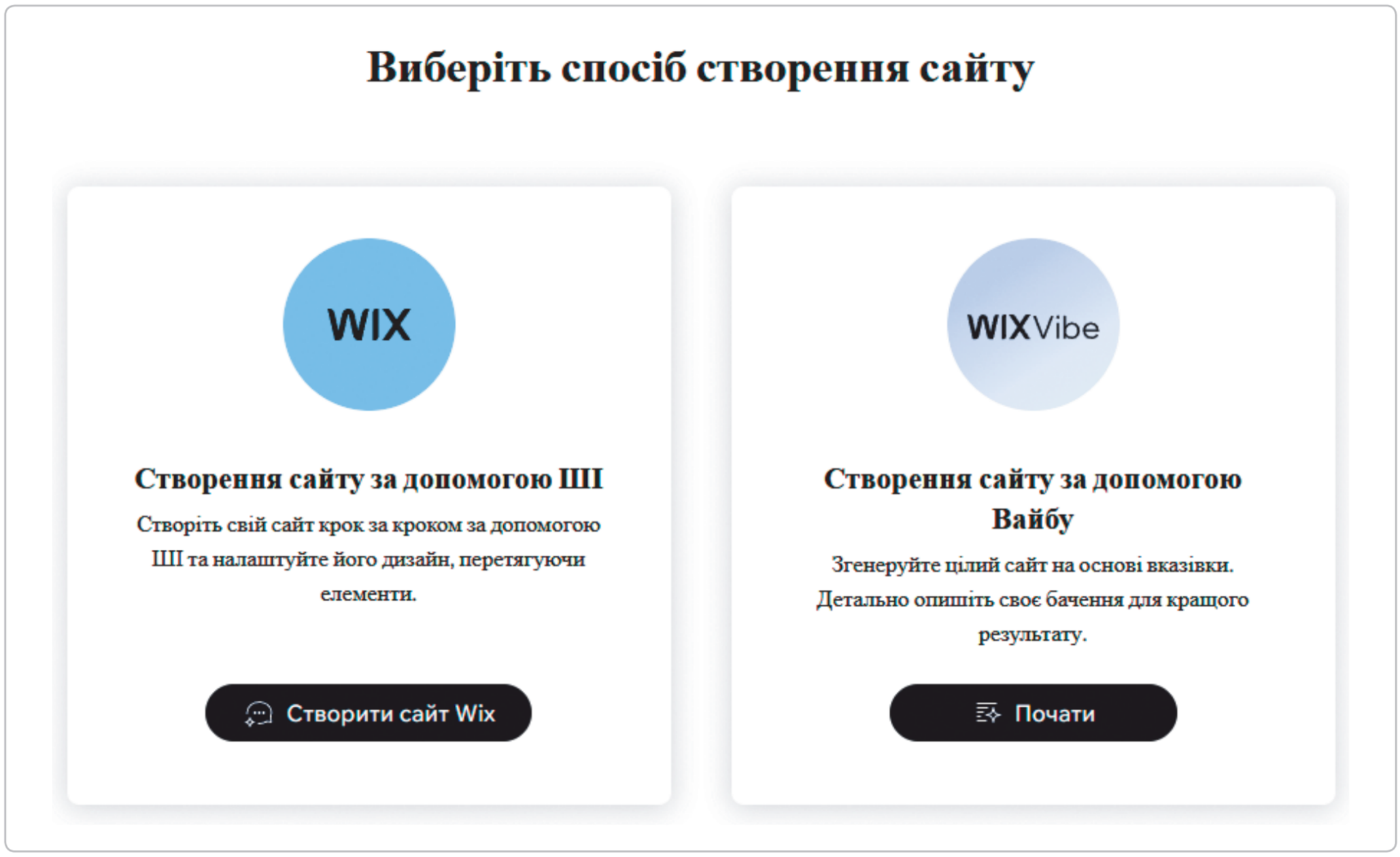
Увага!

Імена сайтів в усьому інтернеті мають бути унікальними. Якщо ти введеш просте ім'я сайту, то, ймовірно, воно вже буде кимось зайняте. Тоді його потрібно ускладнити, додавши, наприклад, якесь число в кінці.

Що таке платформа WIX і як скористатися засобами ШІ на ній?

WIX — це більш потужна, ніж Сайти Google, платформа для хостингу та автоматизованого створення сайтів. Зокрема, щоб створити сайт, ти можеш скористатися асистентом на основі штучного інтелекту, що відкривається за адресою https://www.wix.com/ai-website-builder. Пропонується два режими:

- WIX — майстер, що створює вебсайт на основі твоїх відповідей на серію запитань;
- WIXVibe — генератор на основі ШІ, що створює вебсайт за текстовим описом.



Увага!

Щоб почати працювати з WIX, потрібно створити свій акаунт на платформі (можна ввійти через акаунт Google). Також рекомендуємо вибрати на початковому екрані українську мову інтерфейсу за допомогою значка .

ПЕРЕВІР СЕБЕ

ЗНАННЯ — СТВОРЕННЯ — ОЦІНЮВАННЯ

1. Чи підтримують Google Сайти технології HTML та CSS?
2. Дані яких типів можна розміщувати на сторінках сайту Google?
3. Сайти якої тематичної спрямованості неможливо створити у службі Google Сайти?
4. Створи сайт з інформацією про свої вподобання за допомогою штучного інтелекту. Для цього скористайся двома інструментами, наприклад Чатом GPT (https://chatgpt.com/) та штучним інтелектом на платформі WIX (https://www.wix.com/ai-website-builder). Про конструктор Wix можеш дізнатися з рубрики «Запитання-відповіді».
5. Порівняй два способи створення сайтів, які ми вивчали: безпосередньо за допомогою HTML та CSS, а також у конструкторі сайтів. Назви переваги та недоліки кожного з них.

РОЗДІЛ V

3D-ГРАФІКА

§ 20. Тривимірна графіка. Програми для роботи з тривимірною графікою. Основи створення тривимірної моделі

§ 21. Екструдування об’єктів

§ 22. Змінення форми об’єкта

§ 23. Створення анімації в середовищі Blender

Додаток. Створення 3D-моделей за допомогою генеративного ШІ

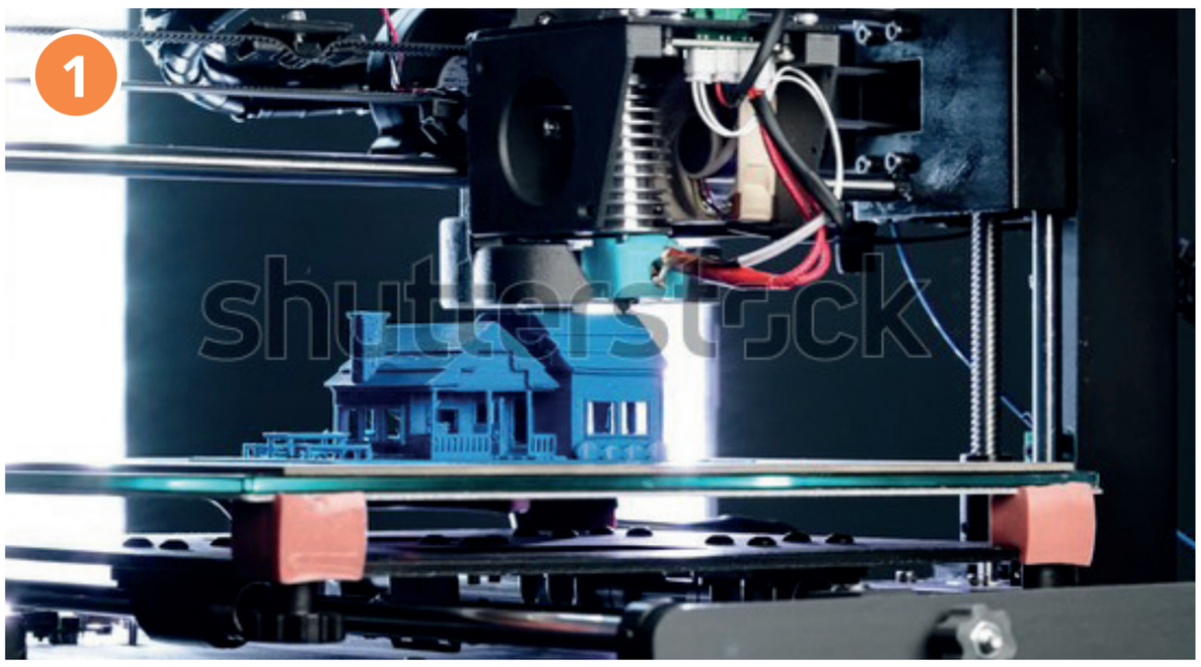
§ 20. Тривимірна графіка. Програми для роботи з тривимірною графікою. Основи створення тривимірної моделі

Як створити об’ємне зображення?

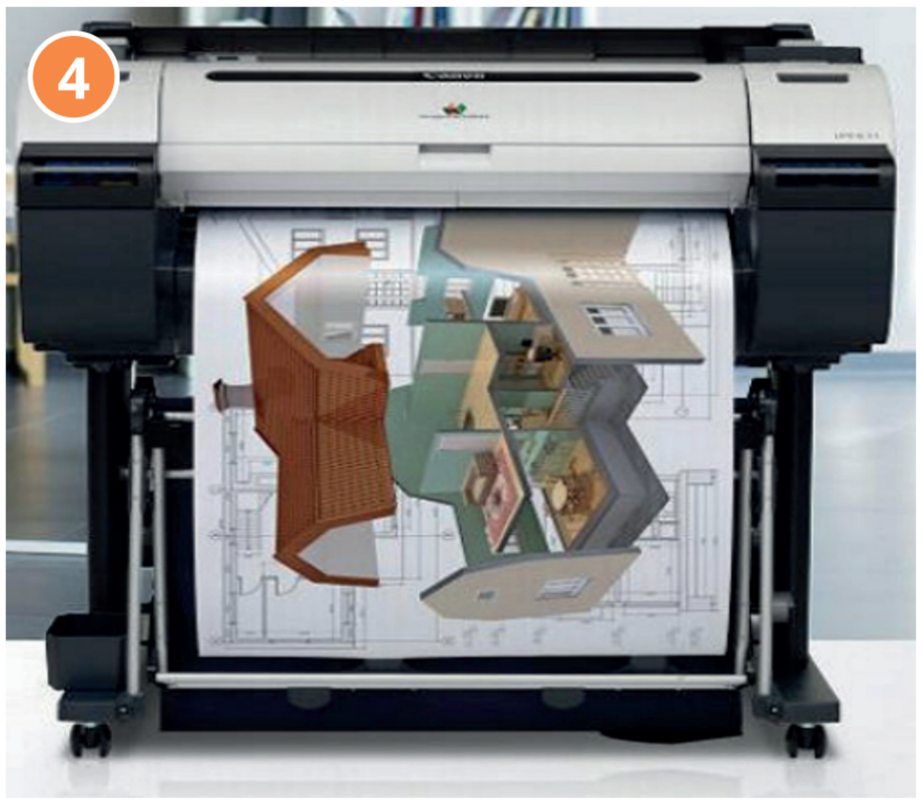
Завдання № 1

Розглянь світлини. Порівняй процеси, які на них зображені.

1. Об’єднай світлини у дві групи та поясни принцип поділу.
2. Визнач, на яких світлинах зображено:
   - інформаційну модель;
   - матеріальну модель;
   - реальний об’єкт (оригінал).
3. Поясни, чим відрізняється 2D-друк від 3D-друку.
4. Встанови послідовність процесів: від виникнення ідеї до створення реального будинку.



vse.ee/cxfj
е-додаток 20.1



Завдання № 2

Ознайомся з інформацією. Поміркуй, що тобі відомо на задану тему і чим ти можеш її доповнити.

Світ тривимірної графіки — це простір, у якому ідеї перетворюються на об'ємні об'єкти. Фахівці з 3D-моделювання створюють проєкти мостів і будівель, транспорт майбутнього, персонажів для фільмів і відеоігор, а також віртуальні світи для VR і метавсесвітів. Вони працюють на межі мистецтва й технологій, поєднуючи інженерну точність із творчою уявою.

3D-моделювальник працює в команді з художниками текстур, аніматорами, геймдизайнерами, архітекторами та програмістами, відповідаючи за свій етап створення цифрового продукту. Саме така співпраця дає змогу створювати фільми з вражаючими спецефектами, ігри з реалістичними персонажами та сучасні архітектурні проєкти.

Найбільші світові компанії — Meta, Google, Amazon, Netflix — інвестують мільярди доларів у розвиток віртуальної та доповненої реальності, 3D-контенту та цифрових технологій. Це означає, що попит на фахівців із тривимірної графіки постійно зростає.

- Об'єднайтесь у пари і складіть список мультфільмів та фільмів, у яких використано 3D-зображення.
- Поясни, як 3D-графіка використовується у галузях людської діяльності:
  - Міське планування
  - Промисловість
  - Комп'ютерні ігри
  - Кінематограф
  - Архітектура та дизайн інтер'єрів
  - Медицина

Більше в е-додатку 20.2.

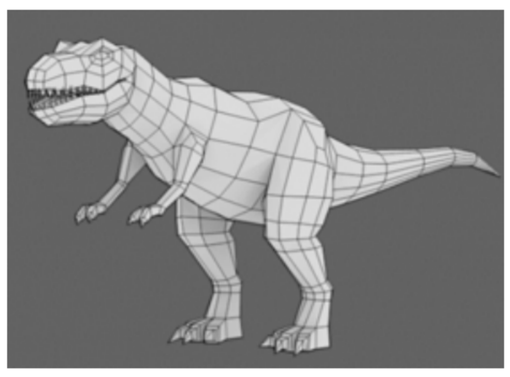
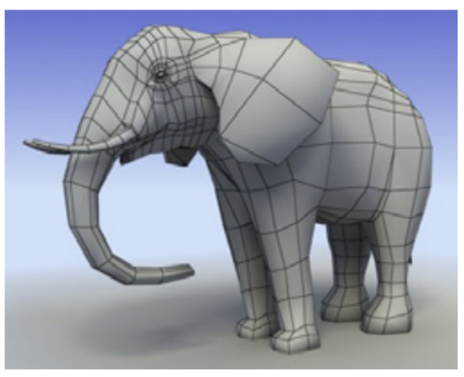
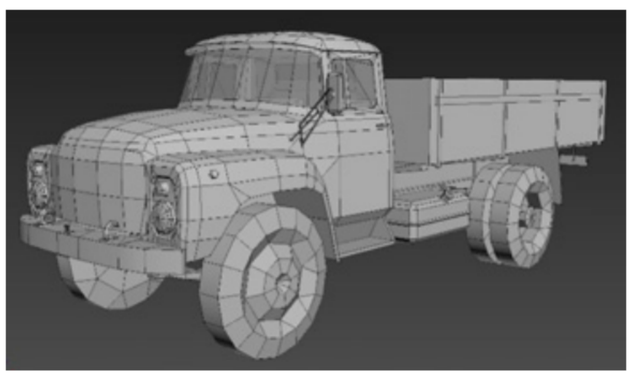
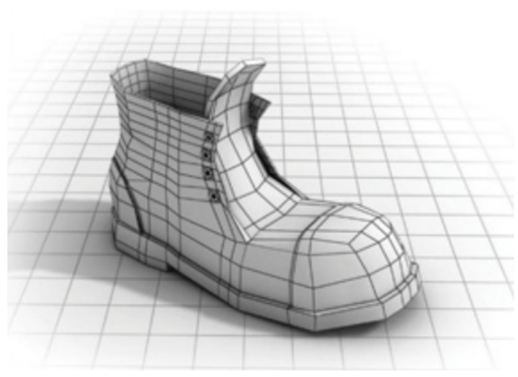
ТЕХНОЛОГІЇ БЕЗ МЕЖ

Сучасний цифровий світ дарує унікальні можливості для самореалізації кожному, незалежно від стану здоров'я чи фізичних обмежень. Професії програміста, 3D-художника або 3D-модельєра є одними з найбільш безбар'єрних, адже для створення складних баз даних, написання коду чи побудови реаліс-

тичних тривимірних моделей потрібні передусім інтелект, креативність та володіння інструментами, а не фізична сила. Соціалізація людей з інвалідністю через такі професії є критично важливою для суспільства: вона дозволяє залучити талановитих фахівців до розробки асистентських технологій та інновацій, що роблять наше життя зручнішим.

Завдання № 3

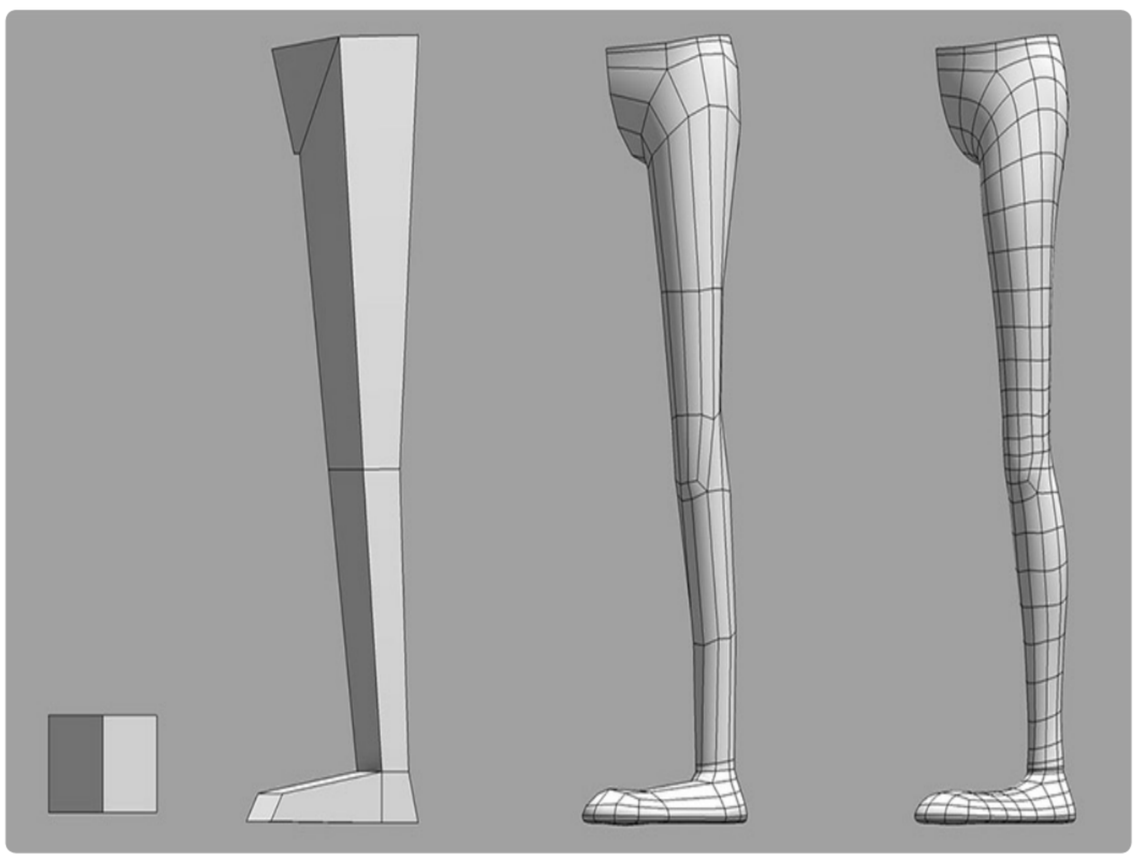
Розглянь подані 3D-моделі. Що спільного у способі їх створення?

Завдання № 4

Розглянь зображення 3D-моделі і поясни, від чого залежить її реалістичність.

Дізнайся більше (е-додаток (20.3)).



Завдання № 5

Увага!

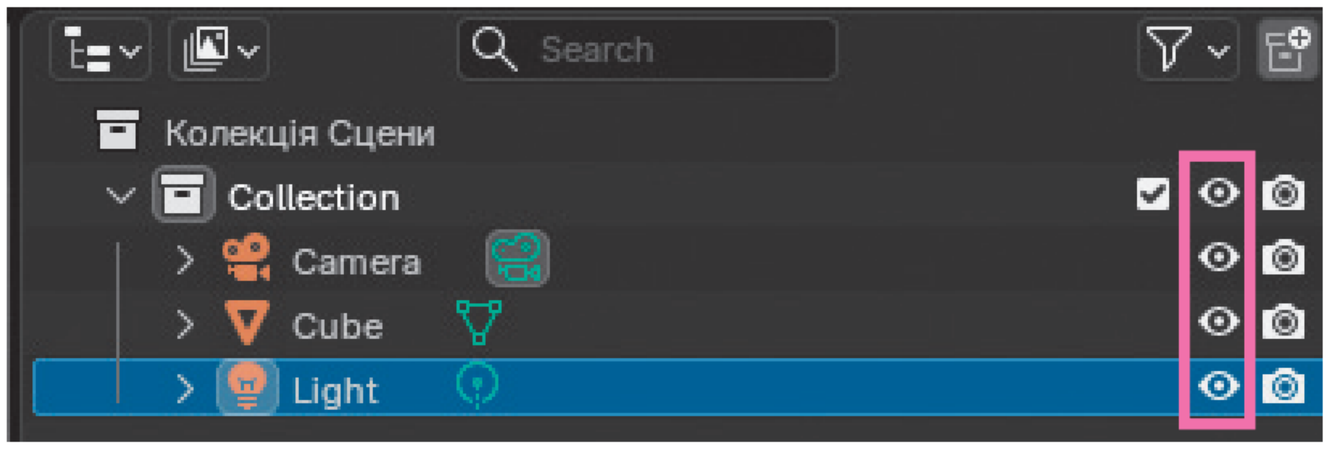
Завантажити останню версію Blender можна за посиланням https://www.blender.org/download/ (Е-додаток (20.4))

1. Запусти програму Blender через меню «Пуск» або за допомогою значка на Робочому столі. Після запуску Blender з'являється картинка-заставка. Її можна прибрати, натиснувши мишею в довільному від заставки місці або за допомогою клавіші Esc.
2. Якщо інтерфейс Blender англомовний, українізуй його. Як це зробити, описано в рубриці «Запитання-відповіді» (стор. 207). Також можна обрати мову інтерфейсу під час першого запуску програми.
3. Відкрий е-додаток (20.4) та опрацюй розділ Інтерфейс Користувача – User Interface інструкції з використання Blender.

Завдання № 6

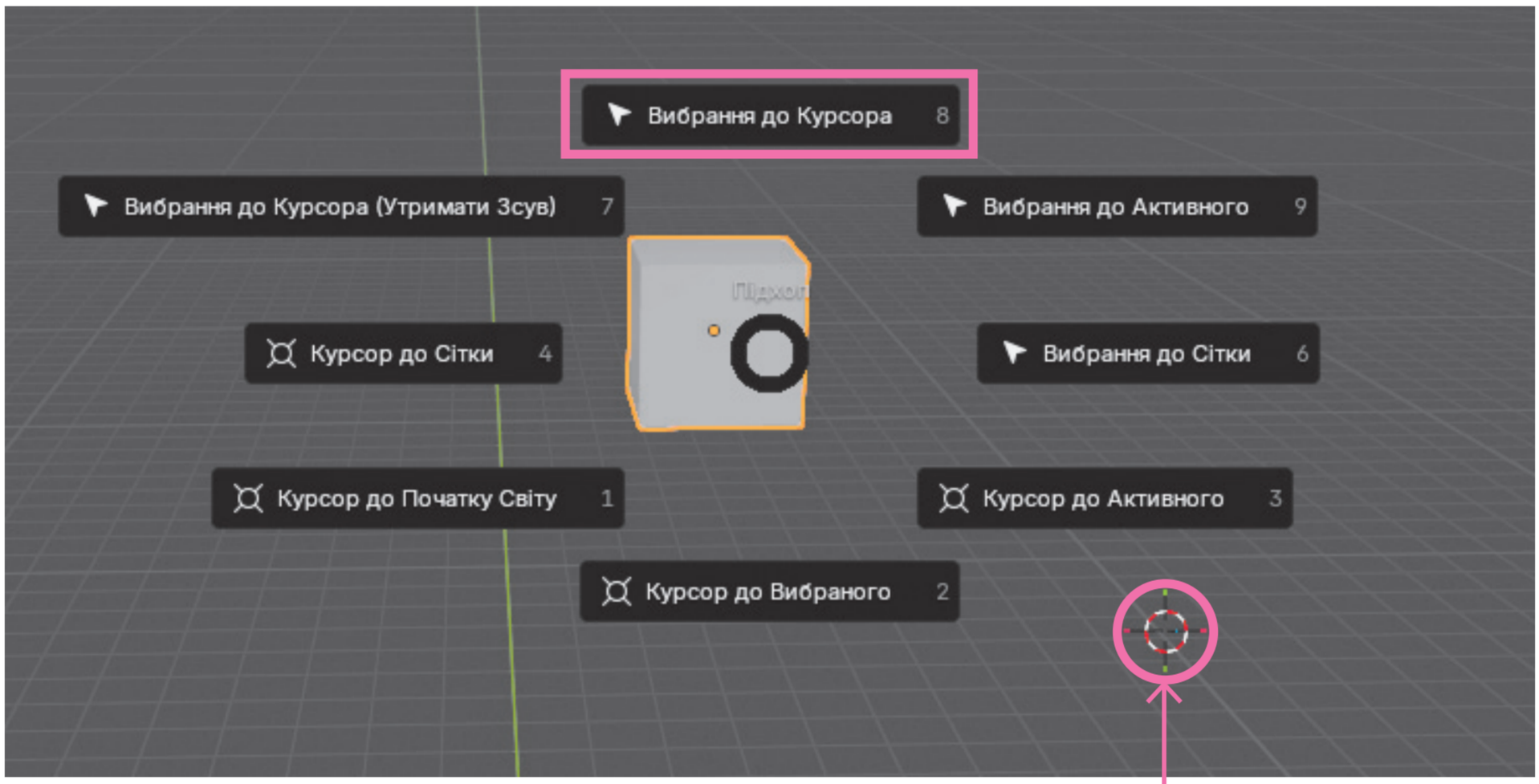
Опрацюй теоретичні відомості з рубріки «Запитання-відповіді» (стор. 212–215), здійсни такі дії з елементами сцени:

1. Знайди область інтерфейсу із назвою **Outliner (Структуратор)**. Прибери зі сцени зображення лампи та камери (видимість об'єкту вмикається та вимикається кнопкою із зображенням ока).

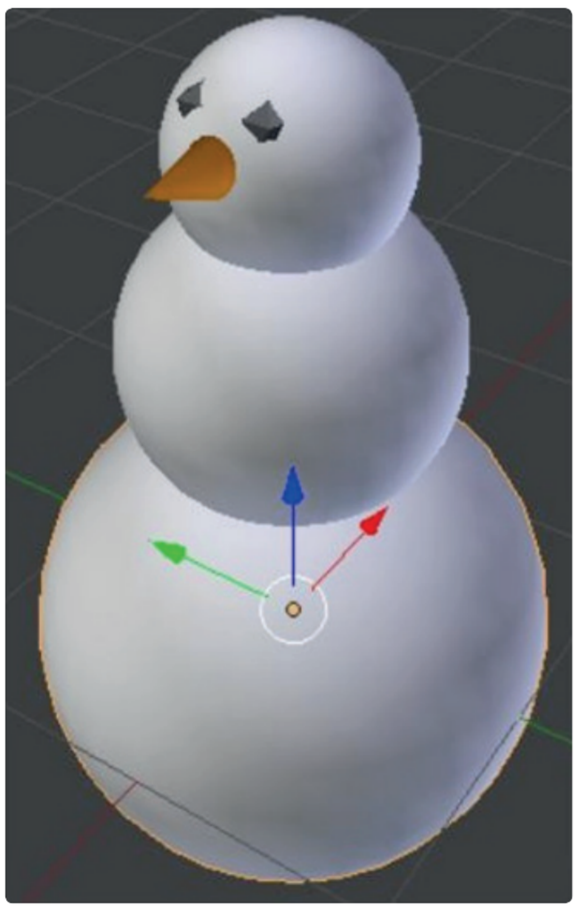


vse.ee/dbgz

2. Спробуй за допомогою коліщатка миші змінити розмір куба — ця дія називається масштабуванням.
3. Розглянь сцену з усіх боків: переміщуй мишу з натиснутим коліщатком. Для зміни положення та масштабу сцени використовуй коліщатко миші разом із клавішами **Ctrl** або **Shift**.
4. Виділи куб, клацнувши по ньому **лівою кнопкою миші (ЛКМ)**.
5. Перемісти куб за допомогою миші: виділи куб і відпусти кнопку миші. Натисни **G**, порухай мишу, а потім клацни **ЛКМ**, щоб завершити переміщення.
6. Перемісти куб іншим способом. Клацни **Shift + права клавіша миші (ПКМ)** у тому місці сцени, куди потрібно перемістити куб — так ти встановиш 3D-курсор. Потім виділи куб, натисни **Shift + S** і вибери команду **Вибране до курсора**.



Завдання № 7



Об’єднайтесь у пари. Розгляньте модель сніговика, побудовану в Blender. Уявіть, що її створює учень або учениця, який/ка уперше працює в Blender. Потрібно відтворити цього сніговика якомога швидше, не втративши подібності.

Обговоріть у парі, які частини моделі можна створити одним і тим самим способом, де доцільно використати копіювання, а де — окремі об’єкти, а також які рішення спростять роботу, але не зіпсують вигляд моделі. Спробуйте передбачити типові помилки новачка й визначити, а також визначте, які частини моделі можуть викликати труднощі під час створення. Запропонуйте рішення, яке ви вважаєте оптимальним, і поясніть свій вибір.

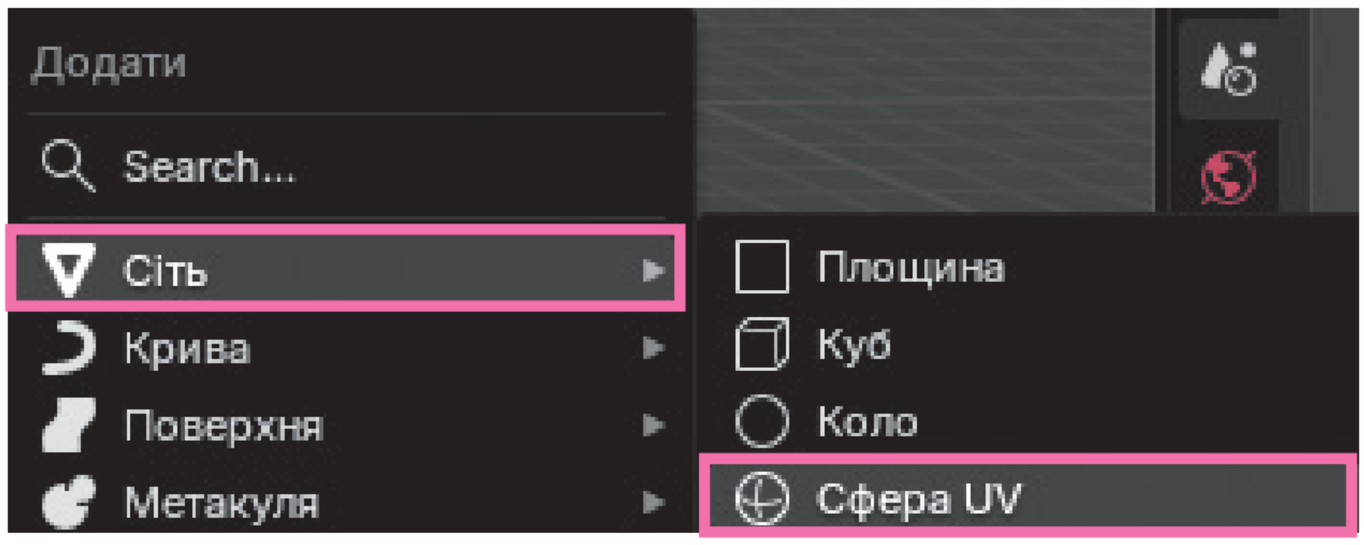
Завдання № 8

Створіть покроково зображення тулуба сніговика за зразком.

Детальніше про виконання операцій у Blender — у рубриці «Запитання-відповіді».

Крок 1 Створення тулуба сніговика.

1. Звільни простір сцени — видали куб: виділи куб **лівою кнопкою миші (ЛКМ)** і натисни клавішу **Delete** або клацни по кубу **ПКМ** і в контекстному меню вибери команду **Видалити**.
2. Додай сферу: натисни комбінацію клавіш **Shift + A** обери:



* *У даному інтерфейсі не коректний переклад з англійської (англ. Mesh — «Сітка») . У Blender Mesh — це 3-d об'єкт, що складається з вершин, ребер і граней.*

3. Побудуй зображення тулуба сніговика скориставшись операціями описаними у таблиці:

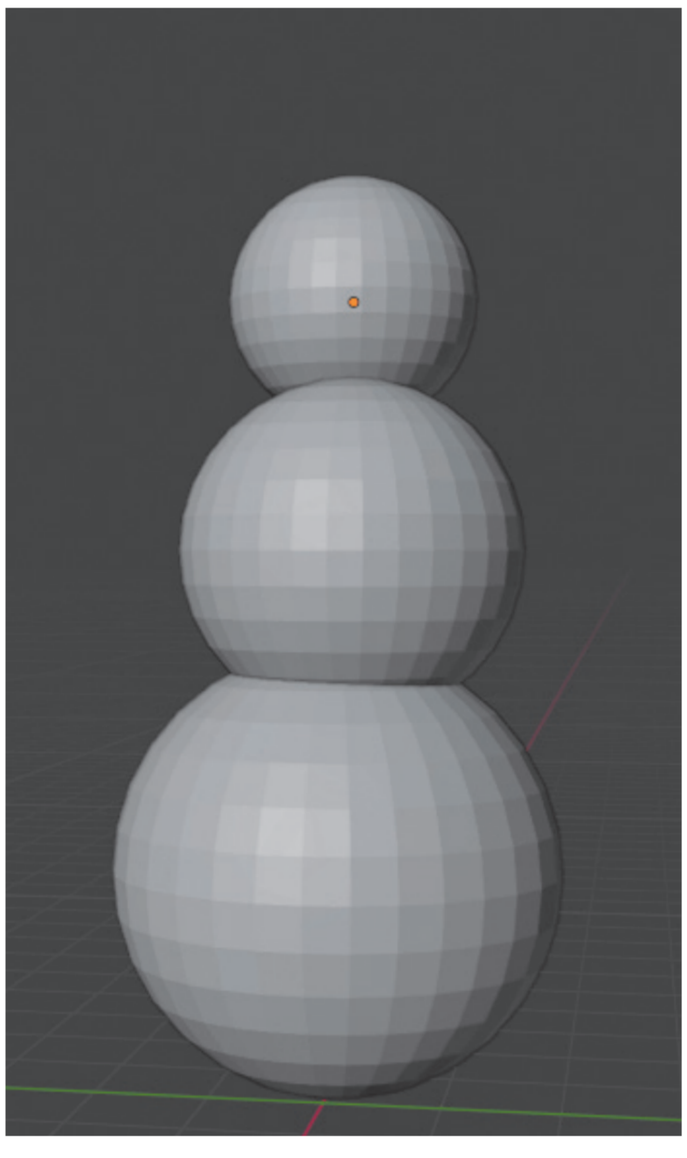
Операції, які є основою для моделювання в 3D-редакторі Blender.

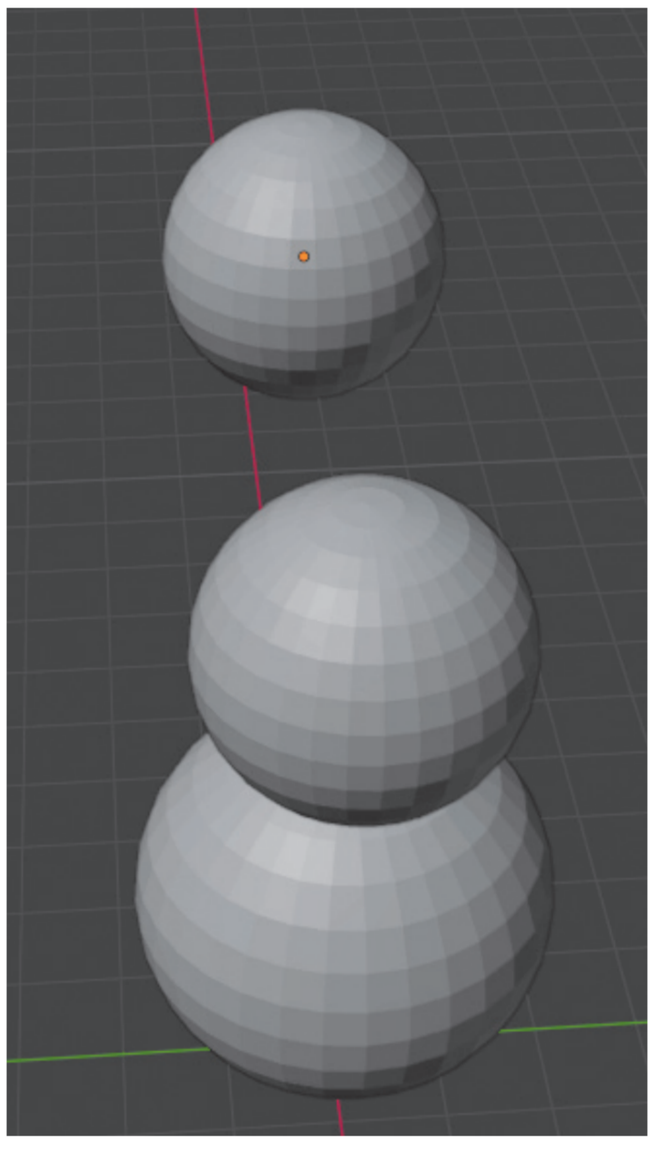
Клавіша	Дія	Пояснення
G	**Переміщення** (Grab/Грабінг)	Рух об’єкта по сцені. Після натискання **G** можна додатково натиснути **X**, **Y**, або **Z**, щоб обмежити рух однією віссю.
R	**Обертання** (Rotate/Ротація)	Поворот об’єкта навколо його центру. Можна обмежити віссю (**R** + **X**, **Y**, або **Z**).
S	**Масштабування** (Scale/Скейлінг)	Зміна розміру об’єкта. Можна змінити розмір вздовж однієї осі (**S** + **X**, **Y**, або **Z**).
E	**Екструдування** (Extrude)	Витиснення нової геометрії з виділених вершин, ребер або граней (використовується в Режимі редагування).
Shift + D	**Дублювати**	Створення копії об’єкта.

Порада:

Під час огляду сцени з різних боків перевіряй, чи об’єкти справді з’єднані в тривимірному просторі, а не лише виглядають дотичними з одного ракурсу. Для правильної геометрії розміщуй об’єкти так, щоб вони частково перетиналися, а не лише торкалися поверхнями. Щоб перевірити це, обертай сцену, утримуючи коліщатко миші, та розглядай модель спереду, збоку й зверху.

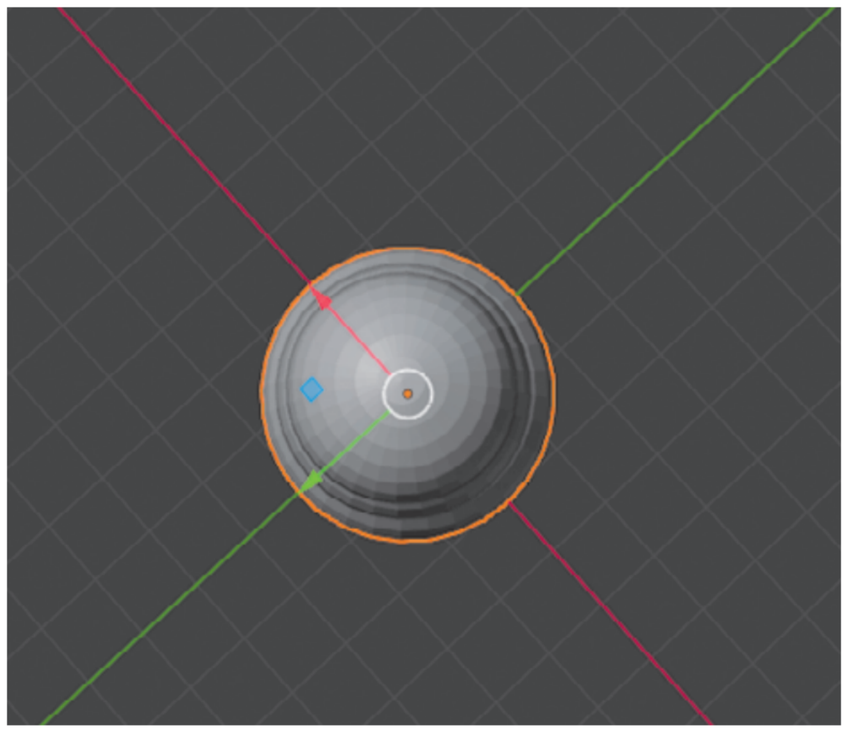
Неправильне розташування об’єктів у 3D-просторі може бути непоміт-ним з одного ракурсу, але добре видно з іншого. На прикладі показано два ракурси перегляду однієї сцени.





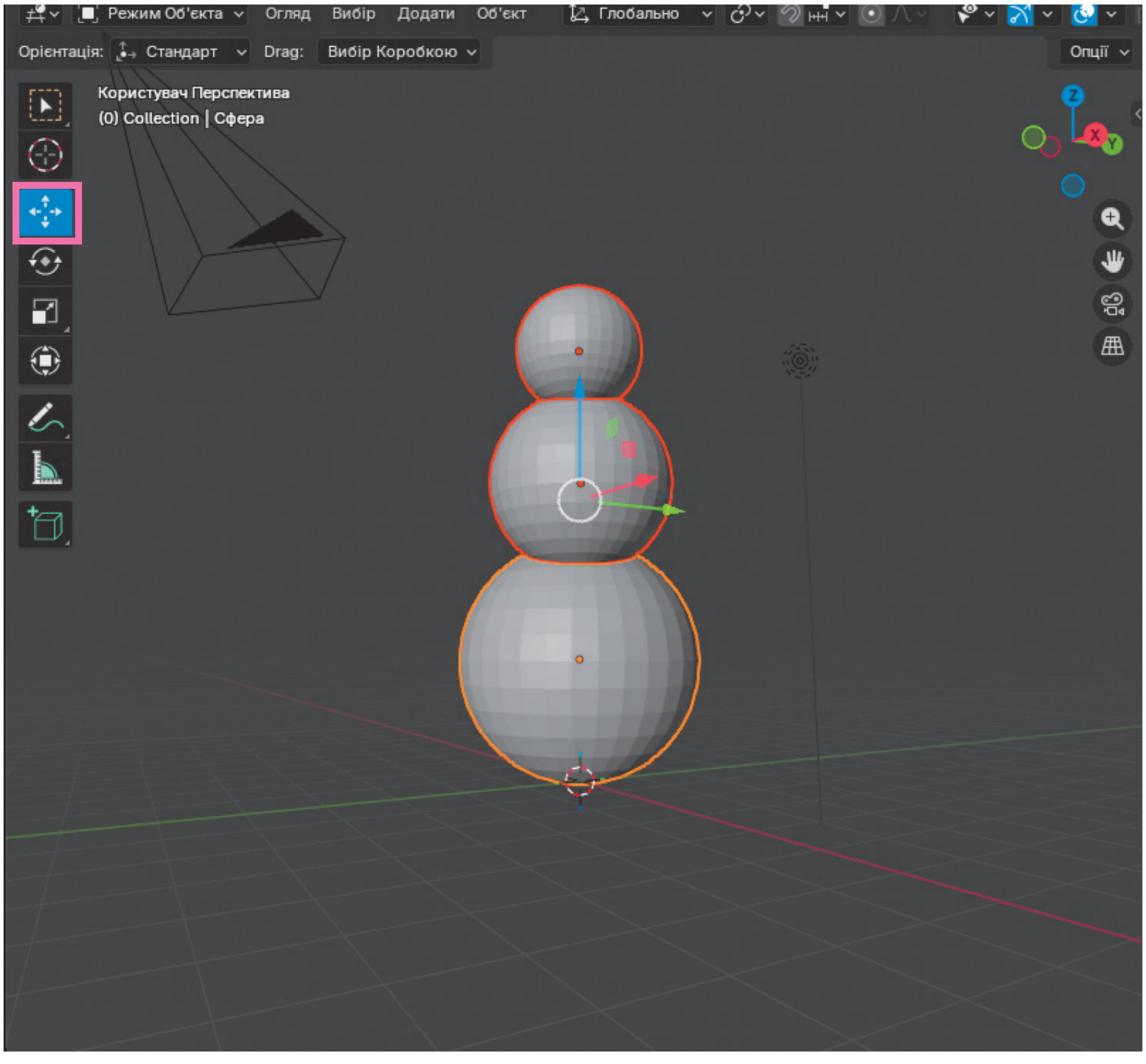
Для редагування дотичності об’єктів увімкни переміщення (**G**) і злегка посунь один об’єкт усередину іншого, знову контролюючи результат з кількох ракурсів.

На малюнку правіше правильний вигляд всіх трьох куль зверху — положення об’єкта зрозуміле відносно червоної (X) і зеленої (Y) осей, центр об’єкта легко перевіряється й коригується.



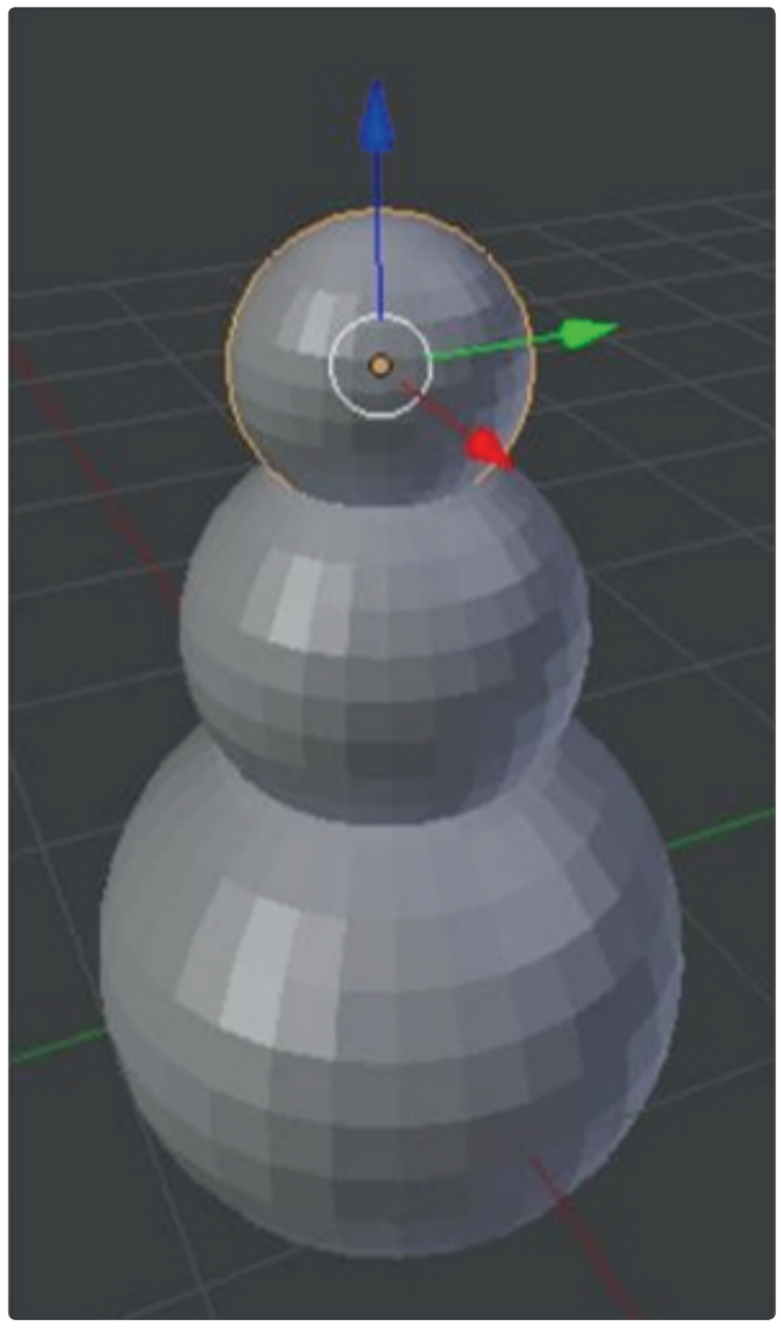
Користуйся таким правилом: положення по X і Y перевіряй зверху, а положення по Z — з вигляду спереду або збоку.

Поекспериментуй. Виділи всі три кулі, на лівій панелі інструментів 3D-вікна обери інструмент «Переміщення» . Після цього біля об’єкта з’являться червона, зелена й синя стрілки — маніпулятор переміщення (gizmo), який допомагає розміщувати об’єкти в тривимірному просторі. За допомогою цього маніпулятора розмісти сніговика в центрі сцени.



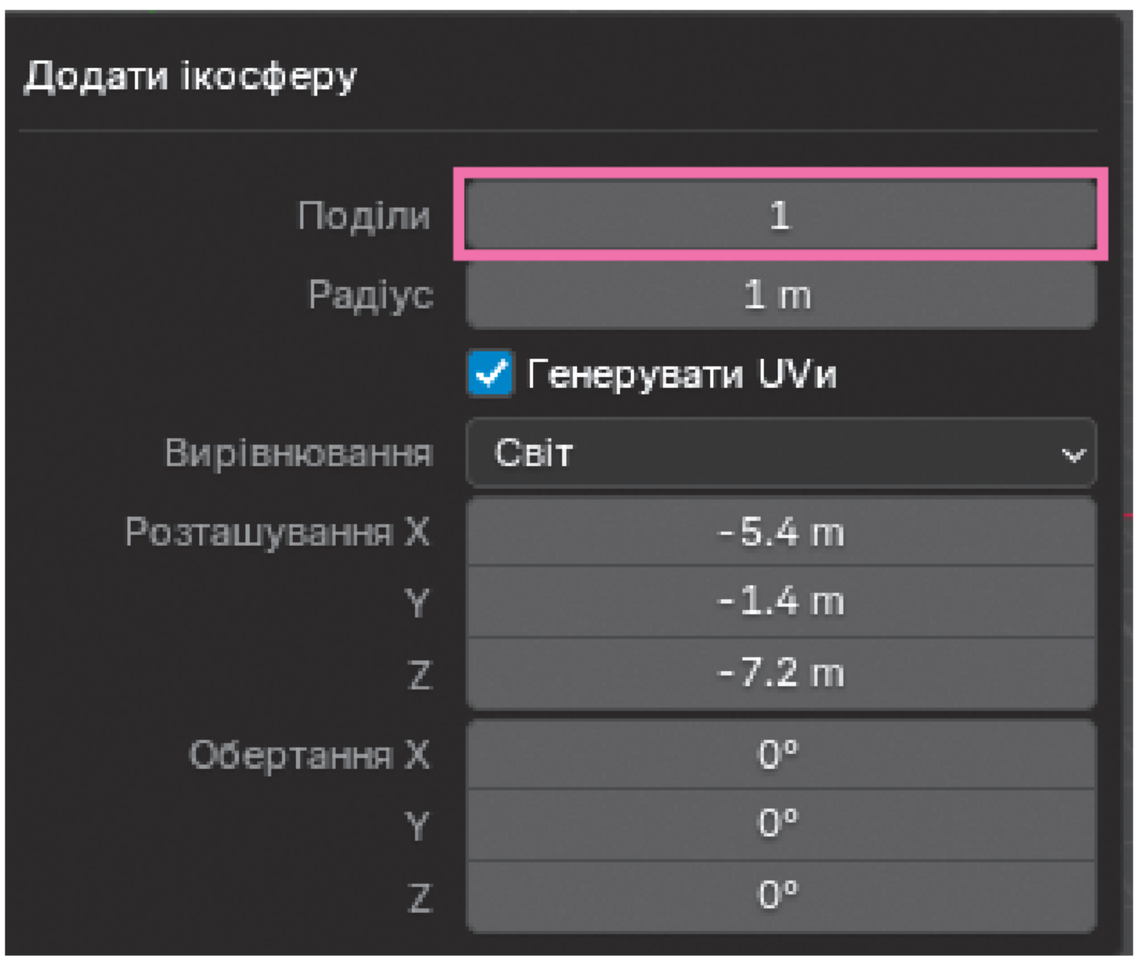
Крок 2 Додавання об’єкту «ніс».

1. Встанови 3D-курсор у місце, де потрібно створити ніс сніговику: підведи вказівник миші та натисни **Shift + ПКМ**.
2. Додай конус: натисни **Shift + A** → **Сітка (Mesh)** → **Конус**.
3. Встанови ніс у потрібному напрямку (орієнтації) та задай необхідний розмір, скориставшись таблицею дій із попереднього кроку.



Крок 3 Додавання об’єктів «очі»

1. Додай фігуру ікосферу (**Shift + A** → **Сітка (Mesh)** → **Ікосфера**). Сфера Іко
2. Натисни **F9** та відкрий панель **Параметри додавання ікосфери** та встанови значення Поділи = 1.



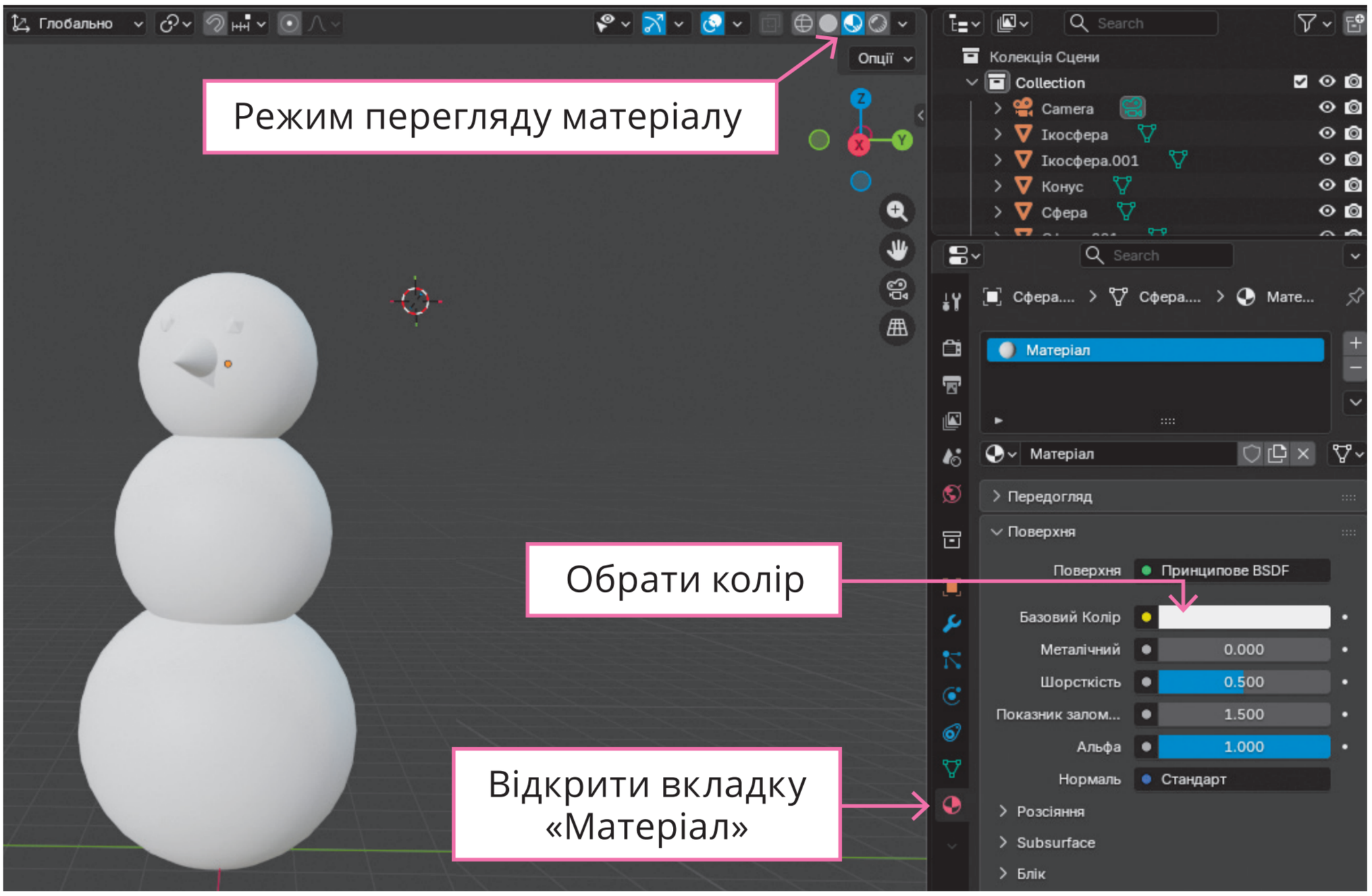
3. Зменш розмір ікосфери та розташуй її, використовуючи інструменти масштабування й переміщення.
4. Створи копію ікосфери для другого ока.

За допомогою різних фігур ви створили зображення сніговика. Розгляньте його: а чи насправді сніговик має такий вигляд? Невже кулька із снігу має гострі кути? А колір носа-моркви має бути, як у снігової кульки?

Крок 4 Згладжування та матеріали

1. Виділи кулі тулуба, клацни правою кнопкою миші та вибери команду **Згладжене тонування**.
2. Відкрий вкладку **Матеріал** у редакторі властивостей.

3. Натисни кнопку **Новий**, щоб створити матеріал.
4. Для снігу встанови білий колір і високу шорсткість поверхні.



5. Для носа задай помаранчевий колір, а для очей — темний.

Для вибору кольору зручно використовувати шістнадцятковий код, наприклад:

Яскраво-бурштиновий	Amber (SAE|ECE)	#FF7E00	255 126 0

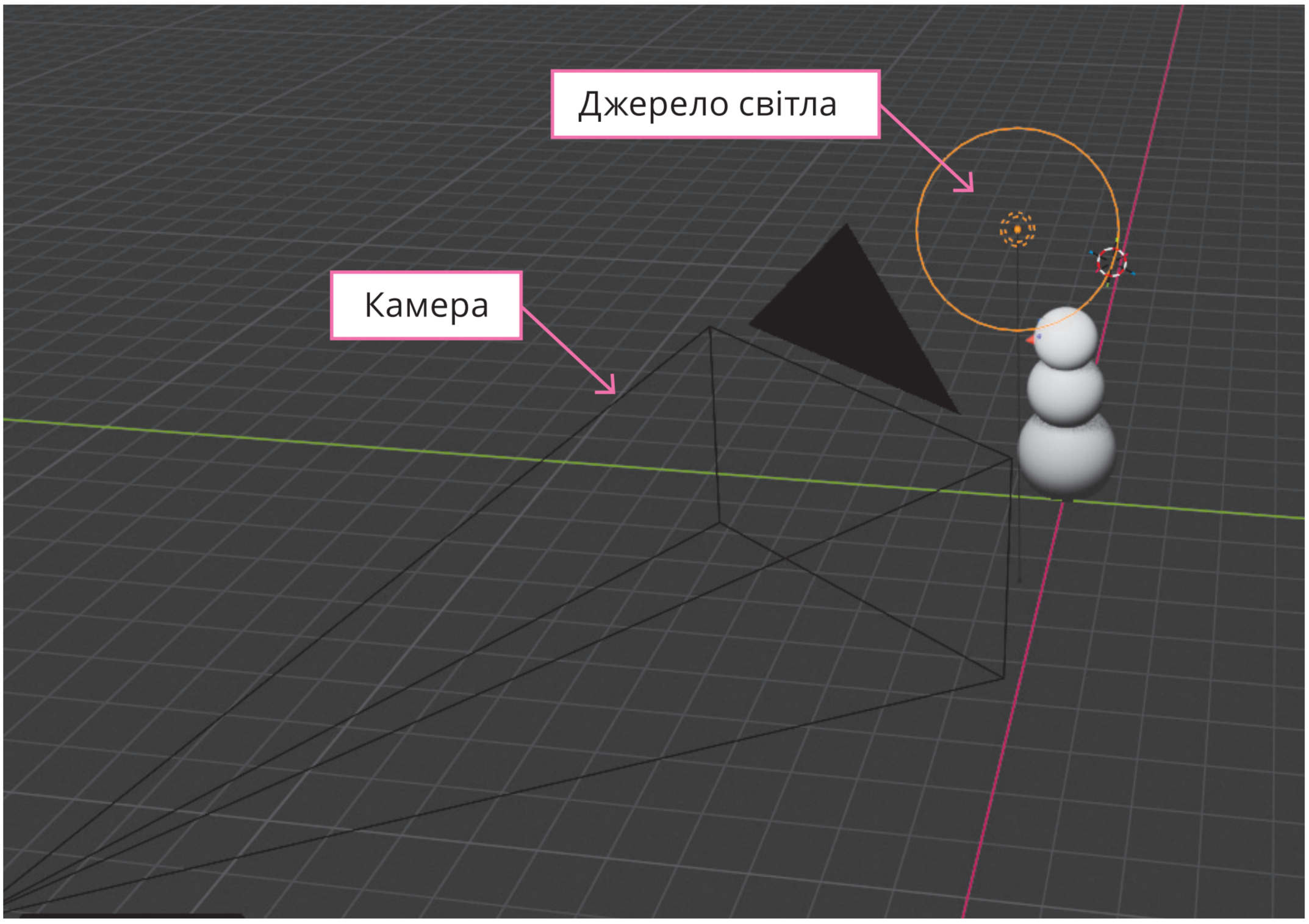
https://www.colorome.com/colors

Завдання № 9

1. Дізнайся більше про можливості збереження зображень із рубрики «Запитання — відповіді».
2. Ознайомся із особливостями процесу Рендеринг та налаштуй камеру та джерело світла для збереження готового зображення.

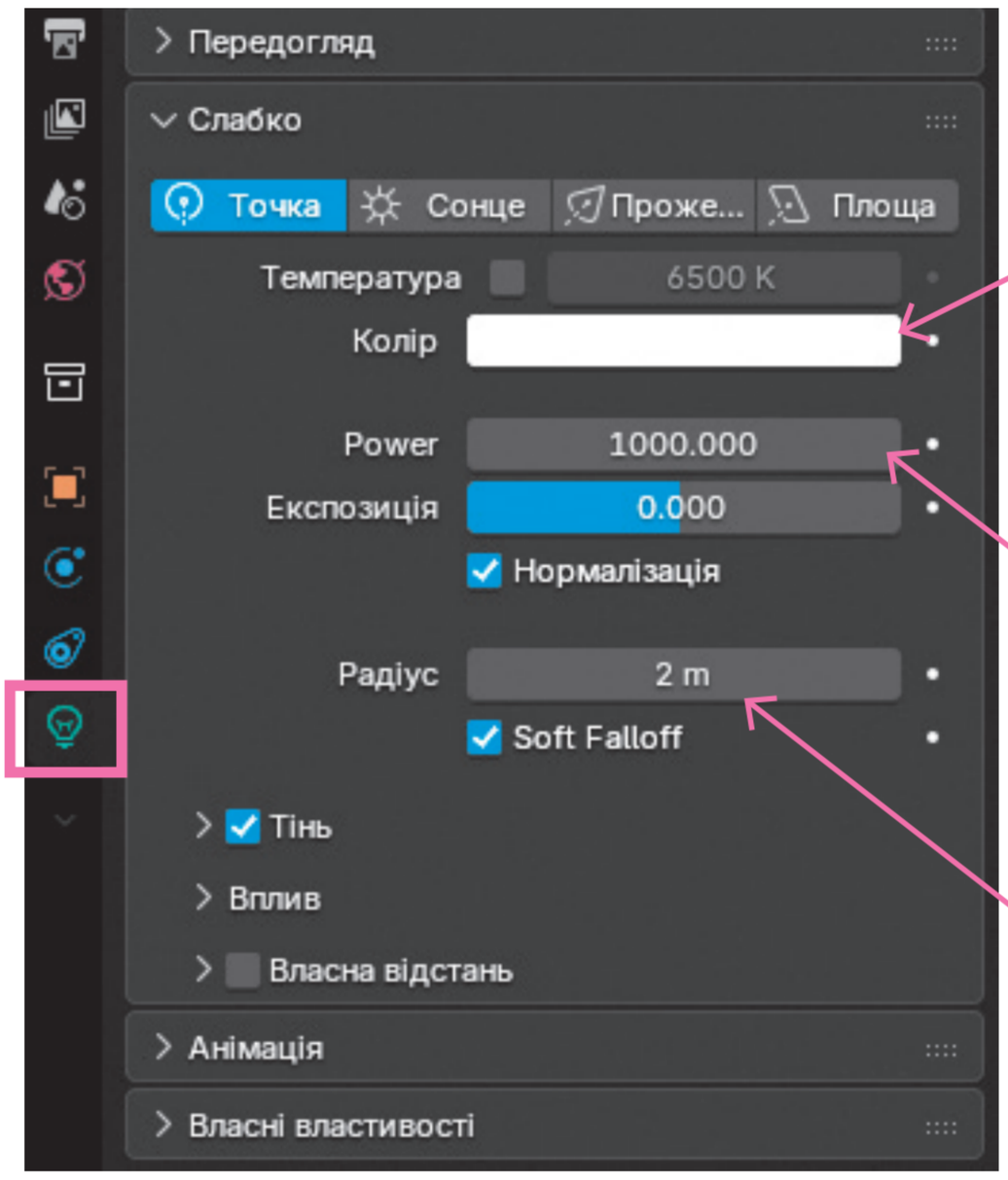
Рендеринг — це процес створення готового зображення 3D-сцени. Під час рендерингу Blender обчислює, як об'єкти виглядатимуть з точки зору камери, з урахуванням освітлення, матеріалів і тіней. Камера визначає, яку частину сцени ми бачимо і з якого ракурсу. Освітлення визначає, наскільки об'єкти помітні, їхній об'єм, тіні та настрій зображення.

Тому для правильного рендерингу важливо коректно розташувати камеру перед об'єктом, а також додати джерела світла, розмістити їх у просторі та налаштувати їхні властивості.



Якщо джерело світла було випадково вилучено с проєкту, то додай його: ***Shift + A*** → ***Світло*** → ***Точкове***.

Налаштування властивостей світла:



Колір. Визначає відтінок освітлення. Білий або трохи теплий колір підходить для нейтрального освітлення сцени.

Power (Потужність). Показує, наскільки яскраве світло. Замале значення → рендер темний; завелике → об’єкт «вигорає».

Радіус. Впливає на м’якість тіней. Малий радіус → різкі тіні. Більший радіус → м’які, природні тіні.

3. Збрежи створене зображення як проєкт і як готове зображення.

ЗАПИТАННЯ-ВІДПОВІДІ

Що є основою створення об’ємного зображення?

Щоб описувати положення об’єктів у 3D, використовують три осі координат *X, Y, Z*. У площині *XY* точку задають двома координатами (*x, y*) — це положення на «підлозі» сцени. Третя вісь *Z* додає висоту: піднімає або опускає точку над площиною. Тому будь-яка точка в просторі має три координати (*x, y, z*).

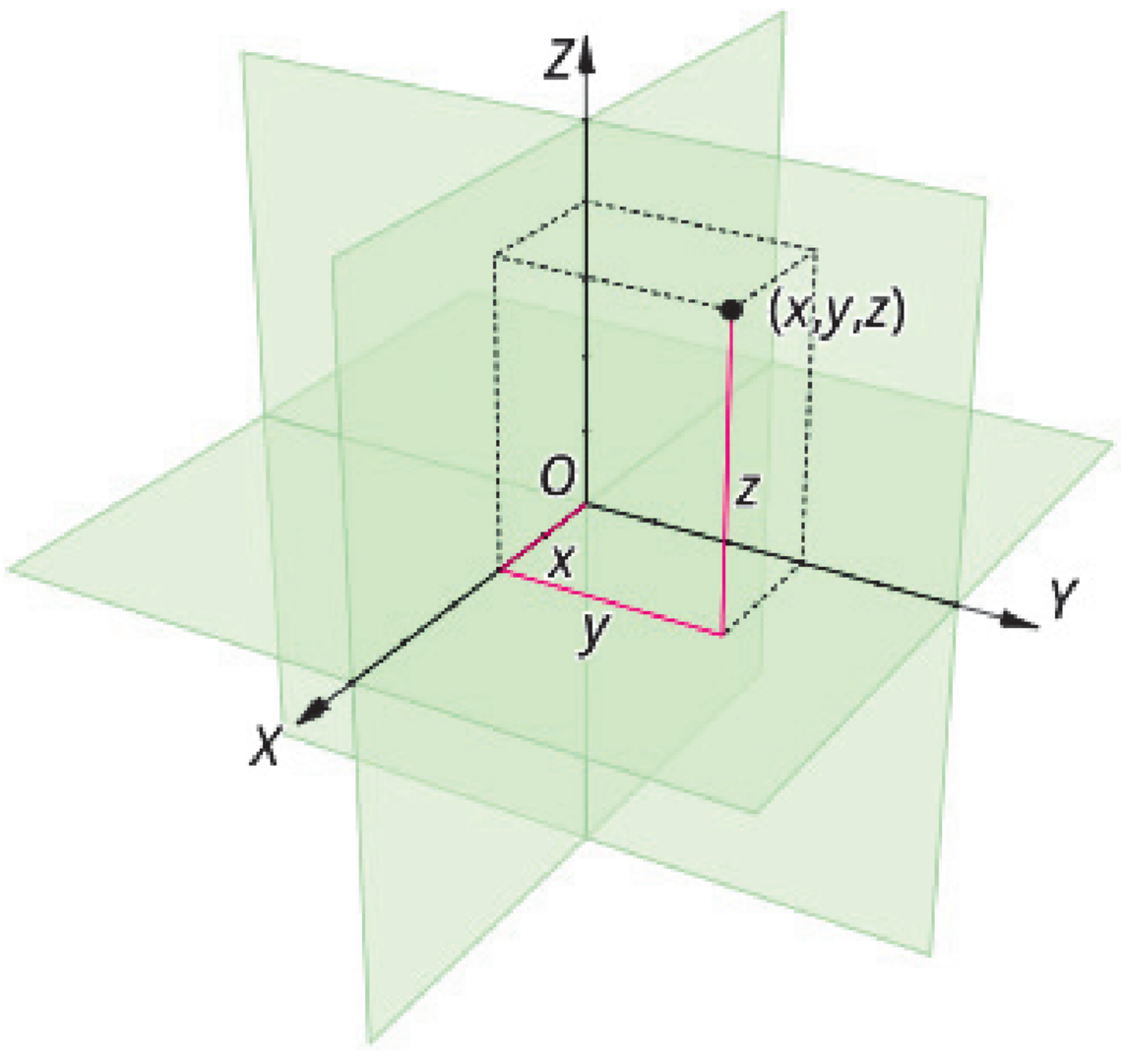


Рис. 2.1. Зображення точки (x, y, z) у тривимірній декартовій системі координат

Точки в тривимірному просторі (вершини) з'єднуються лініями (ребрами), а з кількох ліній можна побудувати багатокутник (грань, або полігон).

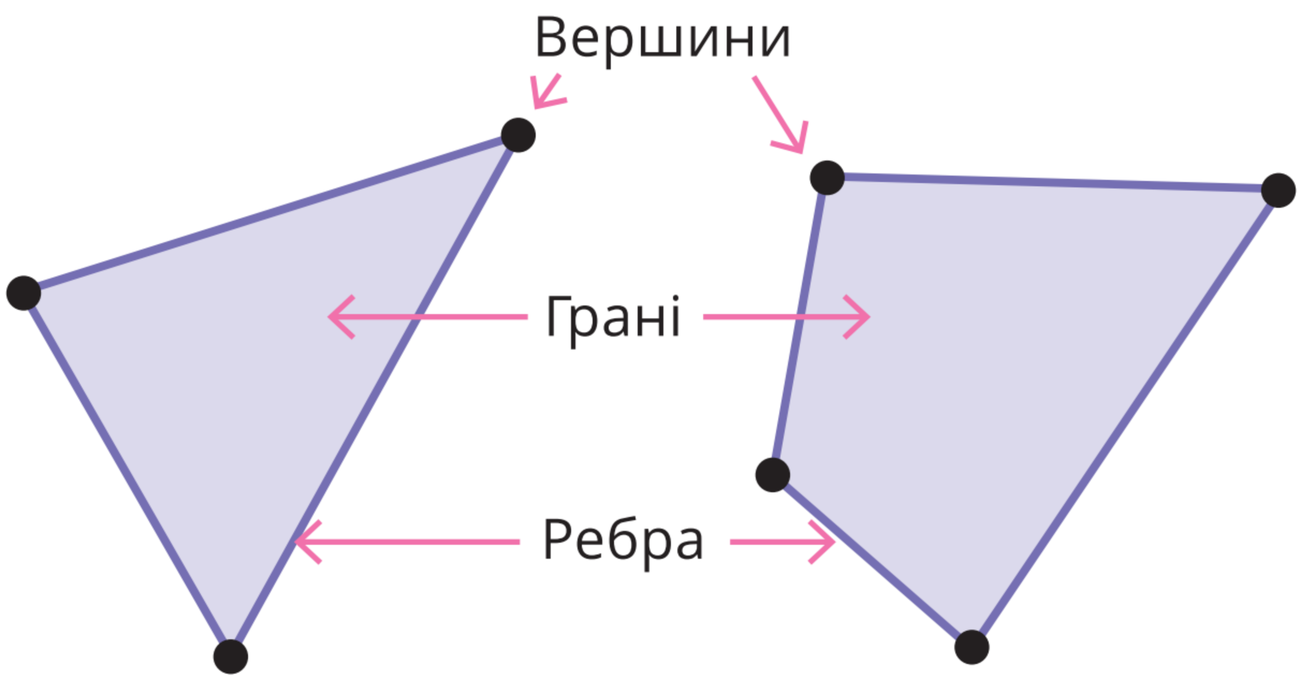


Рис. 2.2. Полігональна структура моделі в 3D-редакторі

ТЕРМІНОЛОГІЧНИЙ СЛОВНИК

Примітив: базова геометрична форма, з якої починають моделювання. Наприклад: куб, сфера, циліндр, конус, тор.

Об'єкт: будь-який елемент, який знаходиться на сцені. Об'єктом може бути примітив, камера, джерело світла або вже змодельована складна фігура.

3D-модель: Математична репрезентація об'єкта у просторі, що складається з вершин (точок), ребер (ліній) та полігонів (граней, зазвичай трикутників або чотирикутників). Чим більше полігонів, тим детальніша модель.

3D-сцена: Віртуальний простір, де розміщуються об'єкти, джерела світла та камера, що відображає перспективу.

Моделювання: Процес створення об'єктів. Основні техніки включають полігональне моделювання, роботу з примітивами (куб, куля) та скульптінг (цифрове ліплення).

Матеріали та текстури: Визначають зовнішній вигляд поверхні: колір, прозорість, відблиски, шорсткість.

Освітлення: Налаштування джерел світла для створення реалістичних тіней, відблисків та настрою сцени.

Рендеринг (Візуалізація): Комп'ютерний процес обчислення 3D-сцени та перетворення її на 2D-зображення або відео, враховуючи світло, матеріали та камеру.

ЦЕ ЦІКАВО

Об’ємне зображення персонажів та сюжетів було вперше використано в мультфільмі «Історія іграшок», який вийшов на екрани в 1995 р. А прем’єра першого фільму з використанням окремих елементів тривимірної графіки «Світ майбутнього» (англ. Future world) відбулася в 1976-му; перші редактори 3D-графіки з’явилися пізніше. Е-додаток (20.5).

Які програми популярні для 3D-моделювання на початковому рівні та чи дозволяють вони створювати анімацію?

Для 3D-моделювання використовують спеціальні програми — 3D-редактори. На початковому рівні популярними є Blender, Tinkercad, SketchUp. Ці програми дозволяють створювати тривимірні об’єкти, а деякі з них, зокрема Blender, також дають змогу додавати анімацію та налаштовувати її параметри.

Серед професіоналів найпоширенішими є Autodesk Maya, 3ds Max, Cinema 4D, ZBrush, Houdini та Blender. Ці редактори використовують у кіноіндустрії, ігровій розробці, анімації, дизайні та візуальних ефектах.

Чим растрова та векторна графіка відрізняється від тривимірної?

Векторні та растрові зображення будують у двовимірному просторі, і вони є плоскими. Тривимірна модель будується на основі стереометрії — геометрії у тривимірному просторі. Однак векторні й растрові зображення можуть бути елементами тривимірної моделі. Побудова такої моделі базується на векторній графіці, а для зафарбування граней використовується растрова графіка.

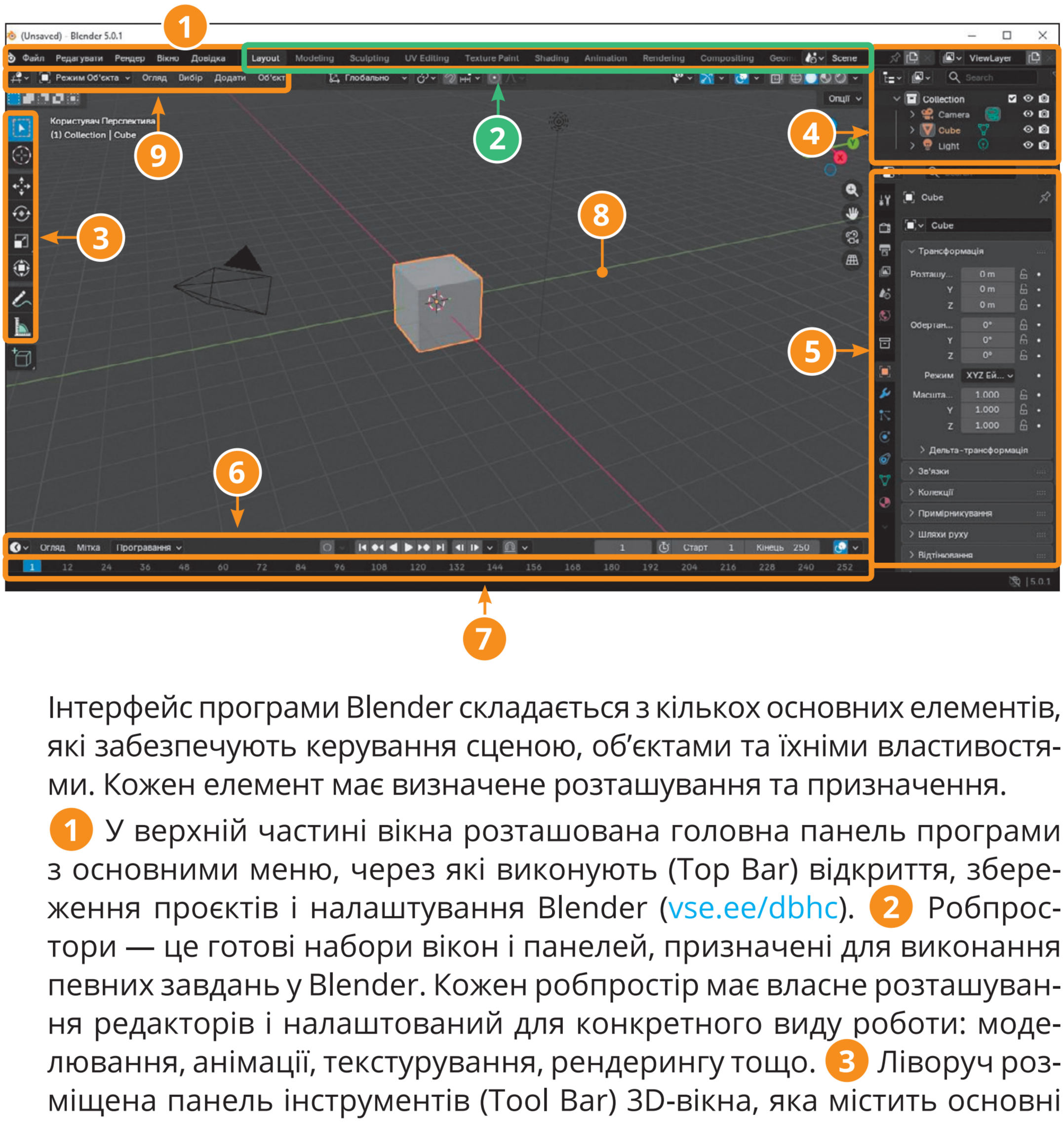
Де використовують тривимірне моделювання?

- У медицині — для комп’ютерної томографії, МРТ-сканування та створення моделей органів.
- В архітектурі та будівництві — для проєктування будівель і візуалізації інтер’єрів.
- У кінематографі та анімації — для створення візуальних ефектів, поєднання відеозйомки з тривимірними сценами та анімаційних фільмів.
- У відеоіграх і віртуальній реальності — для створення ігрових світів, персонажів і симуляцій.
- У промисловому дизайні — для проєктування виробів, техніки та деталей перед виготовленням

Як українізувати інтерфейс Blender?

1. Відкрий меню **Edit (Редагування)**.
2. Вибери команду **Preferences (Уподобання)**.
3. У вікні налаштувань перейди до розділу **Interface (Інтерфейс)**.
4. Знайди параметр **Language (Мова)**.
5. У списку мов обери **Ukrainian (Українська)**.
6. За потреби увімкни додаткові параметри перекладу (назви інструментів, підказки тощо).
7. Натисни кнопку **Save Preferences**, щоб зберегти налаштування.

Які області містить вікно програми Blender?



Інтерфейс програми Blender складається з кількох основних елементів, які забезпечують керування сценою, об’єктами та їхніми властивостями. Кожен елемент має визначене розташування та призначення.

1 У верхній частині вікна розташована головна панель програми з основними меню, через які виконують (Top Bar) відкриття, збереження проєктів і налаштування Blender (vse.ee/dbhc). 2 Робпростори — це готові набори вікон і панелей, призначені для виконання певних завдань у Blender. Кожен робпростір має власне розташування редакторів і налаштований для конкретного виду роботи: моделювання, анімації, текстурування, рендерингу тощо. 3 Ліворуч розміщена панель інструментів (Tool Bar) 3D-вікна, яка містить основні інструменти для вибору, переміщення, обертання та масштабування об’єктів. 4 У правій верхній частині розташована панель Outliner,

де відображається структура сцени — всі об’єкти, колекції та їхня видимість. 5 Під нею знаходиться редактор властивостей (Properties Editor), який містить вкладки для налаштування об’єкта, матеріалів, освітлення, сцени та інших параметрів (vse.ee/dbhe). 6 Унизу вікна розміщена шкала часу, що використовується для роботи з анімацією та керування кадрами. 7 Під шкалою часу знаходиться панель керування відтворенням анімації та навігації по кадрах. 8 Центральну частину екрана займає 3D-вікно — основна робоча область, у якій відображається сцена та виконуються всі дії з об’єктами. 9 Перемикач режимів і меню: Огляд, Вибір, Додати, Об’єкт. Детальніше про режими за посиланням vse.ee/dbid.

Які об’єкти є на робочому полі нового проєкту Blender?

Лампа (джерело світла) — освітлює об’єкти сцени, визначає загальний вигляд зображення під час рендерингу.

Камера — визначає, з якого ракурсу відбувається зйомка під час рендерингу.

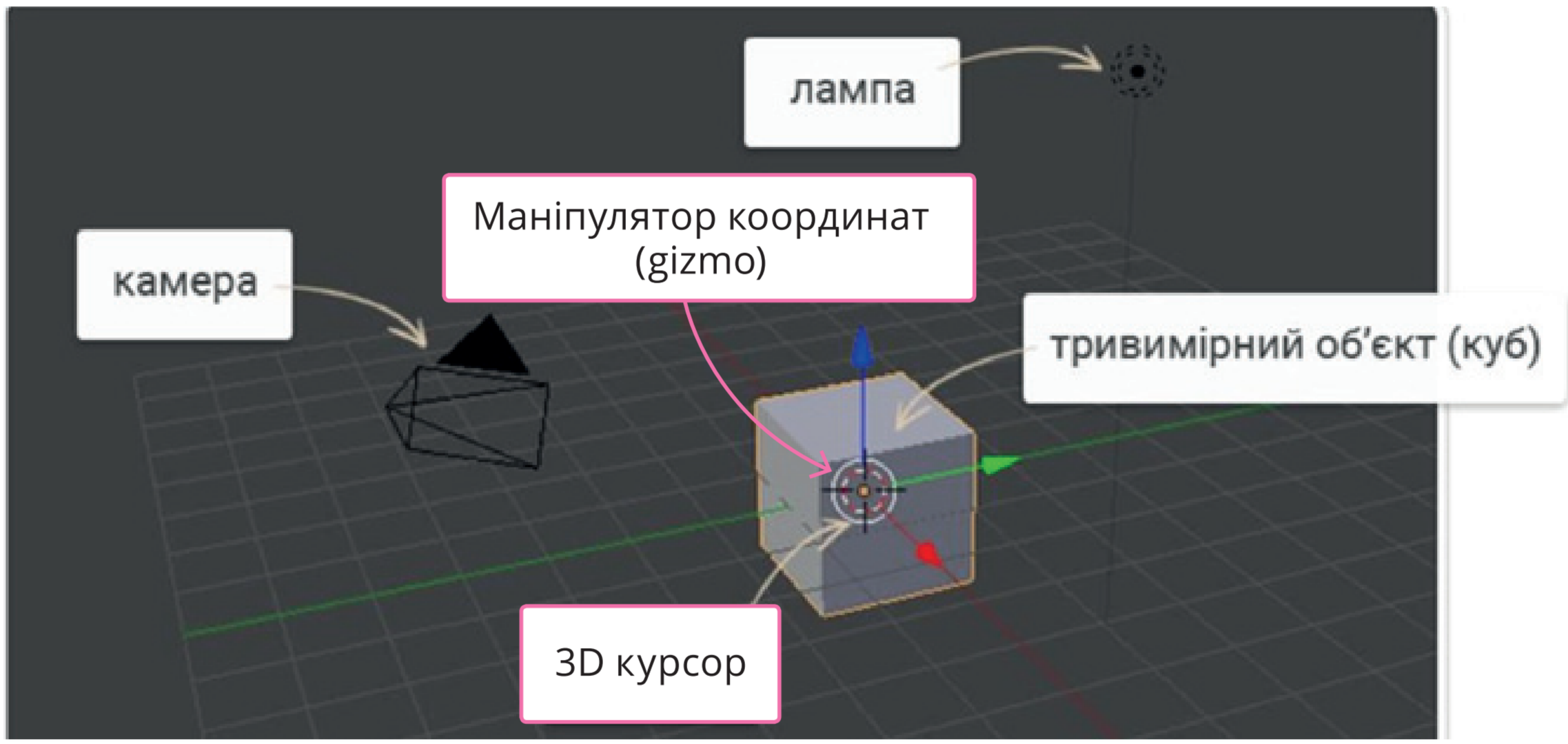
Куб — стандартний тривимірний об’єкт, який з’являється в новому проєкті за замовчуванням.

3D-курсор — точка, у місці якої додаються нові об’єкти.

Осі координат *X, Y, Z* — показують напрямки в тривимірному просторі (*X* — червона, *Y* — зелена, *Z* — синя).

Сітка координат у площині *XY* — допомагає орієнтуватися в просторі сцени та вирівнювати об’єкти.

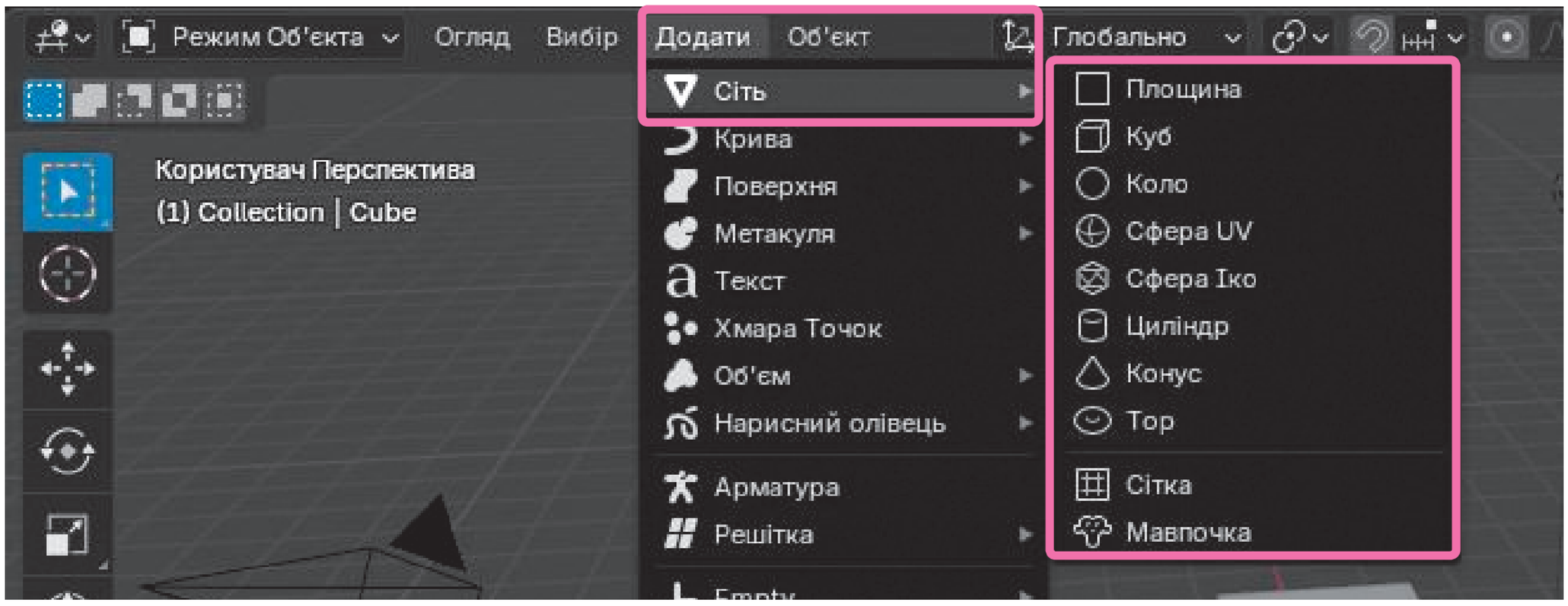
Маніпулятор координат (gizmo) — використовується для переміщення, обертання та масштабування об’єктів у 3D-просторі.



Як додати до сцени об’єкт?

Щоб додати об’єкт до сцени, скористайся меню **Додати** на верхній панелі. У списку обери **Сітка (Mesh)** та потрібний тривимірний об’єкт (куб, сфера, циліндр тощо).

Також можна скористатися клавіатурною комбінацією **Shift + A**.



Важливо!

Новий об’єкт завжди додається в позицію 3D-курсора. Після додавання об’єкт автоматично стає активним і виділеним, тому одразу готовий до подальшого редагування.

Які операції можна виконувати з об’єктами?

Базовими операціями з об’єктами в Blender є переміщення, обертання та масштабування. Їх можна виконати за допомогою Панелі інструментів або використовуючи гарячі клавіші, перелічені в таблиці нижче. Виділений об’єкт можна також перемістити в стандартні розташування, натиснувши клавіші **Shift + S** та вибравши одну з команд меню, що відкриється.

Які дії можна виконувати з об’єктом?

Як виділити об’єкт?

Клацни на об’єкті лівою кнопкою миші. Об’єкт буде підсвічено помаранчевою рамкою.

Як виділити кілька об’єктів у Blender?

Клацни лівою кнопкою миші кожен об’єкт, утримуючи клавішу **Shift**.

Як виділити всі об'єкти?

Натисни клавішу **A**. Щоб зняти все виділення, потрібно клацнути в будь-яке місце сцени.

Як змінити розмір об'єкта?

Спосіб 1. За допомогою клавіатури

Виділи об'єкт. Натисніть клавішу **S** (Scale — масштаб). Перемісти мишу для збільшення або зменшення розміру. Підтвердь дію клавішею **Enter** або клацанням лівої кнопки миші.

Масштабування можна здійснювати і по певній осі. Після натискання **S** можна вибрати вісь:

- **X** — зміна розміру по осі X;
- **Y** — по осі Y;
- **Z** — по осі Z.

Наприклад: **S** → **X** — масштабування лише по осі **X**.

Спосіб 2. За допомогою інструмента масштабування

Виділи об'єкт. На лівій панелі інструментів вибери інструмент . З'явиться маніпулятор із кольоровими осями. Перетягуй осі для зміни розміру.

Спосіб 3. Через властивості об'єкта

Виділіть об'єкт. У **Редакторі властивостей** відкрийте вкладку **Властивості об'єкта /панель «Трансформація»**. У полі **Масштаб** введи числові значення.

Як перемістити об'єкт?

Спосіб 1. За допомогою клавіатури

Виділи об'єкт. Натисни клавішу **G** (Grab — перемістити).
Перемісти мишу для зміни положення об'єкта. Підтвердь дію клавішею **Enter** або клацанням лівої кнопки миші.

Переміщення вздовж певної осі

Після натискання **G** можна обрати вісь:

- **X** — переміщення вздовж осі X;
- **Y** — вздовж осі Y;
- **Z** — вздовж осі Z.

Наприклад: **G** → **Z** — переміщення тільки по осі Z.

Спосіб 2. За допомогою інструмента переміщення

Виділи об'єкт. На лівій панелі інструментів (Toolbar) вибери інструмент . З'явиться маніпулятор із трьома кольоровими осями. Перетягуйте стрілки осей для переміщення об'єкта.

Спосіб 3. Через властивості об'єкта

Виділи об'єкт. У **Редакторі властивостей** відкрий вкладку **Властивості об'єкта/панель «Трансформація»**. У полі **Розташування** введи точні координати положення об'єкта.

Як обертати об'єкт?

Спосіб 1. За допомогою клавіатури

Виділи об'єкт. Натисніть клавішу **R** (Rotate — обертати).
Перемісти мишу для обертання. Підтвердь дію клавішею **Enter** або клацанням лівої кнопки миші.

Обертання навколо певної осі

Після натискання **R** можна обрати вісь:

- **X** — обертання навколо осі X;
- **Y** — навколо осі Y;
- **Z** — навколо осі Z.

Наприклад: **R** → **Z** — обертання по осі Z.

Спосіб 2. За допомогою інструмента обертання

Виділіть об'єкт. На лівій панелі інструментів оберіть інструмент .
З'явиться кольоровий маніпулятор осей.
Перетягуйте кола навколо осей для обертання.

Спосіб 3. Через панель властивостей

Виділи об'єкт. **У редакторі властивостей** відкрий вкладку **Властивості об'єкта/панель «Трансформація».** Перейди до вкладки **Властивості об'єкта**. У полі **Обертання** введи значення кутів повороту.

Як змінити ракурс огляду сцени?

Переміщувати мишу з натиснутим коліщатком миші (середньою кнопкою). Сцена обертається у довільному напрямку.

Як змінити ракурс за допомогою маніпулятора осей?

Натиснути на маніпулятор осей координат управому верхньому куті сцени та перетягувати його мишею для обертання сцени.

Як перейти до стандартних видів (зверху, спереду, збоку)?

Натиснути на літери осей *X*, *Y* або *Z* на маніпуляторі осей координат.

Як перемістити сцену вліво–вправо та вгору–вниз?

Переміщувати мишу з натиснутим коліщатком, утримуючи **Shift** (панорамування).

Як змінити масштаб перегляду сцени?

Прокручувати коліщатко миші вперед або назад.

Як плавно змінити масштаб перегляду?

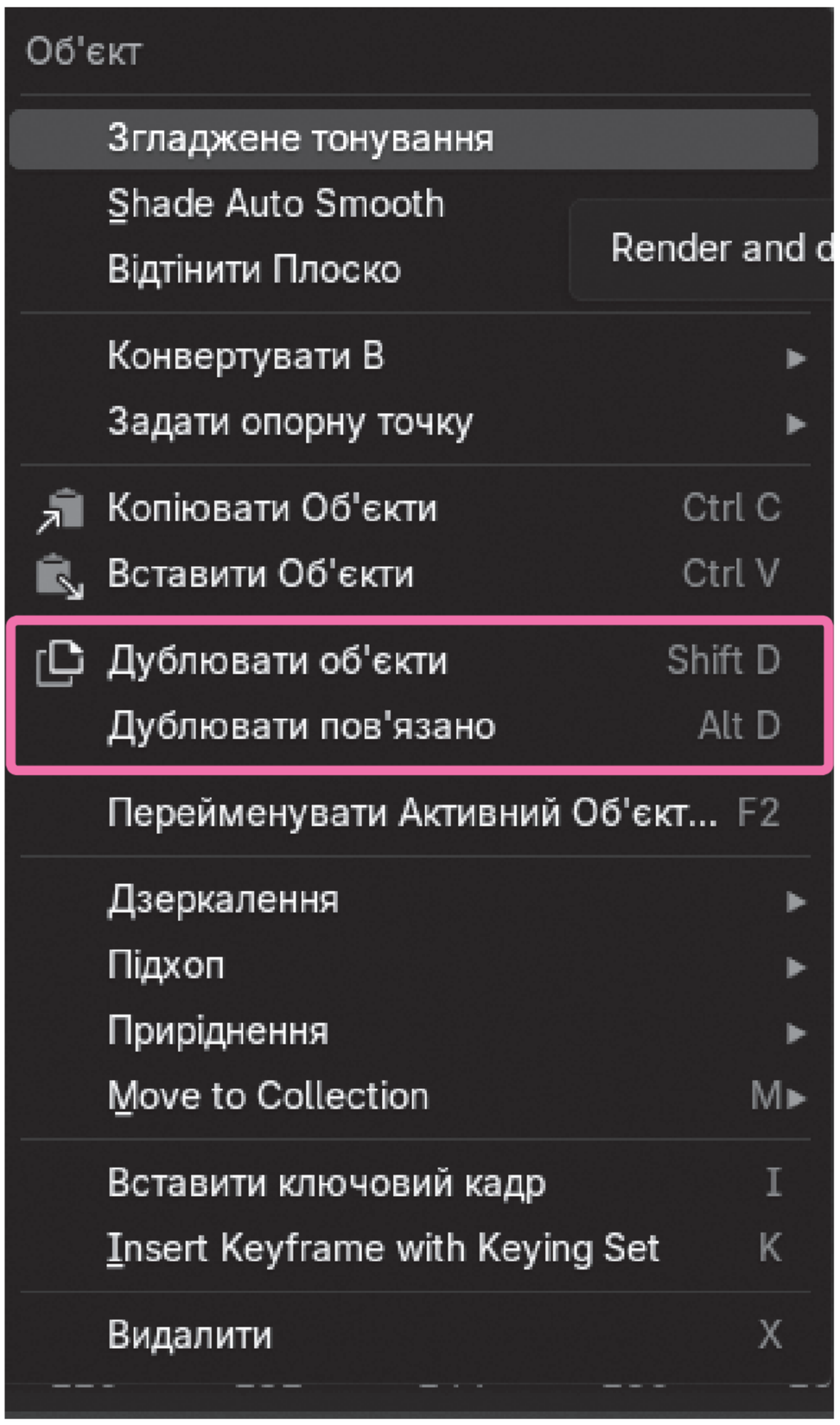
Утримувати **Ctrl** і переміщувати мишу з натиснутим коліщатком.

Як створити копію об’єкта?

У Blender копія об’єкта створюється комбінацією **Shift + D** (незалежна копія) або **Alt + D** (зв’язана копія), що дозволяє ефективно працювати зі складними сценами.

Зв’язана копія — це копія об’єкта, яка ділить одну й ту саму форму (геометрію) з оригіналом. Якщо змінити форму одного з об’єктів (наприклад, у режимі редагування), ці зміни автоматично з’являться у всіх зв’язаних копіях. Водночас положення, обертання та масштаб кожної копії можна змінювати незалежно.

Крім того, копію можна створити через контекстне меню вибраного об’єкта.

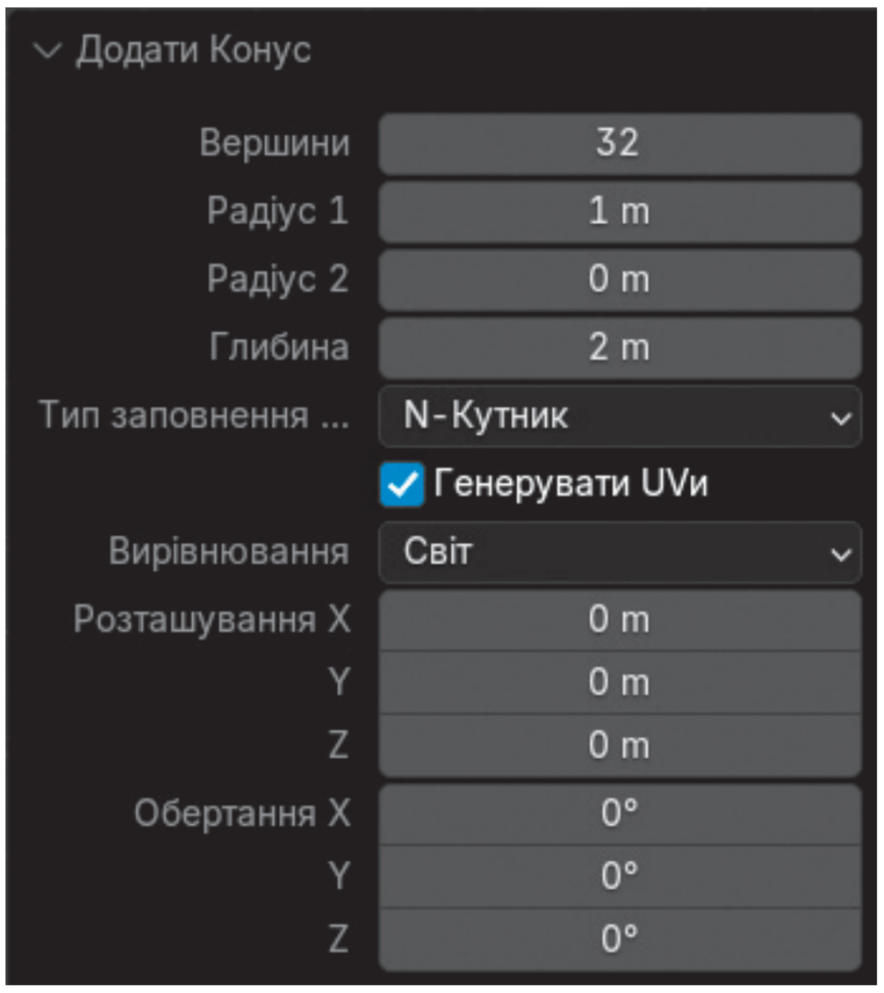


Як змінити гостроту конуса чи кількість граней ікосфери?

Під час створення конуса можна змінити гостроту фігури, встановлюючи значення радіусів основи та вершини в параметрах додавання об’єкта, які з’являються внизу ліворуч у вікні 3D-сцени.

Кількість граней ікосфери визначається параметром «Поділи» у цій самій панелі параметрів.

Якщо панель параметрів зникла, її можна повторно відкрити, натиснувши клавішу **F9** (одразу після додавання об’єкта).



Увага!

Такі параметри, як радіуси конуса та кількість поділів ікосфери, можна змінювати лише одразу після створення об’єкта. Після виконання інших дій ці параметри стають недоступними.

Як згладжують поверхні?

Поверхні створених об’єктів складаються з граней, тому й не є гладкими. Щоб згладити їх, обери **Згладжене тонування** у контекстному меню об’єкта.

Як фарбують поверхні для надання їм реалістичного вигляду?

У сучасних версіях Blender налаштування матеріалів виконують у редакторі властивостей на вкладці **Матеріали**. Тут можна вибрати основний колір поверхні, встановити рівень блиску, шорсткість, прозорість, відбивання світла та інші параметри. Матеріал у Blender схожий на конструктор: кожен блок додає свою властивість — наприклад, колір, шорсткість або блиск, — а разом вони формують вигляд поверхні.

Для створення матеріалу натисни кнопку **Новий**.

Основний колір задається на панелі **Поверхня** у полі **Базовий колір**.

Для створення складнішого вигляду використовують текстури — зображення або процедурні візерунки, що імітують асфальт, траву, пісок, воду, метал, тканину тощо.

Як зберегти свою роботу у файлі?

У Blender можна зберігати проєкт і готове зображення. Це різні типи файлів, тому важливо розуміти їхню відмінність.

Проєкт — це робочий файл із розширенням .blend. У ньому зберігається вся сцена: об’єкти, їхнє розташування, матеріали, освітлення, камера та всі налаштування. Такий файл можна знову відкрити в Blender і продовжити редагування.

Щоб зберегти проєкт, відкрийте меню **Файл** і виберіть команду **Зберегти** або **Зберегти як**. Відкрити збережений проєкт можна за допомогою команди **Відкрити**.

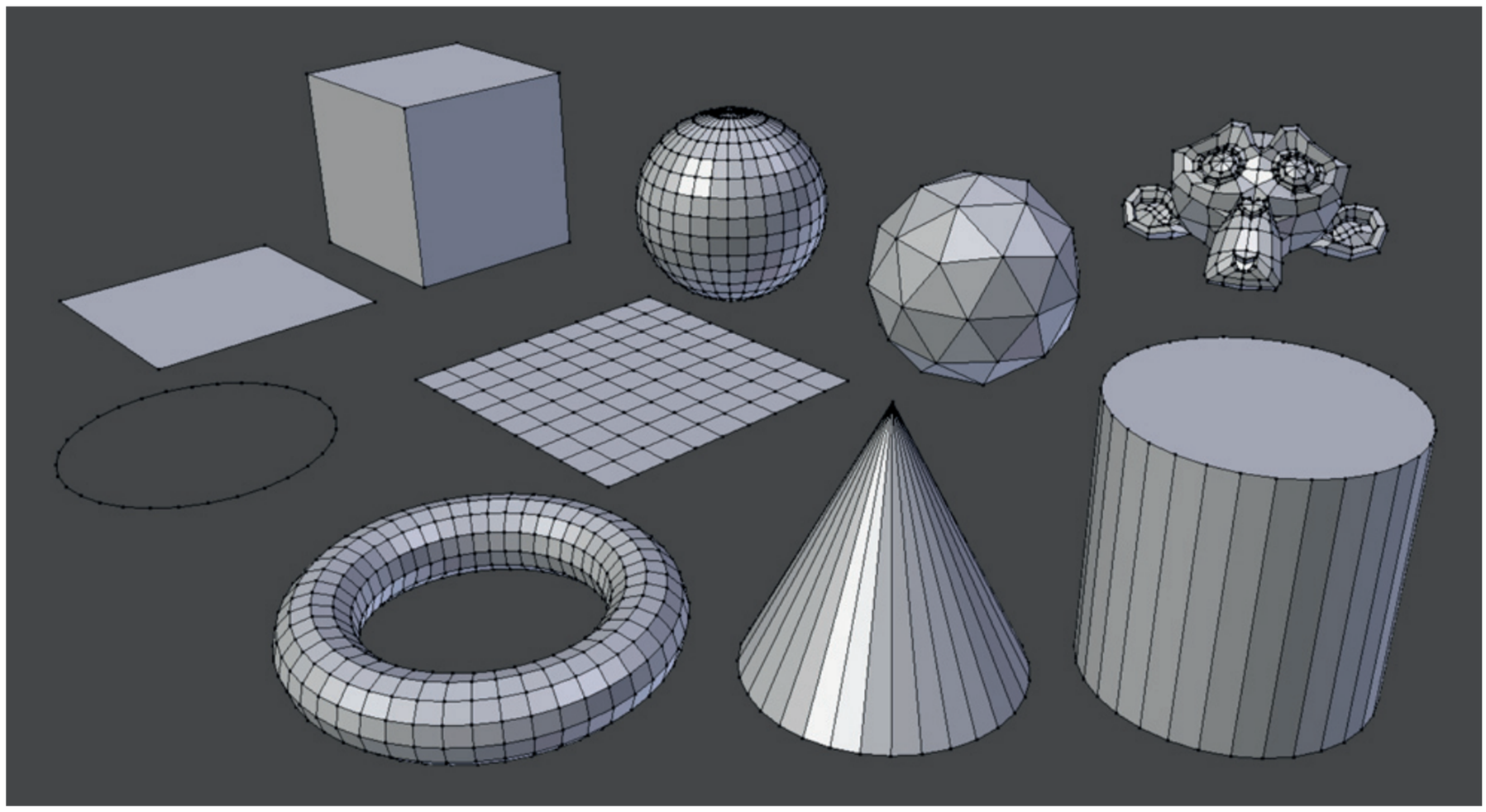
Готове зображення — це результат рендерингу сцени. Воно є плоским (2D) зображенням створеної вами тривимірної моделі. На відміну від файлу .blend, таке зображення вже не містить об’єктів для редагування — його можна лише переглядати або використовувати в інших програмах.

Щоб зберегти готове зображення, спочатку виконайте команду **Рендеринг**. Після завершення рендерингу у редакторі зображень відкрийте меню **Зображення** та виберіть команду **Зберегти як**. За замовчуванням файл зберігається у форматі .png, але можна обрати й інший графічний формат.

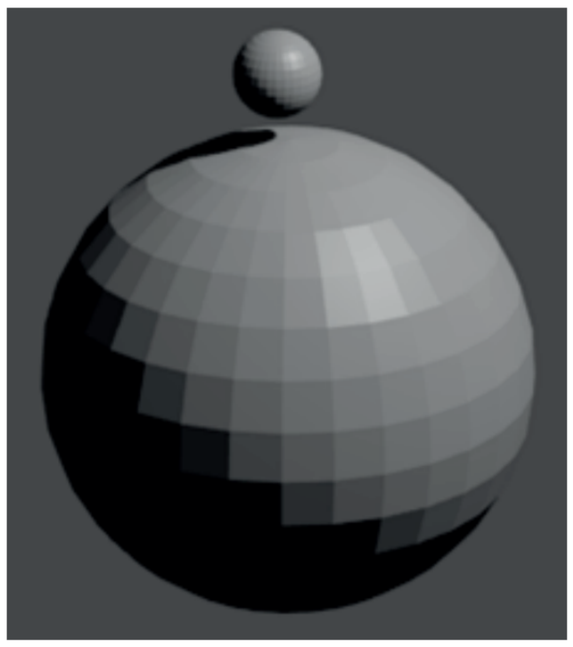
ПЕРЕВІР СЕБЕ

РОЗУМІННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Опиши послідовність дій зі створення та розміщення об’єкта на сцені.
2. Назвіть примітиви, на основі яких створено зображені об’єкти. Де в інтерфейсі Blender можна змінити їхні параметри після створення?



3. Яку команду потрібно виконати, щоб надати поверхні гладкого вигляду?
4. Розглянь малюнок. Яку послідовність команд виконано для створення зображення? Чи можна впевнено стверджувати, що маленька сфера розташована точнісінько над великою? З якого ракурсу потрібно розглянути обидві сцени, щоб визначити розташування центрів сфер? Чому верхня сфера має більш гладкий вигляд?



5. Побудуй два об’єкти за допомогою примітивів UV-Сфера та Ікосфера. Розглянь ці об’єкти в збільшеному вигляді та проаналізуй відмінності між ними.
6. Додай до зображення сніговика руки у вигляді невеликих кульок зі снігу.
7. Оціни створену модель сніговика. Які рішення ти вважаєш вдалими, а що можна було б покращити? Обґрунтуй свою думку, звертаючи увагу на форму, пропорції, матеріали, освітлення та загальний вигляд моделі.

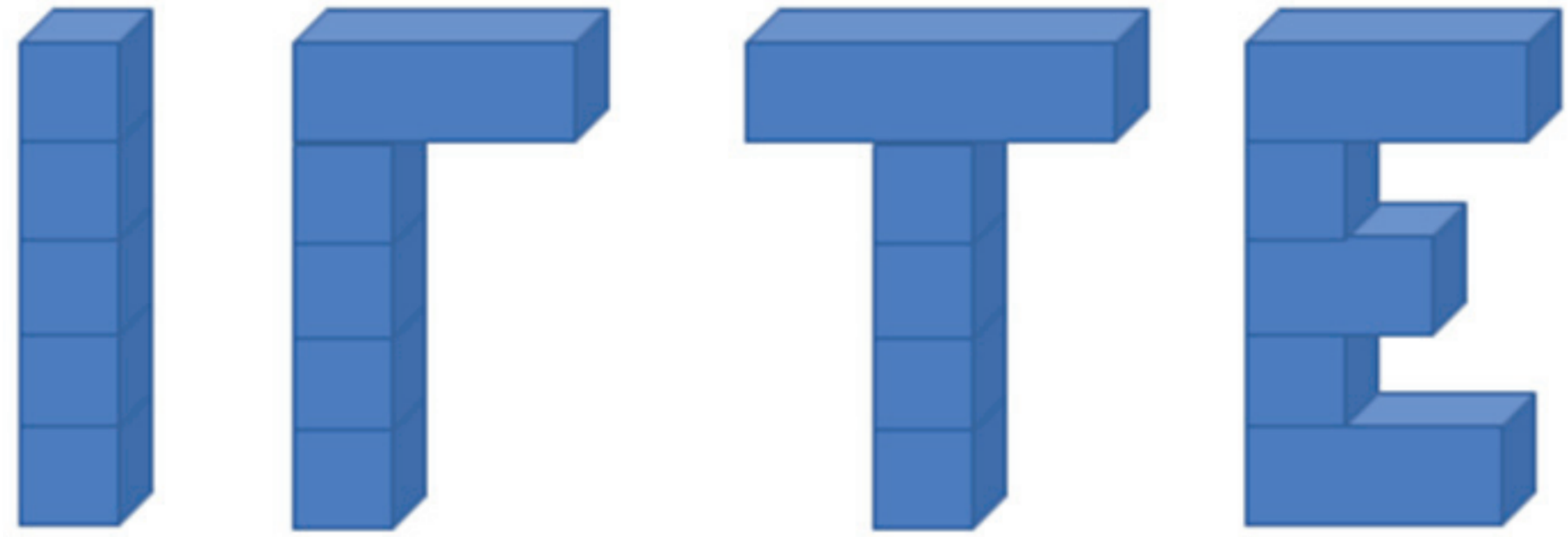
§ 21. Екструдування об’єктів

Як створювати складні геометричні об’єкти?

Для створення деяких зображень у Blender немає стандартних інстру ментів, однак є можливість трансформувати складові простого об’єкта, щоб отримати складніший. Усі об’єкти в Blender мають такі складові: вершини, ребра, грані. Переміщуючи та модифікуючи їх, можна отри-мати практично будь-які 3D-зображення.

Завдання № 1

Уважно подивись на малюнок і поясни, які зміни відбулися з об’єктом. Чи є в Blender інструменти створення об’єктів, за допомогою яких мож-на відразу отримати кожну із зображених літер?



а) б) в) г)

Слово «модифікація» походить від латинського modificatio — зміна, перетворення.

У широкому значенні модифікація — це зміна форми або властивостей об’єкта без втрати його сутності.

У різних галузях це поняття означає:

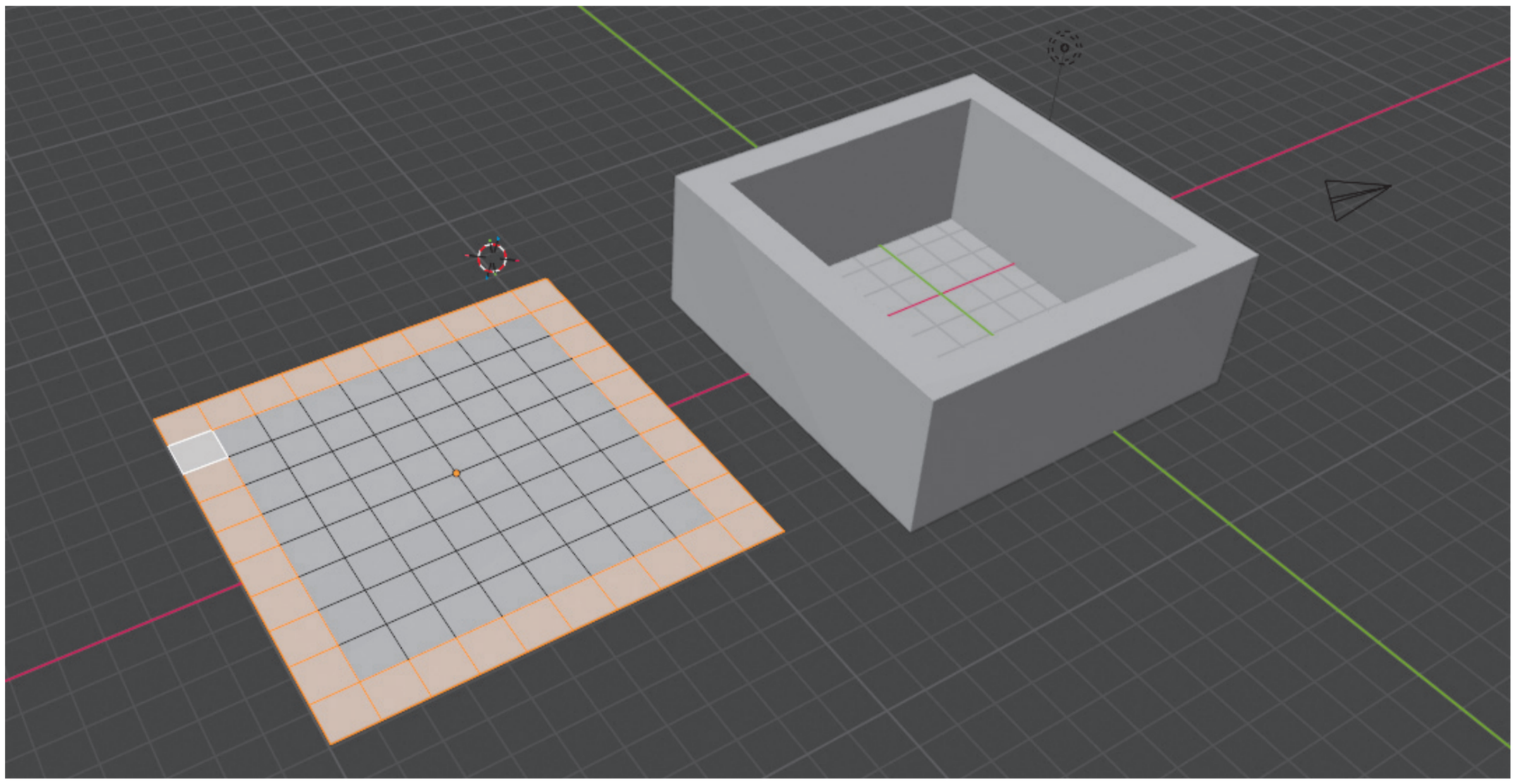
- у повсякденному житті — зміну предмета (наприклад, модернізація пристрою);
- у науці — зміну властивостей явища або об’єкта;
- у біології — зміну ознак організму, яка не передається у спадок.

Поясни, як ти розумієш висловлювання:

- модифікував велосипед;
- модифікував програму;
- модифікував форму намальованої фігури.

Завдання № 2

Об’єднайтесь у пари. Поміркуйте, як з намальованої у Blender поверхні (зліва) отримати зображення ящика (справа).



Скориставшись для цього рубрикою «Запитання-відповіді» (стор. 223), дайте відповіді на запитання:

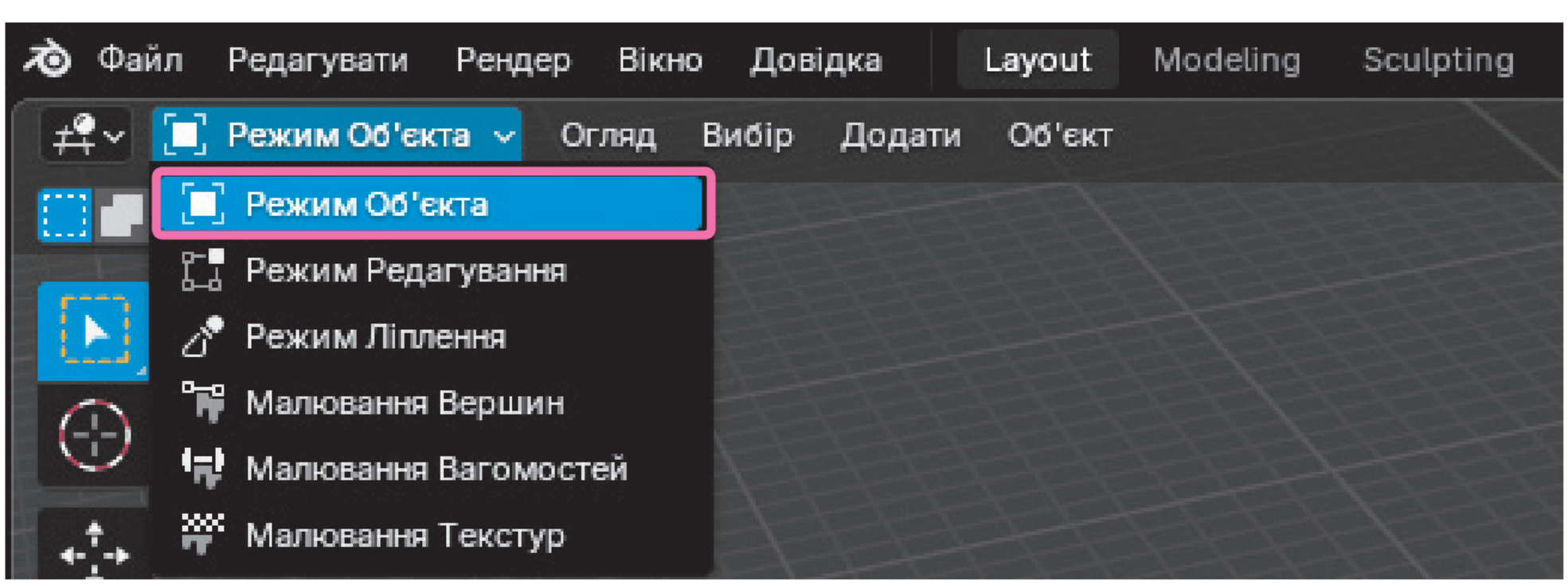
- Яким інструментом створена поверхня на малюнку?
- Що виділено на поверхні помаранчевим кольором?
- Що потрібно зробити з виділенням, щоб отримати стінки ящика?

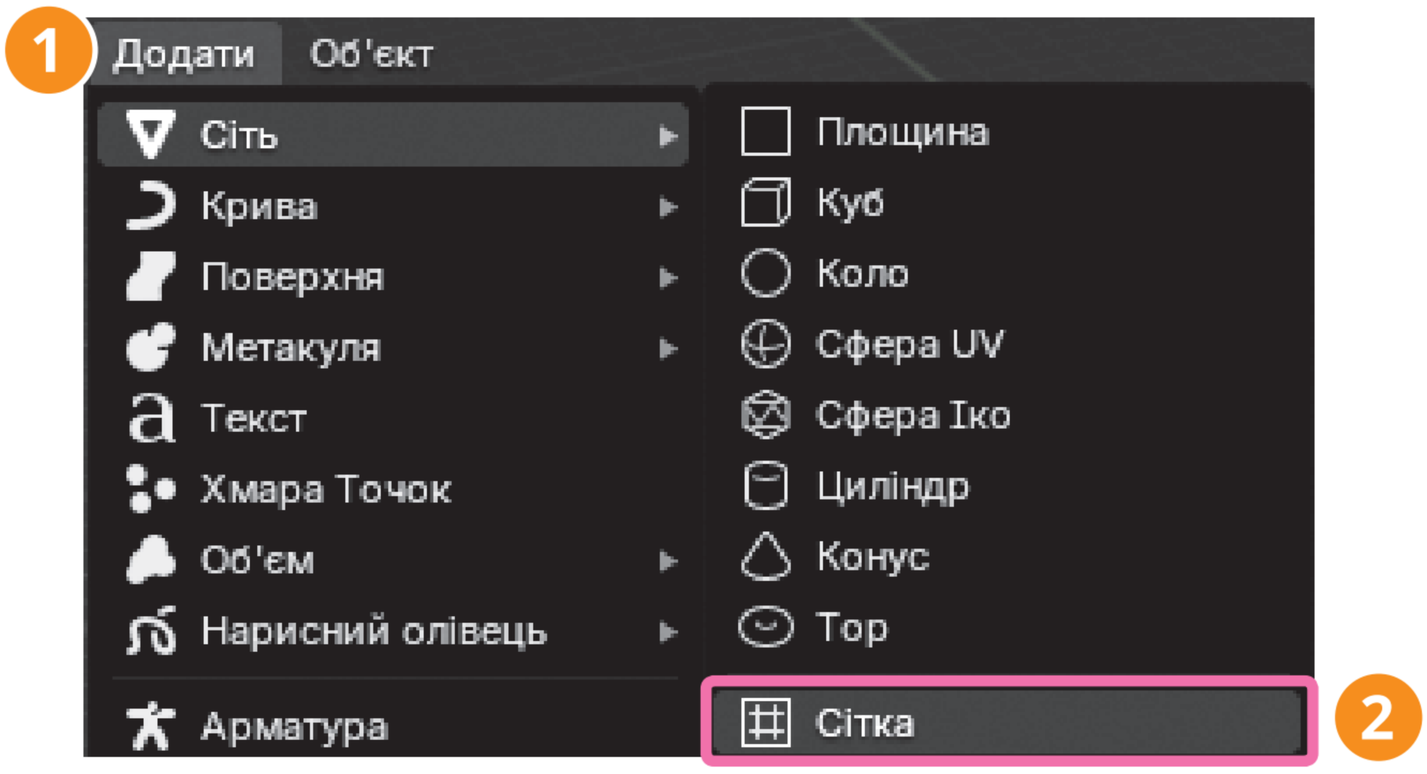
Завдання № 3

Побудуй у редакторі Blender зображення ящика на кшталт того, що на малюнку у завданні 2.

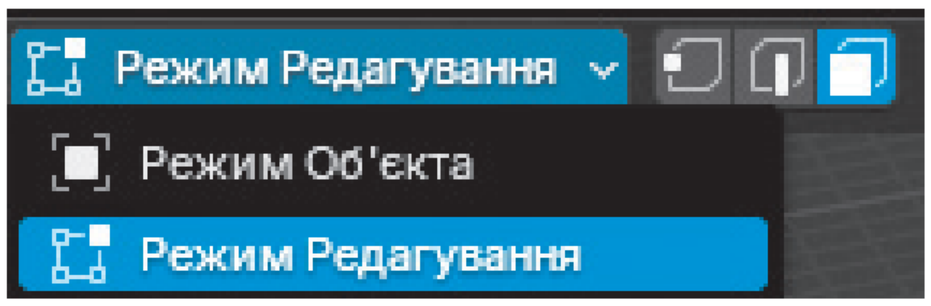
Скористайся інструкцією:

1. Встанови **Режим об'єкта (Object Mode)** та за допомогою меню **Додати** (1) додай об’єкт **Сітка (Grid)** (2).





2. Встанови **Режим редагування**.



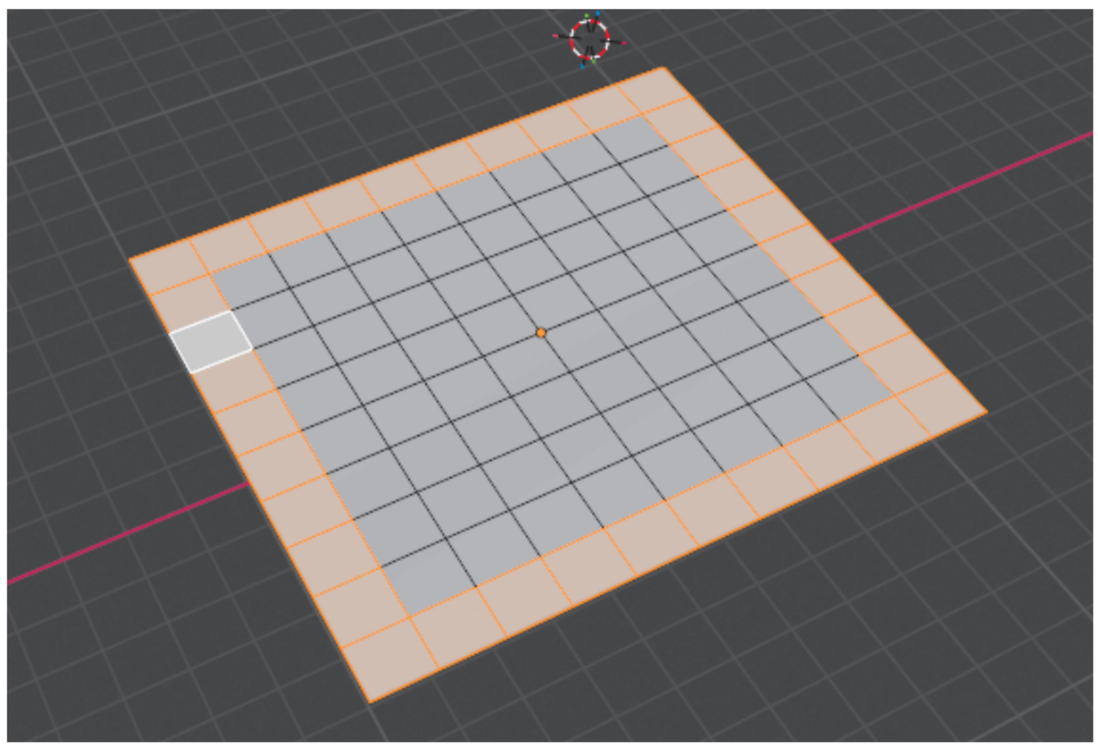
> *Також до режиму редагування можна перейти натиснувши клавішу **Tab**.*

3. Обери **Режим граней** та утримуючи клавішу **Shift** виділи грані сітки по периметру.

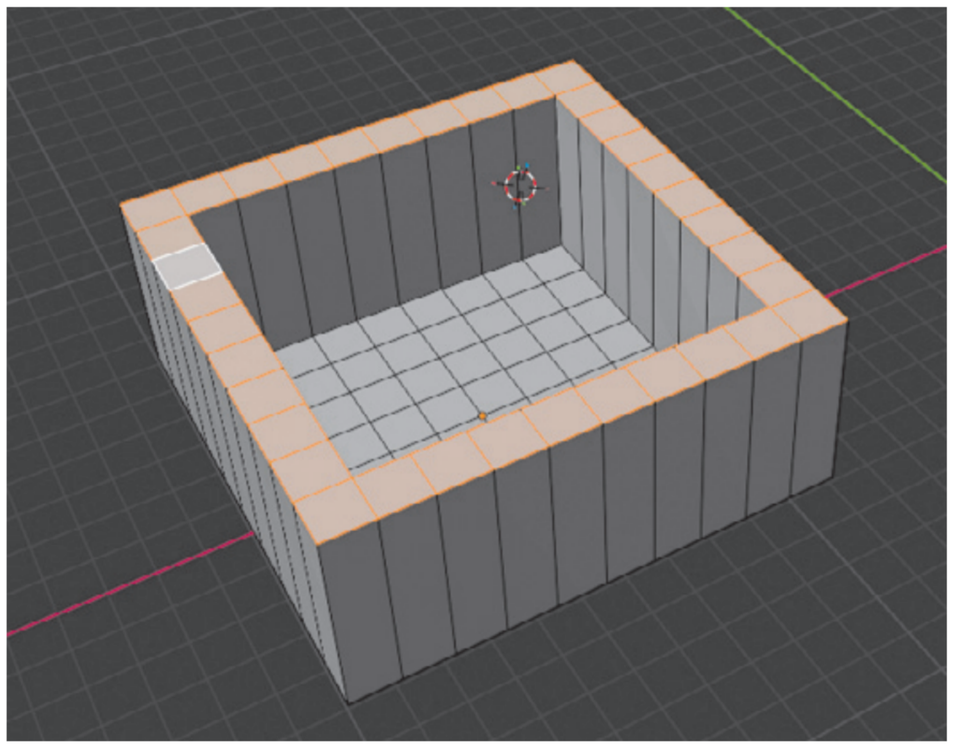
Режим вершин (Vertex Select)
Дозволяє виділяти **окремі вершини.**
Гаряча клавіша **1**

Режим ребер (Edge Select)
Дозволяє виділяти **ребра** (лінії між вершинами)
Гаряча клавіша **2**

Режим граней (Face Select)
Дозволяє виділяти **грані** (поверхні).
Гаряча клавіша **3**



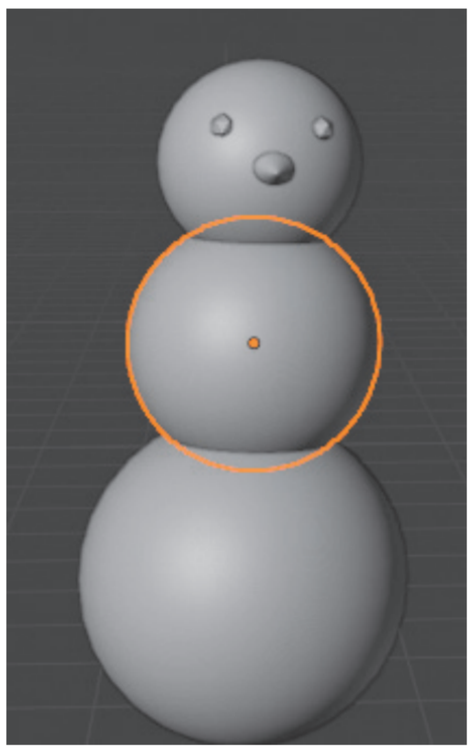
4. Натисни клавішу **E** та «витягни» грані відповідно до зразку. →

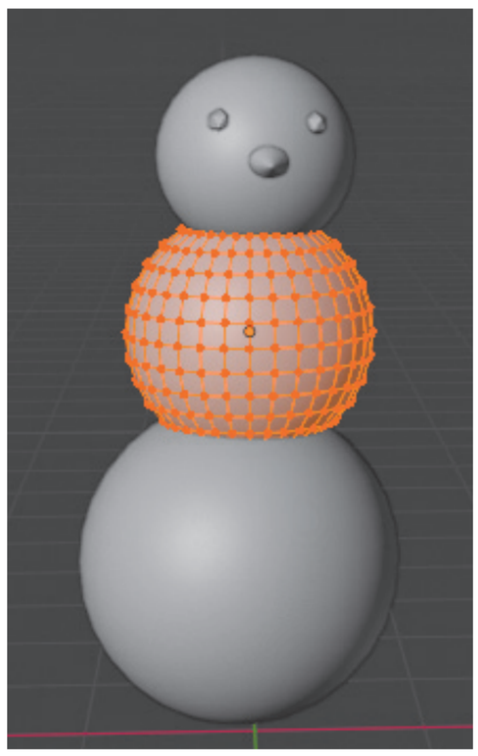


- Розглянь отриманий об’єкт з усіх сторін, опиши його форму.
- Дізнайся, що таке екструдування та як його виконати у Blender за допомогою рубрики «Запитання відповіді» (стор. 223).

Завдання № 4

Відкрий зображення сніговика, створене на попередньому уроці, і за допомогою покрокової інструкції та рубрики «Запитання-відповіді» додайте сніговикові руки-гілки.

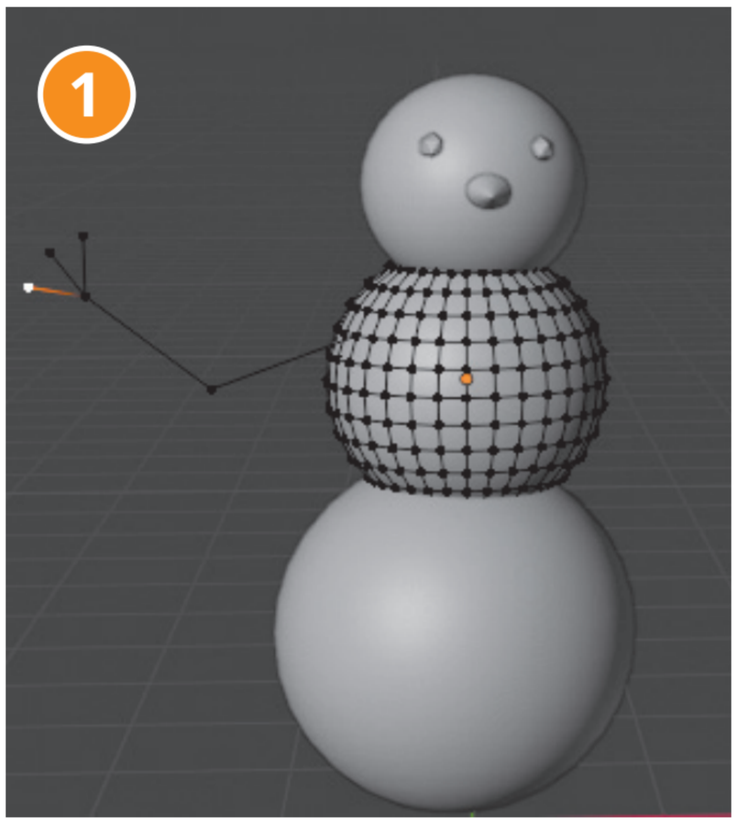


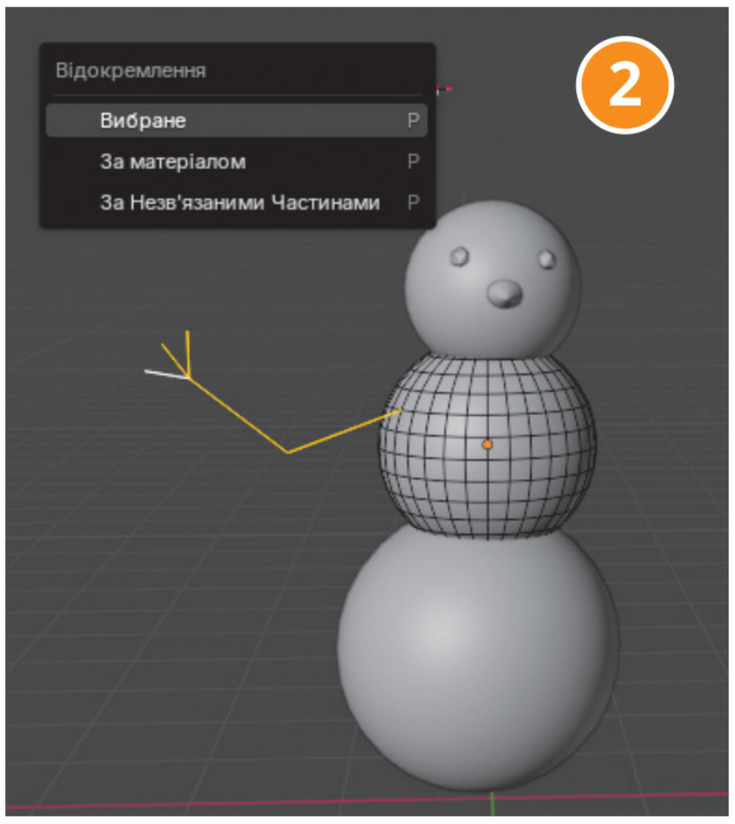


Інструкція:

1. Виділи середню кулю сніговика та увімкни **Режим редагування (Edit Mode)**.
2. Перейди в режим виділення вершин (клавіша **1**) і обери одну у місці, де має починатися рука.
3. Сформуй руку, витягуючи вибрані вершини: натисни **E** (Extrude), за потреби змінюй напрямок і довжину витягування, щоб надати руці бажаної форми.
4. Повтори витягування, поступово формуючи сегменти руки та пальці відповідно до зразка або власного задуму. 1
5. Перетвори форму руки на окремий об’єкт. Для цього:
   - Перейди в **Режим редагування**.
   - Увімкни режим виділення граней (клавіша **2**).
   - Виділи всі грані, з яких складається рука.
   - Натисни **P** і вибери команду **Вибране (Selection)** або ті самі дії можна виконати з контексного меню.

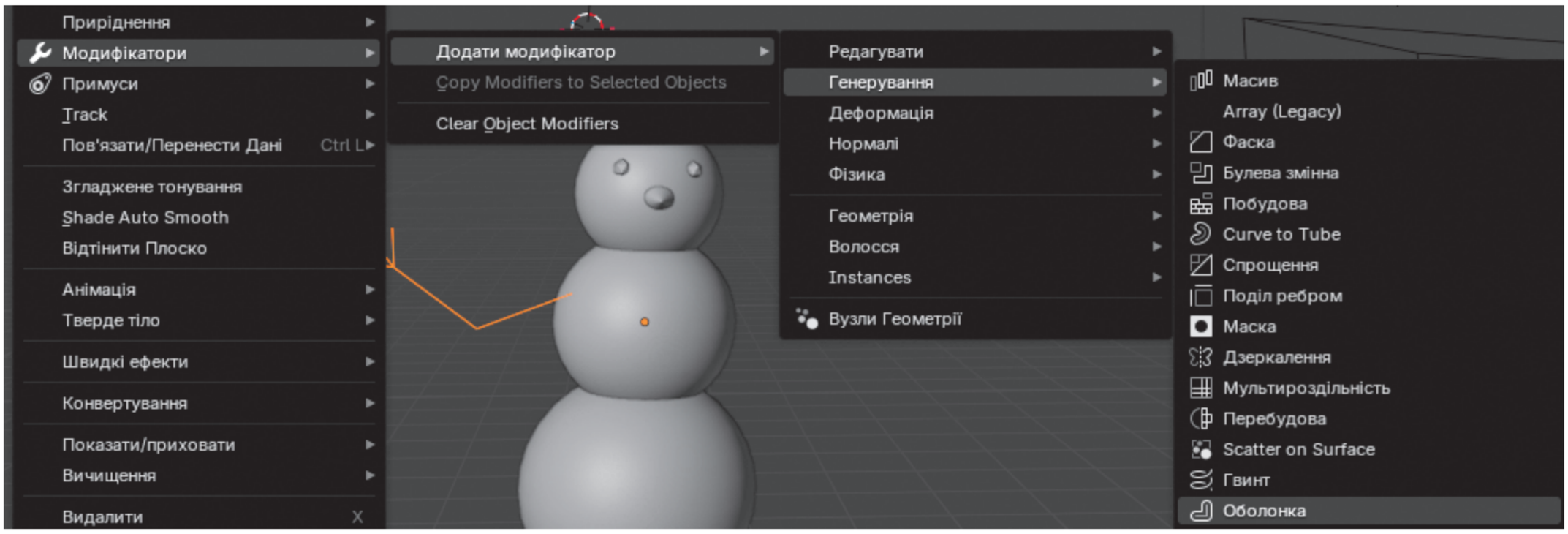
   У результаті рука стане окремим об’єктом, з яким можна працювати незалежно від тулуба сніговика 2





*Команда **P** → **Вибране** дозволяє відокремити частину сітки в новий об'єкт, не порушуючи геометрію решти моделі.*

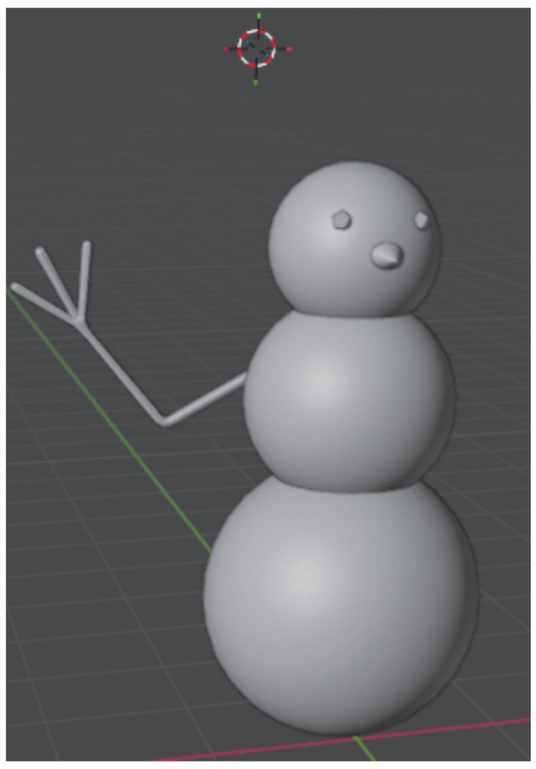
6. Перейди в **Режим об'єкта** і послідовно застосуй модифікатори **Оболонка** та **Поділ поверхні**.



***Модифікатори** — це інструменти, які автоматично змінюють форму або вигляд об'єкта, не руйнуючи його початкову геометрію.*

Після додавання модифікатора **Оболонка** рука спочатку виглядатиме як тонка форма й ще не буде схожою на гілку. Щоб надати їй природного вигляду, потрібно:

1) Перейти в **Режим редагування**.
2) Виділити вершини гілки.
3) Натиснути **Ctrl + A** і, рухаючи мишу, задати товщину: біля тулуба — більшу, на кінчиках — меншу.
4) Повернутися в **Режим об'єкта** та додати модифікатор **Поділ поверхні** для згладжування форми.



У результаті рука набуде об'ємної, плавної форми, подібної до справжньої дерев'яної гілки.

7. Розфарбуй руку на власний розсуд, застосувавши матеріал у редакторі властивостей.
8. Створи копію руки, віддзеркаль її та розташуй симетрично відносно першої.
9. Збережи проєкт.

ЗАПИТАННЯ-ВІДПОВІДІ

Які є режими редагування у Blender?

У Blender передбачено кілька режимів роботи, які визначають, що саме ми можемо редагувати.

Режим об’єкта (Object Mode) використовується для роботи з об’єктами загалом: їх переміщення, обертання, масштабування, копіювання та застосування модифікаторів.

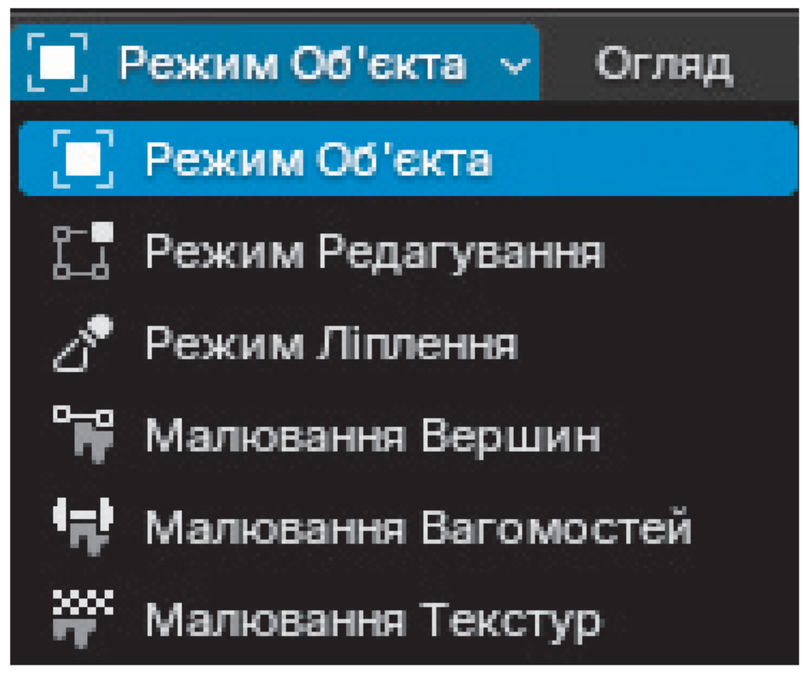
Режим редагування (Edit Mode) дозволяє змінювати форму об’єкта. У цьому режимі редагують його складові — вершини, ребра та грані.

Режим ліплення (Sculpt Mode) призначений для художнього формування поверхні, ніби з глини.

Режим фарбування текстур (Texture Paint) використовується для розфарбовування поверхні об’єкта безпосередньо на моделі.

Режим перегляду камери (Camera View) показує сцену так, як вона виглядатиме під час рендерингу.

i *Режим у Blender визначає, на якому рівні ми працюємо: з об’єктом цілком, з його формою або з виглядом поверхні.*



Як вибирають складові об’єкта для подальших трансформацій?

У **Режимі редагування** над робочою областю розташована група кнопок для вибору вершин, ребер або граней.

Це вибір того, що саме ти будеш редагувати:

- Вершини — коли потрібно потягнути окрему точку;
- Ребра — коли треба змінити край;
- Грані — коли працюєш з ділянкою поверхні.

Щоб виділити вершину, ребро або грань, потрібно:

1. Обрати відповідний режим виділення.
2. Клацнути лівою кнопкою миші по потрібній складовій об’єкта.

Як виділити кілька вершин або граней?

Існує кілька способів:

- Утримуючи клавішу **Shift** — для додавання до виділення окремих складових.
- Прямокутною областю (клавіша **B**) — натисни **B** і обведи прямокутником потрібну ділянку.
- Круглою областю (клавіша **C**) — натисни **C** і води мишкою, ніби пензлем; коліщатком миші змінюй розмір круга.

Що таке екструдування і як його виконати?

Екструдування (від англ. *Extrude* — витягування) — це спосіб зміни форми об'єкта шляхом створення нових складових із уже наявних. Наприклад, з плоскої сітки можна отримати об'ємну форму, а з окремої грані — виступ або заглиблення.

Екструдування виконують у **Режимі редагування** і застосовують до виділених складових об'єкта: вершин, ребер або граней. Щоб виконати екструдування, спочатку потрібно виділити потрібні складові, натиснути клавішу **E**, а потім, рухаючи мишу, задати напрямок і довжину витягування.

Чому для отримання тривимірного об'єкта використовують об'єкт «Сітка (Grid)»?

Поверхня може складатися лише з однієї грані, тоді як сітка (grid) містить багато граней, з'єднаних між собою в єдину структуру. Завдяки цьому під час екструдування окремих граней або їхніх ділянок можна формувати різноманітні об'ємні форми.

Як звести всі вершини об'єкта в одну точку?

У Режимі редагування виділи потрібні вершини, а потім скористайся командою **Об'єднання вершин (Merge)**, обравши один із варіантів: **До центру**, **До курсора** або **Колапс**.

Навіщо це потрібно?

Зведення вершин в одну точку використовують, коли потрібно:

- створити початкову точку для подальшого витягування форми. Наприклад, для створення гілки-руки сніговика можна додати об'єкт Куб, звести всі його вершини в одну точку, а потім за допомогою екструдування поступово побудувати геометрію гілки.;
- спростити геометрію, об'єднавши зайві вершини;
- підготувати основу для модифікаторів або екструдування;
- виправити помилки моделювання, коли вершини накладаються одна на одну.

Як організувати роботу з об’єктами у Blender за допомогою колекцій?

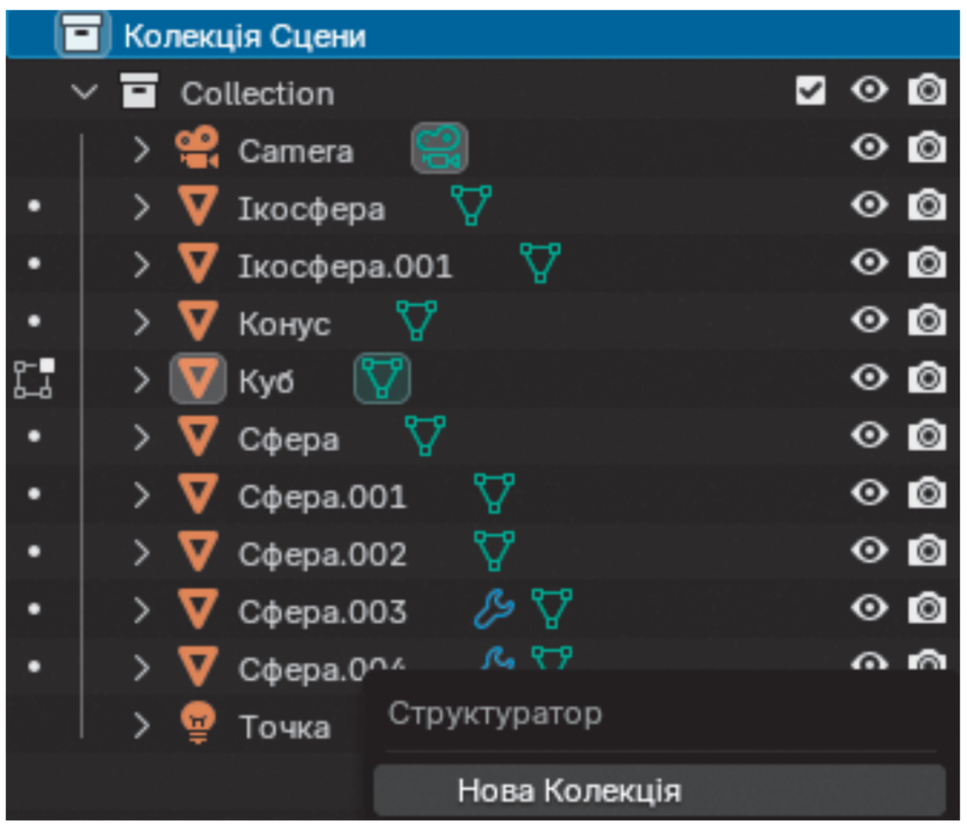
Під час створення тривимірних моделей сцена може містити багато об’єктів. Щоб зручно організувати роботу, у програмі Blender використовують колекції.

Колекції виконують роль груп об’єктів (аналог шарів у графічних редакторах) і дозволяють впорядковувати сцену за призначенням об’єктів.

Завдяки колекціям можна:

- групувати об’єкти (наприклад, будівлі, рослини, персонажів);
- тимчасово приховувати частини сцени;
- працювати лише з потрібними об’єктами, не заважаючи іншим;
- керувати тим, які об’єкти будуть видно під час рендерингу.

Колекції відображаються у правій частині вікна програми — в панелі **Список об’єктів (Outliner)**.



Як створити нову колекцію?

1. Відкрий панель **Outliner**.
2. Натисни праву кнопку миші.
3. Обери команду **Нова колекція**.
4. Задай їй назву.

Як перемістити об’єкт до колекції?

1. Виділи об’єкт у сцені.
2. Натисни клавішу **M**.
3. Обери потрібну колекцію зі списку.

Керування колекціями

Поруч із назвою кожної колекції є значки:

око — показати або приховати об’єкти колекції;

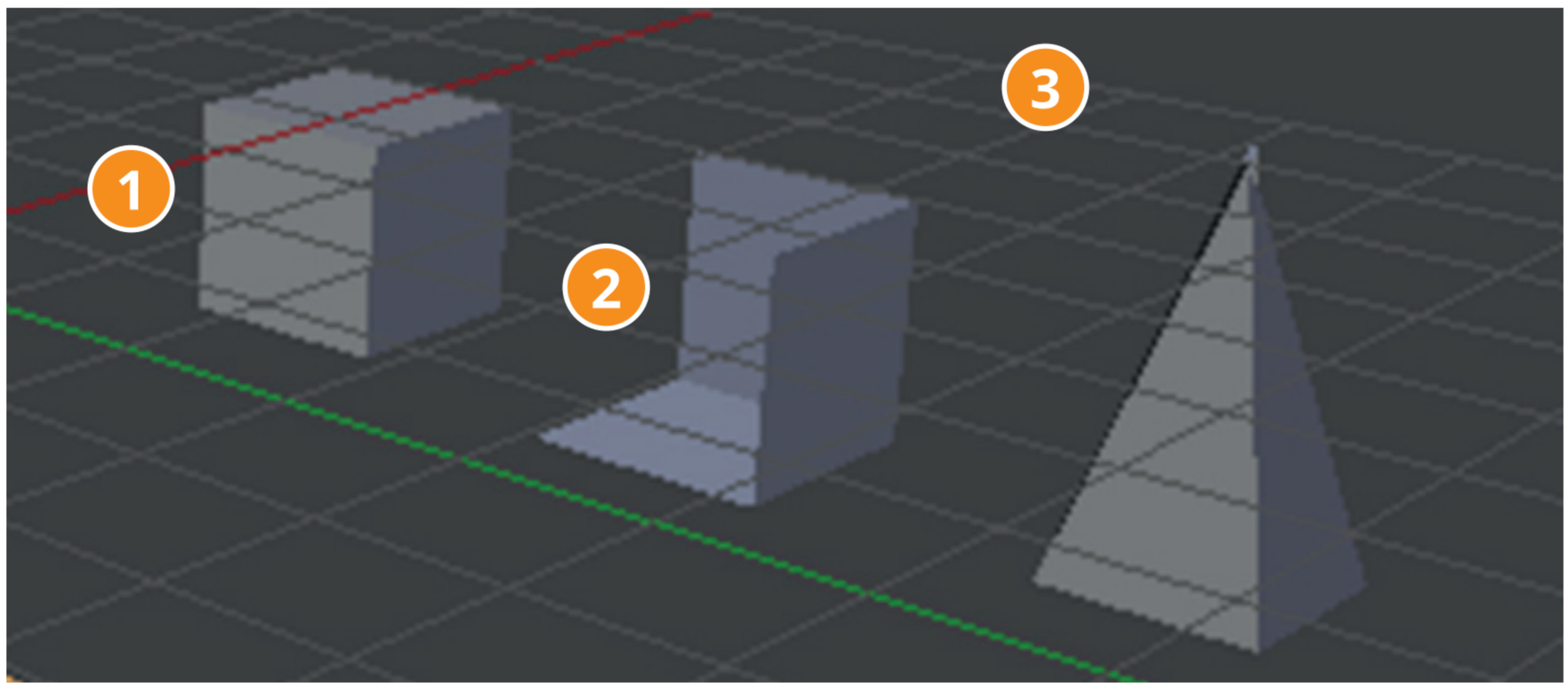
курсор — дозволити або заборонити вибір об’єктів;

камера — визначити, чи буде колекція відображатися під час рендерингу.

ПЕРЕВІР СЕБЕ

РОЗУМІННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Які є складові об’єкта та способи їхнього виділення?
2. Чим відрізняється виділення об’єкта від виділення його складових (вершин, ребер, граней)?
3. Поясни, що таке модифікатор у Blender і чим він відрізняється від редагування сітки вручну.
4. Для чого в Blender використовують різні режими роботи з об’єктами: об’єктний і редагування?
5. У яких випадках зручніше працювати в режимі об’єкта, а в яких — у режимі редагування? Наведи приклади.
6. Чому для створення об’ємних форм частіше використовують об’єкт типу «Сітка (Mesh)», а не інші типи об’єктів?
7. На малюнку показано куб (1) та дві його копії (2 і 3), які було трансформовано для отримання зображень 2 і 3 із зображення 1? Перевір відповідь на практиці.



8. Використай інструмент **Матеріал** для зафарбування однієї з побудованих у попередньому завданні фігур.
9. Побудуй зображення стола на чотирьох ніжках.
10. Порівняй можливості опрацювання об’єктів у Blender та в інших відомих тобі редакторах. Як ти вважаєш, чи зручним є механізм екструдування? Які його недоліки?
11. Які інструменти Blender здалися тобі найскладнішими? Чому саме?
12. Як ти вважаєш, що важливіше під час 3D-моделювання: точність чи творчість? Обґрунтуй відповідь.

§ 22. Змінення форми об’єкта

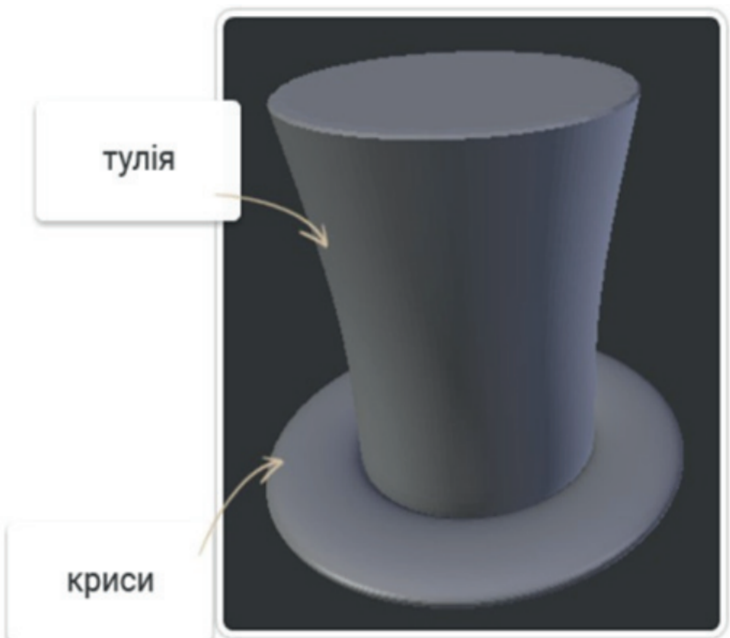
Чи завжди складні 3D-об’єкти створюють з нуля, чи їх можна отримати зі простих форм?

Створений на попередніх уроках сніговик достатньо реалістичний, однак чогось йому не вистачає. Зробимо сніговикові капелюх! Причому не простий, а у вигляді циліндра, такого як носять із фраком. Циліндр — це складений об’єкт, який можна створити з кількох простих, застосовуючи різні операції в режимі редагування.

Завдання № 1

Розглянь зображення капелюха справа.

- Як називаються частини капелюха?
- Які примітиви найкраще підходять для створення цих частин?
- Що потрібно змінити в кожному з обраних примітивів, щоб отримати форму тулії та крисів?



Завдання № 2

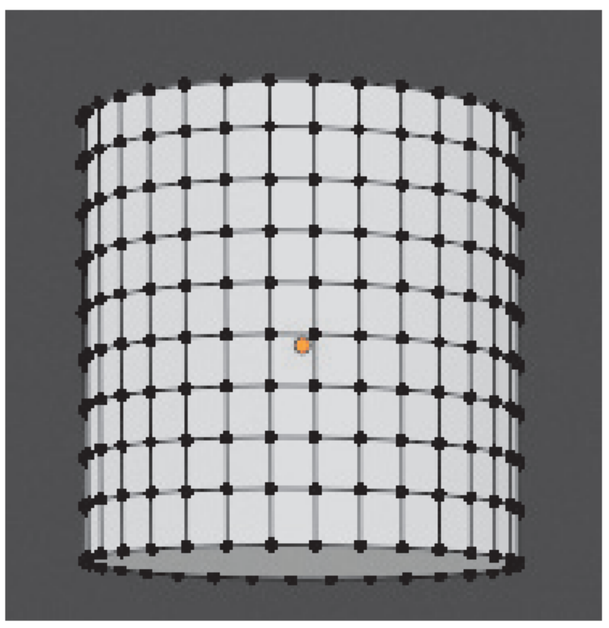
Об’єднайтесь у пари та побудуйте капелюх для сніговика скориставшись інструкцією.

Інструкція:

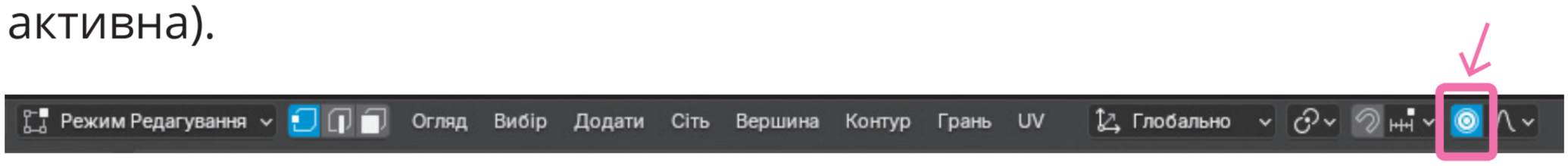
1. Відкрий файл з малюнком сніговика.
2. Створи нову колекцію «Шляпа».
3. Додай новий примітив Циліндр.
4. Перетвори циліндр на тулію шляпи.

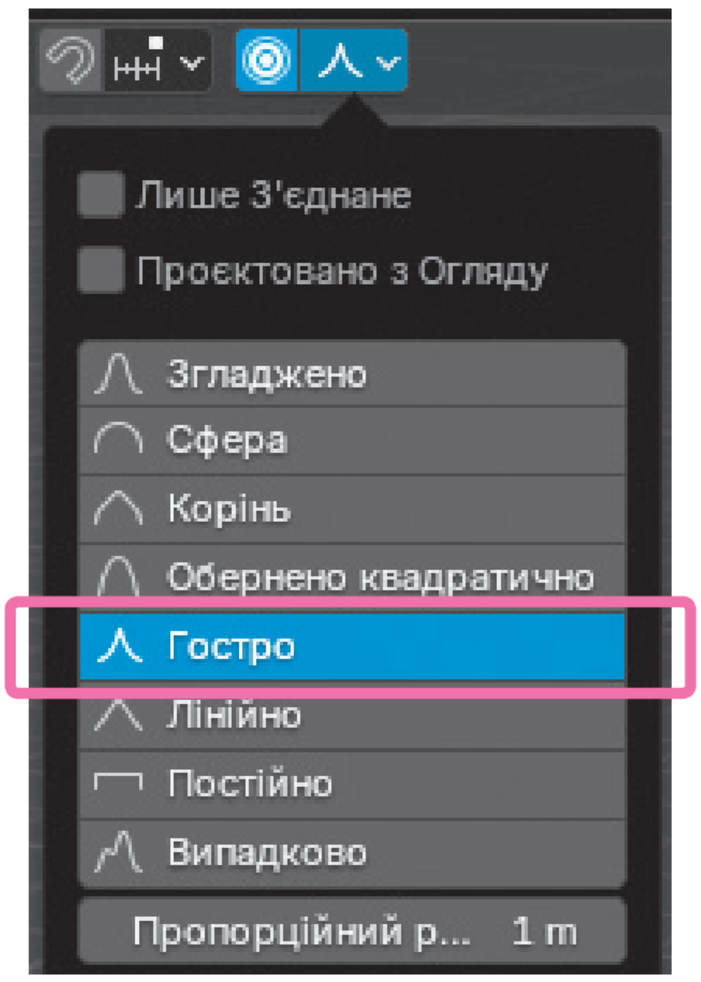
Додай горизонтальні розрізи:

- Натисни **Ctrl + R**;
- Наведи курсор на бік циліндра — з’явиться жовта лінія розрізу;
- Прокрути коліщатко миші (вгору/вниз), щоб змінити кількість розрізів;
- Один клік лівою кнопкою миші — підтверджує кількість;
- Другий клік лівою кнопкою миші — фіксує положення розрізів.



5. Увімкни пропорційне редагування:
   Натисни клавішу **O** (у верхній панелі переконайся, що іконка кола активна).

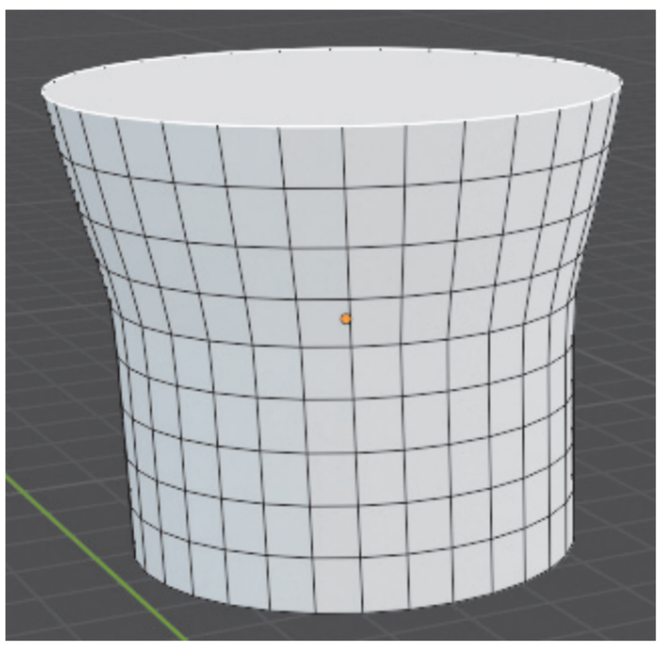




Інші режими:

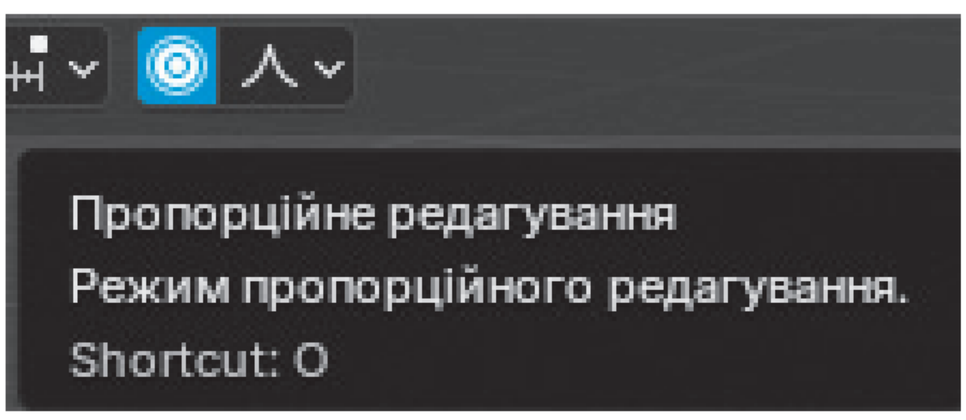
- **Лінійно** — м’яка, рівномірна деформація (добре для плавних форм).
- **Сфера** — округлі, «надути» об’єми (обличчя, кулі).
- **Згладження** — пом’якшення нерівностей.
- **Корінь** — дуже плавний перехід, рідко потрібен на старті.
- **Випадково** — для органічних, «рваних» форм (не для цієї задачі).

6. Зміни форму тулії:
   - Перейди в режим виділення граней (клавіша **3**);
   - Виділи верхню грань циліндра;
   - Натисни **S** (масштабування) і збільш трохи верх.

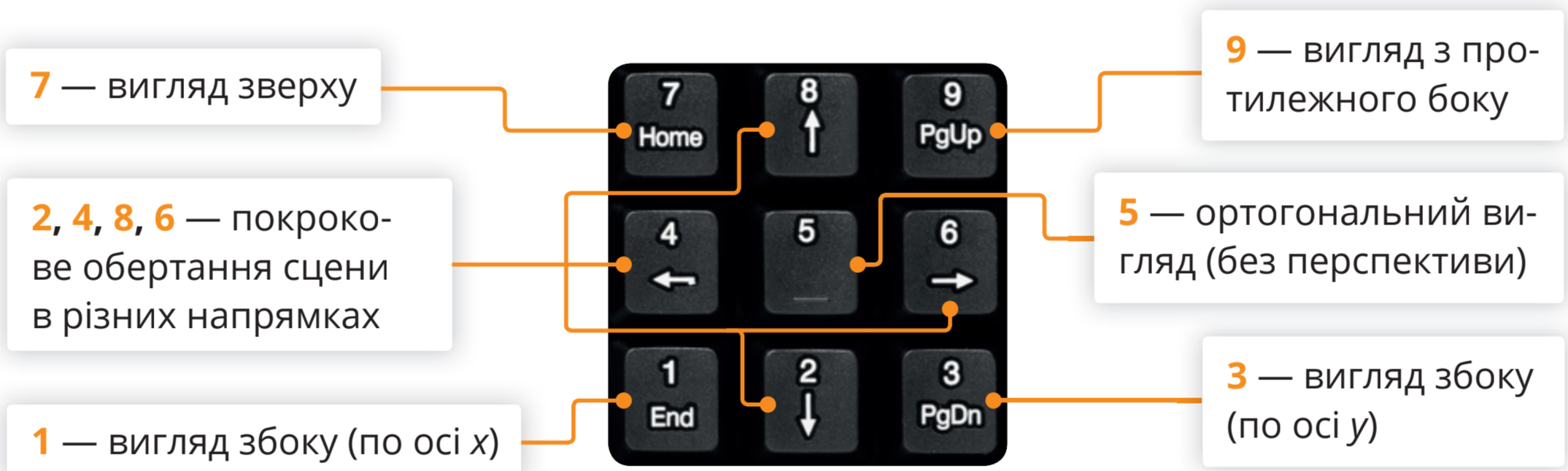


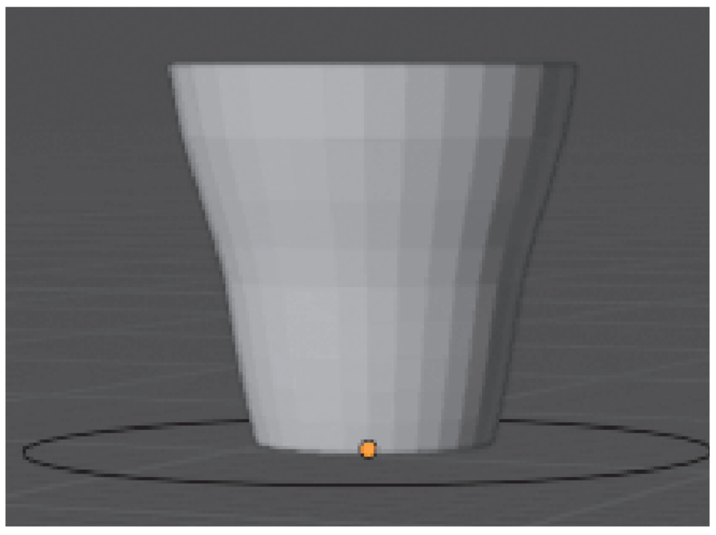
За потреби: **G + Z** — зсунути вгору/вниз, коліщатком — керувати плавністю переходу. Тулія має трохи звужуватись униз і розширюватися вгору.

Пропорційне редагування забезпечує плавну зміну форми об’єкта: під час трансформації виділені вершини змінюються найбільше, а сусідні — у меншій мірі, з поступовим зменшенням впливу з відстанню.

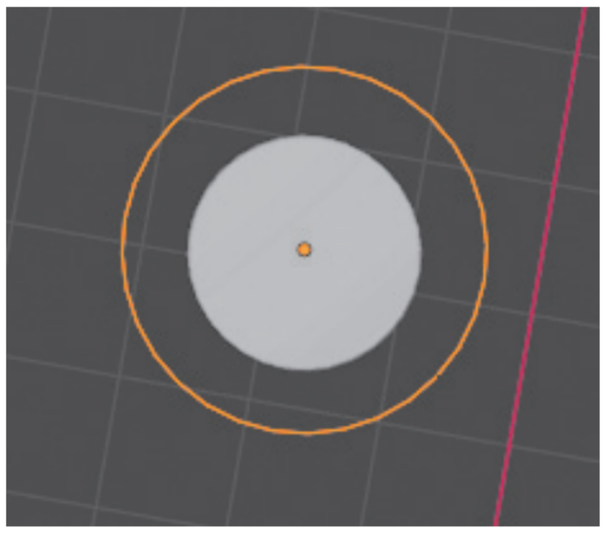


7. Перейди в **Режим об'єкта (Object Mode)** і додай новий об’єкт Коло.
8. Вирівняй коло точно під тулією.
   Це зручно зробити використовуючи ортогональний перегляд:

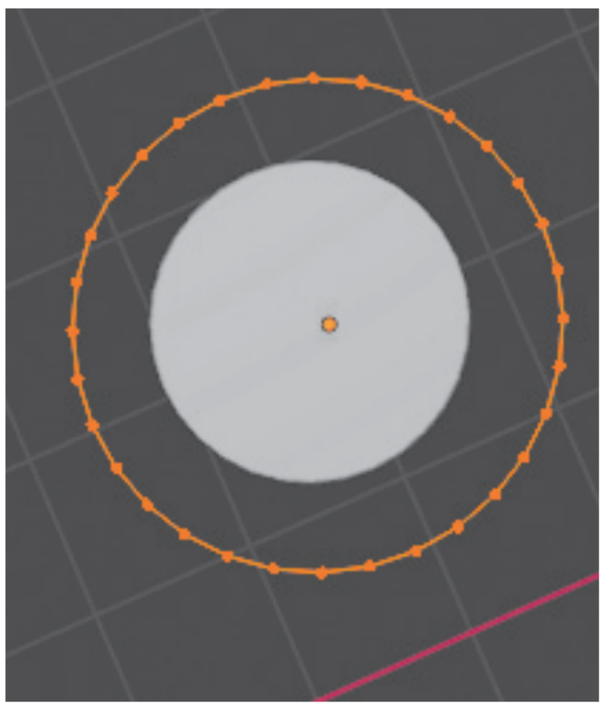




Ортогональний вигляд справа

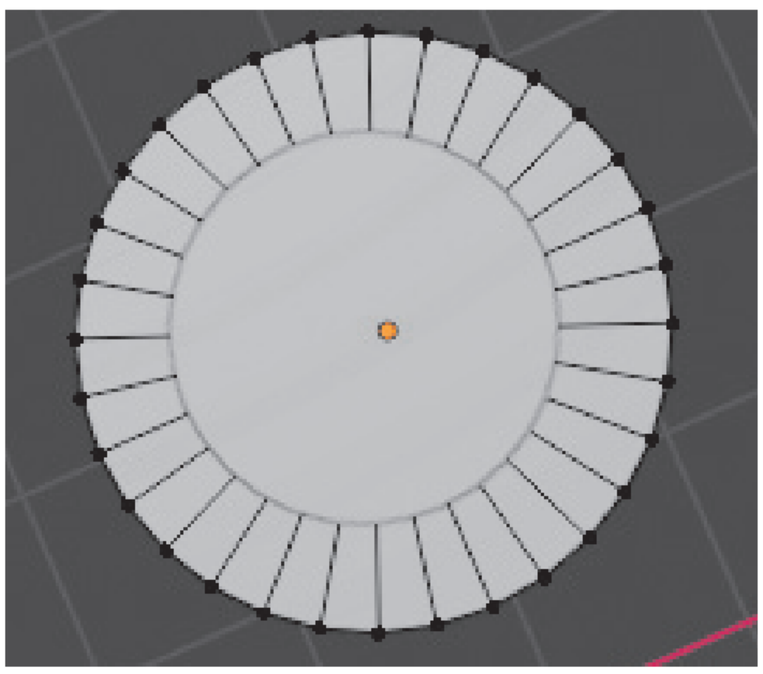


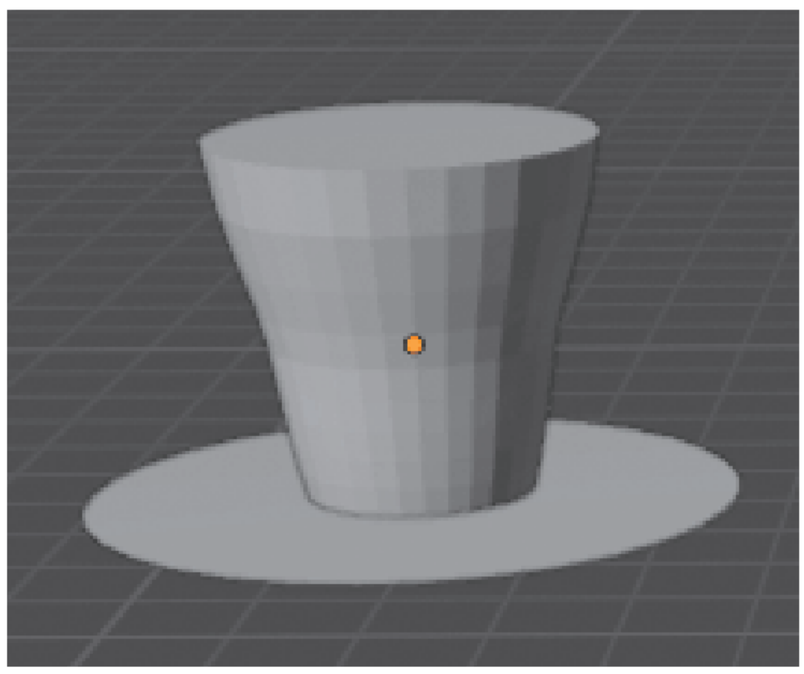
Ортогональний вигляд збоку



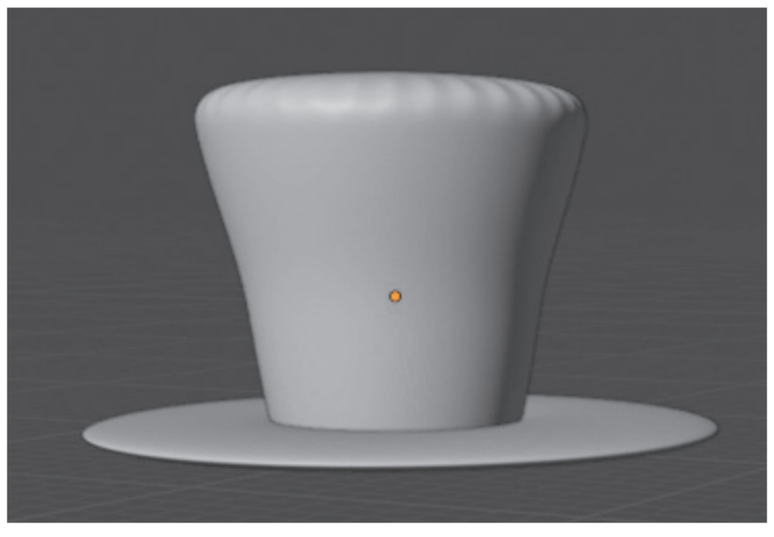
9. Перейди в режим редагування кола (редагування вершин) та натисни **A**, щоб виділити всі вершини.

10. Зроби криси об'ємними: натисни **E** (екструдувати) та одразу натисни **S** (масштаб).

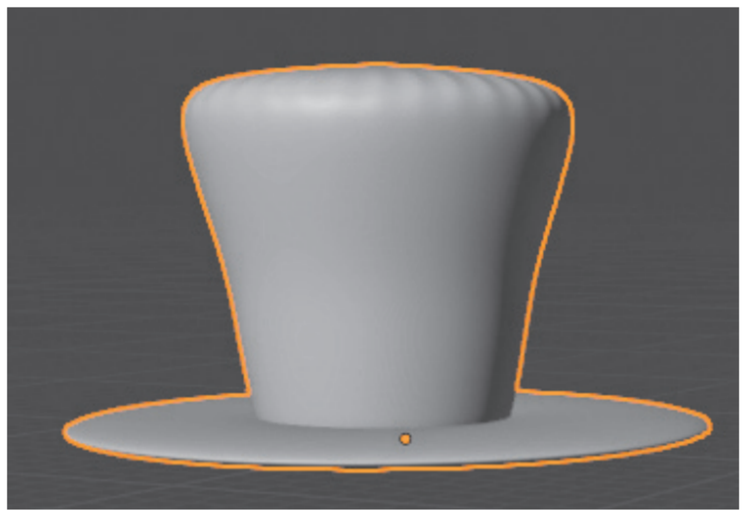




11. Щоб надати капелюху правдоподібнішого вигляду, застосуйте до тулії та до крисів модифікатори:
    - потовщення (оскільки реальний капелюх має товщину);
    - поділ поверхні (щоб зробити поверхні гладкішими);
    - поділ ребром (тільки до тулії, щоб зробити її верхній край «гострим»).



12. Об’єднай тулію і криси у один об’єкт:
    - Перейди в **Режим об’єкта (Object Mode)**;
    - Виділи перший об’єкт;
    - Утримуючи **Shift**, виділи другий об’єкт;
    - Натисни **Ctrl + J**.



13. Розфарбуй та одягни капелюх на сніговика.
14. Збережи проєкт.

ЗАПИТАННЯ-ВІДПОВІДІ

Для чого в Blender додають розрізи (петлі ребер) до 3D-об’єктів?

Додавання розрізів (петель ребер) — це спосіб ущільнити сітку об’єкта, щоб мати більше вершин і ребер для точного редагування форми. Під час цього об’єкт не розділяється на частини, а лише отримує додаткові лінії, які дозволяють:

- точніше змінювати форму об’єкта;
- створювати плавні або різкі переходи;
- підготувати модель до екструдування, згладжування та інших перетворень.

Як додати розрізи (петлі ребер) в Blender?

1. Перейдіть у **Режим редагування**.
2. Натисніть **Ctrl + R** (інструмент Loop Cut).
3. Наведіть курсор на поверхню об’єкта — з’явиться лінія розрізу.
4. Прокрутіть коліщатко миші, щоб змінити кількість розрізів.
5. Клацніть лівою кнопкою миші для підтвердження та ще раз — щоб зафіксувати положення.

Як у Blender використовують модифікатори для зміни форми об’єктів?

Модифікатори — це інструменти Blender, які автоматично змінюють форму або вигляд об’єкта без руйнування його початкової геометрії. Вони працюють як «шари ефектів»: об’єкт залишається тим самим, але його вигляд змінюється в реальному часі.

Головна перевага модифікаторів — завжди можна повернутися назад або змінити параметри, не перемальовуючи модель.

Як збільшити товщину стінок об'єкта?

Для потовщення поверхні використовують модифікатор **Потовщення (Solidify).**

Алгоритм дій:

1. Перейдіть у **Режим об'єкта**.
2. Виділіть потрібний об'єкт.
3. У **Редакторі властивостей** відкрийте вкладку **Модифікатори**.
4. Натисніть **Додати модифікатор** → **Потовщення (Solidify)**.
5. Змініть параметр **Товщина (Thickness)**:
   - додатне значення — потовщення назовні;
   - від'ємне значення — потовщення всередину.
6. За потреби увімкніть параметр Рівномірна товщина.

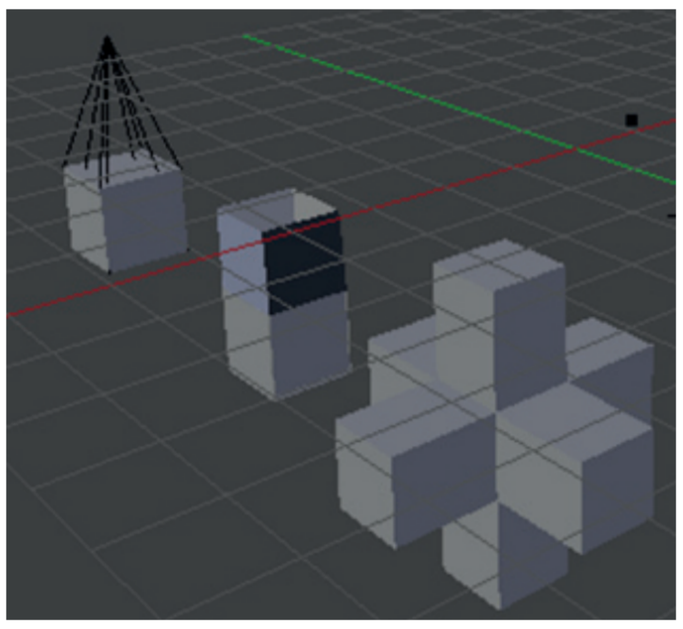
Для реалістичних моделей зазвичай спочатку застосовують **Потовщення**, а потім — **Поділ поверхні**, щоб отримати і об'єм, і плавність.

Детальніше про модифікатори у е-додатку (22.1).

ПЕРЕВІР СЕБЕ

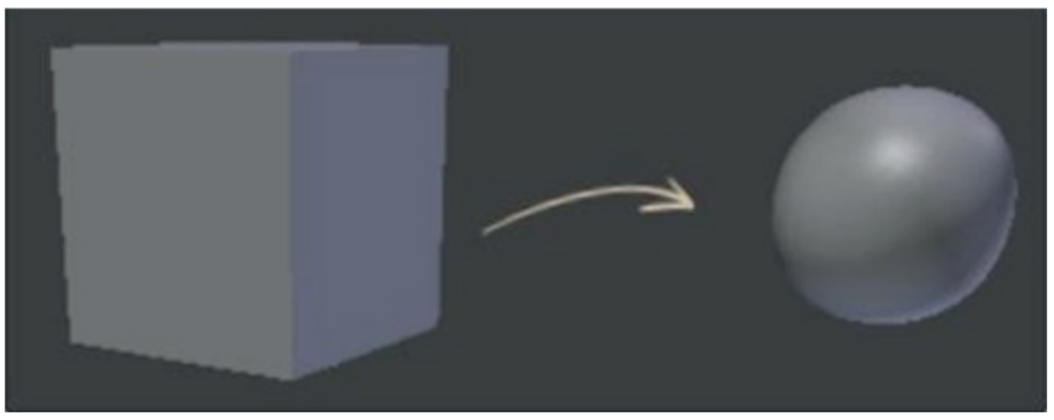
РОЗУМІННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Що трапиться, якщо екструдувати множину вершин, але не переміщувати їх?



2. Проаналізуй, які частини куба потрібно екструдувати для отримання з нього показаного зображення. Спробуй відтворити цей процес.

3. Використай правильно модифікатор та отримай із зображення куба сферу. →

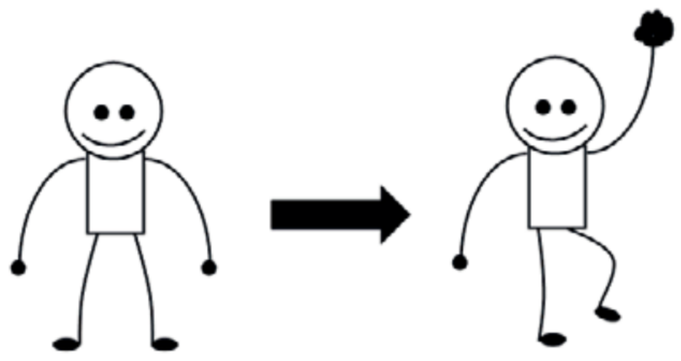


4. Відкрий е-додаток (22.3) та створи об'єкт за відео інструкцією.

5. Порівняй можливості екструдування вершин та граней. Чи можна замінити одну складову на іншу? У яких випадках це більш зручно, а в яких неможливо?

§ 23. Створення анімації в середовищі Blender

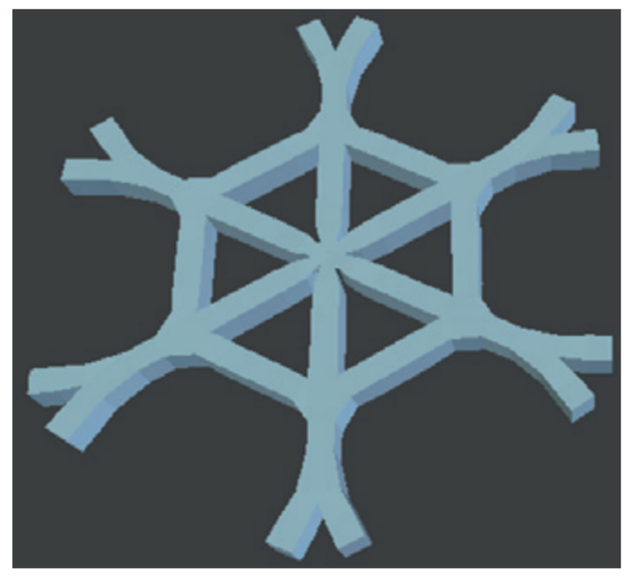
Як «оживити» картинку?



Як прикрасити зображення сніговика, створене на попередніх уроках? Можна створити полотно білого кольору, наче снігова поверхня, на якій стоїть сніговик. Або розмістити сніговика у вирі сніжинок. Оберемо останнє.

Завдання № 1

Щоб створити враження снігопаду, потрібно змоделювати сніжинку, «розмножити» її та додати анімаційні ефекти. Стандартних інструментів для створення сніжинки такої форми немає. Спробуй описати алгоритм її створення за допомогою стандартних примітивів і їх трансформацій, використовуючи відомі тобі інструменти.



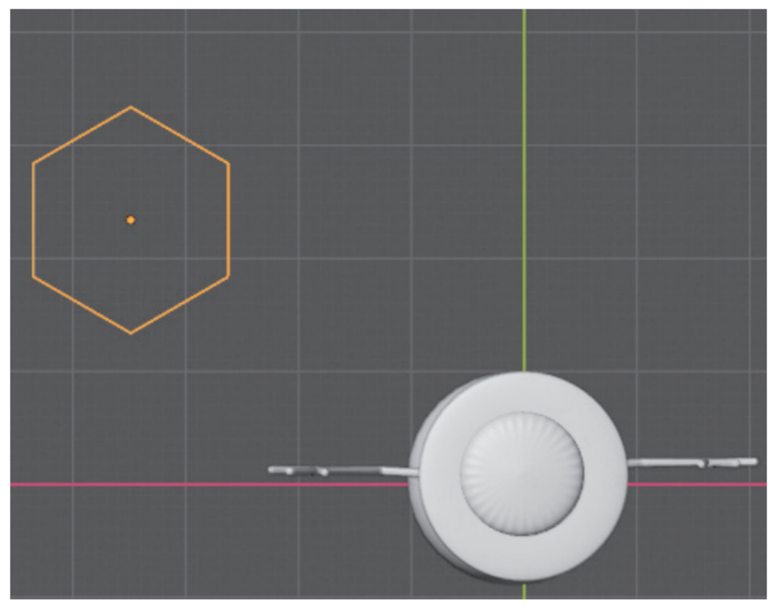
Завдання № 2

Об’єднайтесь у пари та побудуйте зображення сніжинки.

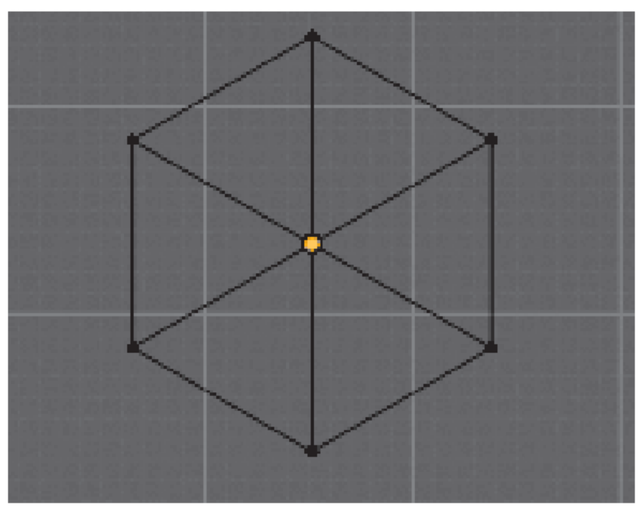
Інструкція:

Відкрийте файл із зображенням сніговика.

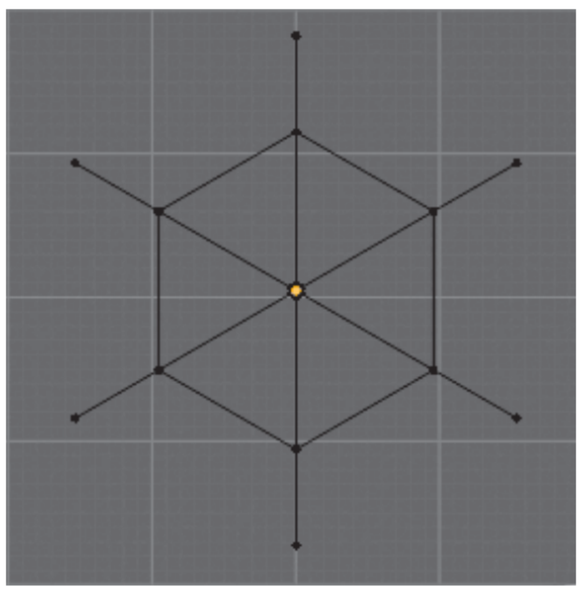
1. Створіть нову колекцію «Сніг».
2. Перейдіть у ортогональний режим перегляду зверху (**NumPad 7**, за потреби **NumPad 5**).
3. У **Режимі об’єкта** додайте примітив **Коло** (**Add** → **Mesh** → **Circle**).



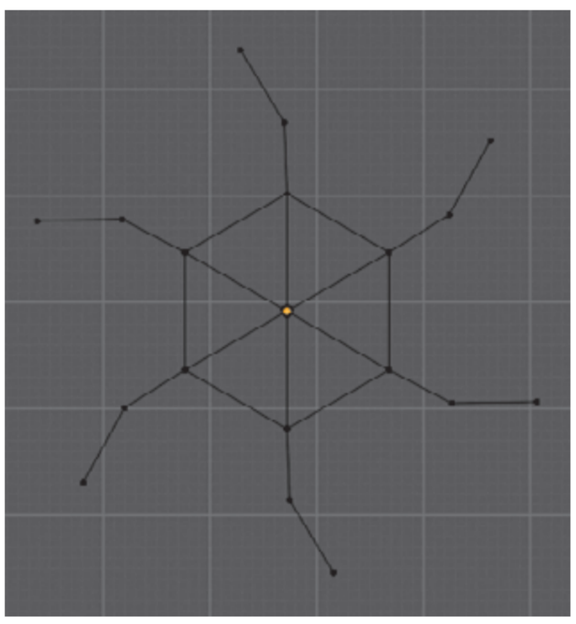
4. У панелі **Налаштувати останню операцію** (**F9**) встановіть кількість вершин — **6**.
5. Перейдіть у **Режим редагування**.
6. Виділіть усі вершини **A**.
7. Виконайте команду **Alt + E** → **Тільки вершини**.
8. Не переміщуючи вершини, натисніть **S** і масштабуйте їх до центра, щоб утворилися ребра від вершин до середини шестикутника. Підтвердьте дію лівою кнопкою миші.



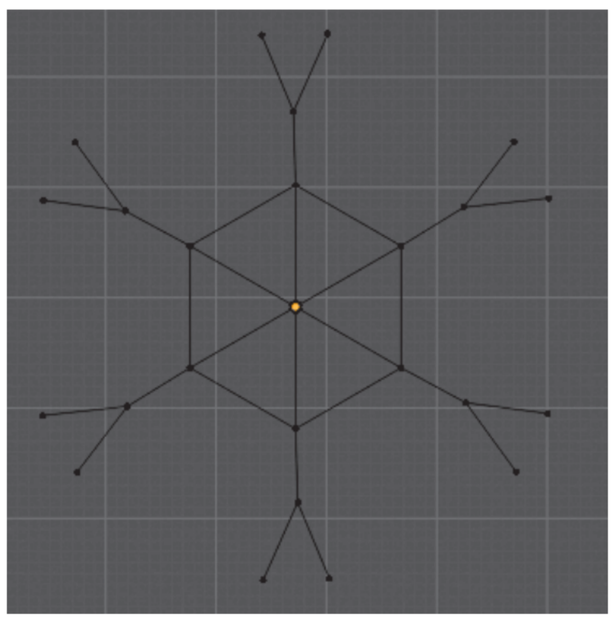
9. Утримуючи **Shift** виділіть зовнішні вершини шестикутника, виконайте команду **Alt + E** → **Тільки вершини**, не переміщуючи їх, натисніть **S** і масштабуйте вершини назовні, щоб утворилися промені сніжинки. Підтвердьте дію лівою кнопкою миші.



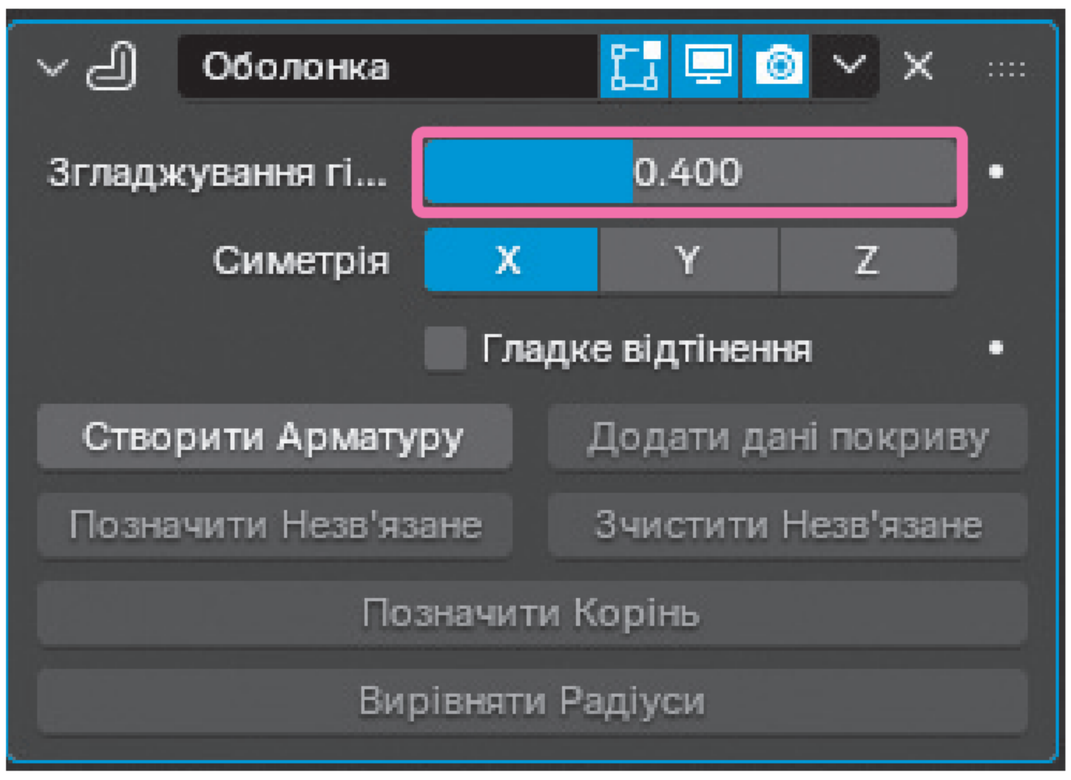
10. Натисніть **R** та змістіть вершини.

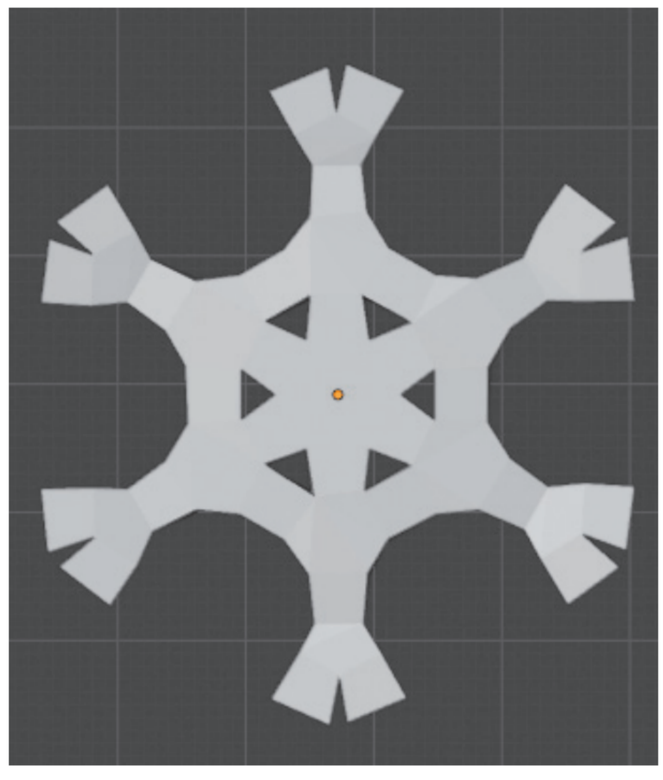


11. Повторіть пункт 11, щоб отримати симетричну гілку.

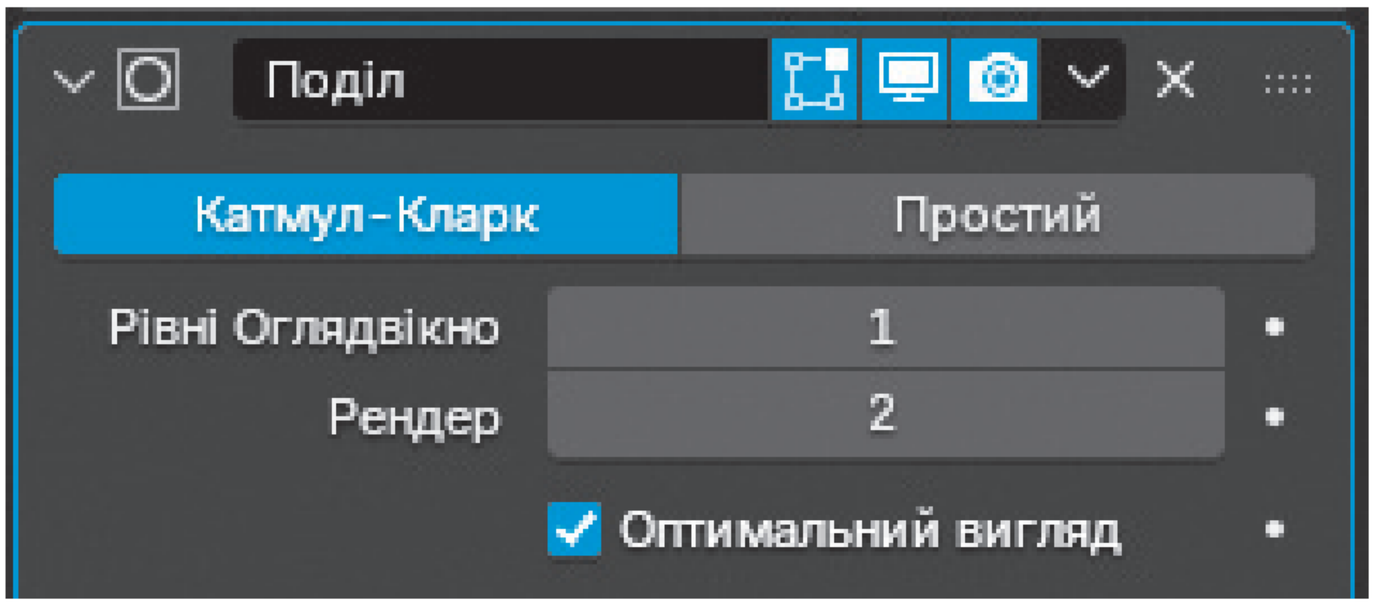


12. Перейдіть у **Режим об'єкта** та і застосуйте модифікатор **Skin (Генерування — Оболонка)**.

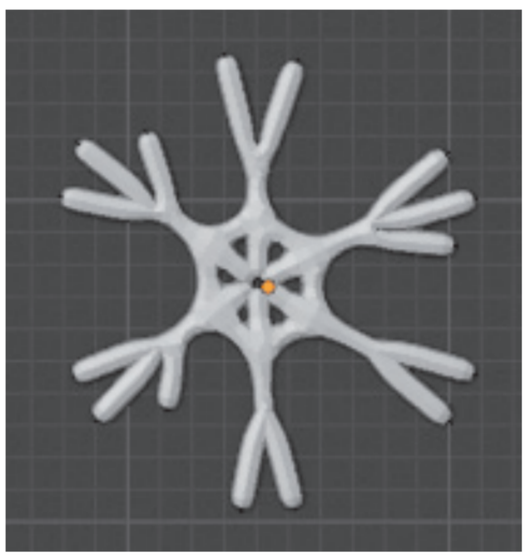




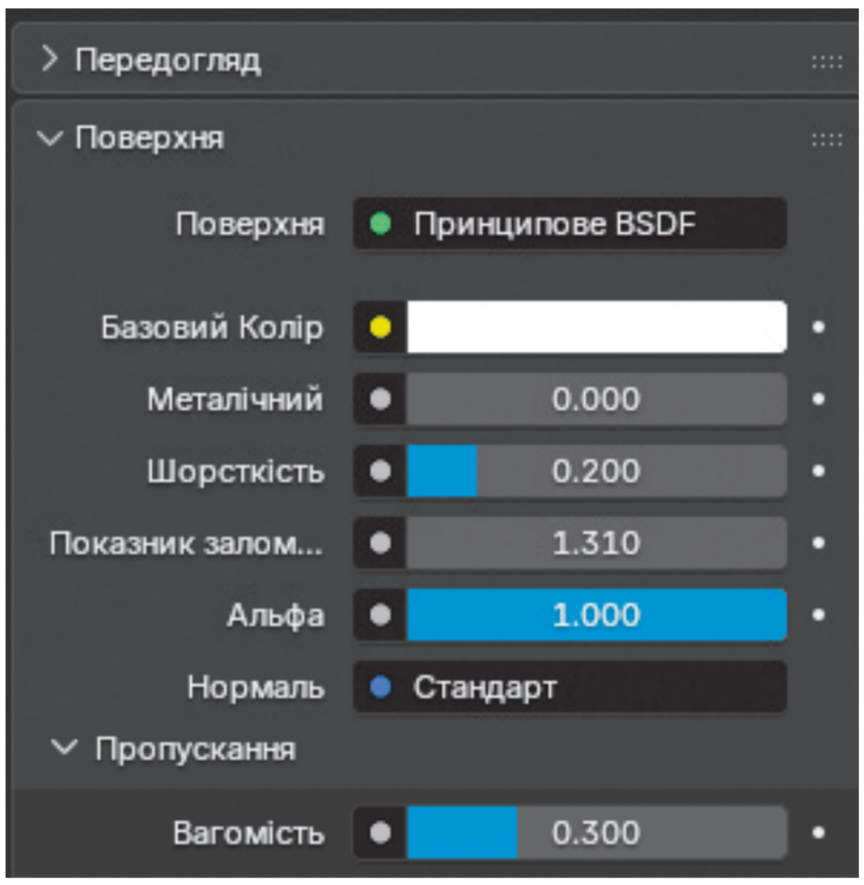
13. Виділіть об'єкт, застосуйте модифікатор **Subdivision Surface (Генерування — Поділ поверхні)**.



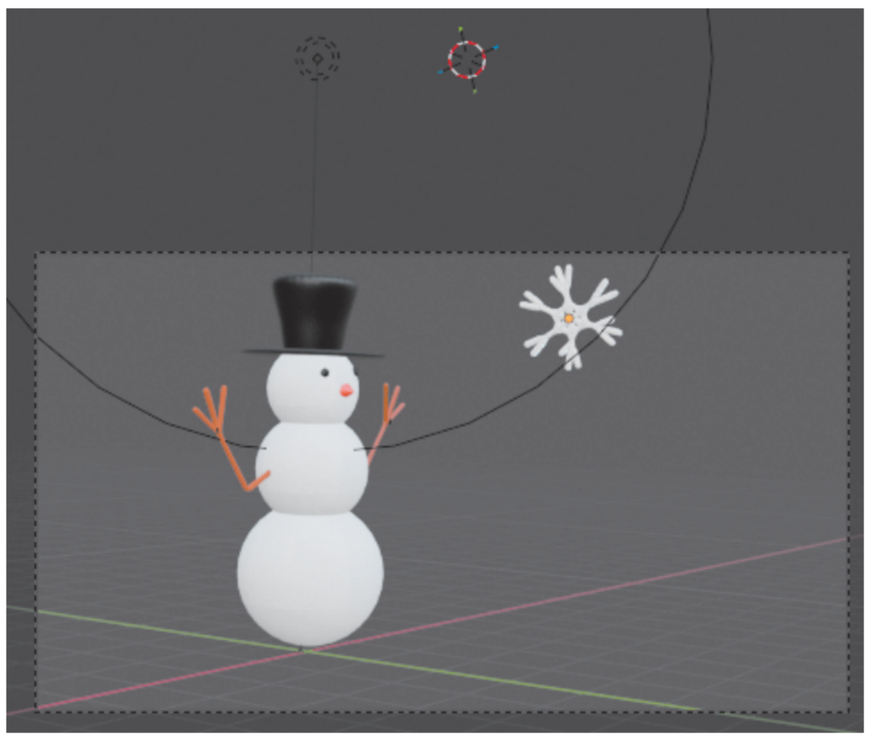
14. Відрегулюйте товщину гілочок. Виділіть об’єкт, у контекстному меню виберіть команду **Shade Smooth (Згладжене тонування)**.
15. У режимі редагування:
    - трохи варіюйте довжину гілочок (не всі мають бути однаковими);
    - зробіть кінчики трохи вужчими (масштабуванням **S**);
    - додайте 1–2 додаткові екструдування для кристалічної структури.



16. Відкрийте вкладку **Матеріал (Material Properties)** та установіть для параметра **Базовий колір (Base Color)** майже білий із легким блакитним відтінком. Виконайте налаштування:



17. Налаштуйте положення всіх об’єктів так, щоб у виді через камеру сніговик і сніжинка були розташовані в потрібному для рендерингу положенні, а світло правильно освітлювало всі об’єкти сцени.



Для цього:

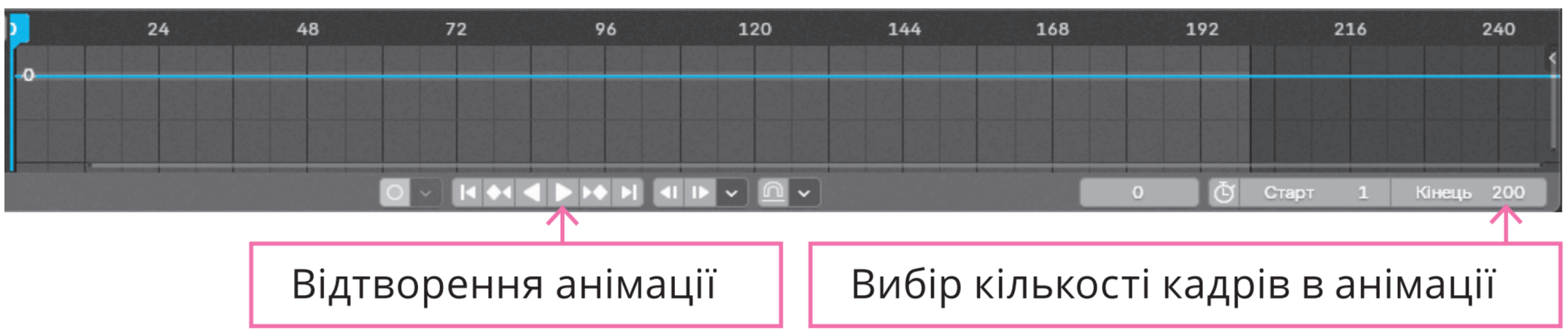
Перейдіть у вид через камеру (**NumPad 0**) та виділяйте й переміщуйте об’єкти клавішею **G**:
G + X — переміщення вздовж осі X;
G + Y — переміщення вздовж осі Y;
G + Z — переміщення вздовж осі Z.

За потреби використайте команду **Ctrl + Alt + NumPad 0**, щоб вирівняти камеру за поточним виглядом.

Після налаштування перевірте результат у режимі **Rendered (Рендеринг)** та за потреби скоригуйте положення об’єктів або освітлення.

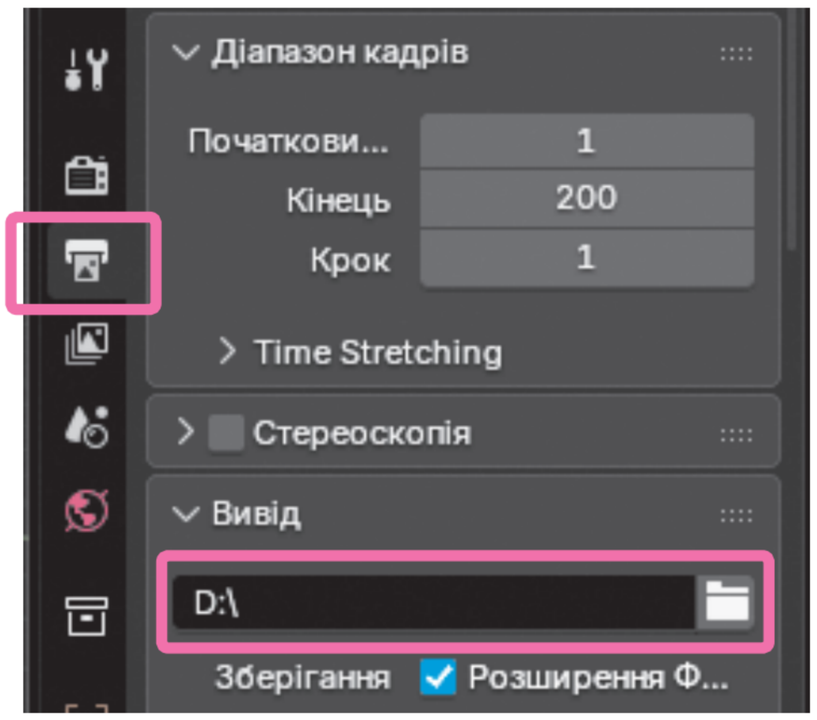
Завдання № 3

1. Виділіть сніжинку та натисніть **Shift + D** — створіть копію. Перемістіть її в інше місце (**G**).
2. Повторіть цю операцію кілька разів. Щоб падіння виглядало природно, для кожної копії змініть початкову висоту, розмір і виконайте невелике обертання.
3. На **шкалі часу (Timeline)** встановіть перший кадр (наприклад, 1). Виділіть сніжинку та натисніть **I** (ай) — буде створено початковий ключовий кадр.



4. Перейдіть на кінцевий кадр (наприклад, 100).
   Перемістіть сніжинку вниз уздовж осі Z (**G → Z**).
   Знову натисніть **I**, щоб створити кінцевий ключовий кадр.
5. Перегляньте анімацію.
6. Збережіть анімацію попередньо налаштувавши шлях для збереження:
   Виконайте команду **Render → Render Animation** (**Ctrl + F12**).

   Е-додаток (23.2).



ЗАПИТАННЯ-ВІДПОВІДІ

Як у Blender побудувати багатокутник?

Для побудови правильного багатокутника використовують примітив **Коло** у якому змінюють кількість вершин. Наприклад, у разі вибору 4 вершин буде зображено квадрат. Цю кількість вибирають на лівій панелі в області **Додати коло**.

Увага!

Кількість вершин можна вибрати лише під час побудови кола (F9). Після виконання певної дії з об'єктом, наприклад переміщення або переходу в режим редагування, змінити кількість вершин буде вже неможливо.

Як діє анімація у Blender?

Анімація у Blender — це зміна властивостей об’єктів у часі. Програма працює за принципом послідовності кадрів: кожен кадр — це окремий стан сцени. Коли кадри швидко змінюються один за одним, ми бачимо рух. Щоб створити цей рух, у Blender використовують ключові кадри. Ключовий кадр фіксує певний стан об’єкта в конкретний момент часу: його положення, обертання, масштаб або інші параметри. Між ключовими кадрами Blender автоматично обчислює проміжні зміни — цей процес називається інтерполяцією.

Анімувати можна різні властивості: переміщення (Location), обертання (Rotation), масштаб (Scale), зміну матеріалу, світла або інших параметрів. Якщо на початку об’єкт знаходиться в одному місці, а в кінці — в іншому, Blender створить плавний перехід між цими положеннями.

У Blender існує кілька способів створення анімації. Найпоширеніший — покадрова анімація з використанням ключових кадрів. Також можна створювати анімацію за допомогою модифікаторів і обмежень (Constraints), які автоматично керують рухом об’єктів. Для складніших ефектів використовують системи частинок або Geometry Nodes, де рух генерується процедурно. Окремим напрямом є скелетна анімація (Armature), що застосовується для персонажів.

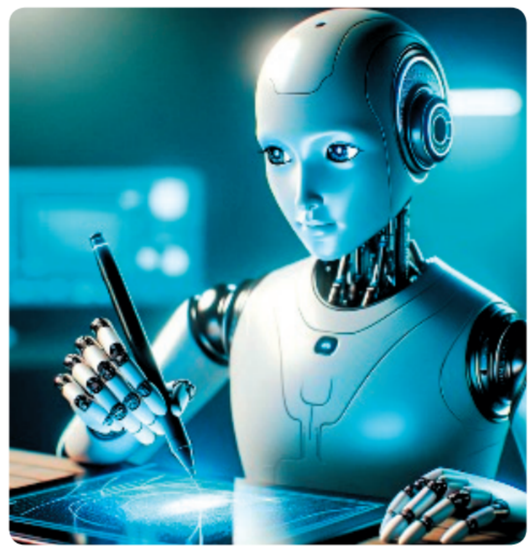
ПЕРЕВІР СЕБЕ

ЗНАННЯ — АНАЛІЗ — СТВОРЕННЯ — ВЛАСНІ ДУМКИ

1. Якою клавішею вставляють ключовий кадр анімації?
2. Якою командою виконують рендер всієї анімації?
3. Чому під час створення сніжинки зручно працювати у виді зверху?
4. Чому для природного ефекту падіння сніжинок потрібно зміщувати їхні ключові кадри?
5. Опишіть власний алгоритм створення іншої фігури (наприклад, зірки або квітки) на основі примітива Коло.
6. Запропонуйте спосіб удосконалення анімації снігопаду.
7. Складіть план створення нової сцени (наприклад, падаюче листя або дощ).
8. Як ти вважаєш, наскільки зручно використовувати анімаційні ефекти редактора Blender для створення мультфільму?

Додаток. Створення 3D-моделей за допомогою генеративного ШІ

Як штучний інтелект може допомогти створити 3D-модель?



Ти вже знаєш, як створювати тривимірні об’єкти у Blender і вручну керувати їхньою формою, розмірами та матеріалами. Але сьогодні з’явилися інструменти, які здатні «намалювати» 3D-модель за лічені хвилини — достатньо описати її словами або завантажити зображення. Так працюють сервіси генеративного штучного інтелекту.

Уяви, що тобі потрібно швидко створити модель робота для навчального проєкту. Раніше ти би витратив чимало часу на моделювання кожної деталі. Тепер достатньо ввести опис: *«маленький робот-помічник із круглою головою та антеною»,* — і через хвилину AI запропонує тобі готовий варіант. Його можна завантажити у форматі OBJ, FBX чи GLB і продовжити редагувати у Blender.

Але, як і будь-яка технологія, він має не тільки переваги, а й певні обмеження та етичні виклики.

Переваги генерації 3D-моделей за допомогою ШІ

- Модель створюється за хвилини, а не години.
- Деякі сервіси мають безкоштовні можливості, тому їх легко використовувати в навчальних проєктах.
- Згенеровані моделі можна одразу імпортувати у Blender чи ігрові рушії.
- Навіть новачок може зробити складну модель без спеціальних навичок.

Недоліки та обмеження

- Часто моделі потребують ручного доопрацювання: виправлення сітки, корекції текстур.
- Деякі сервіси обмежують кількість запитів або якість експорту.
- AI іноді створює надто складні моделі, які доводиться спрощувати для анімації чи ігор.
- Для комерційних проєктів важливо уважно читати ліцензію.

Розвиток генеративного інтелекту не лише відкриває нові можливості для створення 3D-моделей, а й порушує важливі етичні питання.

Одне з них стосується авторського права: якщо система навчається на великих масивах даних з інтернету, виникає сумнів, кому належить створена нею модель — користувачу чи авторам вихідних зображень. Не менш важливою є проблема справедливості та безпеки. Потрібно розуміти, які саме дані використовувалися під час навчання моделі та чи не порушують вони права людей, чиї роботи або персональна інформація могли бути включені до навчального набору.

Ще одне суттєве питання — вплив на професії. Дехто побоюється, що генеративні інструменти можуть замінити 3D-художників або дизайнерів. Проте більшість фахівців вважає, що штучний інтелект радше стане корисним помічником, який бере на себе рутинні операції, тоді як творчі рішення та контроль якості залишатимуться за людиною.

Завдання

Відкрий е-додаток (24.1–24.3) та послідовно ознайомся із інформацією поданою у розділах 24.1–24.3. Порівняйте можливості цих трьох сервісів виконавши дії:

1. Оберіть один промт (текстовий опис), наприклад: «Small friendly robot assistant with round head and antenna» або створіть власний опис.
2. Створіть 3D-модель за цим самим промтом у трьох описаних у е-додатку сервісах.
3. Завантажте отримані моделі та відкрийте їх у Blender.
4. Оцініть результати за такими критеріями:

   Якість моделі. Чіткість форми, реалістичність або відповідність стилю, коректність пропорцій, наявність артефактів чи помилок.

   Якість текстур. Наскільки акуратно накладені, чи відповідають опису, чи не «з'їжджають» на окремих частинах моделі.

   Структура та топологія. Чи акуратно побудована 3D-сітка, чи легко модель адаптувати під анімацію, наявність зайвих або проблемних елементів.

   Зручність використання сервісу. Наскільки інтуїтивний інтерфейс, чи легко вводити промти, швидкість генерації, чи потрібна реєстрація, чи є обмеження безкоштовної версії

   Експорт і сумісність. Які формати доступні, чи без проблем відкривається модель у Blender, чи потребує додаткової конвертації

   Відповідність очікуваному результату.

5. Створіть порівняльну таблицю або короткий звіт.

Зміст

РОЗДІЛ I
ОПРАЦЮВАННЯ ТАБЛИЧНИХ ДАНИХ

§ 1. Табличні дані в сучасному світі 5
§ 2. Сортування. Прості та розширені фільтри 17
§ 3. Візуальний аналіз даних 32
§ 4. Дані та аналітичні висновки 40
§ 5. Основи машинного навчання в електронних таблицях 52

РОЗДІЛ II
БАЗИ ДАНИХ. СИСТЕМИ КЕРУВАННЯ БАЗАМИ ДАНИХ

§ 6. Основні поняття баз даних 67
§ 7. Ознайомлення із системою керування базами даних 75
§ 8. Обмеження цілісності 84
§ 9. Вибір даних 92

РОЗДІЛ III
АЛГОРИТМИ ТА ПРОГРАМИ

§ 10. Поняття списку 106
§ 11. Операції з елементами списку 113
§ 12. Введення елементів списку 120
§ 13. Елемент керування ListBox 127
§ 14. Операції з рядками. Шифр Цезаря 135
§ 15. Упорядкування елементів списку 144

РОЗДІЛ IV
СТВОРЕННЯ ТА ПУБЛІКАЦІЯ ВЕБРЕСУРСІВ

§ 16. Структура вебсторінки. Мова HTML 154
§ 17. Як змінити вигляд вебсторінки? Основи CSS 165
§ 18. Вебсторінка як єдина система 173
§ 19. Автоматизовані засоби для створення та публікації вебресурсів 185

РОЗДІЛ V
3D-ГРАФІКА

§ 20. Тривимірна графіка. Програми для роботи з тривимірною графікою. Основи створення тривимірної моделі 198
§ 21. Екструдування об’єктів 219
§ 22. Змінення форми об’єкта 228
§ 23. Створення анімації в середовищі Blender 233
Додаток. Створення 3D-моделей за допомогою генеративного ШІ 238

ПОСІБНИК ДЛЯ ПОЧАТКІВЦІВ

vse.ee/cyot

ГАРЯЧІ КЛАВІШІ BLENDER

Операції, які є основою для моделювання в 3D-редакторі Blender.

Клавіша	Дія	Пояснення
G	**Переміщення** (Grab/Грабінг)	Рух об’єкта по сцені. Після натискання **G** можна додатково натиснути **X**, **Y**, або **Z**, щоб обмежити рух однією віссю.
R	**Обертання** (Rotate/Ротація)	Поворот об’єкта навколо його центру. Можна обмежити віссю (**R** + **X**, **Y**, або **Z**).
S	**Масштабування** (Scale/Скейлінг)	Зміна розміру об’єкта. Можна змінити розмір вздовж однієї осі (**S** + **X**, **Y**, або **Z**).
E	**Екструдування** (Extrude)	Витиснення нової геометрії з виділених вершин, ребер або граней (використовується в Режимі редагування).
Shift + D	**Дублювати**	Створення копії об’єкта.

Глобальні гарячі клавіші

Клавіша	Дія
Ctrl + O	Відкрити файл
Ctrl + S	Зберегти файл
Ctrl + N	Новий файл
Ctrl + Z	Скасувати

Клавіша	Дія
Shift + Ctrl + Z	Повторити
F12	Вималювати зображення
F11	Показати/Сховати вимальовуване
Q	Швидкий доступ

Загальні клавіші редактора та об’єктний режим

Клавіша	Дія
ПКМ (Права Кнопка Миші)	Виділити об’єкт
Shift + S	Меню розміщення
A	Виділити/Зняти виділення все
Ctrl + M + Y/X/Z	Дзеркальне відображення
Ctrl + G	Групувати
Delete	Видалити

Клавіші режиму редагування

Клавіша	Дія
Tab	Режим редагування
1 / 2 / 3	Вершини/Ребра/Грані
Ctrl + R	Розрізання (Петльове різання)
B	Прямокутна область
C	Кругла область
Alt + A	Запуск анімації

Клавіші перегляду сцени

Клавіша	Дія
Коліщатко миші (натиснути і рухати)	Обертати сцену
Ctrl + Коліщатко миші	Зміщення вліво-вправо
Shift + Коліщатко миші	Зміщення вгору-вниз
Коліщатко миші (прокручувати)	Масштабування сцени
7 (на цифровій клавіатурі)	Вигляд зверху
1 (на цифровій клавіатурі)	Вигляд збоку (по осі X)
3 (на цифровій клавіатурі)	Вигляд збоку (по осі Y)
5 (на цифровій клавіатурі)	Ортогональний вигляд