

Генеза

НОВА УКРАЇНЬСЬКА
ШКОЛА

ІНФОРМАТИКА

Частина 2

8
КЛАС



ІНФОРМАТИКА

Частина 2

Навчальний посібник для 8 класу
закладів загальної середньої освіти, які беруть участь
в інноваційному освітньому проєкті всеукраїнського
рівня за темою «Розроблення і впровадження
навчально-методичного забезпечення для закладів
загальної середньої освіти в умовах реалізації
Державного стандарту базової середньої
освіти» у 2024/2025 навчальному році

Схвалено для використання в освітньому процесі

Київ
«ГЕНЕЗА»
2024

УДК 004(075.3)

I-74

Авторський колектив:

Й. Я. Ривкінд, Т. І. Лисенко, Л. А. Чернікова, В. В. Шакоцько

*Схвалено для використання в освітньому процесі
(рішення експертної комісії з інформатики від 07.10.2024,
протокол № 11; № 3.0661-2024 у Каталогі надання грифів
навчальній літературі та навчальним програмам)*

Відповідає модельній навчальній програмі
«Інформатика. 7–9 класи» для закладів загальної середньої освіти
(автори: *Ривкінд Й. Я., Лисенко Т. І., Чернікова Л. А., Шакоцько В. В.*)

Переглянути електронний додаток до навчального посібника можна за посиланням <https://sites.google.com/view/informatika-8class> або QR-кодом



Додатки до посібника можна знайти за адресою <https://cutt.ly/benwo0jJ> або QR-кодом



I-74 Інформатика : навч. посіб. для 8-го кл. закл. заг. серед. освіти, які беруть участь в інновац. освіт. проєкті всеукр. рівня за темою «Розроблення і впровадження навчально-методичного забезпечення для закладів загальної середньої освіти в умовах реалізації Державного стандарту базової середньої освіти» у 2024/2025 навч. р. У 2 ч. Ч. 2 / Ривкінд Й. Я. [та ін.]. — Київ : Генеза, 2024. — 208 с. : іл.

ISBN 978-____-____-____-____.

Посібник підготовлено для забезпечення пілотних 8-х класів навчально-методичними матеріалами з інформатики для продовження реалізації реформи «Нова українська школа».

Частина 2 призначена для організації роботи з вивчення курсу «Інформатика» у другому семестрі.

УДК 004(075.3)

ISBN 978-____-____-____-____

© Ривкінд Й. Я., Лисенко Т. І.,
Чернікова Л. А., Шакоцько В. В.,
2024
© «Генеза», оригінал-макет, 2024

Навчальне видання

**РИВКІНД Йосиф Якович, ЛИСЕНКО Тетяна Іванівна,
ЧЕРНІКОВА Людмила Антонівна, ШАКОТЬКО Віктор Васильович**

ІНФОРМАТИКА

Частина 2

**Навчальний посібник для 8 класу
закладів загальної середньої освіти, які беруть участь
в інноваційному освітньому проєкті всеукраїнського рівня
за темою «Розроблення і впровадження навчально-методичного
забезпечення для закладів загальної середньої освіти в умовах
реалізації Державного стандарту базової середньої освіти»
у 2024/2025 навчальному році**

Схвалено для використання в освітньому процесі

У навчальному посібнику використано ілюстративний матеріал з відкритих джерел інтернету, зокрема сайтів *vecteezy.com*, *depositphotos.com*. Усі матеріали в посібнику використано з навчальною метою відповідно до законодавства України про авторське право і суміжні права.

**Редактор *Наталія Дашко*
Обкладинка *Олександра Павленка*
Макет, художнє оформлення *Василя Марущинця*
Комп'ютерна верстка *Юрія Лебедева*
Коректор *Інна Борік***

Формат 70×100/16.

Ум. друк. арк. __,__. Обл.-вид. арк. __,__.

Тираж _____ пр. Вид. № _____.

Зам. № _____

**ТОВ «Гене́за», 01133, Україна, місто Київ,
вул. Генерала Алмазова, 18/7 (літ. В), офіс 404.
Свідоцтво суб'єкта видавничої справи серія ДК
№ 7692 від 24.10.2022.**

Виготовлено

Дорогі восьмикласниці та восьмикласники! Шановні вчительки та вчителі!

Учні та учениці восьмого класу продовжуватимуть вивчення найсучаснішого шкільного предмета – **інформатики**.

Підручник, який підготував наш авторський колектив, сприятиме успішному навчанню інформатики. Він допоможе навчитися логічно мислити, обґрунтовувати власні дії та свій вибір цифрових засобів для розв'язування задач з різних галузей з використанням інформаційних технологій. А також бути креативними, розвивати навички розв'язування проблем, генерувати власні ідеї та добросовісно використовувати чужі для створення нових інформаційних об'єктів.

На уроках інформатики у 8 класі ви дізнаєтеся про особливості кодування даних, одиниці вимірювання довжини двійкового коду, апаратне забезпечення персонального комп'ютера. Ознайомитеся з поняттями «смарттехнології», «Інтернет речей», особливостями доцільного використання штучного інтелекту. Навчитесь виконувати операції з архівами даних, створювати та публікувати власні вебресурси. Продовжите створювати проекти однією з сучасних мов програмування – **Python** або **Object Pascal**, опрацьовувати дані, подані в електронних таблицях. Набуті компетентності ви застосовуватимете під час виконання навчальних проєктів на уроках інформатики та інших предметів.

Матеріал підручника поділено на вісім розділів. Розділи складаються з пунктів, а кожний пункт – з підпунктів. Корисними та цікавими для вас будуть такі рубрики:



Пригадайте



Працюємо з комп'ютером



Інтеграція (поєднання) наук



Для тих, хто працює з Android



Поміркуйте



Практична робота



Працюємо у групах/ в парах



Цікаві факти з історії



Обговоріть і зробіть висновки



Найважливіше в цьому пункті



Дослідіть



Виконайте завдання



Для тих, хто хоче знати більше



Дайте відповіді на запитання



Готуємось до вивчення нового матеріалу



Оцініть свої знання та вміння

Значна увага в підручнику приділяється практичній діяльності, у процесі якої й формуються відповідні інформатичні компетентності. Учні та учениці будуть запропоновано обдумати та обговорити в парах або в невеликих групах відповіді на деякі запитання, виконати завдання зі створення чи удосконалення певного інформаційного продукту.

Матеріали для виконання завдань, вправ, у тому числі файли-заготовки, розміщено на сайті **Інформатика** для всіх за адресою <http://sites.google.com/pu.org.ua/allinf> або за QR-кодом.

Зичимо плідного та захопливого навчання!

Авторський колектив



У цьому розділі ви отримаєте нові, а також поглибите та розширите свої знання й удосконалисте навички з таких тем:

- ▶ етапи розв'язування задач з використанням алгоритмів і проєктів, створених мовою програмування;
- ▶ елементи керування **кнопки, написи, поля, прапорці, перемикачі**, їх властивості, події, що можуть відбуватися з ними, їх використання у проєктах;
- ▶ встановлення та змінення значень властивостей елементів керування під час виконання проєкту;
- ▶ лінійні алгоритми та проєкти;
- ▶ логічні величини, логічні вирази, логічні операції заперечення, кон'юнкція, диз'юнкція;
- ▶ алгоритми та проєкти з розгалуженнями;
- ▶ алгоритми та проєкти з циклами;
- ▶ алгоритми та проєкти з розгалуженнями та циклами;
- ▶ використання проєктів для проведення комп'ютерних експериментів.

У підручнику цей розділ викладено з використанням мови програмування **Python** і середовища створення та редагування проєктів **IDLE**.

Текст цього розділу з використанням мови програмування **Object Pascal** і середовища створення та редагування проєктів **Lazarus** знаходиться за посиланням <https://cutt.ly/ceL9HV2G> або QR-кодом.



4.1. ЕТАПИ РОЗВ'ЯЗУВАННЯ ЗАДАЧ З ВИКОРИСТАННЯМ АЛГОРИТМІВ І ПРОЄКТІВ

У цьому пункті йтиметься про:

- ▶ етапи розв'язування задач з використанням алгоритмів і проєктів, створених мовою програмування на основі складених алгоритмів;
- ▶ тестові набори даних;
- ▶ види помилок під час виконання проєкту;
- ▶ тестування проєктів, налагодження проєктів, проведення комп'ютерних експериментів.

ЕТАПИ РОЗВ'ЯЗУВАННЯ ЗАДАЧ З ВИКОРИСТАННЯМ АЛГОРИТМІВ І ПРОЄКТІВ МОВОЮ ПРОГРАМУВАННЯ, СТВОРЕНИХ НА ОСНОВІ СКЛАДЕНИХ АЛГОРИТМІВ



Пригадайте

- Які ви знаєте етапи роботи над проєктом, що створюється мовою програмування?
- Що таке *модель об'єкта*? Для чого її створюють?
- Які види моделей ви знаєте? Чим характерна математична модель?

У попередніх класах ви вже розв'язували різноманітні задачі, створюючи алгоритми та проєкти мовою програмування на основі цих алгоритмів.

Розглянемо детальніше, з яких етапів має складатися розв'язування задач на обчисленнями з використанням алгоритмів і проєктів мовою програмування на основі цих алгоритмів.

Етап 1. Уважно прочитати умову задачі, проаналізувати її та створити математичну модель задачі. Тобто визначити:

1. Які дані є вхідними?
2. Які дані будуть кінцевими результатами?
3. Формули, використовуючи які можна отримати значення кінцевих результатів.

Етап 2. Створити алгоритм розв'язування задачі. Найчастіше алгоритм подають у вигляді блок-схеми. Але використовують й інші способи подання алгоритму, зокрема, словесний або комбінований.

Етап 3. Створити комп'ютерний проєкт мовою програмування, у якому використовується створений алгоритм.

Етап 4. Запустити проєкт на виконання для кількох довільних наборів значень вхідних даних, щоб перевірити, чи видає він значення кінцевих результатів. Якщо виконання проєкту переривається, то це означає, що проєкт містить **синтаксичні помилки**. У такому разі, потрібно їх виправити і знову запустити проєкт на виконання.

Етап 5. Запустити проєкт на виконання для таких наборів значень вхідних даних, щоб можна було б знайти значення кінцевих результатів без виконання проєкту і порівняти їх з тими значеннями кінцевих результатів, які видає комп'ютер у результаті виконання проєкту.

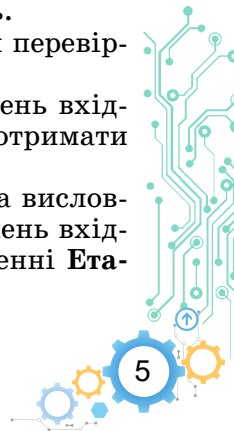
Якщо очікувані нами значення кінцевих результатів не збігаються з тими, які видає комп'ютер, то це означає, що проєкт містить **логічні помилки**. Потрібно їх визначити, виправити і знову запустити проєкт на виконання для тих самих наборів значень вхідних даних.

Такі набори значень вхідних даних мають бути дібрані так, щоб перевірити правильність виконання всіх складових частин проєкту. Зокрема, якщо проєкт містить розгалуження, то потрібно дібрати такі набори значень вхідних даних, щоб перевірити правильність роботи проєкту для всіх можливих випадків виконання розгалужень.

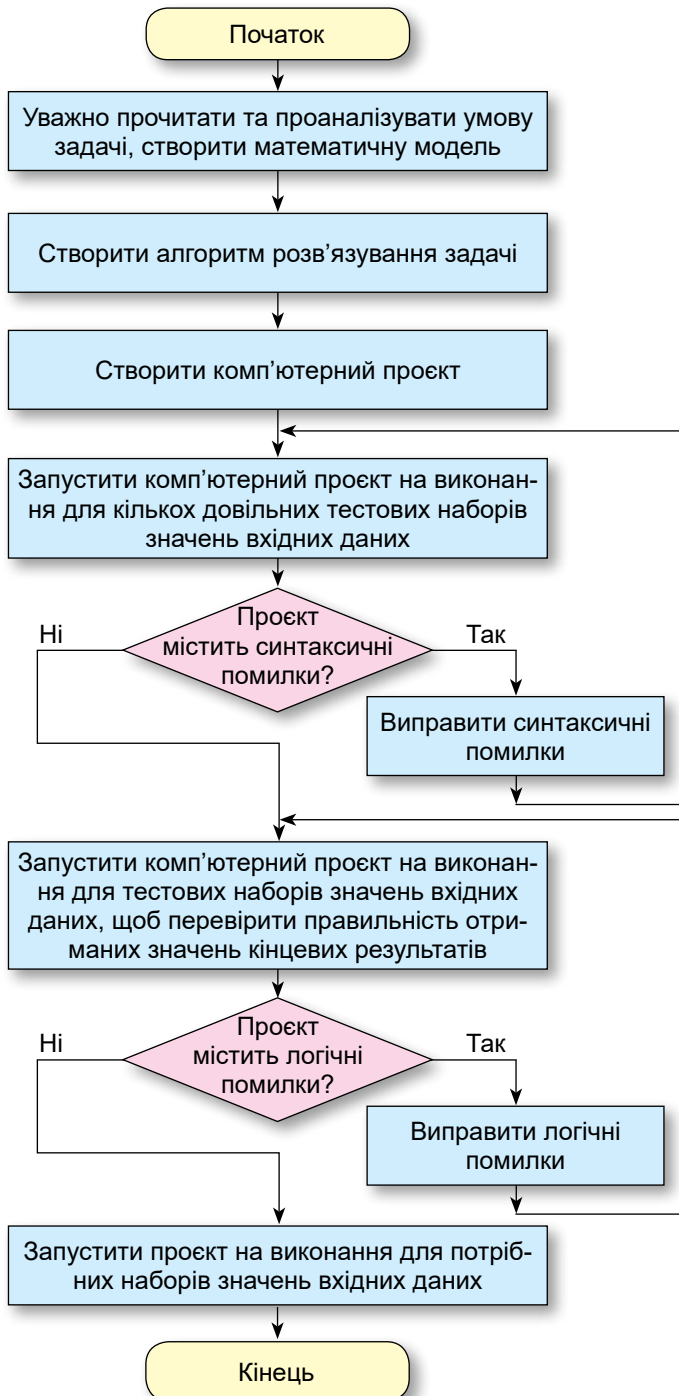
Набори значень вхідних даних, які використовуються для перевірки правильності проєкту, називаються **тестовими наборами**.

Етап 6. Запустити проєкт на виконання для наборів значень вхідних даних, які вказані в умові задачі, або інших значень і отримати значення кінцевих результатів.

Звертаємо вашу увагу: під час проведення **Етапу 6** можна висловлювати гіпотези і, виконуючи проєкт для різних наборів значень вхідних даних, підтвердити їх або заперечити. Тобто при проведенні **Етапу 6** можна здійснювати **комп'ютерний експеримент**.



Перераховані вище етапи та послідовність їх виконання можна подати у вигляді блок-схеми (мал. 4.1).



Мал. 4.1. Етапи розв'язування задач з використанням алгоритмів і проєктів



Працюємо з комп'ютером

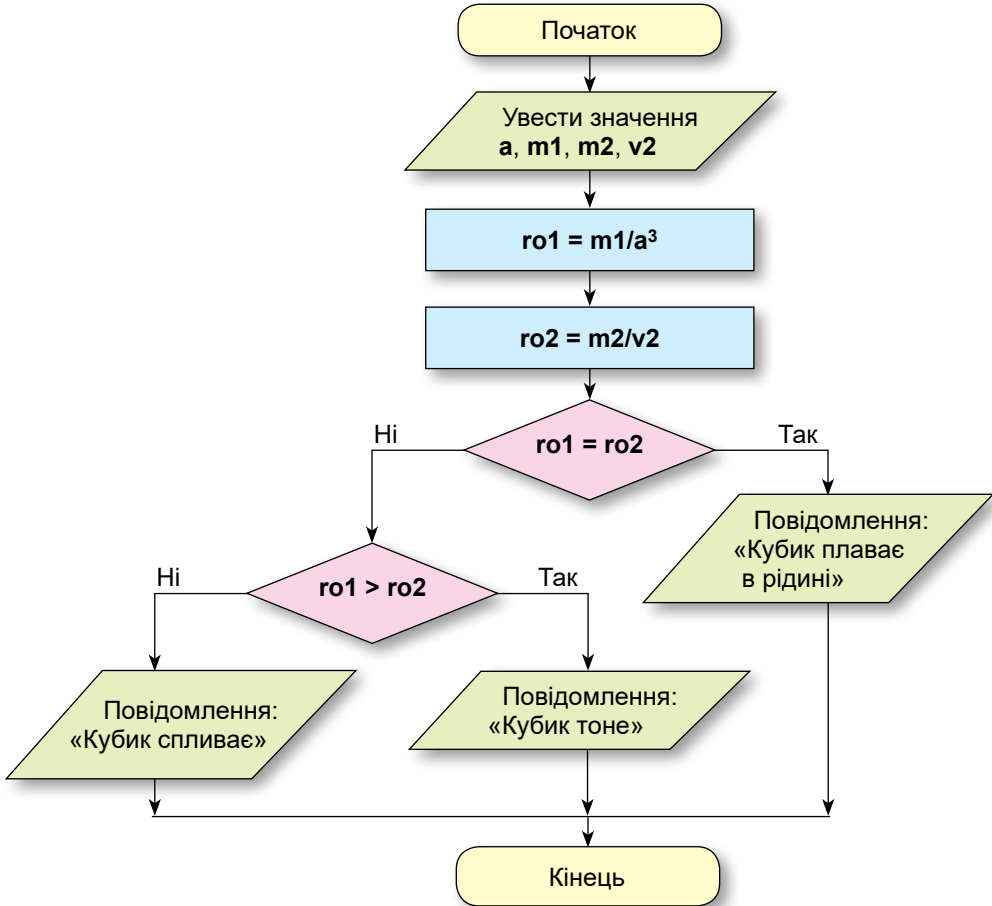
Увага! Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

Задача. У Марічки є кубик, для якого відомі довжина його ребра a і маса m_1 . Марічка занурює кубик у посуд з рідиною. Відомі маса витісненої рідини m_2 та її об'єм v_2 . Визначити, чи буде плавати кубик у рідині, чи потоне, чи буде спливати.

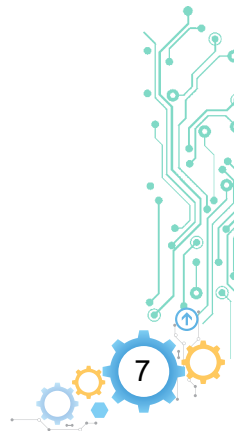
Етап 1. Уважно прочитайте і проаналізуйте умову задачі, створіть математичну модель задачі:

1. **Вхідні дані:** довжина ребра кубика a , маса кубика m_1 , маса витісненої рідини m_2 , об'єм витісненої рідини v_2 .
2. **Кінцеві результати:** Повідомлення: «Кубик плаває в рідині», або «Кубик тоне», або «Кубик спливає».
3. **Формули:** густина речовини, з якої виготовлено кубик, $ro1 = m_1/a^3$; густина рідини: $ro2 = m_2/v_2$. Відповідно до закону Архімеда, якщо $ro1 = ro2$, кубик плаває в рідині; якщо $ro1 > ro2$, кубик тоне; якщо $ro1 < ro2$, кубик спливає.

Етап 2. Створіть алгоритм розв'язування задачі (мал. 1).



Мал. 1. Алгоритм



Етап 3. Створіть комп'ютерний проєкт.

1. Відкрийте середовище розробки проєкту.

2. Уведіть у середовище розробки проєкту такі команди проєкту:

```
a = float(input("Уведіть довжину ребра кубика: "))
m1 = float(input("Уведіть масу кубика: "))
m2 = float(input("Уведіть масу витісненої рідини: "))
v2 = float(input("Уведіть об'єм витісненої рідини: "))
ro1 = m1/a**3
ro2 = m2/v2
if ro1 == ro2:
    print("Кубик плаває в рідині")
elif ro1 > ro2:
    print("Кубик тоне")
else:
    print("Кубик спливає")
```

3. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.1**.

Етап 4. Запустіть проєкт на виконання для довільних наборів значень вхідних даних, щоб перевірити, чи видає він значення кінцевих результатів.

Наприклад, уведіть значення **a = 1, m1 = 5, m2 = 3, v2 = 2**.

Якщо виконання проєкту переривається, то це означає, що проєкт містить **синтаксичні помилки**. У такому разі, знайдіть і виправіть їх та знову запустіть проєкт на виконання.

Звертаємо вашу увагу: на цьому етапі значення вхідних даних довільні, вони можуть не мати ніякого фізичного змісту, адже головне на цьому етапі – перевірити, що виконання проєкту не переривається.

Етап 5. Запустіть проєкт на виконання для таких тестових наборів значень вхідних даних, для яких ви заздалегідь визначили результат виконання проєкту. Оскільки проєкт містить розгалуження і можливі три різні повідомлення як результат виконання проєкту, то на цьому етапі потрібно виконати проєкт як мінімум для трьох тестових наборів значень вхідних даних. Причому вони мають бути підібрані так, щоб перевірити правильність виконання проєкту в кожному з трьох можливих випадків.

Наприклад, можна взяти такі три тестових набори вхідних даних:

1. **a = 1, m1 = 5, m2 = 10, v2 = 2**. Очікуваний результат: «Кубик плаває».
2. **a = 1, m1 = 5, m2 = 3, v2 = 2**. Очікуваний результат: «Кубик тоне».
3. **a = 1, m1 = 5, m2 = 20, v2 = 2**. Очікуваний результат: «Кубик спливає».

Виконайте проєкт з такими значеннями наборів вхідних даних і переконайтеся, що результати виконання проєкту збігаються з очікуваними. Якщо ні, знайдіть у проєкті **логічні помилки**, виправіть їх і знову виконайте проєкт з наведеними значеннями вхідних даних.

Етап 6. Запустіть проєкт на виконання для наборів значень вхідних даних, які ви підберете для конкретних матеріалів, рідин і розмірів кубиків. Значення вхідних даних і результати виконання проєкту запишіть у зошит.



Найважливіше в цьому пункті

Етапи розв'язування задач з використанням алгоритмів і проектів мовою програмування, створених на основі складаних алгоритмів:

Етап 1. Уважно прочитати умову задачі, проаналізувати її та створити математичну модель задачі. Тобто визначити:

1. Які дані є вхідними?
2. Які дані будуть кінцевими результатами?
3. Формули, використовуючи які можна отримати значення кінцевих результатів.

Етап 2. Створити алгоритм розв'язування задачі. Найчастіше алгоритм подають у вигляді блок-схеми.

Етап 3. Створити комп'ютерний проект мовою програмування, у якому використовується створений алгоритм.

Етап 4. Запустити проект на виконання для кількох довільних наборів значень вхідних даних, щоб перевірити, чи видає він значення кінцевих результатів.

Етап 5. Запустити проект на виконання для таких наборів значень вхідних даних, щоб можна було б знайти значення кінцевих результатів без виконання проекту і порівняти їх з тими значеннями кінцевих результатів, які видає комп'ютер у результаті виконання проекту.

Етап 6. Запустити проект на виконання для наборів значень вхідних даних, які вказані в умові задачі, або інших значень і отримати значення кінцевих результатів.

Якщо виконання проекту переривається, значить проект містить **синтаксичні помилки**.

Якщо значення результатів виконання проекту не збігаються з очікуваними, значить проект містить **логічні помилки**.

Набори значень вхідних даних, які використовуються для визначення правильності роботи проекту, називаються **тестовими наборами**.



Дайте відповіді на запитання

1. З яких етапів складається розв'язування задач на обчислення з використанням алгоритмів і проектів мовою програмування, створених на основі відповідних алгоритмів?

2. Що потрібно виконати на кожному з етапів розв'язування задач на обчислення з використанням алгоритмів і проектів мовою програмування, створених на основі відповідних алгоритмів? Коротко поясніть.

3. Які види помилок може містити проект? У чому полягає кожний з них?

4. Які набори значень вхідних даних називають тестовими?



Обговоріть і зробіть висновки

1. Як визначити мінімальну кількість тестових наборів значень вхідних даних, щоб якісно протестувати проект?

2. Чи можна дібрати набори значень вхідних даних і стільки їх потрібно, щоб на 100 % гарантувати, що проект завжди працюватиме правильно?

3. Що буде, якщо не виправити синтаксичну помилку? Логічну помилку?





Виконайте завдання

1. Щоб дістатися з дому на тренування, Василю потрібно дійти до автобуса і потім проїхати ним до стадіону. Який час він витратить на шлях від дому до стадіону, якщо відомо відстані, які проходить Василь і проїжджає автобус, і швидкості руху Василя й автобуса? Виконайте всі етапи розв'язування задачі. Математичну модель і алгоритм запишіть у зошит. Висловіть гіпотезу. Перевірте її, використовуючи створений проєкт. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.1.1.

2. Туристам потрібно за три дні пройти s км. У I день вони йшли t_1 год зі швидкістю v_1 км/год, у II день вони йшли t_2 год зі швидкістю v_2 км/год. З якою швидкістю їм потрібно рухатися в III день, щоб за t_3 год пройти потрібний шлях? Виконайте всі етапи розв'язування задачі. Математичну модель і алгоритм запишіть у зошит. Висловіть гіпотезу. Перевірте її, використовуючи створений проєкт. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.1.2.

3. В Улянки є кулька, для якої відомі її радіус R і маса m_1 . Улянка занурює кульку в посуд з рідиною. Відомі маса витісненої рідини m_2 та її об'єм v_1 . Визначити, чи буде плавати кулька в рідині, чи потоне, чи спливатиме. Об'єм кульки обчислюється за формулою $\frac{4}{3}\pi R^3$. Виконайте всі етапи розв'язування задачі. Математичну модель і алгоритм запишіть у зошит. Висловіть гіпотезу. Перевірте її, використовуючи створений проєкт. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.1.3.

4. Петрик хоче запросити свого друга в кіно і пригостити його морозивом. Мати дала йому для цього x грн. Чи вистачить грошей у Петрика на 2 квитки в кіно і на 2 порції морозива? Виконайте всі етапи розв'язування задачі. Математичну модель і алгоритм запишіть у зошит. Висловіть гіпотезу. Перевірте її, використовуючи створений проєкт. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.1.4.



Готуємось до вивчення нового матеріалу

1. Наведіть приклади властивостей вікна виконання проєкту та їх значень.
2. Які події можуть відбуватися з вікном виконання проєкту?
3. Що таке *обробник події*? Як він записується і виконується у проєкті?
4. Для чого у проєктах використовується елемент керування *кнопка*?

4.2. КНОПКА

У цьому пункті йтиметься про:

- ▶ кнопку як елемент керування, її властивості та їх значення;
- ▶ створення кнопки та способи її розміщення у вікні;
- ▶ події, які можуть відбуватися з кнопкою;
- ▶ обробники подій з кнопкою.

ВЛАСТИВОСТІ КНОПКИ



Пригадайте

● Які ви знаєте властивості вікна? Наведіть приклади їх значень. ● Якою командою створюється та відкривається вікно? ● Як змінити значення властивостей вікна? ● Для чого використовують *кнопку* в проєктах?

Одним з елементів керування, який можна використати у віконних проєктах, є **кнопка**.

Щоб розмістити кнопку в уже створеному та відкритому вікні, потрібно:

1. Створити новий об'єкт типу **Button** (англ. *button* – кнопка), пов'язати його зі змінною, яка визначатиме ім'я об'єкта.
2. Установити значення властивостей кнопки або залишити їх за замовчуванням.
3. Розмістити створений об'єкт у вікні.

Для створення кнопки та пов'язування її зі змінною використовують команду вигляду:

```
<ім'я_кнопки> = Button()
```

або

```
<ім'я_кнопки> = Button(<набір властивостей та їх значень>)
```

У першому випадку буде створено кнопку зі значеннями властивостей **висота**, **ширина** і **колір**, установленними за замовчуванням, без тексту на ній. У другому випадку буде створено кнопку, значення властивостей якої задано у цій команді. У дужках через кому вказують пари **<властивість> = <значення>**.

Для кнопки можуть бути встановлені значення таких властивостей:

- **text** (англ. *text* – текст) – текст на кнопці; значенням цієї властивості може бути довільний текст; цей текст потрібно брати в лапки; наприклад, **text = 'Обчислити'**;
- **width** (англ. *width* – ширина) – ширина кнопки; значенням цієї властивості може бути натуральне число – кількість символів, які можуть бути розміщені поруч по горизонталі; наприклад, **width = 20**;
- **height** (англ. *height* – висота) – висота кнопки; значенням цієї властивості може бути натуральне число – кількість символів, які можуть бути розміщені поруч по вертикалі (кількість рядків); наприклад, **height = 2**;
- **bg** або **background** (англ. *background* – задній план, фон) – колір фону; значенням цієї властивості може бути назва кольору англійською мовою; цю назву потрібно брати в лапки; наприклад, **bg = 'red'**;



- **fg** або **foreground** (англ. *foreground* – передній план) – колір символів тексту; значенням цієї властивості може бути назва кольору англійською мовою; цю назву потрібно брати в лапки; наприклад, **fg = ‘green’**;
- **font** (англ. *font* – шрифт) – шрифт символів тексту на кнопці; значення цієї властивості може містити назву шрифту, розмір символів і накреслення: **bold** (англ. *bold* – жирний) – напівжирний, **italic** (англ. *italic* – курсив) – курсивний, **underline** (англ. *underline* – підкреслений) – підкреслений; наприклад, **font = ‘Arial 14’** або **font = (‘Verdana’, 13, ‘bold’)**;
- **state** (англ. *state* – стан) – стан кнопки; значеннями цієї властивості можуть бути: **normal** (англ. *normal* – нормальний) – звичайний, **disable** (англ. *disable* – відключити, зробити неспроможним) – недоступний; ці значення потрібно брати в лапки, наприклад, **state = ‘normal’**

та інших.

Наприклад, кнопка з іменем **bt** з текстом *Змінити* завширшки 15 символів може бути створена командою:

```
bt = Button(text = ‘Змінити’, width = 15).
```

Звертаємо вашу увагу: якщо під час створення кнопки значення властивості **width** не задається, то ширина кнопки автоматично визначається довжиною тексту, який є значенням властивості **text**.

Значення властивостей кнопки також можна задавати окремими командами виду: **<ім’я кнопки>[‘<назва властивості>’] = <значення властивості>**. Наприклад, установити для кнопки з іменем **btn** ширину 25 символів можна командою **btn[‘width’] = 25**, установити шрифт *Times New Roman* з висотою символів 28 можна командою **btn[‘font’] = ‘Times New Roman, 28’**, задати синій колір символів – командою **btn[‘fg’] = ‘blue’**. Такого виду команди можна використовувати і для змінення значень властивостей кнопки під час виконання проекту.

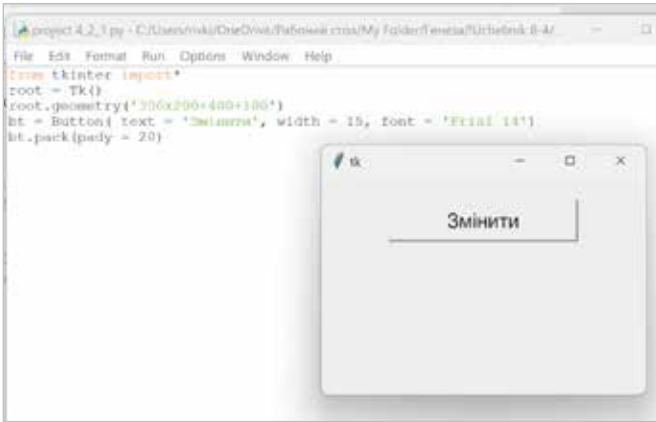
Для розміщення кнопки у вікні можна використати один з двох методів:

- **pack()** (англ. *pack* – упаковка) – кнопка автоматично розміщується під рядком заголовка вікна з однаковими відступами від лівої та правої межі вікна. Якщо у вікні розміщується кілька кнопок, то вони будуть розташовані одна під одною без відступів між собою. Якщо потрібно розмістити кнопку з деяким відступом від рядка заголовка або від попередньої кнопки, то в дужках можна вказати цей відступ у пікселях як значення властивості **pady** (англ. *pady* – відступ). Наприклад, після виконання команди **button.pack(pady = 20)** кнопка буде розміщена з відступом 20 пікселів від верхньої межі вікна або від попередньої кнопки;
- **place(x = <відступ від лівої межі вікна>, y = <відступ від верхньої межі вікна>)** (англ. *place* – місце) – верхня ліва вершина кнопки розміщується в точці вікна із заданими координатами **x** та **y**.

Нагадуємо: початок системи координат у вікні розміщується в його лівій верхній вершині, вісь **x** розташована горизонтально вправо, вісь **y** – вертикально вниз.

Наприклад, після виконання команди `button.place(x = 200, y = 100)` верхня ліва вершина кнопки з іменем `button` буде розміщена в точці вікна з координатами (200, 100).

На малюнку 4.2 наведено текст проєкту зі створення та розміщення кнопки з іменем `bt` у вікні та результат його виконання.



Мал. 4.2. Кнопка у вікні

Для тих, хто хоче знати більше

Під час виконання проєкту може виникнути потреба, наприклад у іграх, зробити кнопку невидимою. Для цього, залежно від методу розміщення кнопки, можна використати методи `pack_forget()` або `place_forget()`. Щоб зробити кнопку знову видимою, слід повторно застосувати метод `pack()` або `place()`.

ПОДІЇ З КНОПКОЮ. ОБРОБНИКИ ПОДІЙ ДЛЯ КНОПКИ

Пригадайте

- Які події можуть відбуватися з вікном проєкту?
- Що таке *обробник події*?
- Як записується обробник події у проєкті та як він виконується?

Як і з вікном, з кнопкою можуть відбуватися різні події. І з кожною такою подією можна пов'язати її обробник, який буде виконуватися одразу після настання відповідної події.

Пов'язати з кнопкою обробник будь-якої події можна так само, як з вікном, використовуючи метод `bind()`. Наприклад, на малюнку 4.3 наведено приклад обробника події `Click` для кнопки з іменем `button`, який містить команду встановлення жовтого кольору вікна.

```

from tkinter import *
def click(event):
    root['bg'] = 'yellow'
root = Tk()
root.geometry('600x100+350+200')
button = Button(text = 'Змінити', width = 15)
button.pack(pady = 30)
button.bind('<1>', click)
  
```

Мал. 4.3. Обробник події `Click` для кнопки

Для кнопки подією за замовчуванням є *клацання лівої кнопки миші*. Тому пов'язати з кнопкою обробник саме цієї події можна простіше:

- під час створення кнопки вказати ім'я відповідного обробника події як значення властивості **command**;
- обробник події з указаним іменем описати перед створенням кнопки; у заголовку обробника події не потрібно вказувати посилання на подію, що відбулася – **event**.

На малюнку 4.4 наведено приклад обробника події **Click** для кнопки – процедури з іменем *click()*, яка містить команди:

- установлення для кнопки *червоного* кольору фону;
- установлення шрифту *Arial* розміром *14* пікселів;
- виведення на кнопці вказаного тексту;
- установлення для кнопки стану *недоступна*.

```
from tkinter import *
def click():
    button['bg'] = 'red'
    button['font'] = 'Arial 14'
    button['text'] = 'Змінюється значення властивостей кнопки'
    button['state'] = 'disable'
root = Tk()
root.geometry('600x400+350+200')
button = Button(text = 'Вибери мене', command = click)
button.pack(pady = 30)
```

Мал. 4.4. Обробник події **Click** для кнопки

Звертаємо вашу увагу:

1. Ім'я обробника події – значення властивості **command** – записувати потрібно обов'язково без лапок.
2. Значення властивостей кнопки в обробнику подій можна змінювати так само, як значення властивостей вікна (вказати назву властивості у квадратних дужках у лапках, а нове значення властивості – після знаку =).
3. В обробнику події можна змінювати значення властивостей як самої кнопки, так й інших об'єктів (вікна, інших кнопок).
4. Розташування кнопки у вікні можна змінювати в обробнику події, якщо застосувати до кнопки метод **place** з новими значеннями *x* та *y*.



Працюємо з комп'ютером

Увага! Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

Задача. Створіть проект, у якому потрібно:

- 1) відкрити вікно і в ньому розмістити кнопку зі значеннями властивостей за замовчуванням;
- 2) відкрити вікно і в ньому розмістити кнопку зі значеннями властивостей **text**, **height**, **width**;

3) створити обробник події для кнопки, у якому змінити текст на кнопці, її ширину, колір символів, доступність.

1. Відкрийте вікно середовища розробки проектів.
2. Уведіть команди для створення та відкриття вікна розмірами 600 на 400 пікселів.

3. Створіть кнопку зі значеннями властивостей за замовчуванням і розмістіть її у вікні. Для цього введіть команди:

```
button = Button()
button.pack()
```

4. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.2**.
 5. Запустіть проєкт на виконання.
 6. Закрийте вікно виконання проєкту після ознайомлення з результатом його виконання.
 7. Установіть для кнопки такі значення властивостей:
- Напис на кнопці – *Кнопка 1*;
 - Висота – 3 символи;
 - Ширина – 15 символів.

Для цього змініть команду створення кнопки на таку:

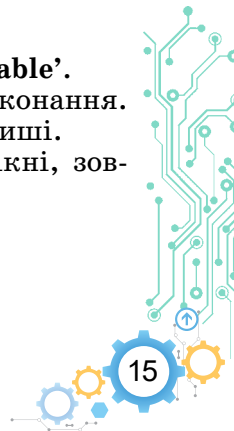
```
button = Button(text = 'Кнопка 1', height = 3, width = 15).
```

8. Установіть для кнопки відступ зверху – 30 пікселів. Для цього змініть команду розміщення кнопки на таку: **button.pack(pady = 30)**.
 9. Збережіть проєкт під тим самим іменем і запустіть його на виконання.
 10. Проаналізуйте, як змінилися значення властивостей кнопки порівняно з її значеннями за замовчуванням.
 11. Закрийте вікно виконання проєкту.
 12. Створіть обробник події з іменем **click** і пов'яжіть його з кнопкою. Для цього:

1. Уведіть після першого рядка тексту проєкту рядок заголовка обробника події: **def click():**
2. Уведіть після рядка заголовка команди:
 - **button['text'] = '8 А клас'** – установити на кнопці текст *8 А клас*;
 - **button['width'] = 30** – установити для кнопки ширину 30 символів;
 - **button['fg'] = 'red'** – установити для кнопки *червоний* колір символів.
3. Додайте до команди створення кнопки властивість **command**, що пов'яже з кнопкою створену процедуру. Команда повинна набути вигляду:

```
button = Button(text = 'Кнопка 1', height = 3, width = 15, command = click)
```

13. Збережіть змінену версію проєкту та запустіть його на виконання.
 14. Наведіть вказівник на кнопку та клацніть ліву кнопку миші.
 15. Зверніть увагу, як змінився вигляд кнопки.
 16. Закрийте вікно виконання проєкту.
 17. Додайте до процедури **click** команду **button['state'] = 'disable'**.
 18. Збережіть змінену версію проєкту та запустіть його на виконання.
 19. Наведіть вказівник на кнопку та клацніть ліву кнопку миші.
 20. Зверніть увагу, чи змінилося розташування кнопки у вікні, зовнішній вигляд кнопки, її доступність. Поясніть, чому.
 21. Закрийте вікно виконання проєкту.
 22. Закрийте вікно середовища розробки проєкту.





Найважливіше в цьому пункті

Щоб розмістити кнопку у вікні, потрібно створити новий об'єкт типу **Button**, пов'язати його зі змінною, що визначатиме ім'я кнопки, установити значення властивостей кнопки, розмістити створений об'єкт у вікні.

Для створення кнопки та пов'язування її зі змінною використовують команду вигляду `<ім'я_кнопки> = Button()`.

Під час створення кнопки можуть бути встановлені значення її властивостей **text, width, height, bg, fg, font, state**. Для цього потрібно ввести в дужках через кому пари `<властивість> = <значення>`. Для кнопки можуть бути встановлені значення властивостей **text, width, height, bg, fg, font, state** та інших.

Для розміщення кнопки у вікні можуть бути використані методи **pack()** або **place()**.

Пов'язати з кнопкою обробник будь-якої події можна так само, як з вікном, використовуючи метод **bind()**.

Щоб пов'язати з кнопкою метод опрацювання події **Click**, можна в команді створення кнопки вказати ім'я відповідної процедури – обробника події як значення властивості **command**. У заголовку цієї процедури не потрібно вводити ключове слово **event**.



Дайте відповіді на запитання

1. Як створити кнопку?
2. Як розмістити створену кнопку у вікні?
3. Які ви знаєте властивості кнопки? Що визначає кожна з них?
4. Як задати значення властивостей кнопки?
5. Якими способами можна призначити для кнопки обробника подій?
6. Як можна призначити для кнопки обробник події **Click**?
7. Як у проєкті змінити значення властивостей кнопки?



Обговоріть і зробіть висновки

1. Чим відрізняється у проєкті створення та відкриття вікна від створення та розміщення кнопки?
2. Що спільного і відмінного в обробниках події для вікна і для кнопки?



Виконайте завдання

1. Створіть проєкт, у якому в результаті настання події **Click** для кнопки її ширина стане 30 символів, висота 3 символи і зміниться текст на кнопці на назву вашого населеного пункту. Початкові значення властивостей кнопки залиште за замовчуванням. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.2.1**.

2. Створіть проєкт, у якому в результаті настання події **MouseMove** для кнопки її ширина стане 20 символів, висота 4 символи і зміниться текст на кнопці на ваше прізвище. Початкові значення властивостей кнопки виберіть самостійно. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.2.2**.

3. Створіть проєкт, у якому в результаті настання події **DoubleClick** для кнопки зміниться її ширина і висота, текст на кнопці, колір фону і колір тексту, а для вікна – його розміри, колір фону, текст у рядку заголовка. Значення цих властивостей виберіть самостійно. Початкові значення властивостей кнопки залиште за замовчуванням. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.2.3**.

4. Створіть проєкт, у якому розташуйте у вікні три кнопки однакових розмірів одна під одною. Початковий колір верхньої кнопки – червоний, двох інших – чорний. Створіть обробники події **Click** для кожної кнопки такі, щоб:

- вибір середньої кнопки робив її жовтою, а двох інших – чорними;
- вибір нижньої кнопки робив її зеленою, а двох інших – чорними;
- вибір верхньої кнопки робив її червоною, а двох інших – чорними.

Збережіть проєкт у вашій папці у файлі з іменем завдання 4.2.4.

5. Створіть проєкт, у якому розташуйте у вікні три кнопки горизонтально одну поруч з іншою. Установіть такі значення їх властивостей, щоб на початку виконання проєкту перша кнопка була видима і доступна, друга – видима і недоступна, третя – невидима і недоступна. Для кожної кнопки створіть обробник події **Click**, щоб:

- вибір першої кнопки робив її невидимою і недоступною, другу кнопку – видимою і доступною, третю кнопку – видимою і недоступною;
- вибір другої кнопки робив її невидимою і недоступною, третю кнопку – видимою і доступною, першу кнопку – видимою і недоступною;
- вибір третьої кнопки робив її невидимою і недоступною, першу кнопку – видимою і доступною, другу кнопку – видимою і недоступною.

Збережіть проєкт у вашій папці у файлі з іменем завдання 4.2.5.



Готуємось до вивчення нового матеріалу

1. Які ще елементи керування, крім кнопки, ви використовували під час роботи з різними програмами?
2. Для чого використовується графічний об'єкт **Напис** у текстовому процесорі **Word**?

4.3. НАПИС

У цьому пункті йтиметься про:

- ▶ напис, як елемент керування у вікні, його властивості та їх значення;
- ▶ створення напису та способи його розміщення у вікні;
- ▶ події, які можуть відбуватися з написом;
- ▶ обробники подій з написом.

НАПИС. ВЛАСТИВОСТІ НАПISУ



Пригадайте

- Які ви знаєте властивості кнопки? Наведіть приклади їх значень.
- Якими командами створюється кнопка і розміщується у вікні?
- Як змінити значення властивостей кнопки?

Ще одним елементом керування, який можна використати у проєкті, є **напис**. **Напис** – це елемент керування, який використовується для виведення текстових повідомлень.

Щоб розмістити напис у вже створеному і відкритому вікні, потрібно:



1. Створити новий об'єкт типу **Label** (англ. *label* – напис), пов'язати його зі змінною, яка визначатиме ім'я об'єкта.
2. Установити значення властивостей напису або залишити їх за замовчуванням.
3. Розмістити створений об'єкт у вікні.
Для створення напису та пов'язування його зі змінною використовують команду вигляду:

```
<ім'я_напису> = Label()
```

або

```
<ім'я_напису> = Label(<набір властивостей та їх значень>)
```

У першому випадку буде створено напис зі значеннями властивостей **висота**, **ширина** і **колір**, установленими за замовчуванням, без напису в ньому. У другому випадку буде створено напис, значення властивостей якого задані у цій команді. У дужках через кому вказують пари **<властивість> = <значення>**.

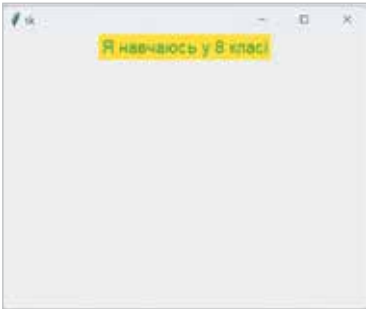
Для напису можуть бути встановлені значення властивостей **text**, **width**, **height**, **bg**, **fg**, **font**, **state** та інших, аналогічно до того, як вони встановлювалися для кнопки.

Наприклад, після виконання команди:

```
lb = Label(text = 'Я навчаюсь у 8 класі', bg = 'yellow', fg = 'green', font = 'Arial 14')
```

буде створено напис з іменем *lb*, у якому буде зеленими літерами на жовтому фоні виведено текст *Я навчаюсь у 8 класі*, шрифт символів *Arial*, розмір символів 14 (мал. 4.5).

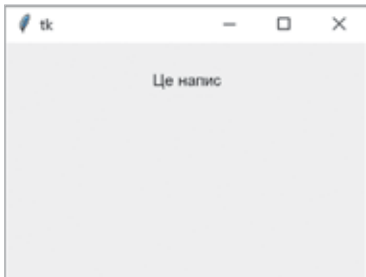
Звертаємо вашу увагу: якщо під час створення напису значення властивості **width** не задається, то ширина напису автоматично визначається довжиною тексту, який є значенням властивості **text**.



Мал. 4.5. Напис у вікні

Значення властивостей напису також можна задавати окремими командами виду: **<ім'я напису>['<назва властивості>'] = <значення властивості>**.

Наприклад, установити для напису з іменем *label* ширину 25 символів можна командою **label['width'] = 25**, установити шрифт *Times New Roman* з висотою символів 10 можна командою **label['font'] = ('Times New Roman', 10)**, задати синій колір символів – командою **label['fg'] = 'blue'**. Такого виду команди можна використовувати і для змінення значень властивостей напису під час виконання проекту.



Мал. 4.6. Вікно проекту з написом

Для розміщення напису у вікні проекту використовують вже відомі вам методи **pack()** і **place()**. Наприклад, для розміщення у вікні напису з іменем *label* з відступом 20 пікселів від рядка заголовка вікна можна використати команду **label.pack(pady = 20)** (мал. 4.6).

ОБРОБНИКИ ПОДІЙ ДЛЯ НАПISУ



Пригадайте

- Які події можуть відбуватися з кнопкою?
- Що таке *обробник події*?
- Як записується обробник події у проєкті та як він виконується?

Хоча написи призначені, переважно, для виведення текстів, але з ними, як і з вікнами і кнопками, можна пов'язувати обробники різних подій.

З написом, як і з вікном проєкту або кнопкою, можна пов'язати процедуру – обробник деякої події, використовуючи метод **bind()**:

`<ім'я напису>.bind(<подія>, <ім'я процедури>)`

Наприклад, потрібно, щоб після настання події **Click** з написом з іменем **label** колір фону напису ставав *жовтим*, колір символів – *червоним*, розмір символів *14*, у напис виводився заданий текст *Я навчаюсь у 8 класі* та результат його виконання, у тому числі результат виконання процедури – обробника події **Click** для напису.

```
from tkinter import *
def label_click(event):
    label['text'] = 'Я навчаюсь у 8 класі'
    label['font'] = 14
    label['fg'] = 'red'
    label['bg'] = 'yellow'
root=Tk()
root.geometry('300x200')
label = Label(text = 'Це напис')
label.bind('<1>', label_click)
label.pack(pady = 20)
```



Мал. 4.7. Обробник події **Click** для напису та результат його виконання



Для тих, хто хоче знати більше

Текст у написі можна вивести в кілька рядків. Для цього на місці, де повинен розпочинатися новий рядок, потрібно ввести символи `\n`. Наприклад, у написі, створеному командою `label = Label(text = 'Я навчаюсь \n у 8 класі')`, у першому рядку буде відображатися текст *Я навчаюсь*, а у другому – *у 8 класі*. Результат виконання цієї команди наведено на малюнку 4.8.



Мал. 4.8. Виведення тексту в написі у два рядки





Працюємо з комп'ютером

Увага! Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

Задача. Створіть проєкт, у якому потрібно:

- 1) відкрити вікно й у ньому розмістити напис з текстом *Це напис* і зі значеннями інших властивостей за замовчуванням;
 - 2) відкрити вікно й у ньому розмістити напис із заданими значеннями властивостей **text, bg, fg, font**;
 - 3) створити обробник події для напису, у якому змінити текст у написі, колір символів, колір фону, шрифт.
1. Відкрийте середовище розробки проєктів.
 2. Відкрийте вікно текстового редактора для введення тексту проєкту.
 3. Уведіть команди для створення вікна розміром 600 на 400 пікселів.
 4. Створіть напис з текстом *Це напис* і розмістіть його у вікні. Для цього введіть команди:

```
label = Label(text = 'Це напис')
label.pack()
```

5. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.3**.
6. Запустіть проєкт на виконання. Зверніть увагу на розташування напису у вікні, на колір його фону, на текст у написі, на колір і розмір символів.
7. Закрийте вікно виконання проєкту.
8. Установіть такі значення властивостей напису: колір фону – *зелений*, колір символів – *білий*, вид шрифту – *Arial*, розмір символів – *16*, накреслення – *підкреслений*. Для цього додайте в команду створення напису властивості та їх значення:

```
label = Label(text = 'Це напис', bg = 'green',
              fg = 'white', font = 'Arial 16 underline')
```

9. Збережіть змінену версію проєкту в тому самому місці з тим самим іменем.
10. Запустіть проєкт на виконання. Зверніть увагу на змінення вигляду напису.
11. Закрийте вікно виконання проєкту.
12. Створіть процедуру з іменем **click** для змінення значень властивостей напису. Для цього введіть після першого рядка тексту проєкту такі команди (текст після символу **#** – це коментар, пояснення; його вводити не потрібно):

```
def click(event):
    label['text'] = 'Моя прізвище - Семеновко' # змінення тексту в написі
    label['bg'] = 'yellow' # змінення кольору фону напису на жовтий
    label['fg'] = 'blue' # змінення кольору символів на синій
    # шрифт Times New Roman, розмір символів 12, накреслення курсивне
    label['font'] = ('Times New Roman', 12, 'italic')
```

13. Пов'яжіть створену процедуру з написом, зазначивши як подію клацання лівої кнопки миші. Для цього введіть після останнього рядка проєкту команду:

```
label.bind('<1>', click)
```

14. Збережіть змінену версію проєкту.
15. Запустіть проєкт на виконання.
16. Наведіть вказівник на напис і клацніть ліву кнопку миші. Зверніть увагу на змінення вигляду напису.
17. Закрийте вікно виконання проєкту.
18. Закрийте вікно середовища розробки проєктів.



Найважливіше в цьому пункті

Напис – це елемент керування, який використовується для виведення у вікні текстових повідомлень.

Для створення напису та пов'язування його зі змінною використовують команду вигляду:

```
<ім'я напису>= Label()
```

або

```
<ім'я напису> = Label (<набір властивостей та їх значень>)
```

У першому випадку буде створено напис зі значеннями властивостей **висота**, **ширина** і **колір**, установленими за замовчуванням, без тексту в ньому. У другому випадку буде створено напис, значення властивостей якого задано у цій команді. У дужках через кому вказують пари **<властивість> = <значення>**.

Для напису можуть бути встановлені значення властивостей **text**, **width**, **height**, **bg**, **fg**, **font**, **state** та інших, аналогічно до того, як вони встановлювалися для кнопки.

Значення властивостей також можна задавати та змінювати окремими командами виду:

```
<ім'я напису>[<назва властивості>] = <значення властивості>
```

Для розміщення напису у вікні проєкту використовують методи **pack()** і **place()**.

З написом можна пов'язати процедуру – обробник деякої події, використовуючи метод **bind()**:

```
<ім'я напису>.bind(<'подія'>, <ім'я процедури>)
```



Дайте відповіді на запитання

1. Для чого використовують напис?
2. Як створити напис?
3. Як можна встановити значення властивостей напису?
4. Які ви знаєте властивості напису? Поясніть їх.
5. Як розмістити напис у вікні?



Обговоріть і зробіть висновки

1. Що відбуватиметься, якщо довжина тексту, який виводиться на кнопку і в напис, більша за їх ширину?
2. З якою метою у вікні розміщують порожній напис?



Виконайте завдання

1. Створіть проєкт, у якому в результаті настання події **Click** для напису буде встановлено ширину напису 30 символів і зміниться текст у написі на назву вулиці, на якій знаходиться ваш заклад освіти. Початкові значення властивостей напису залиште за замовчуванням. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.3.1**.

2. Створіть проєкт, у якому в результаті настання події **DbClick** для напису буде встановлено висоту напису 3 символи, збільшиться розмір символів і зміниться текст у написі на ваше ім'я. Початкові значення властивостей напису залиште за замовчуванням. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.3.2**.

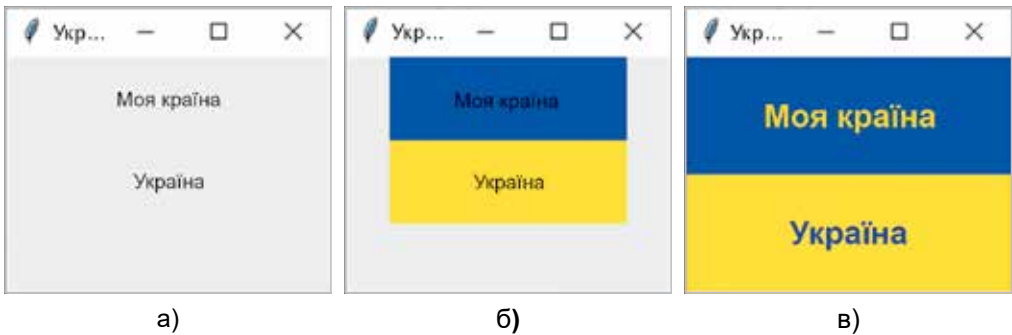


3. Створіть проєкт, у якому в результаті настання події **MouseMove** для напису буде встановлено новий текст у ньому, іншим шрифтом, іншим розміром символів, іншим кольором символів. Початкові значення властивостей напису виберіть самостійно. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.3.3**.

4. Створіть проєкт, у якому в результаті настання події **Click** для вікна ширина напису збільшиться на 5 символів, у напис виведеться назва вашого закладу освіти жовтим кольором на червоному тлі. Початкові значення властивостей напису виберіть самостійно. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.3.4**.

5. Створіть проєкт, у якому в результаті настання події **Click** для кнопки розмір вікна проєкту стане 500 на 400 і в напис виведеться назва вашого закладу освіти синім кольором на жовтому тлі. Початкові значення властивостей вікна і напису виберіть самостійно. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.3.5**.

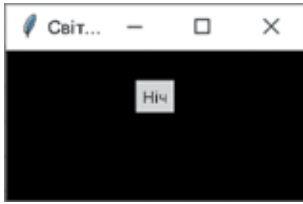
6. Створіть проєкт, у якому розташуйте у вікні два написи *Моя країна* та *Україна*. Установіть заголовок вікна – *Україна*. Після настання події **Click** з першим написом колір тла першого напису повинен стати *синім*, другого напису – *жовтим*. Після настання події **Click** з другим написом колір символів першого напису повинен стати *жовтим*, другого напису – *синім*, шрифт символів у обох написах – *Arial, 14, напівжирний*. Вигляд вікна проєкту наведено на малюнку 4.9. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.3.6**.



Мал. 4.9. Вигляд вікна проєкту:

- а) після запуску;
 б) після настання події **Click** для першого напису;
 в) після настання події **Click** для другого напису

7. Створіть проєкт, у якому розташуйте у вікні два написи *Ніч* і *День*, задайте заголовок вікна – *Світло*. Після запуску проєкту колір фону вікна – *чорний*, колір фону першого напису – *сірий*, другий напис *не відображається*. Після настання події **Click** з першим написом колір фону вікна стає *білим*, перший напис *зникає*, другий напис відображається на *сірому* фоні. Після настання події **Click** з другим написом вікно проєкту повертається до початкового стану. Вигляд вікна проєкту наведено на малюнку 4.10. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.3.7**.



а)



б)

Мал. 4.10. Вікно проєкту: а) початковий стан і після настання події **Click** для другого напису; б) після настання події **Click** для першого напису



Готуємось до вивчення нового матеріалу

1. Для чого використовуються поля в текстовому процесорі **Word**?
2. Чим відрізняються об'єкти *поле* та *напис*?

4.4. ПОЛЕ. ПРОЄКТИ З ВХІДНИМИ ДАНИМИ ТА РЕЗУЛЬТАТАМИ

У цьому пункті йтиметься про:

- ▶ поле, його властивості та їх значення;
- ▶ створення поля та способи його розміщення у вікні;
- ▶ події, які можуть відбуватися з полем;
- ▶ обробники подій з полем;
- ▶ використання поля у проєктах з вхідними даними та результатами;
- ▶ використання у проєктах констант.

ПОЛЕ. ВЛАСТИВОСТІ ПОЛЯ



Пригадайте

- Які ви знаєте властивості напису? Наведіть приклади їх значень.
- Якими командами створюється напис і розміщується у вікні?
- Як змінити значення властивості напису?

Ще одним елементом керування, який можна використати у проєкті, є поле. **Поле** – це елемент керування, який використовується, в основному, для введення даних. Поле можна використовувати також для виведення результатів.

Щоб розмістити поле в уже створеному і відкритому вікні, потрібно:

1. Створити новий об'єкт типу **Entry** (англ. *entry* – запис, вхід), пов'язати його зі змінною, яка визначатиме ім'я об'єкта.
2. Установити значення властивостей поля або залишити їх за замовчуванням.
3. Розмістити створений об'єкт у вікні.

Для створення поля та пов'язування його зі змінною використовують команду вигляду:

`<ім'я_поля> = Entry()`

або

`<ім'я_поля> = Entry(<набір властивостей та їх значень>)`



У першому випадку буде створено поле зі значеннями властивостей **висота**, **ширина** і **колір**, установленими за замовчуванням, без тексту в ньому. У другому випадку буде створено поле, значення властивостей якого задані у цій команді. У дужках через кому вказують пари **<властивість> = <значення>**.

Властивості поля **width**, **bg**, **fg**, **font** аналогічні до відповідних властивостей кнопки та напису, а властивості **text** і **height** відсутні (висота поля визначається висотою символів у його вмісті). Для поля можна задавати товщину рамки як значення властивості **bd** (англ. *border* – межа).

Для розміщення поля у вікні проекту можна використовувати відомі вам методи **pack()** і **place()**.

На малюнку 4.11 наведено вигляд вікна проекту з двома полями. Верхнє поле створено командою **ent1 = Entry()** зі значеннями властивостей за замовчуванням, нижнє поле створено командою:

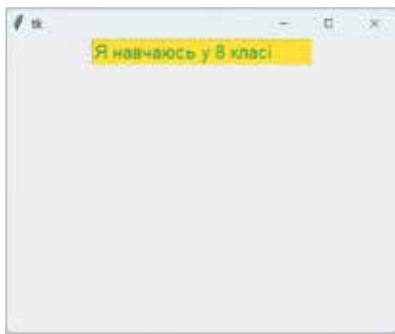
ent2 = Entry(width = 15, bg = 'yellow', fg = 'green', font = 'Arial 20', bd = 3)



Мал. 4.11. Вікно проекту з двома полями

Щоб вивести в поле деякий текст, використовують команду виду:

<ім'я поля>.insert(<позиція>, <текст>)
(англ. *insert* – вставити, вставка),



Мал. 4.12. Поле з текстом у вікні

де *позиція* – це позиція символу в полі, починаючи з якої буде вставлено текст. Якщо поле порожнє, то текст уставляється, починаючи з позиції 0, навіть якщо в команді вказано іншу позицію. Якщо ж у полі вже є деякий вміст, то текст можна вставити, починаючи з будь-якої позиції у цьому вмісті. Для вставлення тексту в кінець існуючого вмісту вказують позицію **END**.

Наприклад, щоб вивести в порожнє поле з іменем **entry** текст *Я навчаюсь у 8 класі*, потрібно виконати команду **entry.insert(0, 'Я навчаюсь у 8 класі')** (мал. 4.12).

Для змінення вмісту цього поля на *Я навчаюсь у 8-А класі* можна після попередньої команди додати команду **entry.insert(14, '-А ')**.

А щоб отримати в полі текст *Я навчаюсь у 8-А класі школи № 1*, можна після попередніх команд додати команду

`entry.insert(END, ' школи № 1')`.

Для очищення вмісту поля з іменем `entry` використовують команду:
`entry.delete(0, END)`.

Отримати з поля його вміст можна методом `get()` (англ. *get* – отримати).

Наприклад, у результаті виконання команди `x = entry.get()` значенням змінної `x` стане текст, який міститься в полі з іменем `entry`.

А щоб у напис з іменем `label` вивести текст, що введено в поле з іменем `entry`, потрібно виконати команду `label['text'] = entry.get()`.

Під час виконання проєкту буває зручно, щоб одразу після запуску проєкту в полі був установлений курсор. У такому разі кажуть, що поле знаходиться у **фокусі**. Для цього використовують метод `focus_set()` (англ. *focus* – фокус, *set* – установити). Наприклад, виконання команди `entry.focus_set()` приводить до встановлення курсора всередині поля з іменем `entry`.

ОБРОБНИКИ ПОДІЙ ДЛЯ ПОЛЯ



Пригадайте

- Які події можуть відбуватися з написом?
- Що таке *обробник події*?
- Як записується обробник події у проєкті і як він виконується?

Як і з іншими об'єктами вікна, з полем можна пов'язати процедури – обробники подій, використовуючи метод `bind()`.

Наприклад, на малюнку 4.13 наведено текст проєкту з процедурою – обробником події `Click` для поля. Ця процедура містить команди:

- встановлення *жовтого* кольору фону поля;
- встановлення *червоного* кольору для символів тексту в полі;
- встановлення розміру *12* для символів тексту в полі;
- виведення тексту, уведеного в полі, у напис;
- присвоювання змінній `x` тексту з поля;
- виведення у вікно повідомлень тексту – значення змінної `x`, яке є текстом з поля, до якого в кінці додано знак оклику.

Після запуску проєкту на виконання в поле можна ввести довільний текст (на малюнку 4.14, *а* в поле введено текст *Працюємо з полем*) і вибрати поле. Результат виконання обробника події `Click` для поля наведено на малюнку 4.14, *б*.

```
from tkinter import*
from tkinter.messagebox import*

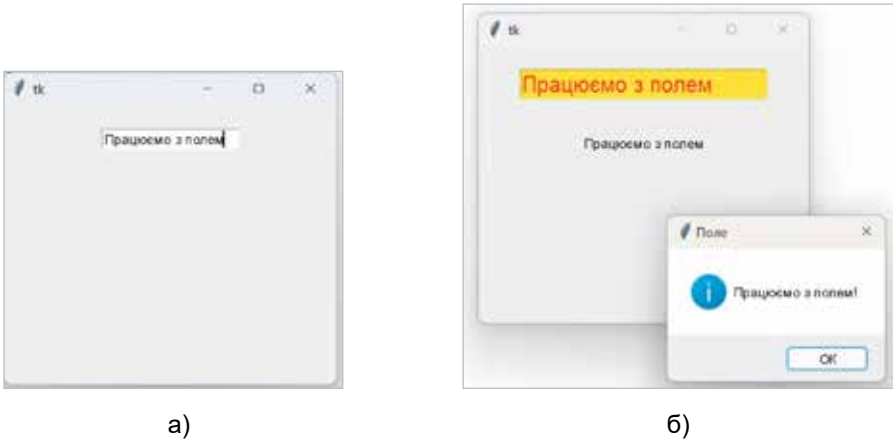
def click(event):
    entry['bg']='yellow'
    entry['fg']='red'
    entry['font']=12
    label['text'] = entry.get()
    x = entry.get()
    showinfo('Поле', x+'!')

root = Tk()
root.geometry('300x250')

entry = Entry()
entry.pack(pady = 20)
entry.focus_set()
entry.bind('<1>', click)

label = Label()
label.pack(pady = 10)
```

Мал. 4.13. Текст проєкту з обробником події `Click` для поля



Мал. 4.14. Вікно виконання проєкту після введення тексту в поле і до настання події **Click** для поля (а) та після настання події (б)

ПРОЄКТИ З УВЕДЕННЯМ ДАНИХ І ВИВЕДЕННЯМ РЕЗУЛЬТАТІВ



Пригадайте

- Для чого використовують команду присвоювання? Який її загальний вигляд?
- Як вивести текст у напис; у поле; у вікно повідомлень? • Які типи змінних ви знаєте? Які значення можуть мати змінні цих типів? Що означає тип змінної?

У ході виконання проєктів можна вводити значення вхідних даних і виводити значення результатів.

Для введення значень вхідних даних використовується **поле**, а для виведення значень результатів можна використовувати **поле**, **напис**, **вікно повідомлень**.

Створимо для прикладу проєкт, у якому обчислюватиметься сума двох довільних чисел.

Створимо математичну модель для цього проєкту:

вхідні дані: два довільних числа x і y ;

кінцеві результати: їх сума z ;

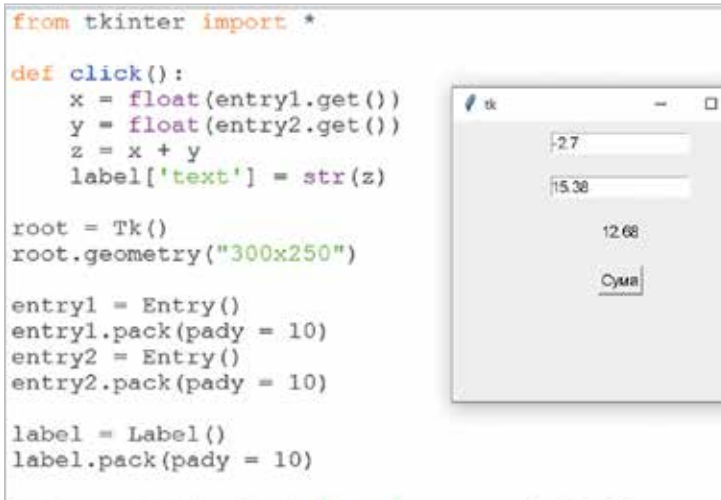
формули: $z = x + y$.

Оскільки в математичній моделі два вхідних даних, використаємо у цьому проєкті два поля, у які користувач під час виконання проєкту вводитиме доданки (вони можуть бути і цілі, і дробові). Кінцевий результат виведемо в напис. Розмістимо також у вікні кнопку, після вибору якої відбуватиметься додавання введених у поля чисел і виведення суми в напис.

Текст проєкту, що створює вікно з розміщеними в ньому необхідними об'єктами, процедура для додавання двох чисел, а також вікно виконання проєкту наведені на малюнку 4.15.

Розглянемо наведену процедуру.

Хоча користувач під час виконання проєкту вводить у поля числа, система тим не менше сприймає їх як тексти. Ці тексти є вмістом полів з іменами **entry1** і **entry2**. Тому перші дві команди процедури $x = \text{float}(\text{entry1.get}())$ і $y = \text{float}(\text{entry2.get}())$ переводять текстове



Мал. 4.15. Вікно з двома полями, написом і кнопкою та текст проєкту з процедурую для додавання двох чисел

подання чисел безпосередньо в числа і присвоюють ці значення змінним x і y відповідно.

Третя команда додає ці числа та присвоює суму змінній z .

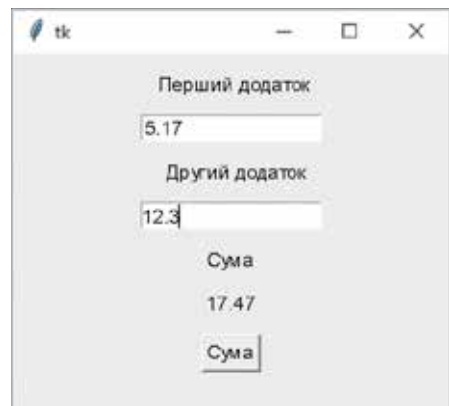
Четверта команда `label['text'] = str(z)` (англ. *string* – рядок) переводить обчислену суму з числа в його текстове подання та присвоює цей текст значенню властивості `text` напису з іменем `label`.

Значення змінних x , y , z , які використані у процедурі, за умовою задачі можуть бути довільними числами, тобто і цілими, і дробовими. Тому для таких змінних використано тип `float`.

Під час виконання створеного проєкту користувач уводить у поля два числа (на малюнку 4.15 це числа $-2,7$ і $15,38$), вибирає кнопку *Сума* і результат ($12,68$) виводиться в напис. Не закриваючи вікно виконання проєкту, користувач може ввести в поля інші числа, знову вибрати кнопку *Сума* та одержати новий результат. І так можна повторювати довільну кількість разів.

Звертаємо вашу увагу: під час уведення в поля дробових чисел ціла і дробова частини відокремлюються крапкою.

Щоб під час виконання проєкту користувачеві було зрозуміліше, у які поля що саме потрібно вводити і де виводитиметься результат, доцільно у вікні розмістити поруч з кожним полем і написом, призначеним для виведення результату, написи з пояснювальними текстами, наприклад, *Перший доданок*, *Другий доданок*, *Сума* (мал. 4.16).



Мал. 4.16. Вікно виконання проєкту додавання двох чисел з пояснювальними текстами



Звертаємо вашу увагу.

1. Для виведення значення кінцевого результату (змінна z) можна замість напису розмістити у вікні ще одне поле, наприклад з іменем `entry3`. І тоді у процедурі команда виведення була б такою: `entry3.insert(0, str(z))`.
2. Перед використанням у проєкті команди виведення в поле доцільно включати у проєкт команду очищення поля від можливих попередніх результатів виконання проєкту: `entry3.delete(0, END)`.
3. Можна було б також вивести значення змінної z у вікно повідомлень. Тоді команда виведення була б такою: `showinfo('Результат', str(z))`.

Якщо у проєкті вхідними даними могли би бути тільки цілі числа, то для читання їх з полів можна використати команди `x = int(entry1.get())` і `y = int(entry2.get())`.

У проєкті на малюнку 4.13 у команді `x = entry.get()` використана змінна x , значенням якої є текст. Така змінна має тип `string`.

ВИКОРИСТАННЯ У ПРОЄКТІ КОНСТАНТ (ПОСТІЙНИХ ВЕЛИЧИН)



Пригадайте

- Чим константи (постійні величини) відрізняються від змінних?
- Які константи вам відомі? Наведіть їх значення.

У проєктах, крім змінних, часто використовують константи (постійні величини). **Константа** – це величина, значення якої не змінюється в ході виконання проєкту.

Наприклад, якщо потрібно створити проєкт для обчислення маси сосни, то потрібно знати її об'єм і густину. Це може бути важливим під час розв'язування питання, які вантажівки потрібні для перевезення сосен на деревообробний завод.

І якщо об'єм сосни може бути різним (змінна величина), то густина сосни є постійною величиною. Значення густини можна знайти в таблиці густин речовин.

Створимо математичну модель для цього проєкту:

вхідні дані: об'єм сосни V ;

кінцеві результати: маса сосни m ;

формула: $m = \rho V$, де ρ – густина сосни – константа, що дорівнює 520 кг/м^3 .

У мові програмування **Python** прийнято записувати імена констант (постійних величин) великими літерами.

Уважатимемо, що у проєкті значення об'єму вводиться в кубічних метрах у поле з іменем `entry`, а значення маси виводиться в кілограмах у напис з іменем `label`.

Тоді відповідний фрагмент проєкту виглядатиме так:

```
RO = 520
v = float(entry.get())
m = v * RO
label['text'] = str(m)
```



Працюємо з комп'ютером

Увага! Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

Задача 1. Створіть проект для порівняння значень властивостей поля за замовчуванням і змінених значень.

1. Відкрийте середовище розробки проектів.
2. Уведіть команди створення і відкриття вікна розміром 400 на 300 пікселів:

```
from tkinter import*
root = Tk()
root.geometry('400x300')
```

3. Уведіть команди для створення поля та розміщення його у вікні:


```
entry = Entry()
entry.pack()
```
4. Збережіть проект у вашій папці у файлі з іменем **вправа 4.4.1**.
5. Запустіть проект на виконання. Зверніть увагу на місце розташування поля у вікні, на його розміри, на колір його фону, на текст у полі, на колір і розмір символів. Запишіть значення відповідних властивостей у зошит.
6. Закрийте вікно виконання проекту.
7. Змініть команду створення поля, установивши значення властивостей поля:

```
entry = Entry(width = 25, bg = 'yellow', fg = 'blue', font = 'Arial 14', bd = 3)
```

8. Уведіть команду для виведення в поле тексту:


```
entry.insert(0, 'Вивчаємо властивості поля')
```
9. Збережіть змінену версію проекту в тому самому місці під тим самим іменем.
10. Запустіть проект на виконання. Зверніть увагу на змінення розташування поля у вікні, на колір його фону, на текст у написі, на колір і розмір символів.
11. Закрийте вікно виконання проекту.

Задача 2. Створіть проект для обчислення добутку двох довільних чисел.

Створимо математичну модель для цього проекту:

вхідні дані: два довільних числа x і y ;

кінцеві результати: їх добуток z ;

формули: $z = x * y$.

Використаємо у проекті два поля для введення значень множників, ще одне поле для виведення значення добутку, кнопку, з якою пов'яжемо обробник події **Click**. Біля кожного поля розмістимо написи з відповідними пояснювальними текстами.

1. Відкрийте нове вікно середовища розробки проектів для створення нового проекту.
2. Уведіть команди створення вікна розміром 400 на 300 пікселів:

```
from tkinter import*
root = Tk()
root.geometry('400x300')
```



3. Уведіть команди для створення та розміщення у вікні з відступом 5 пікселів від попереднього компонента:

- напису з іменем **label1** і текстом *Перший множник*:

```
label1 = Label(text = 'Перший множник')
```

```
label1.pack(pady = 5)
```

- поля з іменем **entry1** для введення першого множника:

```
entry1 = Entry(width = 10)
```

```
entry1.pack(pady = 5)
```

- напису з іменем **label2** та текстом *Другий множник*:

```
label2 = Label(text = 'Другий множник')
```

```
label2.pack(pady = 5)
```

- поля з іменем **entry2** для введення другого множника:

```
entry2 = Entry(width = 10)
```

```
entry2.pack(pady = 5)
```

- напису з іменем **label3** та текстом *Добуток*:

```
label3 = Label(text = 'Добуток')
```

```
label3.pack(pady = 5)
```

- поля з іменем **entry3** для виведення результату множення:

```
entry3 = Entry(width = 10)
```

```
entry3.pack(pady = 5)
```

- кнопки з іменем **button** і текстом *Помножити*:

```
button = Button(text = 'Помножити')
```

```
button.pack(pady = 5)
```

4. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.4.2**.
5. Запустіть проєкт на виконання.
6. Переконайтесь у правильності розміщення компонентів у вікні та закрийте вікно виконання проєкту.
7. Доповніть команду створення кнопки вказівкою імені процедури **click** – обробника події **Click** для кнопки.
8. Створіть процедуру з іменем **click** і введіть такі команди:
- ```
x = float(entry1.get())
y = float(entry2.get())
z = x * y
entry3.delete(0, END)
entry3.insert(0, str(z))
```
9. Збережіть змінену версію проєкту в тому самому місці з тим самим іменем.
10. Запустіть проєкт на виконання.
11. Уведіть у поля два числа, наприклад *25* і *-12*.
12. Виберіть кнопку *Помножити*.
13. Проаналізуйте правильність роботи проєкту.
14. Уведіть у поля числа *7,4* і *-1,25*.
15. Виберіть кнопку *Помножити*.
16. Проаналізуйте правильність роботи проєкту.
17. Закрийте вікно виконання проєкту.



**Задача 3.** Створіть проєкт для обчислення площі круга за відомим його радіусом.

Створимо математичну модель для розв'язування цієї задачі:

**вхідні дані:** радіус круга  $r$ ;

**кінцеві результати:** площа круга  $S$ ;

**формули:**  $S = \pi \times r^2$ ,  $\pi$  – константа,  $\pi \approx 3,1416$ .

1. Відкрийте нове вікно середовища розробки проєктів для створення нового проєкту.
2. Уведіть команди створення вікна розміром 400 на 300 пікселів.  

```
from tkinter import*
root = Tk()
root.geometry('400x300')
```
3. Уведіть команди для створення та розміщення у вікні з відступом 10 пікселів від попереднього компонента:
  - напису з іменем `label1` і текстом *Радіус круга*:  

```
label1 = Label(text = 'Радіус круга')
label1.pack(pady = 10)
```
  - поля з іменем `entry1` для введення радіуса круга:  

```
entry1 = Entry(width = 10)
entry1.pack(pady = 10)
```
  - напису з іменем `label2` та текстом *Площа круга*:  

```
label2 = Label(text = 'Площа круга')
label2.pack(pady = 10)
```
  - поля з іменем `entry2` для виведення площі круга:  

```
entry2 = Entry(width = 10)
entry2.pack(pady = 10)
```
  - кнопки з іменем `button` і текстом *Площа круга*:  

```
button = Button(text = 'Площа круга')
button.pack(pady = 10)
```
4. Доповніть команду створення кнопки вказівкою імені процедури `click` – обробника події `Click` для кнопки.
5. Створіть процедуру з іменем `click` і введіть такі команди:  

```
PI = 3.1416
r = float(entry1.get())
s = PI * r ** 2
entry2.delete(0, END)
entry2.insert(0, str(s))
```
6. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.4.3**.
7. Запустіть проєкт на виконання.
8. Уведіть у поле радіус 10.
9. Виберіть кнопку *Площа круга*.
10. Перегляньте результат виконання проєкту.
11. Уведіть у поле радіус 4.12.
12. Виберіть кнопку *Площа круга*.
13. Перегляньте результат виконання проєкту.
14. Закрийте вікно виконання проєкту.
15. Закрийте вікно середовища розробки проєктів.





### Найважливіше в цьому пункті

**Поле** – це елемент керування, який використовується для введення значень вхідних даних і виведення значень результатів.

Щоб розмістити поле в уже створеному і відкритому вікні, потрібно:

1. Створити новий об'єкт типу **Entry** (англ. *entry* – запис, вхід), пов'язати його зі змінною, яка визначатиме ім'я об'єкта.
2. Установити значення властивостей поля або залишити їх за замовчуванням.
3. Розмістити створений об'єкт у вікні.

Для створення поля та пов'язування його зі змінною використовують команду вигляду:

```
<ім'я_поля> = Entry ()
```

або

```
<ім'я_поля> = Entry (<набір властивостей та їх значень>)
```

У першому випадку буде створено поле зі значеннями властивостей **висота**, **ширина** і **колір**, установленими за замовчуванням, без тексту в ньому. У другому випадку буде створено поле, значення властивостей якого задано у цій команді. У дужках через кому вказують пари **<властивість> = <значення>**.

Властивості поля **width**, **bg**, **fg**, **font** аналогічні до відповідних властивостей кнопки та напису, а властивості **text** і **height** відсутні (висота поля визначається висотою символів у його вмісті). Для поля можна задавати товщину рамки як значення властивості **bd** (англ. *border* – межа).

Для розміщення поля у вікні проєкту можна використовувати відомі вам методи **pack()** і **place()**.

Щоб вивести в поле деякий текст, використовують команду виду:

```
<ім'я поля>.insert(<позиція>, <текст>)
```

Отримати з поля його вміст можна, використовуючи метод **get()**, наприклад **label['text'] = entry.get()**.

Для очищення вмісту поля з іменем **entry** використовують команду:

```
entry.delete(0, END)
```

Як і з іншими об'єктами вікна, з полем можна пов'язати процедури – обробники подій, використовуючи метод **bind()**.

Прочитати число з поля з іменем **entry1** і присвоїти його змінній можна командою **x = float(entry1.get())** або **x = int(entry1.get())**.

Вивести значення змінної **z** у напис можна командою **label['text'] = str(z)**, а в поле – командою **entry.insert(0, z)**.

**Константа** – це величина, значення якої не змінюється в ході виконання проєкту. У мові програмування **Python** прийнято записувати імена констант (постійних величин) великими літерами.



### Дайте відповіді на запитання

1. Для чого може використовуватися поле?
2. Як створити поле і розмістити його у вікні?
3. Які властивості поля ви знаєте? Охарактеризуйте їх.
4. Як задати значення властивості поля в команді його створення?
5. Як змінити значення властивості поля в ході виконання проєкту?
6. Як отримати значення з поля; записати в поле; видалити дані з поля?
7. Якими командами можна перевести текстове подання числа безпосередньо в число і навпаки?
8. Чим константа відрізняється від змінної?



### Обговоріть і зробіть висновки

1. Чим відрізняються команди виведення в напис і в поле?
2. Чим відрізняються команди виведення в поле й у вікно повідомлень?



### Виконайте завдання

1. Створіть проект, у якому в результаті настання події **Click** для поля буде встановлено ширину поля 15 символів і виведено текст у поле «Інформатика» синім кольором на жовтому фоні. Початкові значення властивостей поля залиште за замовчуванням. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.1**.

2. Створіть проект, у якому в результаті настання події **DoubleClick** для поля буде встановлено товщину рамки поля 3 пікселя і змінено текст у полі на ваше ім'я синім кольором. Початкові значення властивостей поля виберіть самостійно. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.2**.

3. Створіть проект, у якому в результаті настання події **MouseMove** для поля буде змінено текст у ньому, встановлено червоний колір тексту, сірий колір фону і ширина поля збільшиться на 5 символів. Початкові значення властивостей поля виберіть самостійно. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.3**.

4. Створіть математичну модель і проект для обчислення значення виразу  $(2*a + b)*(a - b)$ . Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.4**.

5. Створіть математичну модель і проект для обчислення значення виразу  $(3*a - b)*(a + 2*b)$ . Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.5**.

6. Створіть математичну модель і проект для обчислення довжини кола за даним його радіусом, використовуючи константу. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.6**.

7. Створіть математичну модель і проект для обчислення шляху, який проходить світло у вакуумі за певний час. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.7**.

8. Створіть математичну модель і проект для визначення, яку суму грошей має заплатити споживач за використану електроенергію. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.8**.

9. Клієнт поклав у банк  $x$  грн під  $p$  % річних. За умовою банку прибуток за рік додається до вкладу. Створіть математичну модель і проект для визначення, який прибуток клієнт отримає через 2 роки. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.4.9**.





#### Практична робота № 4.

«Проекти з полями, написаними і кнопками, з уведенням даних і виведенням результатів»

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Задача 1.** Створіть проєкт з використанням поля і кнопки. Установіть такі початкові значення властивостей:

- поля:
  - ширина – 15 символів;
  - товщина рамки – 3 пікселі;
  - відступ від верхньої межі вікна – 30 пікселів;
- кнопки:
  - ширина – 10 символів;
  - відступ від поля – 10 символів;
  - текст на кнопці – назва вашого класу;
  - колір тексту – синій;
  - колір фону – жовтий.

Створіть обробник події **Click** для кнопки, виконання якого встановить червоний колір фону поля, шрифт символів *Arial*, розмір символів 14, збільшить ширину поля на 20 символів, установить колір тексту білий і виведе в поле текст *Ми використовуємо властивості поля!*.

Збережіть проєкт у вашій папці у файлі з іменем **практична 4.1**.

**Задача 2.** Створіть проєкт для обчислення значення виразу  $a*(4*b - a)$ .

1. Створіть математичну модель для розв'язування задачі. Запишіть її в зошит.
2. Визначте, які компоненти потрібно розмістити у вікні для введення значень вхідних даних, виведення значень кінцевих результатів. Розмістіть їх у вікні. Підпишіть їх.
3. Створіть обробник події **Click** для кнопки для обчислення значення виразу і виведення результату.
4. Збережіть проєкт у вашій папці у файлі з іменем **практична 4.2**.

**Задача 3.** Мати дала Марійці певну суму грошей і попросила купити 400 г печива, 2 пакети молока і хліб. Складіть проєкт для обчислення суми грошей, яка залишиться в Марійки.

1. Створіть математичну модель для розв'язування задачі. Запишіть її в зошит.
2. Визначте, які компоненти потрібно розмістити у вікні для введення значень вхідних даних, виведення результатів. Розмістіть їх у вікні. Підпишіть їх.
3. Створіть обробник події **Click** для кнопки для обчислення результату і виведення його.
4. Збережіть проєкт у вашій папці у файлі з іменем **практична 4.3**.



#### Готуємось до вивчення нового матеріалу

1. Які типи змінних ви знаєте? Наведіть приклади їх значень.
2. Що визначають типи змінних?

## 4.5. ЛОГІЧНІ ВИРАЗИ. ЗМІННІ ЛОГІЧНОГО ТИПУ. ЛОГІЧНІ ОПЕРАЦІЇ

У цьому пункті йтиметься про:

- ▶ логічні вирази;
- ▶ змінні логічного типу;
- ▶ логічні операції заперечення, кон'юнкція, диз'юнкція.

### ЛОГІЧНІ ВИРАЗИ



#### Пригадайте

• Що таке *висловлювання*? • На яких уроках ви використовували істинні та хибні висловлювання? Наведіть приклади. • Чи будь-яке речення є висловлюванням? Наведіть приклади.



#### Поміркуйте

- Як можна визначити, що висловлювання істинне?
- Як можна визначити, що висловлювання хибне?

Ви вже знаєте, що **висловлювання** – це речення, яке містить твердження про певний об'єкт або про зв'язки між об'єктами та про яке можна однозначно сказати, **істинне** воно чи **хибне**. Приклади **істинних висловлювань**: «Київ – столиця України»,  $2*5 - 4 = 6$ ,  $4 < 12$ . Приклади **хибних висловлювань**: «У січні 30 днів»,  $2*7 + 3 = 12$ ,  $3 > 5$ . Приклади речень, які не є висловлюваннями: «Закрий, будь ласка, вікно», «Тобі сподобався цей фільм?».

Висловлювання можна розглядати як **логічний вираз**.

**Логічним виразом** називається вираз, який може набувати одного з двох значень: **True** (англ. *true* – істина, істинність, правда) або **False** (англ. *false* – хиба, хибність, неправда). Значення **True** та **False** називають **логічними константами**.

**Логіка** (грец. λόγος – слово, смисл, думка, мова) – наука про форми, методи і закони правильного мислення, про способи міркування.

Якщо висловлювання істинне, то вважають, що значення відповідного логічного виразу дорівнює **True**. Тобто, «Київ – столиця України» = **True**,  $(2*5 - 4 = 6) = \mathbf{True}$ ,  $(4 < 12) = \mathbf{True}$ .

Якщо висловлювання хибне, то вважають, що значення відповідного логічного виразу дорівнює **False**. Тобто, «Цей підручник для учнів сьомого класу» = **False**,  $(2*7 + 3 = 12) = \mathbf{False}$ ,  $(3 > 5) = \mathbf{False}$ .

В алгоритмах логічні вирази використовують у розгалуженнях і циклах. Ми розглянемо таке використання в наступних пунктах.

### ЗМІННІ ЛОГІЧНОГО ТИПУ



#### Пригадайте

• Які типи змінних ви знаєте? Наведіть приклади значень змінної кожного з цих типів. • Що визначає тип змінної?

У попередніх пунктах ви використовували змінні типів **float** та **int**, значеннями яких були числа, та типу **str**, значеннями яких були рядки тексту.



В інформатиці використовують також змінні, значенням яких можуть бути **True** або **False**. У мові програмування **Python** ці значення потрібно обов'язково писати з великої літери.

Змінні, які можуть набувати одного з двох значень **True** або **False**, називаються **змінними логічного типу**.

Логічний тип змінних у мові програмування **Python** позначається **bool** (англ. *boolean* – логічний).

*Звертаємо вашу увагу:* самі змінні логічного типу і логічні константи також є логічними виразами.

Надавати змінним логічного типу відповідні значення можна командою присвоювання. Наприклад,  $x = \text{True}$  або  $x = \text{False}$ .

Змінній логічного типу можна присвоювати значення іншої змінної логічного типу, наприклад,  $y = x$ .

Змінній логічного типу можна також присвоювати значення логічного виразу. Тобто можна використовувати такі команди присвоювання:  $x = (32 < 13)$ ,  $y = (2 + 12/3 == 6)$ . У першій команді значення логічного виразу  $32 < 13$  дорівнює **False**, тому і значення змінної  $x$  теж **False**. У другій команді знак  $==$  використано для порівняння, чи дорівнює значення виразу  $2 + 12/3$  числу **6**. Оскільки вони рівні, то значення логічного виразу  $2 + 12/3 == 6$  дорівнює **True**, і тому значення змінної  $y$  теж **True**.

## ЛОГІЧНІ ОПЕРАЦІЇ



### Пригадайте

- Які ви знаєте арифметичні операції? Над якими об'єктами вони виконуються? Що є результатом цих операцій?
- Яку операцію над текстами ви використовували? Що є результатом такої операції?

Найпростіші логічні операції ви вже знаєте. Це **операції порівняння**: дорівнює, не дорівнює, більше, менше, більше або дорівнює, менше або дорівнює. У мові програмування **Python** ці операції позначають так:

|           |             |             |        |              |                     |
|-----------|-------------|-------------|--------|--------------|---------------------|
| <b>==</b> | дорівнює    | <b>&gt;</b> | більше | <b>&gt;=</b> | більше або дорівнює |
| <b>!=</b> | не дорівнює | <b>&lt;</b> | менше  | <b>&lt;=</b> | менше або дорівнює  |

Розглянемо ще три логічні операції: **заперечення**, **кон'юнкція** (лат. *conjunctio* – об'єднання) і **диз'юнкція** (лат. *disjunctio* – роз'єднання).

**Запереченням** логічного виразу  $x$  називається логічна операція, результат якої дорівнює **True**, якщо значення логічного виразу  $x$  дорівнює **False**, і дорівнює **False**, якщо значення логічного виразу  $x$  дорівнює **True**.

Заперечення логічного виразу  $x$  у мові програмування **Python** позначають **not x**. Інколи в математиці заперечення логічного виразу  $x$  позначається  $\bar{x}$  або  $\neg x$ .

Тобто, якщо  $x = \text{True}$ , то **not x = False**, і якщо  $x = \text{False}$ , то **not x = True**. Для операції заперечення можна скласти таку таблицю, яку називають **таблицею істинності**:

| $x$   | $\text{not } x$ |
|-------|-----------------|
| True  | False           |
| False | True            |

Для заперечення висловлювання часто використовують частку **не** до дієслова або до узагальнюючих слів *усі, кожен*. Наприклад, результатом заперечення висловлювання  $A = \text{«Існує найбільше натуральне число»}$  є висловлювання  $\text{not } A = \text{«Не існує найбільшого натурального числа»}$ , а результатом заперечення висловлювання  $B = \text{«Усі натуральні числа парні»}$  є висловлювання  $\text{not } B = \text{«Не всі натуральні числа парні»}$ .

**Кон'юнкцією** двох логічних виразів  $x$  і  $y$  називається логічна операція, результат якої дорівнює **True**, якщо значення кожного з логічних виразів  $x$  і  $y$  дорівнюють **True**, і результат якої дорівнює **False**, якщо значення хоча б одного з логічних виразів  $x$  або  $y$  дорівнює **False**.

Кон'юнкція двох логічних виразів  $x$  і  $y$  у мові програмування **Python** позначається  $x \text{ and } y$  (англ. *and* – і, та). Інколи в математиці кон'юнкція двох логічних виразів  $x$  і  $y$  позначається  $x \wedge y$ .

Таблиця істинності для кон'юнкції виглядає так:

| $x$   | $y$   | $x \text{ and } y$ |
|-------|-------|--------------------|
| True  | True  | True               |
| True  | False | False              |
| False | True  | False              |
| False | False | False              |

Для утворення кон'юнкції двох висловлювань часто використовують сполучники **і, та**. Наприклад, висловлювання  $\text{«Число 36 кратне 3 і число 36 кратне 2»}$  є кон'юнкцією двох висловлювань:  $\text{«Число 36 кратне 3»}$  і  $\text{«Число 36 кратне 2»}$ , а висловлювання  $\text{«По місту курсують автобуси та трамвай»}$  є кон'юнкцією двох висловлювань:  $\text{«По місту курсують автобуси»}$  та  $\text{«По місту курсують трамвай»}$ .

**Диз'юнкцією** двох логічних виразів  $x$  і  $y$  називається логічна операція, результат якої дорівнює **True**, якщо значення хоча б одного з логічних виразів  $x$  або  $y$  дорівнює **True**, і результат якої дорівнює **False**, якщо значення кожного з логічних виразів  $x$  і  $y$  дорівнює **False**.

Диз'юнкція двох логічних виразів  $x$  і  $y$  у мові програмування **Python** позначається  $x \text{ or } y$  (англ. *or* – або). Інколи в математиці диз'юнкція двох логічних виразів  $x$  і  $y$  позначається  $x \vee y$ .

Таблиця істинності для диз'юнкції виглядає так:

| $x$   | $y$   | $x \text{ or } y$ |
|-------|-------|-------------------|
| True  | True  | True              |
| True  | False | True              |
| False | True  | True              |
| False | False | False             |



Для утворення диз'юнкції висловлювань часто використовують сполучник **або**. Наприклад, висловлювання  $11 \leq 21$  (*11 менше або дорівнює 21*) є диз'юнкцією двох висловлювань:  $(11 < 21)$  **або**  $(11 = 21)$ .

Часто використовують логічні вирази, що містять кілька логічних операцій. Наприклад,  $x$  **or**  $y$  **and** **not**  $z$ . У таких виразах для логічних операцій, розглянутих вище, визначено такий **пріоритет виконання логічних операцій**:

- спочатку виконуються операції заперечення;
- потім операції кон'юнкції;
- потім операції диз'юнкції.

Для визначення значень логічного виразу можна скласти **таблицю істинності**. У ній потрібно розглянути всі можливі значення змінних логічного типу, що входять до цього виразу, та виконати, враховуючи пріоритет операцій, усі логічні операції, які входять до цього виразу. Наприклад, для логічного виразу  $x$  **or**  $y$  **and** **not**  $z$  вона буде такою:

| $x$   | $y$   | $z$   | $not\ z$ | $y\ and\ not\ z$ | $x\ or\ y\ and\ not\ z$ |
|-------|-------|-------|----------|------------------|-------------------------|
| True  | True  | True  | False    | False            | True                    |
| True  | True  | False | True     | True             | True                    |
| True  | False | True  | False    | False            | True                    |
| False | True  | True  | False    | False            | False                   |
| True  | False | False | True     | False            | True                    |
| False | True  | False | True     | True             | True                    |
| False | False | True  | False    | False            | False                   |
| False | False | False | True     | False            | False                   |

Для змінення порядку виконання логічних операцій використовують **дужки**. Так, наприклад, у логічному виразі  $(x\ or\ y)$  **and** **not**  $z$  спочатку буде виконано операцію диз'юнкції в дужках, потім операцію заперечення, а потім операцію кон'юнкції.

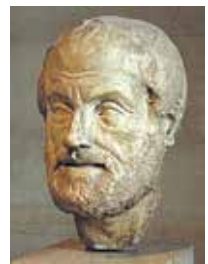


#### Цікаві факти з історії

Логіка як наука зародилася у Стародавній Греції. Засновником цієї науки вважається давньогрецький філософ Арістотель (384–22 р. р. до н. е) (мал. 4.17).

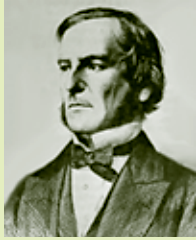
Означення логічних операцій і формулювання їх властивостей були сформульовані англійським математиком і філософом **Джорджем Булем** (1815–1864) (мал. 4.18). У 1854 році вийшла його основна робота «Дослідження законів мислення, на яких засновані математичні теорії логіки й імовірності». У ній досліджується система, яку сьогодні називають *Алгеброю висловлювань*, або *Булевою алгеброю*.

Булева алгебра стала фундаментальною науковою основою для створення комп'ютерів і використовується в усіх мовах програмування. Саме іменем цього вченого названо логічний тип даних **bool** у програмуванні. Існують мови логічного програмування, наприклад **PROLOG** [https://uk.wikipedia.org/wiki/Пролог\\_\(мова\\_програмування\)](https://uk.wikipedia.org/wiki/Пролог_(мова_програмування)).



Мал. 4.17.  
Арістотель





Мал. 4.18. Джордж Буль



Мал. 4.19. Ірина Хоменко

Потрібно також відзначити значний внесок українських учених у розвиток логіки як науки: Яна Лукасевича (1878–1956), Павла Копніна (1922–1971), Клавдії Руденко (1919–1974), Ірини Хоменко (нар. 1961 р.) (мал. 4.19).

У 2019 році **UNESCO** (англ. *United Nations Educational, Scientific and Cultural Organization* – Організація Об'єднаних Націй з питань освіти, науки і культури) оголосило 14 січня **Всесвітнім днем логіки**.



### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Задача 1.** Використовуючи вікно виконання проекту середовища розробки проектів мовою програмування **Python**, обчисліть значення логічного виразу  $x > 0$  and  $y > 0$  or  $z < 0$  для таких наборів значень змінних  $x, y, z$ :

- 1)  $x = 3, y = 5, z = -2$ ;
- 2)  $x = 4, y = -6, z = -5$ ;
- 3)  $x = 9, y = -5, z = 0$ ;
- 4)  $x = -5, y = -7, z = 2$ .

Запишіть значення логічного виразу для кожного набору значень вхідних даних у зошит.

1. Відкрийте середовище розробки проектів.
2. У вікні виконання проектів уведіть послідовно команди:

```
x = 3
y = 5
z = -2
t = x > 0 and y > 0 or z < 0
print(t)
```

3. Запишіть результат виконання цих команд у зошит.
4. Повторіть пункт 2 для інших наборів значень змінних  $x, y, z$ .
5. Запишіть у зошит результати виконання для кожного набору значень вхідних даних.
6. Закрийте вікно середовища розробки проектів.

**Задача 2.** Створіть проект з використанням елементів керування для обчислення значення виразу  $t = x > 0$  and  $(y > 0$  or  $z < 0)$ . Визначте значення виразу за таких значень змінних  $x, y, z$ :

- 1)  $x = 5, y = 8, z = -2$ ;
- 2)  $x = 5, y = -8, z = 3$ ;
- 3)  $x = -5, y = 8, z = -4$ ;
- 4)  $x = -5, y = -8, z = 5$ .



Створіть математичну модель для цієї задачі:

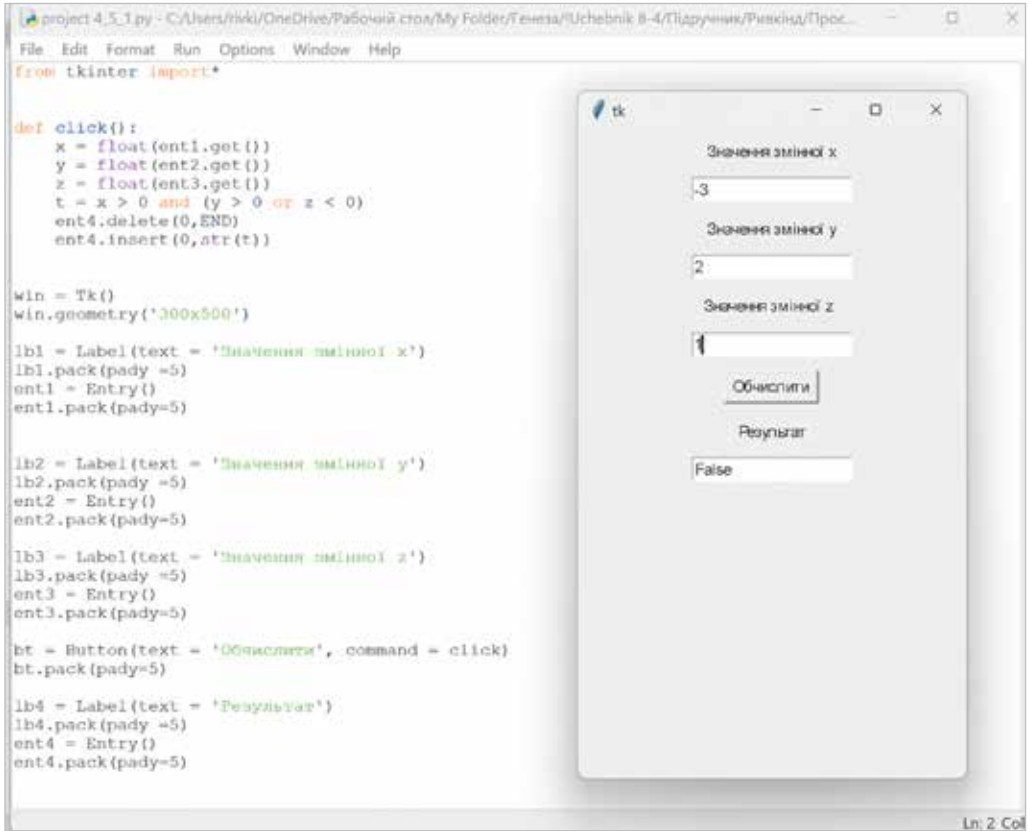
**вхідні дані:**  $x, y, z$ ;

**кінцеві результати:** значення логічної змінної  $t$ ;

**формули:**  $t = x > 0 \text{ and } (y > 0 \text{ or } z < 0)$ .

Оскільки вхідних даних три, розташуємо у вікні три поля для введення значень змінних  $x, y, z$ , четверте поле для виведення значення результату. Підпишемо кожне поле відповідно до його призначення. Розташуємо у вікні також і кнопку.

1. Відкрийте середовище розробки проєктів.
2. Відкрийте вікно створення нового проєкту.
3. Уведіть команди з малюнка 1.



Мал. 1. Текст проєкту для розв'язування **Задачі 2** і вікно виконання цього проєкту

4. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.5**.
5. Запустіть проєкт на виконання.
6. Уведіть у відповідні поля значення першого набору значень змінних  $x, y, z$ .
7. Виберіть кнопку і запишіть у зошит отриманий результат.
8. Уведіть послідовно значення з інших наборів, вибирайте кнопку і записуйте в зошит отримані результати.
9. Закрийте вікно виконання проєкту.
10. Закрийте вікно середовища розробки проєктів.



### Найважливіше в цьому пункті

**Висловлювання** – це речення, яке містить твердження про певний об'єкт або про зв'язки між об'єктами і про яке можна однозначно сказати, **істинне** воно чи **хибне**.

**Логічним виразом** називається вираз, який може набувати одного з двох значень: **True** або **False**. Ці значення називаються **логічними константами**.

Висловлювання можна розглядати як **логічний вираз**. Якщо висловлювання істинне, то вважають, що значення відповідного логічного виразу дорівнює **True**. Якщо висловлювання хибне, то вважають, що значення відповідного логічного виразу дорівнює **False**.

Змінні, які можуть набувати одного з двох значень **True** або **False**, називаються **змінними логічного типу**. Логічний тип змінних у мові програмування **Python** позначається **bool**. Змінні логічного типу і логічні константи також уважаються логічними виразами.

Над логічними виразами можна виконувати **логічні операції заперечення, кон'юнкція, диз'юнкція**.

**Запереченням** логічного виразу **x** називається логічна операція, результат якої дорівнює **True**, якщо значення логічного виразу **x** дорівнює **False**, і дорівнює **False**, якщо значення логічного виразу **x** дорівнює **True**.

**Кон'юнкцією** двох логічних виразів **x** і **y** називається логічна операція, результат якої дорівнює **True**, якщо значення кожного з логічних виразів **x** і **y** дорівнюють **True**, і дорівнює **False**, якщо значення хоча б одного з логічних виразів **x** або **y** дорівнює **False**.

**Диз'юнкцією** двох логічних виразів **x** і **y** називається логічна операція, результат якої дорівнює **True**, якщо значення хоча б одного з логічних виразів **x** або **y** дорівнює **True**, і дорівнює **False**, якщо значення кожного з логічних виразів **x** і **y** дорівнює **False**.

У логічних виразах для логічних операцій, розглянутих вище, визначено такий **пріоритет виконання операцій**: спочатку виконуються операції заперечення, потім операції кон'юнкції, потім операції диз'юнкції. Для змінення цього порядку виконання логічних операцій використовують дужки.

Для обчислення значень логічного виразу можна скласти **таблицю істинності**. У ній потрібно розглянути всі можливі значення змінних логічного типу, що входять до цього виразу, і виконати, враховуючи пріоритет операцій, усі логічні операції, які до нього входять.



### Дайте відповіді на запитання

1. Який вираз називається логічним? Наведіть приклади.
2. Які змінні називаються змінними логічного типу?
3. Що таке *заперечення логічного виразу*?
4. Що таке *кон'юнкція двох логічних виразів*?
5. Що таке *диз'юнкція двох логічних виразів*?
6. Яким є пріоритет виконання логічних операцій?



### Обговоріть і зробіть висновки

1. Як би ви назвали логічну операцію «**Якщо А, то В**»? Чи використовували ви таку логічну операцію на уроках? Якщо так, наведіть приклади. Чи використовували ви таку логічну операцію в повсякденному житті? Якщо так, наведіть приклади.



### Виконайте завдання

1. Наведіть приклади використання логічних виразів і логічних операцій у математиці, українській мові, інших предметах, у життєвих ситуаціях.



2. Сформулюйте заперечення висловлювань і з'ясуйте, істинні вони чи хибні:

- а) Число 3 є дільником числа 545.
- б) Автомобіль не має права їхати на червоне світло.
- в) Існують паралелограми з прямими кутами.
- г) Рівняння  $2x - 3 = 0$  має цілий корінь.
- д) Не існує натурального числа, що ділиться на 2.
- е) Існує ціле число, яке ділиться на всі цілі числа.
- ж) Усі люди вміють грати в шахи.

3. Серед наступних висловлювань знайдіть кон'юнкції та диз'юнкції та визначте, істинні вони чи хибні:

- а) Число 27 кратне 3 і кратне 9.
- б)  $17 < 42 < 18$ .
- в) Число 2 просте або парне.
- г)  $\triangle ABC$  є гострокутним, прямокутним або тупокутним.
- д) Діагоналі будь-якого паралелограму перпендикулярні та точкою перетину поділяються навпіл.
- е)  $7^2 = 49$  і  $(-7)^2 = 49$ .
- ж)  $21 \leq 21$ .
- и)  $21 \leq 18$ .

4. Нехай  $A$  = «Сергієнко вивчає англійську мову»,  $B$  = «Сергієнко має оцінку 8 з інформатики». Сформулюйте висловлювання:

- а)  $A$  and not  $B$ ;
- б) not  $A$  or  $B$ ;
- в)  $A$  or not  $A$  and  $B$ .

5. Запишіть у зошит порядок виконання логічних операцій у логічних виразах зі змінними логічного типу  $a$  і  $b$ :

- а)  $a$  or not  $a$  and  $b$ ;
- б) not  $a$  or not  $b$  and  $a$ ;
- в)  $(a$  or not  $b)$  and (not  $a$  or not  $b$ );
- г) not  $a$  and not  $b$  or  $c$  and  $d$ .

6. Створіть у зошиті таблиці істинності для логічних виразів зі змінними логічного типу  $a$  і  $b$ :

- а)  $a$  or  $a$  and  $b$ ;
- б)  $a$  or not  $a$  and  $b$ ;
- в)  $a$  or  $b$  or not  $b$ ;
- г) not  $a$  or not  $b$ ;
- д)  $(a$  or not  $b)$  and  $(a$  or  $b)$ ;
- е)  $(x$  or  $y)$  and not  $y$ ;
- ж)  $a$  and (not  $b$  or  $a$ ) and  $b$ ;
- и)  $(x$  or  $y)$  and not  $z$ .

7. Створіть проєкт для визначення значення логічного виразу  $(x$  or not  $y)$  and  $z$ , де  $x$ ,  $y$ ,  $z$  – змінні логічного типу. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.5.7. Використайте створений проєкт для заповнення в зошиті таблиці істинності для даного виразу за всіх можливих наборів значень змінних  $x$ ,  $y$ ,  $z$ .

8. Створіть проєкт для визначення значення логічного виразу  $x > 0$  and not  $(y < 0$  or  $x < 0)$  and  $y > 0$ , де  $x$ ,  $y$  – змінні числового типу. Збережіть проєкт у вашій папці у файлі з іменем завдан-

ня 4.5.8. Використайте створений проєкт для заповнення в зошиті таблиці істинності для даного виразу за вказаних наборів значень змінних  $x, y, z$ :

- а)  $x = 5, y = 8$ ;
- б)  $x = 5, y = -8$ ;
- в)  $x = -5, y = 8$ ;
- г)  $x = -5, y = -8$ .

9. На виробництві нафтопереробного заводу є три установки А, В, С. На кожній з них встановлено датчики. Якщо одночасно будуть перевищено граничні показники датчиків на першій установці і на будь-якій іншій або на всіх трьох установках, то виникає надзвичайна ситуація. У такому разі повинен пролунати попереджувальний сигнал. Для створення автоматизованої системи подавання цього сигналу складіть у зошиті логічний вираз і для нього заповніть таблицю істинності, на основі якої буде працювати ця система.

10. Михайлу дозволяють погратися на комп'ютері в комп'ютерні ігри, якщо він виконав домашні завдання *або* навів порядок у своїй кімнаті і завершила свою роботу з комп'ютером його старша сестра. У системі *Розумного будинку* можна фіксувати виконання кожного виду роботи. Це робить можливим автоматично отримувати повідомлення про надання доступу до комп'ютера. Складіть у зошиті логічний вираз і для нього заповніть таблицю істинності, на основі якої працює система надсилання повідомлень.



### Готуємось до вивчення нового матеріалу

- У яких випадках в алгоритмі потрібно використати розгалуження?
- Які види розгалуження ви знаєте? Як вони виконуються?

## 4.6. ЛОГІЧНІ ВИРАЗИ В РОЗГАЛУЖЕННЯХ. ВІКОННІ ПРОЄКТИ З РОЗГАЛУЖЕННЯМИ

У цьому пункті йтиметься про:

- ▶ використання логічних виразів і логічних операцій у розгалуженнях;
- ▶ віконні проєкти з розгалуженнями.

## ЛОГІЧНІ ВИРАЗИ В РОЗГАЛУЖЕННЯХ



### Пригадайте

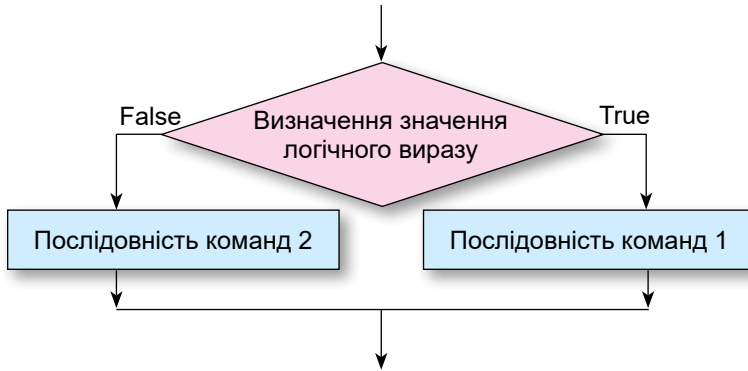
● У яких випадках в алгоритмах використовують розгалуження? ● Які ви знаєте види розгалужень? Як виконується розгалуження кожного з цих видів?

Ви вже знаєте, що кожне розгалуження розпочинається з **команди перевірки умови**, результатом виконання якої може бути або **Так (істина, True)**, або **Ні (хиба, False)**. І залежно від результату виконання команди перевірки умови, будуть виконуватися ті чи інші команди алгоритму.

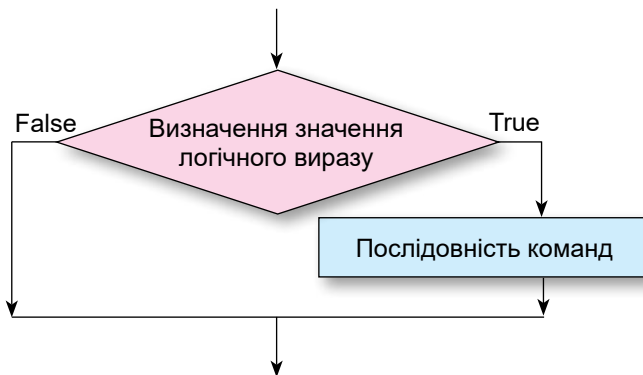
Як умови ви використовували в розгалуженнях висловлювання, математичні рівності та нерівності, запитання, на які потрібно відповісти **Так** чи **Ні**. Усі такі умови є логічними виразами. У розгалу-

женнях можна використовувати й інші логічні вирази: логічні змінні, логічні вирази з логічними операціями тощо.

Тому в загальному випадку кожне розгалуження розпочинається з команди визначення значення логічного виразу. Результатом виконання такої команди можуть бути значення **True** або **False**. І залежно від результату виконання цієї команди, будуть виконуватися ті чи інші команди алгоритму (мал. 4.20, 4.21).



Мал. 4.20. Блок-схема повного розгалуження



Мал. 4.21. Блок-схема неповного розгалуження

Використовуючи поняття логічного виразу, команди розгалуження в мові програмування **Python** виглядатимуть так:

а) команда повного розгалуження:

```
if <логічний вираз>:
 <послідовність команд 1>
else:
 <послідовність команд 2>
```

Виконання цієї команди відбувається так: визначається значення логічного виразу; якщо це значення **True**, то виконується **послідовність команд 1** і після цього виконується команда, наступна за розгалуженням; якщо це значення **False**, то виконується **послідовність команд 2** і після цього виконується команда, наступна за розгалуженням.

б) команда неповного розгалуження:

```
if <логічний вираз>:
 <послідовність команд>
```

Виконання цієї команди відбувається так: визначається значення логічного виразу; якщо це значення **True**, то виконується **послідовність команд** і після цього виконується команда, наступна за розгалуженням; якщо це значення **False**, одразу виконується команда, наступна за розгалуженням.

в) команда багатоальтернативного розгалуження:

```
if <логічний вираз 1>:
 <послідовність команд 1>
elif <логічний вираз 2>:
 <послідовність команд 2>
elif <логічний вираз 3>:
 <послідовність команд 3>
...
else:
 <послідовність команд N>
```

Виконання цієї команди відбувається так:

- визначається значення **логічного виразу 1**;
- якщо це значення **True**, то виконується **послідовність команд 1** і після цього виконується команда, наступна за розгалуженням;
- якщо це значення **False**, то визначається значення **логічного виразу 2**;
- якщо це значення **True**, то виконується **послідовність команд 2** і після цього виконується команда, наступна за розгалуженням;
- якщо це значення **False**, то визначається значення наступного логічного виразу і так далі;
- якщо всі логічні вирази в команді мають значення **False**, то виконується **послідовність команд N**, розміщена після **else**, і після цього виконується команда, наступна за розгалуженням.

*Звертаємо вашу увагу:*

- Команда багатоальтернативного розгалуження може не містити **else**:  
 <послідовність команд N>

У такому разі, якщо всі логічні вирази в команді мають значення **False**, то одразу виконується команда, наступна за розгалуженням.

- Якщо послідовності команд містять тільки по одній команді, то цю команду можна записувати без переходу на новий рядок після двокрапки.

## ПРОЄКТИ З РОЗГАЛУЖЕННЯМИ



### Пригадайте

- У яких випадках використовується команда повного розгалуження?
- У яких випадках використовується команда неповного розгалуження?
- У яких випадках використовується команда багатоальтернативного розгалуження?



Розглянемо кілька задач, для розв'язування яких створюватимемо віконні проекти з розгалуженнями, у яких використовуватимуться різні логічні вирази, у тому числі з логічними операціями.

**Задача 1.** Обчислити значення виразу  $\frac{a+2b}{a-3b}$ , де  $a, b$  – довільні числа.

Створимо математичну модель для цієї задачі:

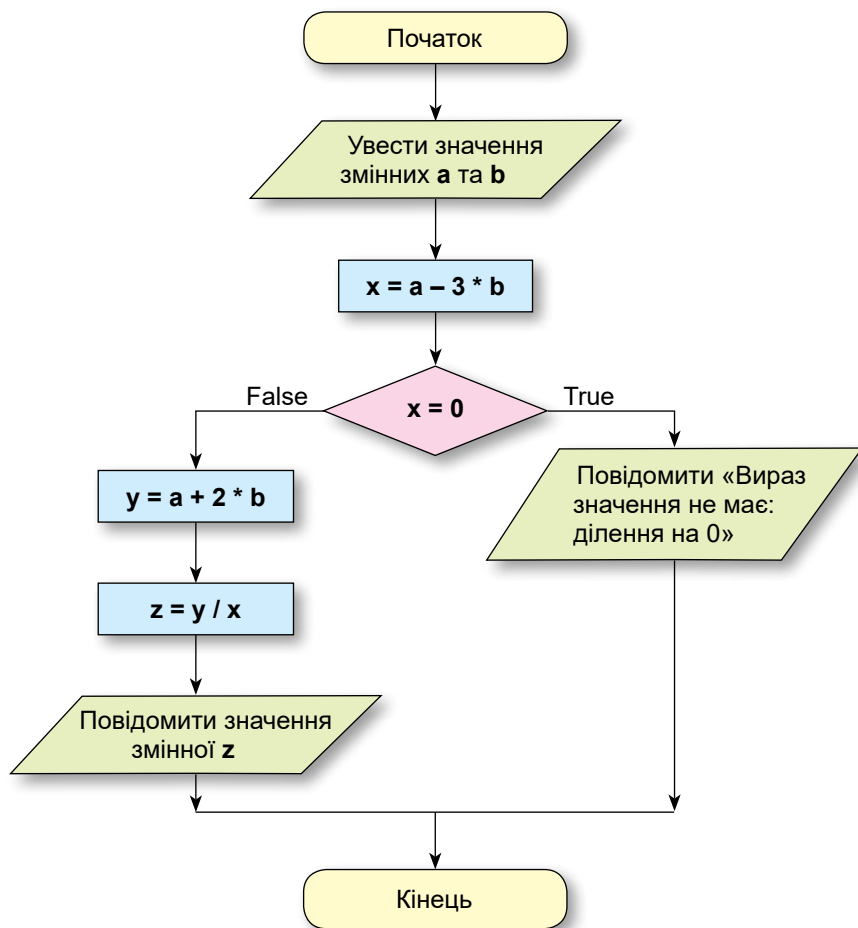
**вхідні дані:** два довільних числа  $a$  і  $b$ ;

**кінцеві результати:** значення виразу  $z$ ;

**формули:**  $z = (a + 2*b) / (a - 3*b)$ .

Оскільки вираз містить дію ділення на вираз, який може дорівнювати нулю (наприклад, при  $a = 3$  і  $b = 1$ ), алгоритм розв'язування цієї задачі повинен містити розгалуження, у якому будуть ураховані два можливих випадки: значення дільника  $a - 3*b$  дорівнює 0 і значення дільника  $a - 3*b$  не дорівнює 0.

Блок-схему алгоритму розв'язування цієї задачі наведено на малюнку 4.22.



Мал. 4.22. Блок-схема алгоритму розв'язування **Задачі 1**



Оскільки математична модель задачі має два вхідних даних і один кінцевий результат, розмістимо у вікні два поля для введення значень вхідних даних, підпишемо їх у відповідних написах, і кнопку, після вибору якої відбуватимуться відповідні обчислення. Результат виводитимемо у вікно повідомлень.

Текст проекту буде такий:

```

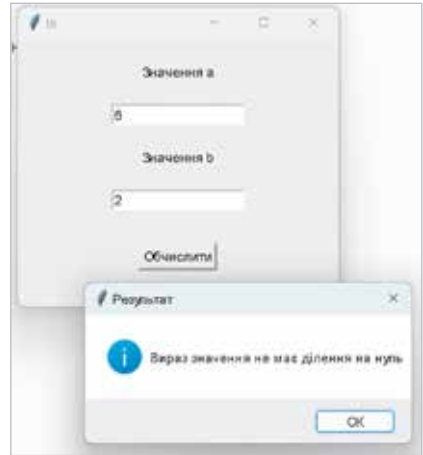
from tkinter import*
from tkinter.messagebox import*
def click():
 a = float(entry1.get())
 b = float(entry2.get())
 x = a - 3 * b
 if x == 0:
 showinfo('Результат', 'Вираз значення не має: ділення на нуль')
 else:
 y = a + 2 * b
 z = y / x
 showinfo('Результат', str(z))
root = Tk()
root.geometry('300x250')
label1 = Label(text = 'Значення a')
label1.pack(pady = 20)
entry1 = Entry()
entry1.pack()
entry1.focus_set()
label2 = Label(text = 'Значення b')
label2.pack(pady = 20)
entry2 = Entry()
entry2.pack()
button = Button(text = 'Обчислити',
command = click)
button.pack(pady = 30)

```

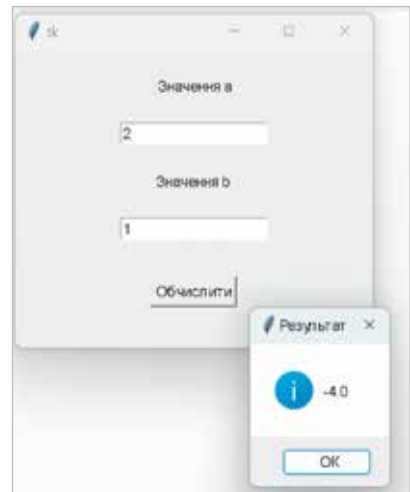
Для перевірки правильності проекту можна виконати його, наприклад для таких наборів вхідних даних:  $a = 6$ ,  $b = 2$ ;  $a = 2$ ,  $b = 1$ . Результати його виконання при цих значеннях вхідних даних наведено на малюнку 4.23.

*Звертаємо вашу увагу.*

Обчислення значення дільника  $a - 3*b$  можна було б не здійснювати в окремій команді та не запам'ятовувати як значення змінної  $x$ , а здійснити безпосередньо в команді розгалуження. Також можна обчислення значення виразу  $(a + 2*b) / (a - 3*b)$  записати не двома командами, а однією. Тоді обробник події був би таким:



а)



б)

Мал. 4.23. Результати виконання проекту для значень вхідних даних  $a = 6$ ,  $b = 2$  (а) і  $a = 2$ ,  $b = 1$  (б)



```
def click():
 a = float(entry1.get())
 b = float(entry2.get())
 if a - 3 * b == 0:
 showinfo('Результат', 'Вираз значення не має: ділення на нуль')
 else:
 z = (a + 2 * b) / (a - 3 * b)
 showinfo('Результат', str(z))
```

Розглянемо задачу, під час розв'язування якої в логічному виразі команди розгалуження використовується диз'юнкція.

**Задача 2.** Обчислити значення функції

$$y = \begin{cases} 2x - 12, & \text{якщо } x < -3 \text{ або } x > 5, \\ 7 - 8x & \text{для всіх інших } x. \end{cases}$$

Побудуємо математичну модель для цієї задачі:

**вхідні дані:** довільне число  $x$ ;

**кінцеві результати:** значення функції  $y$ ;

**формули:**  $y = \begin{cases} 2x - 12, & \text{якщо } x < -3 \text{ або } x > 5, \\ 7 - 8x & \text{для всіх інших } x. \end{cases}$

Оскільки математична модель задачі має одне вхідне дане й один кінцевий результат, розмістимо у вікні два поля: одне для введення значень вхідного даного, друге – для виведення значення кінцевого результату, підпишемо їх у відповідних написах, і кнопку, після вибору якої відбуватимуться відповідні обчислення.

Текст проєкту буде такий:

```
from tkinter import*
def click():
 x = float(entry1.get())
 if x < -3 or x > 5: y = 2 * x - 12
 else: y = 7 - 8 * x
 entry2.delete(0, END)
 entry2.insert(0, str(y))
root = Tk()
root.geometry('300x250')
label1 = Label(text = 'Значення x')
label1.pack(pady = 10)
entry1 = Entry()
entry1.pack()
entry1.focus_set()
label2 = Label(text = 'Значення y')
label2.pack(pady = 10)
entry2 = Entry()
entry2.pack()
button = Button(text = 'Обчислити значення функції', command = click)
button.pack(pady = 20)
```

Для перевірки правильності проєкту можна виконати його, наприклад, для таких значень:  $x = -4$ ;  $x = 0$ ;  $x = 7$ .

*Нагадуємо вам:* тестувати проект з розгалуженням потрібно для таких наборів вхідних даних, щоб перевірити правильність його роботи при всіх можливих варіантах його виконання.

Так, наприклад, проект для обчислення значення виразу  $y = \frac{x - 5}{x^2 - 9}$  містить команду розгалуження, оскільки його виконання залежить від того, дорівнює знаменник нулю чи не дорівнює. Тому його потрібно тестувати для значень  $x$ , при яких знаменник дорівнює нулю, тобто 3 і  $-3$ , і при яких знаменник не дорівнює нулю, наприклад, 5 і 2.



### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Задача 1.** Складіть проект, у якому при кожному переміщенні вказівника по кнопці її ширина збільшується на 5 символів. Якщо після чергового переміщення вказівника ширина кнопки стає більшою за 50 символів, то її розмір має набути початкового значення 15 символів.

1. Відкрийте вікно середовища розробки проекту.
2. Уведіть команди для створення та відкриття вікна, розміщення в ньому кнопки з іменем **button** завширшки 15 символів.
3. Створіть метод – обробник події **<Motion>** для кнопки.
4. Уключіть до цього методу команду, щоб при кожному переміщенні вказівника по кнопці її ширина збільшувалася на 5:

**button['width'] += 5**

5. Уключіть до цього методу команду, щоб у випадку перевищення ширини кнопки значення 50 символів ширина кнопки знову стала 15 символів:

**if button['width'] > 50: button['width'] = 15**

6. Збережіть проект у вашій папці у файлі з іменем **вправа 4.6.1**.
7. Запустіть проект на виконання.
8. Перемістіть вказівник по кнопці і спостерігайте за зміненням її ширини.
9. Повторіть ще кілька разів переміщення вказівника по кнопці, поки її ширина знову не стане 15 символів.
10. Закрийте вікно виконання проекту.
11. Закрийте вікно середовища розробки проектів.

**Задача 2.** Проект для обчислення значень функції  $y = \frac{x^2 - 4}{x^2 - 25}$  містить помилки. Протестувати цей проект, знайти помилки та виправити їх.

1. Відкрийте вікно середовища розробки проектів.
2. Відкрийте проект з іменем **вправа 4.6.2**. Перегляньте його.
3. Запустіть проект на виконання.
4. Уведіть як значення  $x$  число 0. Визначте, який результат виконання проекту ви очікуєте, і порівняйте його з отриманим результатом виконання проекту.

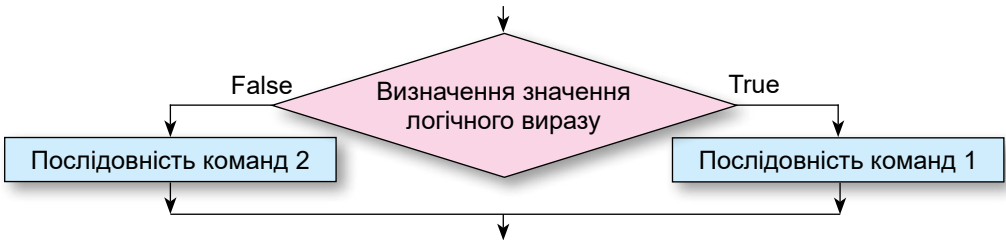


5. Якщо очікуване значення не збігається з результатом виконання, внесіть зміни у проєкт, які виправляють помилки. Після цього знову виконайте проєкт із значенням змінної  $x$ , що дорівнює числу 0.
6. Запустіть проєкт на виконання.
7. Уведіть як значення  $x$  число 5. Визначте, який результат виконання проєкту ви очікуєте, і порівняйте його з отриманим результатом виконання проєкту.
8. Якщо очікуваний результат виконання проєкту не збігається з результатом виконання, внесіть зміни у проєкт, які виправляють помилки. Після цього знову виконайте проєкт зі значенням змінної  $x$ , що дорівнює числу 5.
9. Запустіть ще раз проєкт на виконання.
10. Уведіть як значення  $x$  число  $-5$ . Визначте, який результат виконання проєкту ви очікуєте, і порівняйте його з отриманим результатом виконання проєкту.
11. Якщо очікуваний результат виконання проєкту не збігається з результатом виконання, внесіть зміни у проєкт, що виправляють помилки. Після цього знову виконайте проєкт зі значенням змінної  $x$ , що дорівнює числу 5.
12. Запишіть у зошиті висновок, чому саме для таких значень змінної  $x$  ви тестували проєкт.
13. Протестуйте проєкт ще з двома значеннями  $x$ . Упевніться кожного разу, що результат виконання збігається з очікуваним.
14. Закрийте всі відкриті вікна.

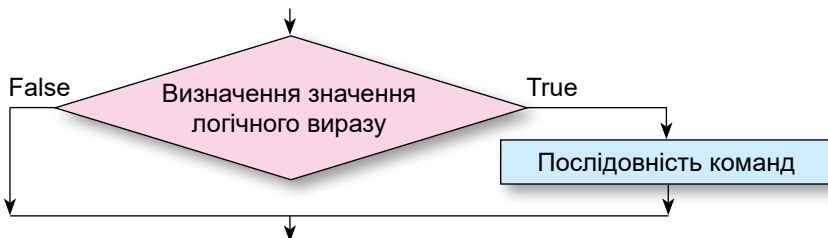


#### Найважливіше в цьому пункті

У загальному випадку кожне розгалуження розпочинається з **команди визначення значення логічного виразу**. Результатом виконання такої команди можуть бути значення **True** або **False**. І залежно від результату виконання цієї команди, будуть виконуватися ті чи інші команди алгоритму (мал. 4.24, 4.25).



Мал. 4.24. Блок-схема повного розгалуження



Мал. 4.25. Блок-схема неповного розгалуження

Використовуючи поняття логічного виразу, команди розгалуження в мові програмування **Python** виглядатимуть так:

а) команда повного розгалуження:

```
if <логічний вираз>:
 <послідовність команд 1>
else:
 <послідовність команд 2>
```

Виконання цієї команди відбувається так: визначається значення логічного виразу; якщо це значення **True**, то виконується **послідовність команд 1** і після цього виконується команда, наступна за розгалуженням; якщо це значення **False**, то виконується **послідовність команд 2** і після цього виконується команда, наступна за розгалуженням.

б) команда неповного розгалуження:

```
if <логічний вираз>:
 <послідовність команд 1>
```

Виконання цієї команди відбувається так: визначається значення логічного виразу; якщо це значення **True**, то виконується **послідовність команд 1** і після цього виконується команда, наступна за розгалуженням; якщо це значення **False**, одразу виконується команда, наступна за розгалуженням.

в) команда багатоальтернативного розгалуження:

```
if <логічний вираз 1>:
 <послідовність команд 1>
elif <логічний вираз 2>:
 <послідовність команд 2>
elif <логічний вираз 3>:
 <послідовність команд 3>
...
else:
 <послідовність команд N>
```

Виконання цієї команди відбувається так:

- визначається значення **логічного виразу 1**;
- якщо це значення **True**, то виконується **послідовність команд 1** і після цього виконується команда, наступна за розгалуженням;
- якщо це значення **False**, то визначається значення **логічного виразу 2**;
- якщо це значення **True**, то виконується **послідовність команд 2** і після цього виконується команда, наступна за розгалуженням;
- якщо це значення **False**, то визначається значення наступного логічного виразу і так далі;
- якщо всі логічні вирази в команді мають значення **False**, то виконується **послідовність команд N**, розміщена після **else**, і після цього виконується команда, наступна за розгалуженням.



### Дайте відповіді на запитання

1. Який вигляд має команда повного розгалуження з використанням логічних виразів? Як вона виконується?
2. Який вигляд має команда неповного розгалуження з використанням логічних виразів? Як вона виконується?
3. Який вигляд має команда багатоальтернативного розгалуження з використанням логічних виразів? Як вона виконується?
4. У яких випадках в алгоритмах потрібно використовувати повне розгалуження, а у яких неповне?



5. Як потрібно підбирати тестові набори значень вхідних даних для тестування проекту з розгалуженням?



### Обговоріть і зробіть висновки

1. Чи можна команду багатоальтернативного розгалуження замінити на інші команди розгалуження? Якщо так, наведіть приклади.



### Виконайте завдання

1. Складіть блок-схему і поясніть виконання фрагмента проекту:

**if** <логічний вираз>:

<команди 1>

**else:**

**if** <логічний вираз 2>:

<команди 2>

**else:**

<команди 3>

2. Складіть блок-схему і поясніть виконання фрагмента проекту:

**if** <логічний вираз>:

<команди 1>

**if** <логічний вираз 2>:

<команди 2>

Поясніть, чим цей фрагмент проекту відрізняється від фрагмента проекту в завданні № 1 і чим відрізняються їх виконання.

3.  $f$  – змінна логічного типу. Чи можна розгалуження **if**  $f = \text{True}$ : <Команда> замінити розгалуженням **if**  $f$ : <Команда>. Відповідь поясніть.

4. Створіть математичну модель і проект для обчислення значення виразу:  $(a + b) - c : (a - b)$ . Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем завдання 4.6.4. Запишіть у зошит набори вхідних даних, для яких ви тестували цей проект.

5. Створіть математичну модель і проект для обчислення значення виразу:  $a + (b - c) : (a + 2b)$ . Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем завдання 4.6.5. Запишіть у зошит набори вхідних даних, для яких ви тестували цей проект.

6. Створіть математичну модель і проект, який визначає менше з двох довільних чисел або виводить повідомлення, що числа рівні. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем завдання 4.6.6. Запишіть у зошит набори значень вхідних даних, для яких ви тестували цей проект.

7. Створіть математичну модель і проект, який визначає найменше з чотирьох довільних чисел. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем завдання 4.6.7. Запишіть у зошит набори значень вхідних даних, для яких ви тестували цей проект.

8. Створіть математичну модель і проект, який обчислює модуль довільного числа. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем завдання 4.6.8. Запишіть у зошит значення вхідних даних, для яких ви тестували цей проект.

9. Створіть проект, у якому у вікні розміщується кнопка завширшки 40 символів. Після кожного вибору кнопки її ширина зменшується на 5 символів. А коли ширина кнопки стає менше, ніж 10 символів, її ширина знову стає 40 символів. Збережіть проект у вашій папці у файлі з іменем **завдання 4.6.9**.

10. Створіть математичну модель і проект для обчислення значення функції  $y = \begin{cases} 15 - 3x, & \text{якщо } -1 < x < 4, \\ 6 + 4x & \text{для всіх інших } x. \end{cases}$  Математичну модель за-

пишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.6.10**. Запишіть у зошит набори значень вхідних даних, для яких ви тестували цей проект.

11. Створіть математичну модель і проект для обчислення значення функції  $y = \begin{cases} 2x - 4, & \text{якщо } x < 2, \\ x^2 + 5, & \text{якщо } 2 \leq x \leq 7, \\ 3 - 5x, & \text{якщо } x > 7. \end{cases}$  Математичну модель запишіть

у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.6.11**. Запишіть у зошит набори значень вхідних даних, для яких ви тестували цей проект.

12. Створіть математичну модель і проект для обчислення значення функції  $y = \begin{cases} 2x - 4, & \text{якщо } x < 2, \\ x^2 + 5, & \text{якщо } 5 \leq x \leq 12, \\ 3 - 5x & \text{для всіх інших } x. \end{cases}$  Математичну модель за-

пишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.6.12**. Запишіть у зошит набори значень вхідних даних, для яких ви тестували цей проект.

13. Створіть математичну модель і проект для визначення, чи можна з трьох відрізків із заданими довжинами утворити трикутник. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.6.13**. Запишіть у зошит набори значень вхідних даних, для яких ви тестували цей проект.

14. Банк пропонує вкладнику два види депозитів: 15 % прибутку за рік і 7 % прибутку за півроку. В обох депозитах прибуток за вказаний час додається до вкладу. Створіть математичну модель і проект для порівняння прибутку за 1 рік по цих депозитах, якщо на обидва покласти одну й ту саму суму грошей. Математичну модель запишіть у зошит. Збережіть проект у вашій папці у файлі з іменем **завдання 4.6.14**. Запишіть у зошит набори значень вхідних даних, для яких ви тестували цей проект.



### Готуємось до вивчення нового матеріалу

1. Які елементи керування у віконних проектах ви знаєте?
2. Для чого використовувався кожний з них?
3. Як використовувалися події з цими елементами керування?

## 4.7. ПРАПОРЦІ ТА ПЕРЕМИКАЧІ У ВІКОННИХ ПРОЄКТАХ

У цьому пункті йтиметься про:

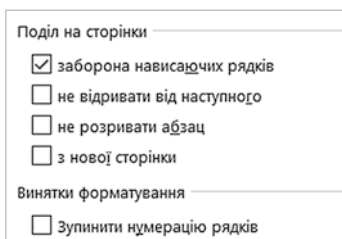
- ▶ прапорці як елементи керування у віконних проєктах;
- ▶ перемикачі як елементи керування у віконних проєктах.

## ПРАПОРЦІ ТА ЇХ ВЛАСТИВОСТІ



## Пригадайте

• Який загальний вигляд команди неповного розгалуження в мові програмування **Python** і як вона виконується? • Для чого використовували у програмах, з якими ви працювали, прапорці як елементи керування? Наведіть приклади.



Мал. 4.26. Приклад використання прапорців

У різних програмах прапорці використовують для вибору одного або кількох варіантів з набору можливих, або не вибору жодного (мал. 4.26). Якщо прапорець вибраний, то в нього всередині з'являється позначка .

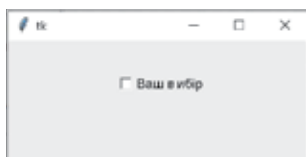
Команда для створення прапорця у вікні проєкту мовою **Python** така:

```
<ім'я прапорця> = Checkbutton(text =
'<текст підпису>'), (англ. checkboxbutton –
прапорець).
```

Після розміщення у вікні прапорця, створеного таким способом, він матиме визначений користувачем підпис. Наприклад, після виконання команд:

```
flag = Checkbutton(text = 'Ваш вибір')
flag.pack()
```

буде створено та розміщено у вікні прапорець з іменем **flag** і підписом *Ваш вибір* (мал. 4.27).



Мал. 4.27. Прапорець у вікні проєкту

Властивості прапорця **width**, **height**, **bg**, **fg**, **font** аналогічні відповідним властивостям напису, поля та кнопки.

Щоб мати можливість дізнаватися під час виконання проєкту стан прапорця (вибраний або невибраний), потрібно пов'язати з ним спеціальну змінну логічного типу. Для створення такої змінної використовують команду виду:

```
<ім'я змінної> = BooleanVar()
```

Пов'язують цю змінну з прапорцем під час його створення командою, що має вигляд:

```
<ім'я прапорця> = Checkbutton(text = '<текст підпису>',
variable = <ім'я змінної>), (англ. variable – змінна).
```

Наприклад, після виконання команд:

```
choice = BooleanVar()
flag = Checkbutton(text = 'Ваш вибір', variable = choice)
flag.pack()
```



буде створено та розміщено у вікні прапорець з іменем **flag** і підписом *Ваш вибір*, а змінна з іменем **choice** зберігатиме стан цього прапорця: **True** – якщо прапорець вибраний, **False** – якщо прапорець не вибраний.

Значення цієї змінної за замовчуванням – **False**, і після запуску проєкту на виконання позначка прапорця буде відсутня. Якщо під час виконання проєкту буде вибрано прапорець, то значення змінної стане **True**. За повторного вибору прапорця його стан зміниться на протилежний, і також зміниться значення змінної.

Щоб під час запуску проєкту одразу прапорець був у стані *вибраний*, потрібно використати команду виду `<ім'я змінної>.set(True)` (англ. *set* – установити), наприклад `choice.set(True)`.

Щоб визначити значення цієї змінної під час виконання проєкту потрібно виконати команду виду `<ім'я змінної>.get()`, наприклад `choice.get()`. Значення цієї змінної можна використовувати у проєктах у команді розгалуження як значення логічного виразу.

Якщо у проєкті використовується кілька прапорців, то для кожного з них повинна бути створена окрема змінна, наприклад:

```
choice1 = BooleanVar()
flag1 = Checkbutton(text = 'Заголовок', variable = choice1)
flag1.pack()
choice2 = BooleanVar()
flag2 = Checkbutton(text = 'Колір', variable = choice2)
flag2.pack()
choice3 = BooleanVar()
flag3 = Checkbutton(text = 'Розміри', variable = choice3)
flag3.pack()
```

Розглянемо, як можна використати прапорці у проєктах з розгалуженнями.

Нехай, наприклад, нам потрібно в ході виконання проєкту змінювати значення або тільки однієї з властивостей вікна (або заголовок, або колір, або розміри), або кількох з них, або не змінювати значення жодної з них.

Для цього використаємо кнопку та три прапорці (мал. 4.28). Оскільки при виконанні проєкту жодний прапорець може бути не вибраний, то доцільно під час запуску проєкту на виконання для кожного з них залишити значення логічної змінної **False**, яке встановлюється за замовчуванням.

Створимо процедуру для опрацювання події **Click** для кнопки і до її тексту помістимо такі три послідовні (не вкладені одна в одну) команди розгалуження:

```
if choice1.get(): root.title('Заголовок змінений')
if choice2.get(): root['bg'] = 'red'
if choice3.get(): root.geometry('500x200')
```

Якщо при настанні події **Click** для кнопки перший прапорець буде вибраним, то значення змінної **choice1** буде **True** і відповідно викону-



Мал. 4.28. Вікно проєкту з кількома прапорцями



ватиметься команда `root.title('Заголовок змінений')`. Якщо ж перший прапорець не буде вибраним, то значення змінної `choice1` буде `False` і відповідно команда `root.title('Заголовок змінений')` не виконуватиметься, а виконуватиметься наступна команда, тобто наступна команда розгалуження для другого прапорця.

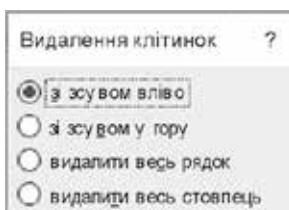
Аналогічно виконуватимуться і дві інші команди розгалуження.

## ПЕРЕМИКАЧІ ТА ЇХ ВЛАСТИВОСТІ



### Пригадайте

• Який загальний вигляд команди повного розгалуження в мові програмування **Python** і як вона виконується? • Який загальний вигляд команди багатоальтернативного розгалуження в мові програмування **Python** і як вона виконується? • Для чого використовувалися у програмах, з якими ви працювали, перемикачі? Наведіть приклади.



Мал. 4.29. Вікно з перемикачами

Перемикачі як елементи керування використовують для вибору тільки одного варіанта з кількох можливих (мал. 4.29).

Якщо у вікні у групі перемикачів розміщено кілька перемикачів, то серед них можна вибрати тільки один. Якщо перемикач вибраний, то в нього всередині з'являється позначка .

Оскільки, на відміну від прапорців, з однієї групи перемикачів може бути вибраний тільки один, то важливо, який саме перемикач вибраний. Тому всі перемикачі з однієї групи повинні бути пов'язані з однією змінною цілого типу. Команда створення цієї змінної така:

`<ім'я змінної> = IntVar()`

Команда створення кожного з перемикачів групи має такий вигляд:

`<ім'я перемикача> = Radiobutton(text = '<тест підпису>', variable = <ім'я змінної>, value = <значення змінної>)`, де:

- властивість `text` визначає текст підпису перемикача;
- властивість `variable` визначає ім'я змінної, яка пов'язана з перемикачем. Повинна бути одна й та сама змінна для всіх перемикачів однієї групи;
- властивість `value` (англ. *value* – значення) визначає значення, якого набуває змінна, якщо буде вибраний саме цей перемикач. Значення повинні бути різними для кожного з перемикачів однієї групи.

На малюнку 4.30 наведено фрагмент проекту, у якому створено три перемикачі однієї групи з іменами `switch1`, `switch2` та `switch3`, для яких змінна `choice` набуває відповідно значень `1`, `2` та `3`.

```
choice = IntVar()
switch1 = Radiobutton(text = "Заруконок", variable = choice, value = 1)
switch1.pack(pady = 10)
switch2 = Radiobutton(text = "Копія", variable = choice, value = 2)
switch2.pack(pady = 10)
switch3 = Radiobutton(text = "Поміпа", variable = choice, value = 3)
switch3.pack(pady = 10)
```

Мал. 4.30. Фрагмент проекту зі створення трьох перемикачів однієї групи

Властивості перемикача **width**, **height**, **bg**, **fg**, **font** аналогічні до відповідних властивостей прапорця та інших елементів керування.

За замовчуванням змінна **choice** має значення **0**. Оскільки це значення не відповідає жодному перемикачу, то після запуску проєкту на виконання жоден перемикач не буде вибраним. Щоб один з перемикачів одразу був вибраним, то потрібно встановити для змінної відповідне значення командою виду:

```
<ім'я змінної>.set(<значення змінної>).
```

Наприклад, після виконання команди **choice.set(1)** стане вибраним перший перемикач.

Щоб дізнатися в ході виконання проєкту, який з перемикачів вибрано, потрібно виконати команду виду **<ім'я змінної>.get()**, наприклад **choice.get()**.

Якщо потрібно розмістити у вікні кілька незалежних груп перемикачів, то кожену групу перемикачів потрібно пов'язати з окремою змінною.

Розглянемо, як можна, використовуючи перемикачі, змінювати значення однієї з властивостей вікна: або заголовок, або колір, або розміри.

Використаємо у проєкті три перемикачі та кнопку (мал. 4.31).

Створимо процедуру – обробник події **Click** для кнопки і до її тексту помістимо таку команду:

```
if choice.get() == 1:
 root.title('Заголовок змінений')
elif choice.get() == 2:
 root['bg'] = 'red'
else:
 root.geometry('600x300')
```



Мал. 4.31. Вікно проєкту з перемикачами

Розпочинається виконання цієї команди буде з визначення значення змінної **choice**.

Якщо при настанні події **Click** для кнопки вибраним буде перший перемикач, то значення логічного виразу **choice.get() == 1** дорівнюватиме **True**. Тому виконається команда змінення заголовка вікна, а інші частини цієї команди розгалуження виконуватися не будуть.

Якщо при настанні події **Click** для кнопки вибраним буде другий перемикач, то значення логічного виразу **choice.get() == 1** дорівнюватиме **False**. Тому буде виконуватися команда після ключового слова **elif**. За її виконання визначається значення змінної **choice**. Оскільки буде вибраний другий перемикач, то це значення дорівнює **2**. Тому виконається команда змінення кольору вікна, а інші частини команди розгалуження виконуватися не будуть.

Якщо ж при настанні події **Click** для кнопки вибраним буде третій перемикач, то значення логічних виразів **choice.get() == 1** і **choice.get() == 2** дорівнюватимуть **False**. Тому не будуть виконуватися команди змінення заголовка вікна та кольору вікна, а виконуватиметься команда після ключового слова **else**, тобто команда змінення розмірів вікна.



Якщо під час виконання проєкту сталося так, що не буде вибраний жодний з перемикачів, то після вибору кнопки зміняться розміри вікна (*подумайте самостійно, чому відбудуться саме такі змінення*). Щоб так не сталося, потрібно замість ключового слова `else` використати ще одну команду `elif`:

```
if choice.get() == 1:
 root.title('Заголовок змінений')
elif choice.get() == 2:
 root['bg'] = 'red'
elif choice.get() == 3:
 root.geometry('600x300')
```



### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Задача.** У класі створюється рейтинг учнів і учениць за підсумками семестру. Якщо середній бал семестрових оцінок учня/учениці більше за 10, то він/вона отримує 10 рейтингових балів; якщо більше за 7 і не більше від 10, то 7 балів; якщо більше за 4 і не більше від 7, то 4 бали; якщо не більше від 4, то 1 бал. Крім того, учень/учениця отримує по 2 додаткових бали, якщо він/вона відвідує шкільні предметні гуртки, шкільні спортивні секції, шкільну театральну студію. Створіть проєкт для визначення рейтингового балу учня/учениці.

Створимо математичну модель для цієї задачі:

**вхідні дані:** середній бал учня/учениці, відомості, чи відвідує він/вона шкільні предметні гуртки, шкільні спортивні секції, шкільну театральну студію;

**кінцеві результати:** рейтинговий бал учня/учениці;

**формули:**

```
якщо середній бал більше за 10, рейтинг = 10
 інакше якщо середній бал більше за 7, рейтинг = 7
 інакше якщо середній бал більше за 4, рейтинг = 4
 інакше рейтинг = 1;
```

```
якщо відвідує шкільні предметні гуртки, збільшити рейтинг на 2;
якщо відвідує шкільні спортивні секції, збільшити рейтинг на 2;
якщо відвідує шкільну театральну студію, збільшити рейтинг на 2.
```

1. Відкрийте вікно середовища створення проєктів і вікно нового проєкту.
2. Уведіть команди створення форми розміром 400 на 450 пікселів.
3. Уведіть команди створення та розміщення у вікні проєкту 4-х перемикачів, які визначатимуть, який середній бал має учень/учениця. Для цього:
  1. Створіть змінну, яка буде пов'язана з усіма перемикачами:
 

```
choice = IntVar()
```
  2. Уведіть команди створення та розміщення перемикача з іменем `switch1`. Значення змінної `choice` для нього 1:
 

```
switch1 = Radiobutton(text = 'Більше 10', variable = choice, value = 1)
switch1.pack(pady = 10)
```

3. Уведіть команди створення та розміщення ще трьох перемикачів з іменами **switch2**, **switch3**, **switch4**. Значення змінної **choice** для них 2, 3 та 4 відповідно, значення властивості **text** як наведено на малюнку 1.
4. Уведіть команди створення та розміщення у вікні трьох прапорців, які визначатимуть, чи відвідує учень/учениця шкільні предметні гуртки, шкільні спортивні секції, шкільну театральну студію. Для цього:
  1. Створіть змінну **choice1**, яка буде пов'язана з першим прапорцем: **choice1 = BooleanVar()**
  2. Уведіть команди створення та розміщення першого прапорця з іменем **flag1**, пов'язаного зі змінною **choice1**:  
**flag1 = Checkbutton(text = 'Предметні гуртки', variable = choice1)**  
**flag1.pack(pady = 10)**
  3. Уведіть команди створення та розміщення ще двох прапорців з іменами **flag2** та **flag3**. Імена змінних для них **choice2** та **choice3** відповідно, значення властивості **text** як наведено на малюнку 1.
5. Уведіть команди створення та розміщення у вікні напису іменем **lb1** для виведення пояснювального тексту, напису іменем **lb2** для виведення сумарного рейтингу і кнопки. Установіть значення їх властивостей, як це наведено на малюнку 1.
6. Збережіть проєкт у файлі з іменем **вправа 4.7**.
7. Запустіть проєкт на виконання, перевірте, як працюють компоненти перемикачі та прапорці.
8. Закрийте вікно виконання проєкту.
9. Створіть процедуру – обробник події **Click** для кнопки та помістіть до неї такі команди:
 

```
s = 0
if choice.get() == 1: s = 10
elif choice.get() == 2: s = 7
elif choice.get() == 3: s = 4
elif choice.get() == 4: s = 1
if choice1.get(): s += 5
if choice2.get(): s += 5
if choice3.get(): s += 5
lb2['text'] = str(s)
```
10. Збережіть оновлену версію проєкту.
11. Запустіть проєкт на виконання.
12. Виберіть перший перемикач, перший і третій прапорець.
13. Виберіть кнопку.
14. Проаналізуйте правильність отриманого результату.
15. Виберіть інший перемикач та інші прапорці.
16. Проаналізуйте правильність отриманого результату.
17. Закрийте вікно виконання проєкту.
18. Закрийте вікно середовища створення проєктів.



Мал. 1. Вікно з прапорцями і перемикачами у проєкті для розв'язування **Задачі**





### Найважливіше в цьому пункті

**Прапорці** використовують для вибору одного або кількох варіантів з набору можливих чи не вибору жодного. З кількох прапорців може бути вибраний або один, або кілька, або всі, або жодний.

Команда для створення прапорця у мові **Python** така:

**<ім'я прапорця> = Checkbutton(text = '<текст підпису>', variable = <ім'я змінної>)**, де змінна логічного типу використовується для визначення стану прапорця та створюється командою **<ім'я змінної> = BooleanVar()**.

Щоб під час запуску проєкту відразу прапорець був у стані *вибраний*, потрібно використати команду виду **<ім'я змінної>.set(True)**. Щоб дізнатися значення цієї змінної під час виконання проєкту, потрібно виконати команду виду **<ім'я змінної>.get()**. Значення цієї змінної можна використовувати у проєктах у команді розгалуження як логічний вираз.

Для опрацювання стану прапорців (вибраний або не вибраний) у проєкті використовують для кожного з них послідовно записані команди розгалуження, які незалежні одна від іншої.

**Перемикачі** використовують для вибору тільки одного з кількох можливих варіантів. З кількох перемикачів однієї групи може бути вибраний тільки один.

Команда для створення перемикача в мові **Python** має вигляд:

**<ім'я перемикача> = Radiobutton(text = '<текст підпису>', variable = <ім'я змінної>, value = <значення змінної>)**, де змінна використовується для визначення вибраного перемикача та створюється командою **<ім'я змінної> = IntVar()**.

Якщо потрібно, щоб один з перемикачів групи відразу був вибраним, то слід установити для змінної відповідне значення командою виду **<ім'я змінної>.set(<значення змінної>)**.

Щоб дізнатися в ході виконання проєкту, який з перемикачів однієї групи вибрано, потрібно виконати команду виду **<ім'я змінної>.get()**.

У вікні можна також розмістити кілька незалежних одна від іншої груп перемикачів. Для цього потрібно кожну групу перемикачів пов'язати з окремою змінною. Тоді в кожній групі можна вибрати по одному перемикачу.

Для опрацювання стану прапорців (вибраний або не вибраний) у проєкті використовують команду багатоальтернативного розгалуження.



### Дайте відповіді на запитання

1. Для чого використовують прапорці, а для чого – перемикачі?
2. Як визначити стан прапорця у проєкті мовою **Python**?
3. Як потрібно записувати команди розгалуження, у яких перевіряється, вибраний прапорець чи ні?
4. Як визначити, який перемикач вибрано у проєкті мовою **Python**?
5. Як доцільно записувати команди розгалуження, у яких перевіряється, вибраний перемикач чи ні?



### Обговоріть і зробіть висновки

1. Чи можна для прапорців використати команду багатоальтернативного розгалуження?
2. Чи можна для перемикачів замість команди багатоальтернативного розгалуження використати команди розгалуження, вкладені одна в одну?
3. Чи можна для перемикачів замість команди багатоальтернативного розгалуження використати команди розгалуження, записані послідовно, одна за одною?



### Виконайте завдання

1. Створіть блок-схему та поясніть хід виконання послідовності команд:

```
if choice 1.get():
 <команди 1>
if choice 2.get():
 <команди 2>
```

де **choice1** і **choice2** – змінні логічного типу, пов'язані з прапорцями.

2. Створіть блок-схему та поясніть хід виконання команди:

```
if choice.get() == 1:
 <команди 1>
elif choice.get() == 2:
 <команди 2>
else:
 <команди 3>
```

де **choice** – змінна цілого типу, пов'язана з групою перемикачів.

3. Поясніть відмінності у виконанні команд розгалуження із завдання № 1 та із завдання № 2.

4. Сформулюйте життєву задачу, у проєкті для розв'язування якої можна використати фрагмент із завдання № 1.

5. Сформулюйте життєву задачу, у проєкті для розв'язування якої можна використати фрагмент із завдання № 2.

6. Створіть математичну модель і проєкт, у якому можна обчислити суму, різницю, добуток, частку від ділення двох довільних чисел залежно від установлених позначок прапорців. Математичну модель запишіть у зошит. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.7.6**.

7. Створіть математичну модель і проєкт, у якому можна обчислити або суму, або різницю, або добуток, або частку від ділення двох довільних чисел залежно від вибраного перемикача. Математичну модель запишіть у зошит. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.7.7**.

8. Робітник виготовляє деталі. Відомо кількість деталей, виготовлених робітником за місяць, і вартість однієї деталі. Якщо робітник виконав план, то він одержує премію в розмірі 10 % від заробленої в результаті виготовлення деталей суми. Якщо робітник перевиконав план, то його премія становить 30 % від заробленої суми. Створіть математичну модель і проєкт з використанням перемикачів, який визначає суму грошей, зароблену робітником за місяць. Математичну модель запишіть у зошит. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.7.8**.

9. Робітник має фіксовану заробітну плату за роботу протягом місяця. Якщо в робітника є діти, він отримує доплату в розмірі 20 % від його місячної заробітної плати. Якщо в робітника протягом місяця були порушення трудової дисципліни, то його заробітна плата зменшується на 15 %. Створіть математичну модель і проєкт з використанням прапорців, який визначає суму грошей, зароблену робітником



за місяць. Математичну модель запишіть у зошит. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.7.9**.

**10.** Створіть проєкт з використанням двох груп перемикачів і кнопки. Перемикачі першої групи визначають колір, у який зафарбовується вікно, а перемикачі другої групи визначають, якими мають стати розміри вікна. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.7.10**.



### Практична робота № 5.

#### «Проєкти з розгалуженнями»

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

- Створіть математичну модель і проєкт для обчислення значення виразу  $x = (a + 3*b)/(2*a - b)$ . Збережіть проєкт у вашій папці у файлі з іменем **практична 5\_1**.
- Створіть проєкт для знаходження найменшого з даних чотирьох чисел. Збережіть проєкт у вашій папці у файлі з іменем **практична 5\_2**.
- Вкладник хоче покласти в банк на депозит певну суму грошей на 2 роки. Банк пропонує два види депозитів: під 15 % річних і під 17 % річних. У першому депозиті прибуток додається до вкладу кожного року, у другому – ні. Створіть математичну модель і проєкт з використанням перемикачів для обчислення суми грошей, яку зможе отримати вкладник через два роки, якщо він вибере один з цих депозитів. Збережіть проєкт у вашій папці у файлі з іменем **практична 5\_3**.



### Готуємось до вивчення нового матеріалу

- Які процеси називаються циклічними? Наведіть приклади.
- Які ви знаєте команди циклу?

## 4.8. ЦИКЛИ З ЛІЧИЛЬНИКОМ

У цьому пункті йтиметься про:

- ▶ цикли;
- ▶ команду циклу з лічильником;
- ▶ використання команди циклу з лічильником у проєктах.

### КОМАНДА ЦИКЛУ З ЛІЧИЛЬНИКОМ



#### Пригадайте

● Які процеси називають циклічними? Наведіть приклади. ● Що таке *цикл в алгоритмі*? ● Який вигляд має команда циклу з лічильником у мові програмування, яку ви вивчали раніше? Як вона виконується?

Ви вже знаєте, що в багатьох алгоритмах одна або кілька команд повинні повторюватися багато разів поспіль. Недоцільно включати в алгоритм ці команди стільки разів, скільки вони повинні повторюватися, тому що кількість разів повторення може бути дуже велика, або навіть заздалегідь невідома.



У таких випадках потрібно в алгоритмах використовувати цикли. Нагадаємо, що **цикл у алгоритмі** – це фрагмент алгоритму, який може повторюватися поспіль більше ніж один раз.

У мовах програмування є кілька команд, які можуть реалізувати цикл. Одна з них – **команда циклу з лічильником**. Її доцільно використовувати в тих випадках, коли кількість повторень команд тіла циклу відома ще до початку виконання циклу. Наприклад, якщо відомо, скільки років гроші будуть зберігатися на рахунок в банку.

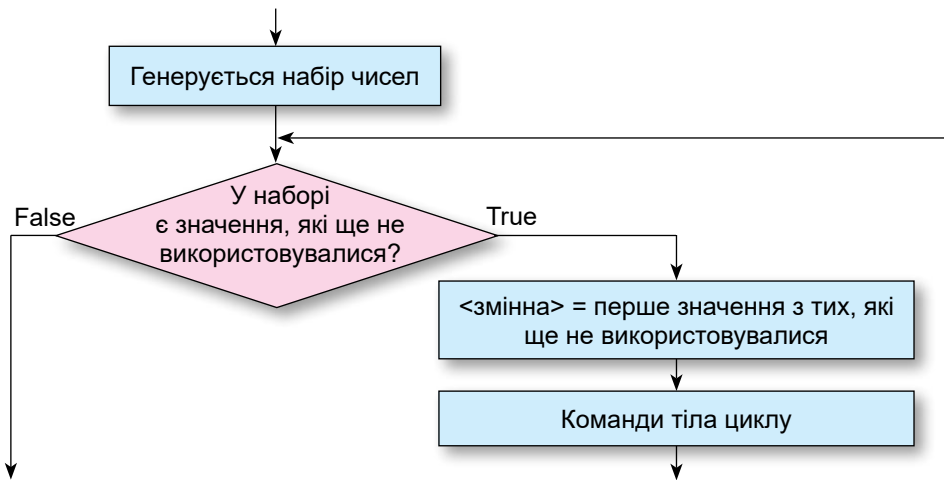
Загальний вигляд команди циклу з лічильником у мові **Python** такий:

```
for <ім'я змінної> in range(<кількість повторень>):
 <команди тіла циклу>
```

Рядок `for <ім'я змінної> in range(<кількість повторень>)` (англ. *for* – для, *in range* – у діапазоні) називається **рядком заголовка команди циклу з лічильником**. Змінна в рядку заголовка команди циклу з лічильником називається **лічильником циклу**.

Функція `range(<кількість повторень>)` утворює (генерує) набір послідовних цілих чисел, перше з яких 0, а останнє – на 1 менше від заданої кількості повторень (усього чисел стільки, як задано кількість повторень). Наприклад, функція `range(3)` генерує набір з 3-х чисел: 0, 1, 2. Змінна – лічильник циклу послідовно набуває значень з цього набору чисел. Тіло циклу виконується, поки лічильник отримує нові значення. Коли нового значення в наборі немає, виконання циклу припиняється.

Блок-схему виконання команди циклу з лічильником в **Python** наведено на малюнку 4.32.



Мал. 4.32. Блок-схема виконання команди циклу з лічильником

Виконується команда циклу з лічильником так:

1. Функція `range(<кількість повторень>)` генерує набір цілих чисел від 0 до **кількість повторень – 1**.
2. Виконується перевірка, чи є в наборі числа, які ще не використовувалися і могли б стати новим значенням змінної – **лічильника циклу**.



3. Якщо результат перевірки **True**, то змінна – лічильник циклу набуває значення першого числа з набору чисел, які ще не використовувалися, і після цього виконується команда 4. Якщо результат перевірки **False**, то виконання команди циклу закінчується і далі виконується команда, наступна за командою циклу.
4. Виконуються команди тіла циклу.
5. Повторюється виконання дії команди 2.

У записі команди циклу з лічильником усі команди тіла циклу повинні бути записані з однаковим відступом праворуч від першої літери слова **for**.



Мал. 4.33. Пірамідка

Розглянемо як приклад проект для розв'язування задачі, у якому використовується цикл з лічильником.

**Задача.** Створити проект для обчислення кількості кульок у пірамідці (мал. 4.33)

Створимо математичну модель для цієї задачі, враховуючи, що в першому зверху шарі пірамідки 1 кулька, у другому – 2 ряди по 2 кульки, у третьому – 3 ряди по 3 кульки і так далі. Врахуємо також, що кількість шарів у пірамідці може бути довільним натуральним числом.

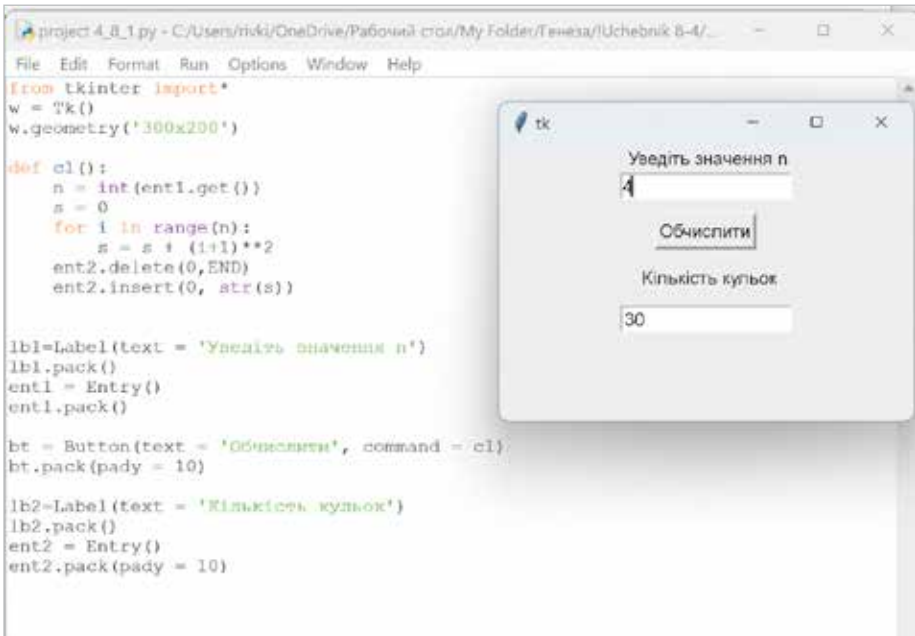
Математична модель:

**вхідні дані:** кількість шарів у пірамідці ( $n$ );

**кінцеві результати:** кількість кульок у пірамідці ( $s$ )

**формули:**  $s = 1^2 + 2^2 + 3^2 + 4^2 + \dots + n^2$ .

Проект для розв'язування цієї задачі та вікно його виконання наведено на малюнку 4.34.



Мал. 4.34. Проект розв'язування **Задачі** та вікно його виконання

Виконаємо цей проєкт для  $n = 4$ .

| Команда                                                                      | Результат виконання                          |                                              |                                               |                                                |              |
|------------------------------------------------------------------------------|----------------------------------------------|----------------------------------------------|-----------------------------------------------|------------------------------------------------|--------------|
|                                                                              | Крок 1                                       | Крок 2                                       | Крок 3                                        | Крок 4                                         | Крок 5       |
| <code>n = int(ent1.get())</code>                                             | <code>n = 4</code>                           |                                              |                                               |                                                |              |
| <code>s = 0</code>                                                           | <code>s = 0</code>                           |                                              |                                               |                                                |              |
| <code>range(4)</code>                                                        | <code>0, 1, 2, 3</code>                      |                                              |                                               |                                                |              |
| У наборі ще є невикористані числа?                                           | <b>True</b>                                  | <b>True</b>                                  | <b>True</b>                                   | <b>True</b>                                    | <b>False</b> |
| Змінна <code>i</code> набуває першого значення з набору невикористаних чисел | <code>i = 0</code>                           | <code>i = 1</code>                           | <code>i = 2</code>                            | <code>i = 3</code>                             |              |
| <code>s = s + (i + 1) * (i + 1)</code>                                       | <code>s = 0 + (0 + 1)<sup>2</sup> = 1</code> | <code>s = 1 + (1 + 1)<sup>2</sup> = 5</code> | <code>s = 5 + (2 + 1)<sup>2</sup> = 14</code> | <code>s = 14 + (3 + 1)<sup>2</sup> = 30</code> |              |
| <code>ent2.insert(0, str(s))</code>                                          |                                              |                                              |                                               |                                                | 30           |

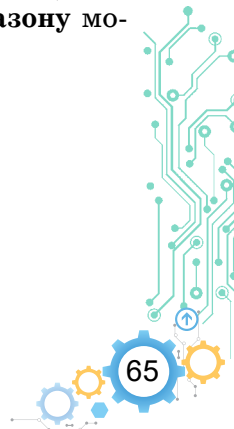
*Звертаємо вашу увагу:* після завершення циклу змінна – лічильник циклу `i` матиме значення 3. Це останнє значення з набору чисел, згенерованого функцією `range(4)`.

Окрім функції `range(<кількість повторень>)`, у команді циклу з лічильником можна використовувати функцію `range(<початок діапазону>, <кінець діапазону>, <крок>)`, яка має 3 аргументи цілого типу:

- **початок діапазону** – перше ціле число, яке потрапить до згенерованого набору;
- **кінець діапазону** – ціле число, яке не увійде до згенерованого набору і яке більше за всі числа з набору, якщо **крок** > 0, і менше всіх чисел з набору, якщо **крок** < 0;
- **крок** – різниця між кожними двома сусідніми цілими числами у згенерованому наборі чисел.

Наприклад, функція `range(3, 8, 2)` згенерує такий набір чисел: 3,  $3 + 2 = 5$ ,  $5 + 2 = 7$ . Наступне число  $7 + 2 = 9$  не увійде в набір чисел, оскільки не менше від другого аргументу функції.

Кожен з аргументів функції `range` може бути як додатним, так і від'ємним цілим числом. **Початок діапазону** та **кінець діапазону** можуть також набувати значення 0.



## ТЕСТУВАННЯ ПРОЄКТУ, ЩО МІСТИТЬ ЦИКЛ З ЛІЧИЛЬНИКОМ



## Пригадайте

- Для чого тестують проєкти? • У чому полягає процес тестування проєкту?
- Як потрібно тестувати проєкт з розгалуженнями?

Під час тестування проєкту, що містить цикл з лічильником, доцільно добирати такі набори вхідних даних, щоб тіло циклу виконувалося 1 раз; 2 рази; 3 рази; 5 разів та за потреби при інших.



## Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Задача.** Велопробіг складається з  $n$  етапів. Довжина першого етапу  $x$  км, а довжина кожного наступного етапу на **30** км більша за довжину попереднього. Створіть проєкт для знаходження довжини велопробігу.

Створимо математичну модель для цієї задачі:

**вхідні дані:** кількість етапів ( $n$ ), довжина першого етапу ( $x$ ), число, на яке довжина кожного наступного етапу більша за довжину попереднього (**30**);

**кінцеві результати:** довжина велопробігу ( $s$ );

**формули:**  $x = x + 30$  – для обчислення довжини наступного етапу;  
 $s = s + x$  – для обчислення наступного значення пройденого шляху.

Отже, для розв'язування цієї задачі потрібно ввести кількість етапів і довжину першого етапу. Очевидно, ці дані вводитиме користувач у поля під час виконання проєкту. У ході виконання проєкту в циклі обчислюватиметься довжина кожного наступного етапу і додаватиметься до загальної довжини всіх попередніх етапів. Кінцевий результат виведемо також у поле.

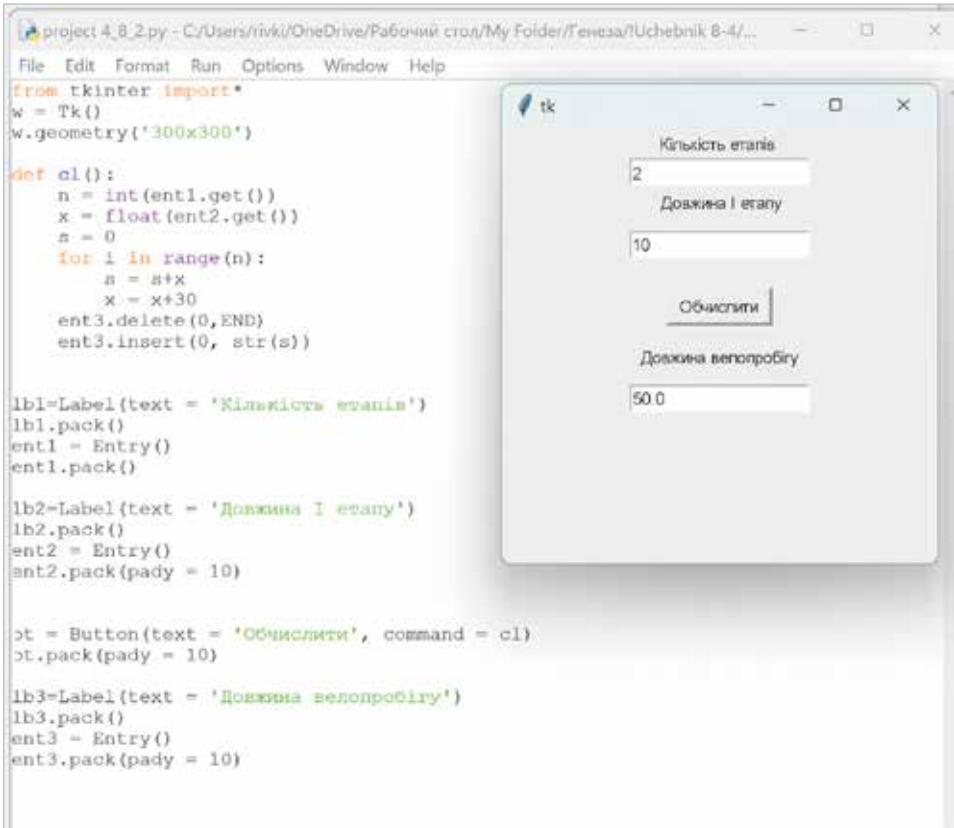
1. Відкрийте вікно середовища розробки проєктів.
2. Створіть новий проєкт, текст якого наведено на малюнку 1.

У цьому проєкті перші три команди обробника події, що знаходяться перед циклом, призначені для введення вхідних даних (кількість етапів  $n$  і довжина I етапу  $x$ ) і надання сумі початкового значення **0**.

Потім розпочинається цикл, у якому команди тіла циклу будуть повторюватись  $n$  разів. У тілі циклу дві команди: додавання довжини етапу до загальної довжини попередніх етапів і знаходження довжини наступного етапу.

Після циклу – команда виведення результату.

3. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.8**.
4. Протестуйте проєкт для таких наборів вхідних даних:
  - а)  $n = 1, x = 10$ ; б)  $n = 2, x = 10$ ; в)  $n = 3, x = 20$ .
5. Висловіть гіпотезу: скільки етапів потрібно провести, якщо довжина I етапу 50 км, а довжина велопробігу повинна бути не менше ніж 600 км. Запишіть гіпотезу в зошит.
6. Перевірте вашу гіпотезу, використовуючи проєкт. Запишіть у зошит, чи справдилася ваша гіпотеза і чому.



Мал. 1. Проєкт для розв'язування **Задачі** та вікно його виконання

7. Виконайте проєкт ще для 2–3 наборів значень вхідних даних, які визначте самостійно.
8. Закрийте вікно виконання проєкту.
9. Закрийте вікно редактора розробки проєктів.



### Найважливіше в цьому пункті

**Цикл в алгоритмі** – це фрагмент алгоритму, який може повторюватися поспіль більше ніж один раз. **Цикл з лічильником** використовується в тих випадках, коли перед його виконанням відомо, скільки разів повинні виконатися команди тіла циклу.

Загальний вигляд команди циклу з лічильником у мові **Python** такий:

```
for <ім'я змінної> in range(<кількість повторень>):
 <команди тіла циклу>
```

Рядок **for <ім'я змінної> in range(<кількість повторень>)** називається **рядком заголовка команди циклу з лічильником**. Змінна в рядку заголовка команди циклу з лічильником називається **лічильником циклу**.

Функція **range(<кількість повторень>)** утворює (генерує) набір послідовних цілих чисел, перше з яких 0, а останнє – на 1 менше від заданої кількості повторень. Змінна – лічильник циклу послідовно набуває значень з цього набору чисел.

Виконується команда циклу з лічильником так:

1. Функція **range(<кількість повторень>)** генерує набір цілих чисел від 0 до **кількість повторень – 1**.



2. Виконується перевірка, чи є в наборі числа, які ще не використовувалися і могли б стати новим значенням **змінної – лічильника циклу**.

3. Якщо результат перевірки **True**, то **змінна – лічильник циклу** набуває значення першого з набору чисел, які ще не використовувалися, і після цього виконується дія 4; якщо **False**, то виконання команди циклу закінчується і далі виконується команда, наступна за командою циклу.

4. Виконуються команди тіла циклу.

5. Повторюється виконання дії 2.

У записі команди циклу з лічильником усі команди тіла циклу повинні бути записані з однаковим відступом праворуч від першої літери у слові **for**.

У команді циклу з лічильником можна також використовувати функцію **range(<початок діапазону>, <кінець діапазону>, <крок>)**, яка має 3 аргументи цілого типу:

- **початок діапазону** – перше ціле число, яке потрапить до згенерованого набору;
- **кінець діапазону** – ціле число, яке не увійде до згенерованого набору й яке більше за всі числа з набору, якщо **крок** > 0, і менше від усіх чисел з набору, якщо **крок** < 0;
- **крок** – різниця між кожними двома сусідніми цілими числами у згенерованому наборі чисел.



### Дайте відповіді на запитання

1. Що таке *цикл в алгоритмі*?
2. Який вигляд має команда циклу з лічильником у мові програмування **Python**?
3. Як виконується команда циклу з лічильником у мові програмування **Python**?
4. Які ви знаєте приклади задач, які можна розв'язати з використанням циклу з лічильником?
5. На яких тестових наборах вхідних даних доцільно тестувати проєкт з циклом з лічильником?



### Обговоріть і зробіть висновки

1. Чи можуть команди тіла циклу з лічильником виконуватися тільки один раз? Наведіть приклад.
2. Чи можуть команди тіла циклу з лічильником не виконуватися жодного разу? Наведіть приклад.
3. Чи може виконання циклу з лічильником ніколи не закінчитися? Поясніть свою відповідь.
4. У чому полягають відмінності у виконанні основних алгоритмічних структур: слідування, розгалуження, цикл?



### Виконайте завдання

1. Створіть у зошиті блок-схему фрагмента проєкту і виконайте його:

а)  $n = 1$   
**for**  $i$  **in** **range**(4):  
 $n = n * (i + 1)$

б)  $p = 0$   
 $a = 2$   
**for**  $i$  **in** **range**(3, 7, 1):  
 $a = 3 * a + 1$   
 $p = p + a$

2. Створіть блок-схему фрагмента проєкту і виконайте його:

а)  $n = 1$

**for**  $i$  **in** **range**(4):

$n = n + (i + 1)**3$

б)  $p = 0$

$a = 4$

**for**  $i$  **in** **range**(2, 6, 1):

$a = 2 * a - 1$

$p = p + a$

3. Створіть математичну модель і блок-схему алгоритму знаходження суми 5 чисел, перше з яких дорівнює 7, а кожне з чотирьох наступних на 3 більше за попереднє. Виконайте складений алгоритм.

4. Створіть математичну модель і блок-схему алгоритму знаходження суми 6 чисел, перше з яких дорівнює 100, а кожне з п'яти наступних дорівнює попередньому, поділеному на 5. Виконайте складений алгоритм.

5. Створіть математичну модель і проєкт для обчислення суми  $n$  чисел, перше з яких дорівнює додатному числу  $x$ , а кожне наступне у 2 рази більше за попереднє. Запишіть у зошит набори вхідних даних, для яких ви тестували проєкт, і отриманий результат. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.8.5.

6. Створіть математичну модель і проєкт для обчислення суми  $n$  чисел, перше з яких дорівнює  $x$ , а кожне наступне дорівнює квадрату попереднього. Запишіть у зошит набори вхідних даних, для яких ви тестували проєкт, і отриманий результат. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.8.6.

7. Людина поклала в банк певну суму грошей. Щорічно банк додає до вкладу 20 % від суми, що зберігається на рахунку після попереднього року. Створіть математичну модель і проєкт для визначення суми, яка буде на рахунку через  $n$  років. Запишіть у зошит набори вхідних даних, для яких ви тестували проєкт, і отриманий результат. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.8.7. Висловіть гіпотезу: на скільки років потрібно покласти гроші, щоб їх сума збільшилася вдвічі. Запишіть гіпотезу в зошит. Використовуючи проєкт, перевірте свою гіпотезу. Запишіть у зошит, чи підтвердилася ваша гіпотеза і чому.

8. Людина поклала в банк певну суму грошей. Щорічно банк додає до вкладу заздалегідь визначений відсоток від суми, що зберігається на рахунку на кінець поточного року. Створіть математичну модель і проєкт для визначення, на скільки зросте сума на рахунку через  $n$  років. Запишіть у зошит набори вхідних даних, для яких ви тестували проєкт, і отриманий результат. Збережіть проєкт у вашій папці у файлі з іменем завдання 4.8.8.

9. Спортсмен, готуючись до участі в марафоні, у перший день тренувань подолав дистанцію в  $s$  км. Кожного наступного дня його денна дистанція збільшувалася на  $p$  % від дистанції попереднього дня. Створіть математичну модель і проєкт для визначення:

- 1) дистанції, яку спортсмен здолає на  $n$ -ий день тренувань;
- 2) сумарної дистанції за  $n$  днів тренувань;
- 3) на скільки більшою буде дистанція в  $n$ -ий день тренувань за дистанцію першого дня?

Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.8.9**.

Висловіть гіпотезу: скільки кілометрів потрібно пробігти в І день і на скільки відсотків потрібно збільшувати дистанцію щодня, щоб за тиждень сумарно пробігти не менше ніж 200 км. Запишіть гіпотезу в зошит. Використовуючи проєкт, перевірте свою гіпотезу. Запишіть у зошит, чи підтвердилася ваша гіпотеза і чому.



### Готуємось до вивчення нового матеріалу

1. Чи знаєте ви інші види циклів, крім циклу з лічильником? Наведіть приклади таких циклів з життя. Як вони записуються, як виконуються?
2. У яких випадках в алгоритмах потрібно використовувати інші цикли, а не цикл з лічильником?

## 4.9. ЦИКЛИ З ПЕРЕДУМОВОЮ

У цьому пункті йтиметься про:

- ▶ команду циклу з передумовою;
- ▶ використання команди циклу з передумовою у проєктах.

### ЦИКЛИ З ПЕРЕДУМОВОЮ



#### Пригадайте

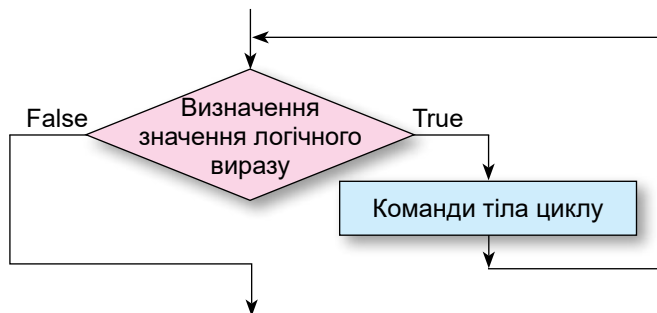
● У яких випадках в алгоритмах використовують цикл з лічильником? ● Який загальний вигляд команди циклу з лічильником в мові програмування **Python** і як вона виконується? ● Як виконується команда **повтори до** в мові програмування **Scratch**?



Ви вже знаєте, якщо перед виконанням циклу невідомо, скільки разів мають виконатися команди тіла циклу, то використовувати цикл з лічильником не можна.

У таких випадках використовують інші цикли, зокрема, **цикл з передумовою**.

Блок-схему циклу з передумовою наведено на малюнку 4.35.



Мал. 4.35. Блок-схема циклу з передумовою



Команда циклу з передумовою в мові програмування **Python**:

**while**<логічний вираз>:  
 <команди тіла циклу>  
 (англ. *while* – поки).

Виконання команди циклу з передумовою відбувається так: визначається значення **логічного виразу**; якщо це значення дорівнює **True**, то виконуються **команди тіла циклу**, після чого знову визначається значення **логічного виразу** і, якщо це значення знову дорівнює **True**, то знову виконуються **команди тіла циклу**, після чого знову визначається значення **логічного виразу**; якщо значення **логічного виразу** дорівнює **False**, то **команди тіла циклу** не виконуються, а виконується команда, наступна за циклом.

*Звертаємо вашу увагу:* серед команд тіла циклу можуть бути лінійні фрагменти, розгалуження, інші цикли.

#### Для тих, хто хоче знати більше

Якщо потрібно достроково перервати виконання циклу, то можна використати команду **break** (англ. *break* – переривати, перерва).

Розглянемо задачу, алгоритм розв’язування якої містить цикл з передумовою.

**Задача.** Фірма щомісячно перераховує кошти на покупку дронів. Першого місяця вона перерахувала  $x$  грн, а кожного наступного – на  $y$  грн більше. Створити математичну модель і проєкт для визначення кількості місяців, через які загальна переведена сума всіх коштів перевищить **100 000** грн.

Створимо математичну модель для цієї задачі:

**вхідні дані:** кошти, переведені в I місяці ( $x$ ); число, на яке сума коштів у наступному місяці більша за суму коштів за попередній місяць ( $y$ ); число, яке має перевищити загальна сума грошей (**100 000**);

**кінцеві результати:** кількість місяців ( $n$ );

**формули:**  $x = x + y$ ,  $s = s + x$ ,  $s \leq 100\,000$ .

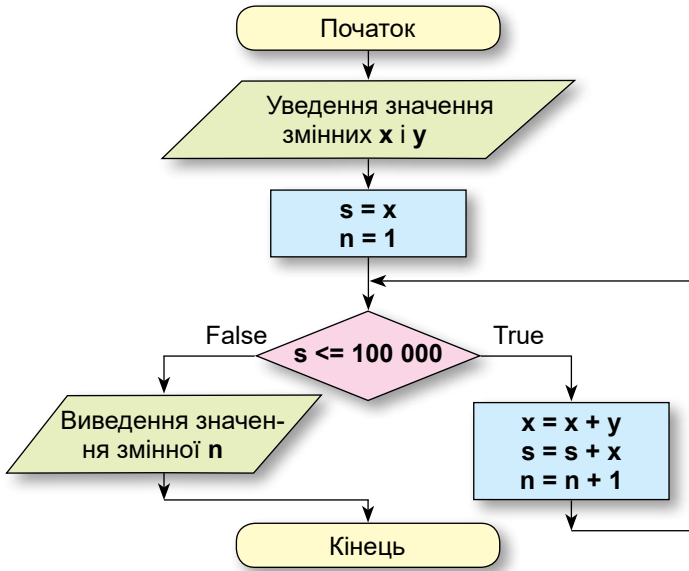
З трьох вхідних даних два потрібно вводити ( $x$  і  $y$ ), а третє – конкретне число. У задачі кінцевий результат, який потрібно виводити, один.

Перед початком циклу присвоїмо змінній  $s$ , яку використаємо для збереження загальної суми, значення першого доданка ( $x$ ). А змінній  $n$ , яку використаємо для збереження кількості потрібних місяців, присвоїмо значення 1, оскільки в сумі вже враховано один (перший) доданок.

До суми потрібно додавати наступні доданки, поки ця сума буде менша або дорівнюватиме 100 000. Тому логічний вираз у заголовку циклу матиме вигляд  $s \leq 100\,000$ . У тілі циклу шукатимемо наступний доданок ( $x = x + y$ ), додаватимемо його до поточного значення суми ( $s = s + x$ ) і збільшуватимемо кількість доданих доданків на 1 ( $n = n + 1$ ).

Блок-схему алгоритму розв’язування цієї задачі наведено на малюнку 4.36.





Мал. 4.36. Блок-схема алгоритму розв'язування **Задачі**

Проект для розв'язування цієї задачі та вікно його виконання наведено на малюнку 4.37.

```

project 4.9_1.py - C:/Users/nvki/OneDrive/Рабочий стол/My Folder/Генеза/Учебник 8-4/...
File Edit Format Run Options Window Help
from tkinter import *
w = Tk()
w.geometry('300x300')

def cl():
 x = float(ent1.get())
 y = float(ent2.get())
 s = x
 n = 1
 while s <= 100000:
 x = x+y
 s = s+x
 n = n+1
 ent3.delete(0, END)
 ent3.insert(0, str(n))

lb1=Label(text = 'У перший місяць')
lb1.pack()
ent1 = Entry()
ent1.pack()

lb2=Label(text = 'Щомісячне збільшення')
lb2.pack()
ent2 = Entry()
ent2.pack(pady = 10)

bt = Button(text = 'Обчислити', command = cl)
bt.pack(pady = 10)

lb3=Label(text = 'Кількість місяців')
lb3.pack()
ent3 = Entry()
ent3.pack(pady = 10)

```

Мал. 4.37. Проект розв'язування **Задачі** та вікно його виконання

Тестувати проєкт, що містить цикл з передумовою, доцільно для таких наборів значень вхідних даних, щоб тіло циклу не виконувалося жодного разу, виконувалося 1 раз; 2 рази; 3 рази; 5 разів і за потреби при інших.



### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Задача.** Створіть математичну модель і проєкт для визначення, на скільки років вкладник має покласти  $x$  грн до банку, щоб отримати від банку не менше ніж  $a$  грн, якщо банк щорічно збільшує вклад на  $p$  %.

Створимо математичну модель для цієї задачі:

**вхідні дані:** сума вкладених грошей ( $x$ ); щорічне збільшення вкладу ( $p$  %); сума грошей, не менше якої потрібно отримати від банку ( $a$ );

**кінцеві результати:** кількість років ( $n$ );

**формули:** сума грошей через кожний рік:  $x = x + x * p / 100$ ;  $n = n + 1$ .

Перед циклом змінній  $n$  для кількості років присвоїмо значення 0, тому що поки що вкладник тільки поклав суму  $x$  грн у банк. Гроші будуть лежати в банку, поки їх сума ще буде менша від потрібної суми  $a$  грн. Вкладник забере гроші з банку тільки тоді, коли сума грошей стане рівною або перевищить  $a$  грн. Тому логічним виразом циклу з передумовою буде нерівність  $x < a$ .

У тілі циклу дві команди:

перша для обчислення суми грошей через рік  $x = x + x * p / 100$ , а друга для збільшення кількості років на 1:  $n = n + 1$ .

1. Відкрийте вікно середовища розробки проєктів.
2. Уведіть фрагмент проєкту, наведений на малюнку 1.
3. Збережіть проєкт у вашій папці у файлі з іменем **вправа 4.9**.
4. Запустіть проєкт на виконання.
5. Уведіть для тестування проєкту набір значень вхідних даних, для якого команди тіла циклу не виконуватимуться жодного разу, наприклад  $x = 1000$ ,  $p = 20$ ,  $a = 500$ .
6. Проаналізуйте результат виконання проєкту і порівняйте його з очікуваним.
7. Уведіть для тестування проєкту набір значень вхідних даних, щоб команди тіла циклу виконувалися 1 раз, наприклад  $x = 1000$ ,  $p = 20$ ,  $a = 1100$ .
8. Проаналізуйте результат виконання проєкту і порівняйте його з очікуваним.
9. Доберіть і введіть для тестування проєкту набір значень вхідних даних, щоб команди тіла циклу виконувалися 2 рази. Запишіть цей набір значень у зошит.
10. Проаналізуйте результат виконання проєкту та порівняйте його з очікуваним.
11. Виконайте проєкт ще для 2–3 наборів вхідних даних. Запишіть у зошит ці набори і відповідні результати виконання проєкту.
12. Закрийте всі відкриті вікна.



```

project 4_9_2.py - C:/Users/nvki/OneDrive/Робочий стіл/My Folder/Тенеза/Учебник 8-4/...
File Edit Format Run Options Window Help
from tkinter import*
w = Tk()
w.geometry('300x300')

def cl():
 x = float(ent1.get())
 p = float(ent2.get())
 a = float(ent3.get())
 n = 0
 while x<a:
 x = x*x*p/100
 n = n+1
 ent4.delete(0,END)
 ent4.insert(0, str(n))

lb1=Label(text = 'Початкова сума')
lb1.pack()
ent1 = Entry()
ent1.pack()

lb2=Label(text = 'Річні відсотки')
lb2.pack()
ent2 = Entry()
ent2.pack(pady = 5)

lb3=Label(text = 'Кінцева сума')
lb3.pack()
ent3 = Entry()
ent3.pack(pady = 5)

bt = Button(text = 'Обчислити', command = cl)
bt.pack(pady = 5)

lb4=Label(text = 'Кількість років')
lb4.pack()
ent4 = Entry()
ent4.pack(pady = 10)

```

Мал. 1. Текст проекту розв'язування **Задачі** та вікно його виконання



### Найважливіше в цьому пункті

Команда циклу з передумовою в мові програмування **Python**:

**while<логічний вираз>:**  
**<команди тіла циклу>**  
 (англ. *while* – поки).

Виконання команди циклу з передумовою відбувається так: визначається значення **логічного виразу**; якщо це значення дорівнює **True**, то виконуються **команди тіла циклу**, після чого знову визначається значення **логічного виразу** і, якщо це значення знову дорівнює **True**, то знову виконуються **команди тіла циклу**, після чого знову визначається значення **логічного виразу**; якщо значення **логічного виразу** дорівнює **False**, то **команди тіла циклу** не виконуються, а виконується команда, наступна за циклом.



**Дайте відповіді на запитання**

1. Який вигляд має блок-схема циклу з передумовою? Поясніть виконання цього циклу.
2. Який вигляд має команда циклу з передумовою в мові програмування Python? Як вона виконується?
3. Чи можуть команди тіла циклу з передумовою не виконуватися жодного разу? Поясніть свою відповідь. Наведіть приклади.
4. Чи може виконання циклу з передумовою ніколи не закінчитися? Поясніть свою відповідь.



**Обговоріть і зробіть висновки**

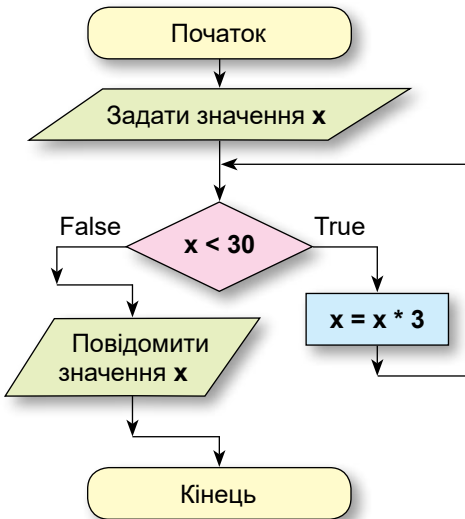
1. Що спільного і що відмінного у циклів з лічильником і з передумовою?
2. Чи можна команду циклу з лічильником замінити командою циклу з передумовою? Якщо так, наведіть приклад.
3. Чи можна команду циклу з передумовою замінити командою циклу з лічильником? Якщо так, наведіть приклад.
4. У чому полягають відмінності у виконанні основних алгоритмічних структур: слідування, розгалуження, цикл з лічильником і цикл з передумовою?



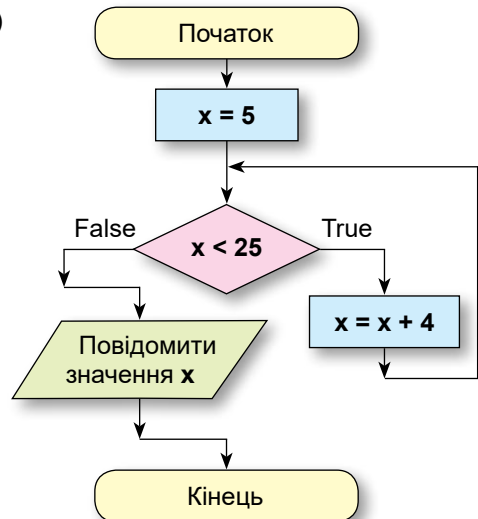
**Виконайте завдання**

1. Виконайте в зошиті алгоритми:

а)



б)



2. Виконайте в зошиті фрагменти проєктів:

```

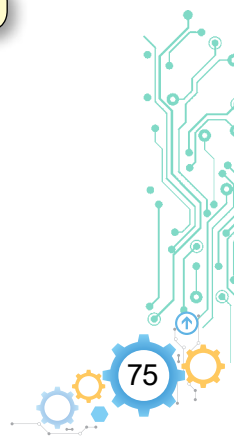
а) k = 1
 a = 12
 while a < 100:
 a = 2 * a - 4
 k += 1

```

```

б) s = 0
 a = 1
 k = 1
 while a < 50:
 s = s + a
 k += 1
 a = k * k

```



3. Виконайте в зошиті фрагменти проєктів:

а)  $k = 1$   
 $a = 100$

**while**  $a > 10$ :

$a = a / 2$

$k += 1$

б)  $s = 0$

$a = 5$

**while**  $a \leq 100$ :

$s = s + a$

$a = a * 2$

4. Складіть текстові задачі за алгоритмами їх розв'язування із завдання № 1.
5. Складіть текстові задачі за алгоритмами їх розв'язування із завдання № 2.
6. Створіть математичну модель і проєкт для визначення, за скільки годин робітник виконає план у  $x$  деталей, якщо за першу годину роботи він виготовив  $y$  деталей, а за кожну наступну годину виготовляє на 1 деталь більше, ніж за попередню. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.9.6**.
7. Створіть математичну модель і проєкт для визначення, за скільки годин турист подолає відстань  $x$  км, якщо за першу годину він пройшов 6 км, а за кожну наступну годину проходить на 1 км менше, ніж за попередню. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.9.7**.
8. Створіть проєкт для обчислення середнього арифметичного додатних чисел, перше з яких дорівнює 100, а кожне наступне менше від попереднього на задане додатне число. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.9.8**.
9. Створіть математичну модель і проєкт для визначення, на скільки років вкладник має покласти  $x$  грн до банку, щоб отримати від банку більше за  $a$  грн прибутку, якщо банк щорічно збільшує вклад на  $p$  %. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.9.9**.
10. Створіть проєкт для знаходження найбільшого спільного дільника (НСД) двох натуральних чисел за алгоритмом Евкліда: поки числа різні, більше з них замінювати різницею більшого та меншого; коли числа стануть рівними, то кожне з них дорівнюватиме їх НСД. Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.9.10**.
11. Спортсменка, готуючись до участі в марафоні, у перший день тренувань пододала дистанцію в  $s$  км. Кожного наступного дня її денна дистанція збільшувалася на  $p$  % від дистанції попереднього дня. Створіть математичну модель і проєкт для визначення:
  - 1) у який день тренувань спортсменка подолає дистанцію в  $s1$  км;
  - 2) у який день тренувань дистанція збільшиться на  $s2$  км порівняно з попереднім днем.
 Збережіть проєкт у вашій папці у файлі з іменем **завдання 4.9.11**.



#### Оцініть свої знання та вміння

Оцініть свої навчальні досягнення з розділу «Алгоритми та програми» (початковий, середній, достатній, високий рівень).

- Я знаю етапи розв'язування задач з використанням алгоритмів і проєктів.
- Я вмію виконувати кожний етап розв'язування задач з використанням алгоритмів і проєктів.
- Я знаю властивості кнопки, напису, поля та яких значень вони можуть набувати.
- Я вмію створювати віконні проєкти з використанням кнопок, написів, полів, їх властивостей.
- Я вмію створювати віконні проєкти з обробниками подій для кнопок, написів, полів.
- Я знаю, які величини називаються логічними.
- Я знаю, що є результатами операцій кон'юнкція, диз'юнкція, заперечення над значеннями логічних величин.
- Я вмію складати таблицю істинності для логічних виразів.
- Я вмію створювати віконні проєкти з використанням логічних величин і операцій над ними.
- Я вмію створювати віконні проєкти з використанням прапорців і перемикачів.
- Я знаю, що таке цикл з лічильником і випадки, у яких його можна застосовувати у проєктах.
- Я вмію створювати віконні проєкти з використанням циклу з лічильником.
- Я знаю, що таке цикл з передумовою і випадки, у яких його можна застосовувати у проєктах.
- Я вмію створювати віконні проєкти з використанням циклу з передумовою.
- Я вмію добирати тестові набори значень вхідних даних для тестування проєктів.

*Повторіть той матеріал, який ви знаєте недостатньо.*



### Практична робота № 6.

#### «Проєкти із циклами»

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

1. Створіть математичну модель і проєкт для розв'язування задачі: *Перед початком повені рівень води в річці становив  $N$  см. Під час повені кожну годину рівень води зростає на  $P$  см. Яким буде рівень води через  $N$  год після початку повені? Через скільки годин після початку повені рівень води буде не менше ніж  $K$  см?*
2. Використайте у проєкті поля для введення вхідних даних, поле для виведення відповіді на перше запитання задачі та дві кнопки. Біля кожного поля розмістіть написи для пояснювальних коментарів.
3. Складіть обробник події **Click** для першої кнопки, виконання якого приведе до виведення в поле відповіді на перше запитання задачі.
4. Складіть обробник події **Click** для другої кнопки, виконання якого приведе до знаходження відповіді на друге запитання задачі та виведення його у вікно повідомлень.
5. Збережіть проєкт у вашій папці у файлі з іменем **практична 6**.



### Готуємось до вивчення нового матеріалу

1. Що таке *дизайн*? Які види дизайну вам відомі?
2. Які ви знаєте вимоги до дизайну слайдів комп'ютерної презентації?



# СТВОРЕННЯ ТА ПУБЛІКАЦІЯ ВЕБРЕСУРСІВ

У цьому розділі ви отримаєте нові, а також поглибите та розширите свої знання й удосконалите навички з таких тем:

- ▶ етапи створення сайтів;
- ▶ тренди вебдизайну;
- ▶ використання гіпертекстових, графічних і мультимедійних елементів на вебсторінках;
- ▶ мова гіпертекстової розмітки;
- ▶ автоматизовані засоби створення та публікації вебресурсів.

## 5.1. ЕТАПИ СТВОРЕННЯ САЙТУ. ДИЗАЙН ВЕБСТОРИНОК. ТРЕНДИ ВЕБДИЗАЙНУ

У цьому пункті йтиметься про:

- ▶ етапи створення сайтів;
- ▶ дизайн вебсторінок;
- ▶ рекомендації до оформлення та структури вебсторінок;
- ▶ тренди вебдизайну;
- ▶ оцінювання вебсайтів.

### ЕТАПИ СТВОРЕННЯ САЙТІВ



#### Пригадайте

- Що таке *вебсторінка*? Що таке *вебсайт*?
- Які об'єкти можуть бути розміщені на вебсторінках?
- Що потрібно знати, щоб відкрити вебсторінку?

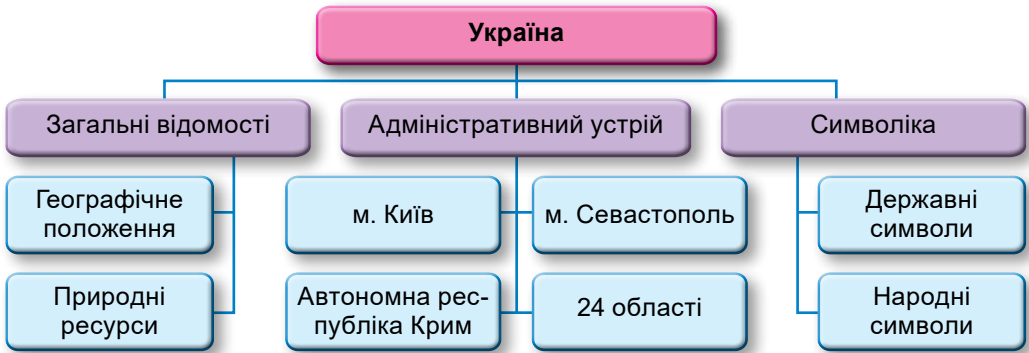
Ви вже знаєте, що таке вебсайти, які об'єкти можуть розміщуватися на вебсторінках, з яких складаються сайти, як здійснювати пошук і перегляд вебресурсів Інтернету. Тепер ви навчитеся їх створювати та розміщувати в Інтернеті.

Створення вебсайту складається з кількох етапів.

1. **Постановка завдання.** Визначається мета створення сайту, його основна тематика, цільова аудиторія – користувачі, які можуть зацікавитись умістом сайту, аналізуються існуючі сайти такої самої або схожої тематики.
2. **Визначення структури сайту.** Складається перелік тематичних розділів сайту та сторінок у них, визначаються зв'язки між сторінками, утворюється **мапа (карта) сайту**, яка пояснює його логічну структуру, формулюються вимоги до функціонування інтерактивних елементів.

Наприклад, на малюнку 5.1 наведено частину мапи сайту, присвяченого Україні.





Мал. 5.1. Мапа сайту України

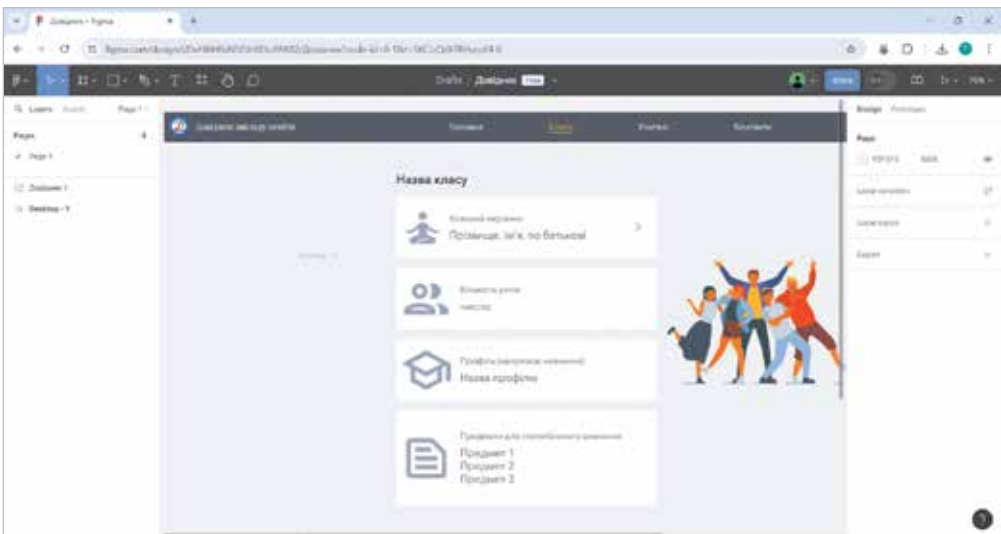
3. **Створення дизайн-макета сторінок сайту.** На цьому етапі обирають, які об'єкти будуть міститися на сторінках, розмір і розташування цих об'єктів, кольорову гаму, набір шрифтів та ін., тобто визначають **стиль сайту**.

У дизайні враховують три складові:

- **ергономіку** – зручність і зрозумілість користування сайтом;
- **композицію** – узгодженість розмірів об'єктів і їх розташування;
- **колористику** – гармонійність поєднання кольорів.

Дизайн-макет створюють, як правило, у спеціалізованому графічному редакторі, наприклад, **Adobe XD, Sketch, Figma** або іншому.

На малюнку 5.2 наведено дизайн-макет вебсторінки шкільного довідника, який створено у спеціалізованому графічному онлайн-редакторі **Figma**.



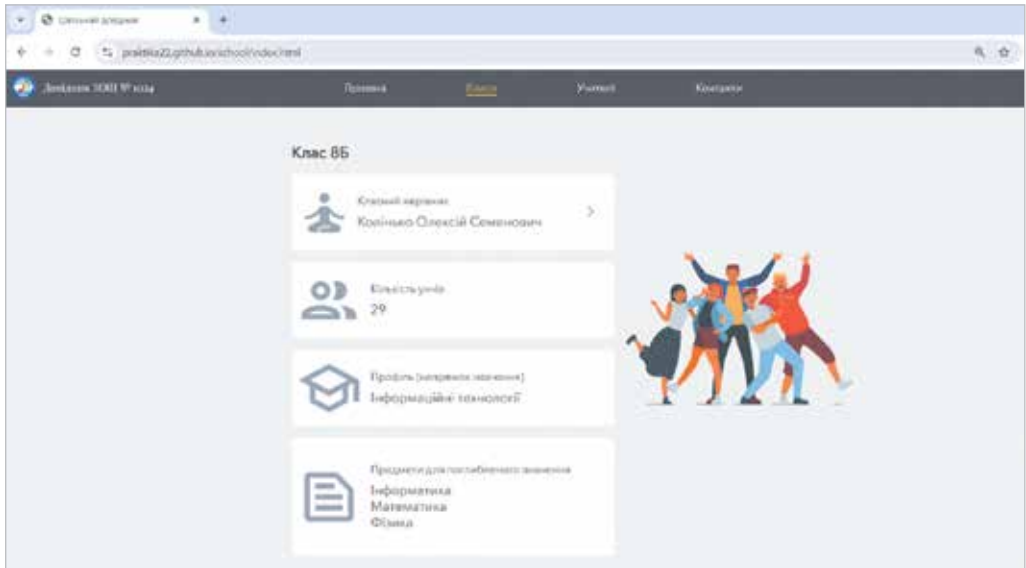
Мал. 5.2. Дизайн-макет сторінки сайту в редакторі **Figma**

4. **Створення сторінок сайту.** На цьому етапі вибирають програмні засоби, які будуть використані для створення сторінок сайту. Під час створення вебсторінок на них розміщують і форматують



**контент** (англ. *content* – вміст) – текстові, графічні, мультимедійні та інтерактивні елементи, а також **елементи навігації** – кнопки, гіперпосилання, та інше таким чином, щоб сторінка отримала вигляд відповідно до розробленого дизайн-макета; забезпечують функціонування інтерактивних елементів – систем пошуку, форм опитування тощо.

На малюнку 5.3 наведено вигляд сторінки, створеної відповідно до наведеного дизайн-макета.



Мал. 5.3. Вебсторінка, створена відповідно до дизайн-макета

5. **Публікація сайту в Інтернеті.** Призначення сторінкам сайту **URL-адрес** і надання дозволу користувачам Інтернету на перегляд вебсторінок відбувається після **публікації** сайту. Перед публікацією сторінки сайту потрібно розмістити на спеціальному сервері, який надає власні апаратні та програмні ресурси для зберігання файлів користувача, забезпечення доступу до них, опрацювання запитів тощо. Такі послуги називають **хостингом** (англ. *hosting* – виявлення гостинності).

## ОФОРМЛЕННЯ ТА СТРУКТУРА ВЕБСТОРИНОК



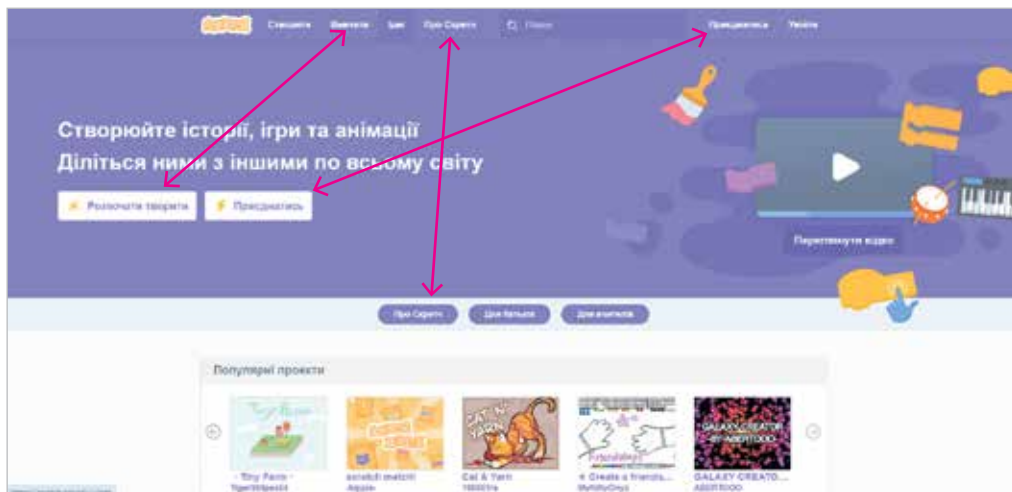
### Поміркуйте

- Які елементи на вебсторінках слугують орієнтиром для користувача, щоб зрозуміти, який сайт і який його розділ переглядається? Чому це зручно?
- Які засоби використовуються для перегляду сторінок, на яких контент не вміщується по висоті на одному екрані?
- Чи доцільно мати кілька різних елементів на вебсторінці, щоб перейти до перегляду іншої сторінки?

Створюючи дизайн-макет для власного сайту, варто мати на увазі рекомендації з оформлення та структури вебсторінок, пов'язані з **ергономікою, композицією та колористикою**.

Вдалий дизайн сайту передбачає зручність користування ним. Будь-яка людина повинна інтуїтивно пов'язувати дії, які їй потрібно виконати на вебсторінці, з об'єктами, які вона бачить. Тому під час розробки дизайну вебсторінок слід урахувувати такі правила *ергономіки*:

- розміщуйте назву сайту в заголовку всіх сторінок;
- зазначайте на кожній сторінці, до якого тематичного розділу вона входить;
- вибирайте розмір шрифту, за якого текст буде сприйматися комфортно;
- створіть можливість переходу з будь-якої сторінки сайту до перегляду головної сторінки;
- створюйте внутрішні гіперпосилання на сторінках з великим обсягом контенту для переходів усередині вебсторінки;
- не допускайте появи горизонтальних смуг прокручування на вебсторінках;
- пропонуйте кілька посилань для переходу на одну й ту саму сторінку, щоб відвідувачі сайту могли вибрати найзручніший для себе спосіб (мал. 5.4).



Мал. 5.4. Приклад сторінки з посиланнями, що дублюються

При розробці дизайну вебсторінок потрібно намагатися надавати додаткові зручності для користувачів з особливими потребами: можливість збільшувати шрифт тексту, розміри зображень, озвучувати призначення окремих елементів.

Будь-який відвідувач сайту завдяки дизайну повинен швидко зрозуміти тематику сайту та легко знайти корисний для себе контент. Тому слід дбати про доцільне розміщення об'єктів на вебсторінках, а для цього бажано дотримуватися таких правил *композиції*:

- великий текст поділяйте на частини, визначайте для них заголовки;
- шрифт у заголовках робіть крупнішим від шрифту основного тексту;
- розміщуйте найважливіші дані на початку сторінки;
- доповнюйте текстові відомості зображеннями, які ілюструють текст;

- не зловживайте анімацією, відео, музичними та графічними елементами – вони можуть відволікати увагу від корисних матеріалів та уповільнюватимуть завантаження сторінок;
- не копіюйте матеріали з інших інтернет-ресурсів, а пропонуйте гіперпосилання на них для уточнення відомостей – так ви забезпечите дотримання закону про захист авторських прав. Обираючи *кольорову гаму* для вашого сайту:
- враховуйте ваші вподобання, тематику сайту та вподобання потенційних відвідувачів, що складають цільову аудиторію;
- вибирайте контрастні кольори для тексту та фону сторінки, щоб текст легше сприймався.

## ТРЕНДИ ВЕБДИЗАЙНУ



### Поміркуйте

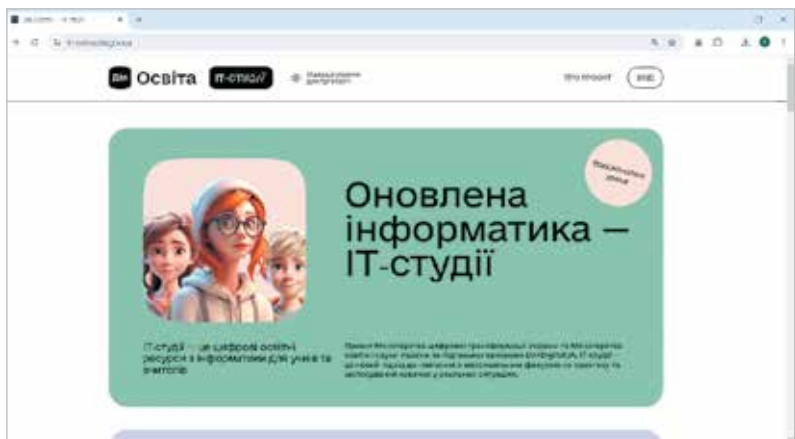
- Чи завжди зручно, щоб вебсторінка виглядала однаково під час її перегляду на екрані смартфона та стаціонарного комп'ютера?
- Сайти якого змісту та з яким оформленням вас найбільше приваблюють?

Під час вибору дизайну для власного вебсайту потрібно враховувати також сучасні **тренди вебдизайну**.

**Тренд** (англ. *trend* – тенденція) – основна тенденція змінення чогонбудь, загальний напрям змін упродовж тривалого часу.

**Адаптація** (лат. *adaptō* – пристосовувати) – процес пристосування до мінливих умов зовнішнього середовища.

Змінення трендів у вебдизайні пов'язано з розвитком цифрових технологій. Так, за даними статистики в наш час більше половини користувачів звертаються до ресурсів Інтернету з мобільних пристроїв. Тому в останні роки особливого значення набув так званий **адаптивний дизайн**, при забезпеченні якого одна й та сама сторінка, можливо, по-різному, але в одному стилі, коректно відображається і на великих, і на маленьких екранах (мал. 5.5).



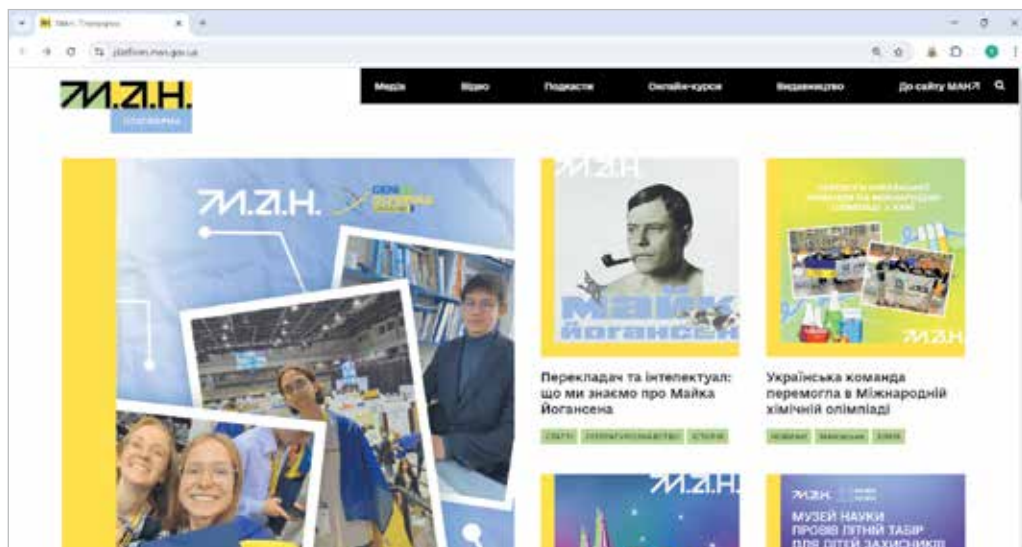
Мал. 5.5. Вигляд сторінки з адаптивним дизайном під час перегляду на смартфоні (ліворуч) і ноутбучі (праворуч)

Ще одним трендом, пов'язаним з особливостями мобільних пристроїв, стало використання **темного режиму**. Чорний колір на екранах деяких видів мобільних пристроїв не потребує затрат енергії, а тому в темному режимі збільшується час використання батареї пристрою. Крім того, темний фон сторінок зменшує напругу очей, оскільки забезпечує контраст між фоном і вмістом сторінки.

Стрімке поширення систем **штучного інтелекту** викликало появу ще одного тренду – використання у веброзробці механізму голосового пошуку та елементів штучного інтелекту.

Також у дизайні вебсторінок упродовж останніх кількох років спостерігаються такі тренди:

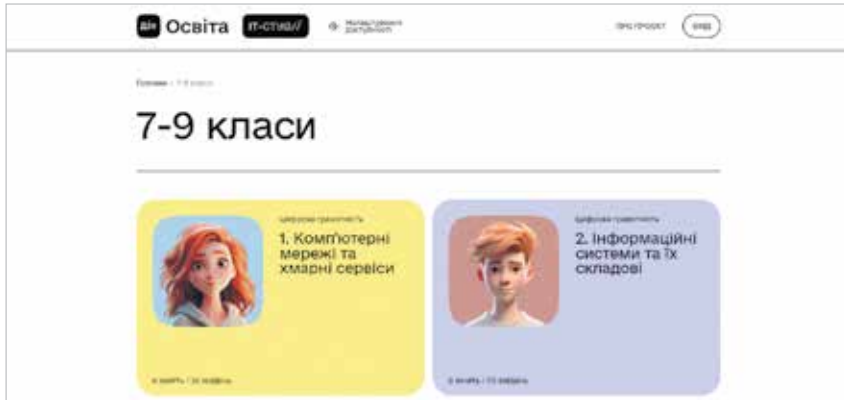
- **типографіка** – застосування нестандартних оригінальних шрифтів (мал. 5.6), великих заголовків, креативного розміщення тексту;



Мал. 5.6. Використання фірмового шрифту на сайті МАН

- **нестандартні колірні рішення** – використання яскравих кольорів і багатоколірної градієнтної заливки фону сторінки або її окремих елементів;
- **анімація та інтерактивність** – використання анімаційних ефектів для привертання уваги до важливих елементів сторінки, застосування плавних переходів під час прокручування вебсторінки, реалізація інтерактивної анімації та вставлення 3D-об'єктів;
- **мінімалізм** – обмеження кількості елементів на екрані, проста композиція, наявність великого порожнього простору. Популярним є структурування контенту на вебсторінках з використанням так званих **карток** – прямокутних блоків, у яких об'єднані взаємопов'язані графічні, текстові та навігаційні елементи (мал. 5.7).

Сайт, який відповідає трендам вебдизайну, виглядає стильно та сучасно, привертає увагу відвідувачів, є зручним для його перегляду на різних пристроях.



Мал. 5.7. Використання карток на вебсторінці

## ОЦІНЮВАННЯ ВЕБСАЙТІВ



### Поміркуйте

- Від чого залежить тривалість перегляду вами тієї чи іншої вебсторінки?
- Чи однаково сприймаються вами всі сайти однакової тематики? Від чого залежить ваше враження від сайту?

Здійснюючи навігацію вебсторінками, ви, напевно, звертали увагу на те, як по-різному сприймаються сайти однакової тематики та спрямованості. На одних сайтах ви затримувалися надовго, а з деяких – одразу переходили на інший ресурс.

Для відвідувача під час оцінювання вебсайту важливими показниками є:

- ступінь інформативності вебсайту, можливість отримання корисних матеріалів, які не повторюють відомості з інших ресурсів;
- візуальна привабливість сайту, дизайн, стиль оформлення;
- зрозумілість і зручність навігації, легкість знаходження потрібних даних, структурованість матеріалу, все те, що забезпечує простоту, комфортність та ефективність у процесі використання сайту;
- наявність інтерактивних засобів та ефектів;
- частота оновлення даних на сайті, що свідчить про увагу розробника до власного проєкту;
- мови, якими доступні матеріали на вебсайті;
- можливість залишити коментарі, отримати відповіді, звернутися до розробника за додатковими відомостями та інше.

Тривалість перегляду відвідувачами сторінок на сайті, повторне звертання до його ресурсів визначають популярність сайту та його рейтинг у базах даних пошукових систем.

Набір стратегій та інструментів для просування сайту в Інтернеті, щоб він відображався на верхніх місцях у видачах пошукових систем, називають **SEO-оптимізацією** сайтів (англ. *Search Engine Optimization* – пошукова оптимізація). Урахування трендів і рекомендацій з вебдизайну є однією зі стратегій пошукової оптимізації.



## Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Завдання.** Перегляньте сторінку з адресою <https://platform.man.gov.ua>, оцініть дотримання рекомендацій щодо оформлення та розміщення об'єктів на сторінці та відповідність трендам вебдизайну.

Для цього:

1. Відкрийте у вікні браузера сторінку з адресою [platform.man.gov.ua](https://platform.man.gov.ua).
2. Ознайомтесь з умістом сторінки, запишіть у зошит відповіді на запитання:
  - а) Яка назва сторінки відображається в заголовку браузера та на відповідній кнопці на **Панелі завдань** у нижній частині **Робочого столу**?
  - б) Для переходу на сторінку з якою адресою призначено гіперпосилання, пов'язане з логотипом МАН у верхньому лівому куті сторінки?
  - в) Які гіперпосилання розміщені на верхній панелі навігації?
  - г) Які групи гіперпосилань розміщені в нижньому колонтитулі?
3. Проаналізуйте, як змінюється вигляд сторінки під час змінення масштабу та ширини області перегляду в браузері. Для цього:
  - а) якщо ви працюєте з ноутбуком або стаціонарним комп'ютером, то змініть масштаб перегляду сторінки, натискаючи клавіші **Ctrl + плюс** та **Ctrl + мінус**;
  - б) якщо ви працюєте зі смартфоном або планшетним комп'ютером, то змініть орієнтацію пристрою з вертикальної на горизонтальну або навпаки.
 Запишіть у зошиті відповіді на запитання:
  - а) Які зміни відбуваються на верхній панелі навігації?
  - б) Які зміни відбуваються в розташуванні карток та їх вмісту на сторінці?
  - в) Які зміни відбуваються в нижньому колонтитулі?
4. Проаналізуйте дотримання на сайті вимог ергономіки, композиції та колористики. Для цього перейдіть за гіперпосиланнями на різні сторінки сайту. Запишіть у зошит відповіді на запитання:
  - а) Чи відображається назва сайту в заголовку всіх сторінок?
  - б) Чи існує можливість переходу з будь-якої сторінки до головної сторінки сайту?
  - в) Чи зручно організована навігація для переміщення по головній сторінці, по сторінці **Media**?
  - г) Як структуровано текст? Чи виділені заголовки?
  - д) Як проілюстровано текстові матеріали?
  - е) Чи комфортний розмір і колір шрифту для читання текстів?
 На основі ваших відповідей зробіть висновок щодо дотримання на сайті вимог ергономіки, композиції, колористики та запишіть його у зошит.
5. Проаналізуйте відповідність сторінки трендам вебдизайну. Запишіть у зошит, які з сучасних трендів прослідковуються на сторінці.
6. Закрийте всі відкриті вікна.



### Найважливіше в цьому пункті

Створення вебсайту складається з кількох етапів: постановка завдання, визначення структури сайту, створення дизайн-макета сторінок сайту, створення сторінок, публікація сайту в Інтернеті.

Створюючи дизайн-макет для власного сайту, варто мати на увазі рекомендації з оформлення та структури вебсторінок, пов'язані з ергономікою, композицією та колористикою. Під час розробки дизайну вебсторінок потрібно намагатися надавати додаткові зручності для користувачів з особливими потребами.

У ході вибору дизайну для власного вебсайту потрібно враховувати також сучасні **тренди вебдизайну**. За **адаптивного дизайну** одна й та сама сторінка, можливо, по-різному, але в одному стилі, коректно відображається і на великих, і на маленьких екранах. Сайт, який відповідає трендам вебдизайну, виглядає стильно та сучасно, привертає увагу відвідувачів, є зручним для його перегляду на різних пристроях.



### Дайте відповіді на запитання

1. За якими етапами відбувається створення сайтів?
2. Що таке *дизайн-макет* сторінок сайту?
3. У чому полягає створення вебсторінок?
4. Які ви знаєте сучасні тренди вебдизайну?
5. Які рекомендації щодо оформлення та структури сторінок варто враховувати під час створення сайту?



### Обговоріть і зробіть висновки

1. Які зручності для користувачів з різними видами особливих потреб можна створити на сайті?
2. Чи всі тренди вебдизайну повинні бути враховані на кожній вебсторінці, щоб вона виглядала сучасно? Відповідь поясніть.
3. Чим може відрізнятись дизайн вебсайтів однієї тематики залежно від цільової аудиторії – від віку, статі, інтересів користувачів?
4. У чому схожість і відмінність вимог до оформлення вебсторінок і слайдів комп'ютерних презентацій?



### Виконайте завдання

1. Перегляньте сайт **Тут цікаво** (*tut-cikavo.com*). Оцініть дотримання рекомендацій щодо оформлення та розміщення об'єктів на сторінці та відповідність сучасним трендам. Проаналізуйте, як видозмінюються сторінки під час змінення масштабу та ширини області перегляду в браузері. Створіть текстовий документ, у який запишіть свої висновки. Збережіть документ у вашій папці у файлі з іменем **завдання 5.1.1**.



2. Порівняйте особливості оформлення та зручність користування сайтами **Український Ерудит** (*ukrerudyt.com*) і **Teztt** (*teztt.com*). Створіть текстовий документ, у якому дайте відповіді на запитання:
- а) Яке основне призначення кожного з цих сайтів?
  - б) Наскільки зручно шукати потрібні відомості (наприклад, вікторини з історії України)?
  - в) Чи реалізований на сайтах адаптивний дизайн? У чому це виявляється?
  - г) Наскільки сучасними виглядають сайти? Які сучасні тренди вебдизайну прослідковуються на сторінках?



Зробіть і запишіть у текстовому документі висновок, який сайт здається вам більш привабливим і чому. Збережіть документ у вашій папці у файлі з іменем **завдання 5.1.2**.



**3.** Об'єднайтеся у пари та розробіть у графічному редакторі дизайн-макети двох варіантів головної сторінки сайту вашого закладу освіти – для відтворення на екрані смартфона та на екрані ноутбука. Враховуйте сучасні тренди вебдизайну. Збережіть дизайн-макети у вашій папці у файлах з іменами **завдання 5.1.3.1** та **завдання 5.1.3.2**. Презентуйте створені дизайн-макети однокласникам/однокласницям.



**4.** Виконайте пошук в Інтернеті матеріалів про тренди вебдизайну поточного року та трирічної давнини. Проаналізуйте, які тренди змінились, а які залишились актуальними. Збережіть опис трендів і результати порівняння у вашій папці у файлі з іменем **завдання 5.1.4**. Презентуйте результати вашого дослідження однокласникам/однокласницям.

**5.** Перегляньте на сайті **Rentafont** ([rentafont.com.ua](http://rentafont.com.ua)) колекції шрифтів, які можуть бути використані на вебсайтах. Створіть у графічному редакторі власні зображення кількох українських літер, які можна було б використати для оформлення заголовків на сайті. Збережіть створене зображення у вашій папці у файлі з іменем **завдання 5.1.5**.

**6.** Знайдіть в Інтернеті відомості про адаптивні логотипи сайтів і зображення логотипів відомих брендів, які використовуються в адаптивному дизайні. Створіть у графічному редакторі власні зображення адаптивного логотипу для вашого персонального сайту. Збережіть створене зображення у вашій папці у файлі з іменем **завдання 5.1.6**.



### Готуємось до вивчення нового матеріалу

1. Для чого створюється розмітка на проїжджій частині дороги та які особливості цієї розмітки? У яких ще випадках використовують розмітку та з якою метою?
2. Як ви структуруєте та розміщуєте текст, коли пишете твір у зошиті або вводите в текстовому процесорі?
3. Як у мові програмування, яку ви знаєте, забезпечити виведення тексту в різних рядках?

## 5.2. ПОНЯТТЯ ПРО МОВУ HTML

У цьому пункті йтиметься про:

- ▶ мову HTML;
- ▶ поняття тегу, атрибута тегу, значення атрибута;
- ▶ базову структуру вебсторінки;
- ▶ створення та збереження вебсторінок.

### МОВА HTML. ТЕГИ, АТРИБУТИ ТЕГІВ, ЗНАЧЕННЯ АТРИБУТІВ

#### Пригадайте

- Які об'єкти можуть бути розміщені на вебсторінках? Які ви знаєте властивості цих об'єктів?
- Як здійснюється форматування об'єктів у текстовому документі?
- Що розуміють під словом *розмітка*? Де та для чого використовується це поняття?



**HTML** (англ. *Hyper Text Markup Language* – мова розмітки гіпертексту) – це мова, якою описують структуру та контент вебсторінки для відображення цієї сторінки у вікні браузера.

Кожна вебсторінка має свій **HTML-код**. Коли ми відкриваємо для перегляду деяку вебсторінку з Інтернету, браузер отримує з інтернет-сервера її HTML-код та супутні файли – графічні, відео та інші. На основі HTML-коду браузер розміщує у вікні об'єкти вебсторінки: заголовки, абзаци тексту, гіперпосилання, списки, таблиці, зображення, відео тощо.

Елементами мови **HTML** є **теги** (англ. *tag* – ярлик, ознака), що вказують браузеру, які об'єкти розмістити на сторінці та як їх відобразити. Імена тегів записують усередині кутових дужок < та >:

`<ім'я_тегу>`

Наприклад, тег `<input>` указує браузеру розмістити на сторінці елемент керування *поле*, а тег `<hr>` (англ. *horizontal row* – горизонтальний ряд) – *горизонтальну лінію* (мал. 5.8).



Мал. 5.8. HTML-теги (ліворуч) і відповідні їм об'єкти сторінки (праворуч)

Наведені теги `<input>` та `<hr>` є **непарними** тегами. Найчастіше непарні теги вказують браузеру вставити певний об'єкт на сторінку.

Існують також **парні** теги, що складаються з **відкриваючої** та **закриваючої** частин, які обмежують фрагмент тексту або HTML-коду. У закриваючій частині перед іменем тегу ставлять символ /. Загальний вигляд парного тегу такий:

`<ім'я_тегу> фрагмент тексту або HTML-коду </ім'я_тегу>`

Тут `<ім'я_тегу>` – відкриваюча частина тегу, а `</ім'я_тегу>` – відповідна йому закриваюча частина.

Парні теги вказують браузеру, як відобразити фрагмент, обмежений частинами парного тегу.

Наприклад:

`<h1>Моя Україна</h1>`

Парний тег з іменем **h1** (англ. *header* – заголовок) указує відобразити обмежений ним фрагмент тексту *Моя Україна* як заголовок першого рівня. Теги з іменами від **h1** до **h6** визначають 6 можливих рівнів заголовків на сторінці.

`<p> Столиця – місто <strong>Київ</strong>. </p>`

Тут парний тег з іменем **p** (англ. *paragraph* – пункт) указує відобразити обмежений ним фрагмент HTML-коду *Столиця – місто <strong>Київ</strong>* як абзац, а парний тег з іменем **strong** (англ. *strong* – сильний) указує відобразити слово *Київ* напівжирним накресленням (мал. 5.9).

```
<h1>Моя Україна</h1>
<p> Столиця - місто Київ. </p>
```

# Моя Україна

Столиця - місто Київ.

Мал. 5.9. Використання парних тегів

Теги можуть містити **атрибути** (англ. *attribute* – характерна риса, ознака), які уточнюють дію цих тегів. У атрибутів можуть бути **значення**. Атрибути записують після імені тегу, відокремлюючи пропуском. Значення атрибутів записують у лапках після знаку =. Загальний вигляд парного тегу з одним атрибутом такий:

```
<ім'я_тегу атрибут="значення"> текст або HTML-код </ім'я_тегу>
```

Наприклад:

```

```

Непраний тег з іменем **img** (англ. *image* – зображення) указує вставити на сторінку зображення, а значенням атрибуту **src** (англ. *source* – джерело) є URL-адреса файлу *https://yvu.com.ua/wp-content/uploads/2020/08/images.jpg*, з якого буде взято це зображення;

```
 Юридичний Вісник України
```

Парний тег з іменем **a** (англ. *anchor* – якор) указує створити гіперпосилання для переходу на сторінку з URL-адресою *https://yvu.com.ua/*, яка є значенням атрибуту **href** (англ. *hypertext reference* – гіпертекстова виноска). Для переходу на цю сторінку потрібно вибрати текст *Юридичний Вісник України*, обмежений частинами парного тегу з іменем **a**.

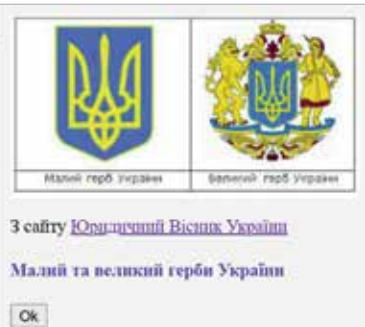
```
<h2 style="color: blue;">Малий та великий герби України</h2>
```

Парний тег з іменем **h2** вказує вивести текст *Малий та великий герби України* як заголовок другого рівня. Атрибут **style** (англ. *style* – стиль) визначає стиль заголовка: колір символів тексту *синій*, відповідно до правила *color: blue* (англ. *color* – колір, *blue* – синій), яке є значенням цього атрибуту.

Тег може мати кілька атрибутів, які відокремлюються один від одного пропусками. Наприклад, тег `<input type="button" value="Ok">` указує розмістити на сторінці елемент керування типу *кнопка* з написом *Ok* (мал. 5.10).

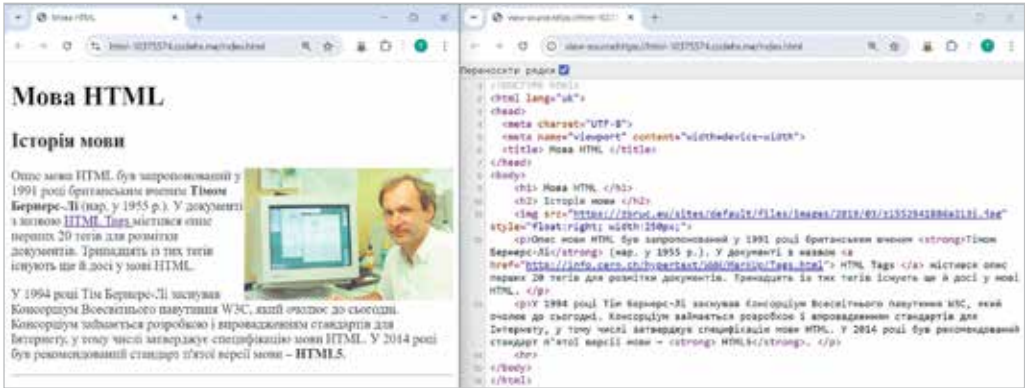
```

<p>
З сайту Юридичний Вісник України
</p>
<h2 style="color: blue;">Малий та великий герби України</h2>
<input type="button" value="Ok">
```



Мал. 5.10. Використання тегів з атрибутами

Переглянути HTML-код будь-якої вебсторінки, відкритої у вікні браузера, можна вибравши в її контекстному меню команду **Переглянути джерело сторінки** або натиснувши сполучення клавіш **Ctrl + U** (мал. 5.11).



Мал. 5.11. Вебсторінка (ліворуч) і її HTML-код (праворуч)

Під час перегляду в браузері HTML-коду сторінки теги, їх атрибути та значення атрибутів виділено різними кольорами, а колір символів тексту чорний. Розглянемо детальніше HTML-код сторінки, наведеної на малюнку 5.11 (табл. 5.1).

Таблиця 5.1

Пояснення тегів HTML-коду сторінки (мал. 5.11)

Номери рядків у HTML-кодi	Теги	Пояснення
1	Непарний тег <b>&lt;!DOCTYPE html&gt;</b> (англ. <i>document type</i> – тип документа)	Указує тип даного документа та версію мови <b>HTML</b> , яку використано. Атрибут <b>html</b> означає, що сторінка створена з використанням актуальної наразі версії мови <b>HTML5</b>
2, 16	Парний тег <b>&lt;html lang="uk"&gt;</b> <b>&lt;/html&gt;</b> (англ. <i>language</i> – мова)	Указує на початок і закінчення HTML-коду сторінки. Атрибут <b>lang</b> зі значенням <b>uk</b> повідомляє, що основна мова в тексті сторінки – українська
3, 7	Парний тег <b>&lt;head&gt;</b> <b>&lt;/head&gt;</b> (англ. <i>head</i> – голова)	Указує на початок і закінчення <b>службової</b> частини коду сторінки з відомостями, потрібними для її правильного відображення браузером
4	Непарний тег <b>&lt;meta charset="UTF-8"&gt;</b> (англ. <i>char set</i> – набір символів)	Указує, яку саме таблицю кодів символів використано в ході створення сторінки. Значення <b>UTF-8</b> атрибута <b>charset</b> означає використання кодової таблиці <b>Юнікод</b>

Номери рядків у HTML-кодi	Теги	Пояснення
5	Непарний тег <b>&lt;meta name="viewport" content="width=device-width"&gt;</b> (англ. <i>viewport</i> – вікно перегляду, <i>content</i> – зміст)	Указує браузеру розміщувати контент сторінки з урахуванням ширини екрана пристрою, який використовується
6	Парний тег <b>&lt;title&gt; Мова HTML &lt;/title&gt;</b> (англ. <i>title</i> – назва)	Указує вивести на вкладці браузера назву сторінки <i>Мова HTML</i>
8, 15	Парний тег <b>&lt;body&gt; &lt;/body&gt;</b> (англ. <i>body</i> – тіло)	Указує на початок і закінчення <b>інформаційної</b> частини, яка містить текст для відображення на сторінці та розмітку
9	Парний тег <b>&lt;h1&gt; Мова HTML &lt;/h1&gt;</b>	Указує відобразити текст <i>Мова HTML</i> як заголовок першого рівня
10	Парний тег <b>&lt;h2&gt; Історія мови &lt;/h2&gt;</b>	Указує відобразити текст <i>Історія мови</i> як заголовок другого рівня
11	Непарний тег <b>&lt;img src = "https://zbruc.eu/sites/default/files/images/2019/03/z1552941886a213i.jpg" style="float: right; width: 250px;"&gt;</b>	Указує вставити на сторінку зображення з файлу з адресою <i>https://zbruc.eu/sites/default/files/images/2019/03/z1552941886a213i.jpg</i> . Стиль зображення (атрибут <b>style</b> ) – вирівнювання по правому краю з обтіканням текстом ( <b>float: right;</b> ), ширина 250 пікселів ( <b>width: 250px;</b> )
12, 13	Парний тег <b>&lt;p&gt; ... &lt;/p&gt;</b>	Указує відобразити текст, що розміщений між частинами парного тегу, як окремий абзац
12, 13	Парний тег <b>&lt;strong&gt; ... &lt;/strong&gt;</b>	Указує відобразити текст, що розміщений між частинами парного тегу, напівжирним накресленням
12	Парний тег <b>&lt;a href = "http://info.cern.ch/hypertext/WWW/MarkUp/Tags.html"&gt; HTML Tags &lt;/a&gt;</b>	Указує створити гіперпосилання для переходу на сторінку з адресою <i>http://info.cern.ch/hypertext/WWW/MarkUp/Tags.html</i> . Для переходу потрібно вибрати текст <i>HTML Tags</i>
14	Непарний тег <b>&lt;hr&gt;</b>	Указує браузеру вставити на сторінку горизонтальну лінію

Теги, розміщені в рядках з 1 по 8 та в 15 і 16, визначають так звану **базову структуру вебсторінки**. Вони використовуються в HTML-кодi кожної вебсторінки з потрібними значеннями атрибутів, щоб браузер правильно відтворювали сторінку за її HTML-кодом.





### Цікаві факти з історії

Опис мови **HTML** був запропонований у 1991 році британським ученим **Тімом Бернерс-Лі** (нар. у 1955 р.) (мал. 5.12). Під його керівництвом були розроблені **концепція Всесвітнього павутиння** (спільно з **Робертом Кайо**), система URL-адрес ресурсів у мережі, створені перший браузер, вебсервер, вебсайт. Завдяки цьому стало можливим існування служби передавання гіпертекстових сторінок **World Wide Web** (англ. *World Wide Web* – всесвітнє павутиння) – **Всесвітнього павутиння**.

У 1994 році Тім Бернерс-Лі заснував **Консорціум Всесвітнього павутиння** (англ. *World Wide Web Consortium*, **W3C** – консорціум всесвітнього павутиння) (<https://www.w3.org>). Консорціум займається розробкою та впровадженням стандартів для різних напрямів веброботи, у тому числі для мови **HTML**.

У 2014 році було рекомендовано до впровадження актуальний на сьогодні стандарт п'ятої версії мови – **HTML5**.



Мал. 5.12.  
Тім Бернерс-Лі

## СТВОРЕННЯ HTML-КОДУ



### Пригадайте

- Як під час збереження текстового документа змінити розширення імені файлу?
- Як у текстовому редакторі під час створення проєктів відомою вам мовою програмування виділяються ключові слова, імена змінних, тексти?

Файли з HTML-кодом можна створювати як текстові файли та зберігати з розширенням імені **html**. Файлу з HTML-кодом головної сторінки сайту зазвичай дають ім'я **index.html**.

Для створення та редагування HTML-коду можна використовувати текстові редактори, наприклад **Блокнот**, **Notepad++**, **Edit Plus** або спеціалізовані редактори коду **Visual Studio Code**, **Sublime Text**, **Atom** та інші.

Також створення HTML-коду можливе на онлайн-ресурсах, наприклад в онлайн-редакторах **Replit** ([replit.com](https://replit.com)), **CodeHS** ([codehs.com](https://codehs.com)), **JS Bin** ([jsbin.com](https://jsbin.com)) та інших. Під час створення HTML-коду в онлайн-редакторі можна переглядати на тій самій вкладці браузера, як виглядатиме вебсторінка.

Створення HTML-коду розпочинається з уведення базової структури вебсторінки з потрібними значеннями атрибутів. Після цього в інформаційній частині коду, що обмежена частинами парного тегу з іменем **body**, вводять текст і теги для відображення на сторінці контенту відповідно до дизайн-макета. Об'єкти на сторінці будуть розташовані в порядку їх опису в HTML-кодi.

У багатьох редакторах є засоби для спрощення процесу написання HTML-коду: підсвітка синтаксису, розпізнавання введених тегів і автоматичне їх доповнення, вставлення готових фрагментів HTML-коду тощо.

Виконання HTML-коду відбувається під час відкривання відповідних файлів у вікні браузера. Якщо файли з HTML-кодом розміщено на носіях даних вашого комп'ютера, то вони не будуть доступні для перегляду користувачам Інтернету до розміщення цих файлів на сервері та їх публікації. В онлайн-редакторах зазвичай існують спеціальні засоби для публікації створених сторінок.

### Для тих, хто хоче знати більше

Стандарт останньої версії мови **HTML5** містить понад 100 тегів. Крім того, під час написання HTML-коду можуть бути використані окремі теги з попередніх версій мови, хоча вони вважатимуться застарілими. Особливо це стосується тегів і атрибутів для форматування об'єктів сторінки: установлення кольору символів і фону, вирівнювання, розмірів об'єктів тощо.

Замість тегів мови **HTML** для форматування об'єктів на сторінці використовують **мову опису стилів CSS** (англ. *Cascading Style Sheets* – каскадні таблиці стилів).

Ознайомитися з тегами мови **HTML** та особливостями мови **CSS** можна на сайті **W3SchoolsUA українською** (<https://w3schoolsua.github.io>).

### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.


1. Відкрийте файл **Розділ 5\Пункт 5.2\вправа 5.2.txt**. Переконайтеся, що він відкрився у вікні текстового редактора.
2. Збережіть відкритий текстовий документ у вашій папці у файлі з іменем **вправа 5.2.html**. Розширення імені файлу **.html** уведіть з клавіатури.
3. Відкрийте у **Провіднику** вікно вашої папки, не закриваючи вікна текстового редактора, та відкрийте файл **вправа 5.2.html**. Переконайтеся, що він відкрився у вікні браузера.
4. Проаналізуйте вміст файлу в текстовому редакторі та вигляд вебсторінки у браузері. Заповніть у зошиті таблицю з відповідями на запитання. Для кожного запитання вкажіть ім'я тегу, який укажує на відповідь.

Запитання	Відповідь	Ім'я тегу
Яка назва сторінки відображена на вкладці браузера?		
Який текст у заголовку першого рівня?		
Які тексти у заголовках другого рівня?		
Скільки абзаців на сторінці?		
Яка адреса файлу, зображення з якого виведено на сторінку?		
На сторінку з якою адресою відбудеться перехід після вибору гіперпосилання?		

5. Унесіть зміни до HTML-коду для виведення слів *Ергономіка*, *Композиція*, *Колористика* у відповідних абзацах напівжирним накресленням. Для цього:
  1. Знайдіть у вікні текстового редактора рядок з тегом створення абзацу **<p>**, у якому пояснюється термін *Ергономіка*.
  2. Обмежте слово *Ергономіка* частинами парного тегу з іменем **strong**. Для цього введіть перед словом *Ергономіка* відкриваючу частину тегу, після слова – закриваючу частину:
 

```
<p>Ергономіка
```

 – це наука, ...

3. Знайдіть у наступному абзаці слово *Композиція* та обмежте його частинами тегу **strong**.
4. Знайдіть у наступному абзаці слово *Колористика* та обмежте його частинами тегу **strong**.
6. Збережіть змінений файл у тій самій папці з тим самим іменем **вправа 5.2.html**.
7. Перейдіть у вікно браузера. Оновіть відкриту сторінку, вибравши кнопку **Перезавантажити цю сторінку**  або натиснувши клавішу **F5**. Переконайтеся у виведенні слів на сторінці в напівжирному накресленні.
8. Перейдіть у вікно текстового редактора.
9. Уведіть атрибут **style** із значенням **color: red** у тег заголовка першого рівня для виведення символів тексту заголовка червоним кольором:  
`<h1 style="color: red"> Розробка дизайну сторінок </h1>`
10. Уведіть тег `<hr>` у вільному рядку перед закриваючою частиною тегу `</body>` для виведення горизонтальної лінії під останнім абзацом на сторінці.
11. Уведіть у новому рядку після тегу `<hr>` ваше прізвище та ім'я та обмежте їх частинами парного тегу з іменем **h3** для виведення вашого прізвища та імені під горизонтальною лінією як заголовка третього рівня:  
`<h3> Прізвище ім'я </h3>`
12. Збережіть змінений файл у тій самій папці з тим самим іменем.
13. Оновіть сторінку у вікні браузера. Переконайтеся, що внесені зміни відобразилися на сторінці.
14. Закрийте всі відкриті вікна.



#### Найважливіше в цьому пункті

**HTML** – це мова, якою описують структуру та контент вебсторінки для відображення цієї сторінки у вікні браузера.

Кожна вебсторінка має свій **HTML-код**. На основі HTML-коду браузер розміщує у вікні об'єкти вебсторінки: заголовки, абзаци тексту, гіперпосилання, списки, таблиці, зображення, відео тощо.

Елементами мови **HTML** є **теги**, які вказують браузеру, які об'єкти розмістити на сторінці та як їх відобразити. Імена тегів записують усередині кутових дужок `<` та `>`.

Теги можуть бути **непарними** або **парними**. Найчастіше непарні теги вказують браузеру вставити певний об'єкт на сторінку. Парні теги складаються з **відкриваючої** та **закриваючої** частин, які обмежують фрагмент тексту або HTML-коду. У закриваючій частині перед іменем тегу ставлять символ `/`.

Теги можуть містити **атрибути**, які уточнюють дію цих тегів. У атрибутів можуть бути **значення**. Атрибути записують після імені тегу, відокремлюючи пропуском. Значення атрибутів записують у лапках після знаку `=`.

Файли з HTML-кодом можна створювати як текстові файли та зберігати з розширенням імені **html**. Файлу з HTML-кодом головної сторінки сайту зазвичай дають ім'я **index.html**.

Для створення та редагування HTML-коду можна використовувати текстові редактори або спеціалізовані редактори коду. Також створення HTML-коду можливе на онлайн-ресурсах. Виконання HTML-коду відбувається після відкриття відповідних файлів у вікні браузера.





### Дайте відповіді на запитання

1. Що визначає HTML-код сторінки?
2. Що таке *тег*? Як записують теги? Як записують парні теги? Як записують теги з атрибутами?
3. Який тег призначено для вставлення зображення з файлу на вебсторінку?
4. Який тег призначено для створення гіперпосилання?
5. Як можна переглянути HTML-код сторінки?



### Обговоріть і зробіть висновки

1. З чим може бути пов'язана поява нових версій мови HTML?
2. Чи може бути доступна для перегляду користувачами Інтернету ваша сторінка, створена засобами текстового редактора, під час розробки сторінки? Чи може бути доступна для перегляду сторінка, що створюється в онлайн-редакторі?



### Виконайте завдання

1. Відкрийте у вікні браузера сторінку з адресою *kievoi.ippo.kubg.edu.ua*. Проаналізуйте HTML-код цієї вебсторінки. Відкрийте файл **Розділ 5\Пункт 5.2\завдання 5.2.1.docx**. Заповніть у файлі таблицю відповідями на запитання щодо вмісту HTML-коду сторінки. Для кожного запитання вкажіть ім'я тегу, який указує на відповідь. Збережіть файл у вашій папці з тим самим іменем.

2. Відкрийте файл **Розділ 5\Пункт 5.2\завдання 5.2.2.txt**. Збережіть файл у вашій папці з іменем **завдання 5.2.2.html**. Упишіть у HTML-код текст і теги для створення на сторінці заголовка першого рівня з назвою вашого населеного пункту, горизонтальної лінії, посилання на сайт адміністрації вашого населеного пункту в Інтернеті. Збережіть змінений файл у тій самій папці з тим самим іменем. Перегляньте його вміст у вікні браузера. Перевірте правильність гіперпосилання.



3. Відкрийте файл **Розділ 5\Пункт 5.2\завдання 5.2.3.txt**. Збережіть файл у вашій папці з іменем **завдання 5.2.3.html**. Перегляньте вміст збереженого файлу у вікні браузера. Змініть HTML-код таким чином, щоб:

- у заголовку браузера відображалось ваше прізвище;
- у заголовку першого рівня відображалось ваше прізвище та ім'я;
- у заголовку другого рівня відображалась назва вашого закладу освіти;
- слова *Українська Вікіпедія* виводились у напівжирному накресленні;
- останнє гіперпосилання вказувало на головну сторінку **Вікіпедії**;
- горизонтальна лінія розмістилась під гіперпосиланням на **Вікіпедію**.

Збережіть змінений файл у тій самій папці з тим самим іменем. Перегляньте його вміст у вікні браузера. Перевірте правильність гіперпосилання.

4. Створіть вебсторінку за зразком (мал. 5.11) та описом (табл. 5.1). Текст та URL-адреси ресурсів містяться у файлі **Розділ 5\Пункт 5.2\завдання 5.2.4.txt**. Збережіть створену сторінку у вашій папці у файлі з іменем **завдання 5.2.4.html**.



-  **5.** Знайдіть в Інтернеті відомості про створення таблиць засобами мови **HTML**. Запишіть у зошит імена тегів, що вказують створити таблицю, заголовок таблиці, рядок таблиці, клітинку таблиці.
-  **6.** Знайдіть в Інтернеті відомості про створення списків засобами мови **HTML**. Запишіть у зошит імена тегів, що вказують створити нумерований список, список з маркерами, елемент списку. Створіть вебсторінку, що містить нумерований список з назвами днів тижня та список з маркерами з назвами пір року. Збережіть створену сторінку у вашій папці у файлі з іменем завдання **5.2.6.html**.



### Готуємось до вивчення нового матеріалу

1. Які етапи розробки вебсайту?
2. Які ви знаєте сучасні тренди вебдизайну?
3. Документи яких типів можуть бути створені з використанням сервісів **Google**?

## 5.3. АВТОМАТИЗОВАНІ ЗАСОБИ СТВОРЕННЯ ТА ПУБЛІКАЦІЇ ВЕБСАЙТІВ

У цьому пункті йтиметься про:

- ▶ автоматизовані засоби створення вебсайтів;
- ▶ створення сайтів у сервісі **Google Сайти**;
- ▶ вибір теми оформлення сайту;
- ▶ створення вебсторінок і додавання елементів на сторінки;
- ▶ попередній перегляд та публікацію сайту.

## АВТОМАТИЗОВАНІ ЗАСОБИ СТВОРЕННЯ ВЕБСАЙТІВ



### Пригадайте

- Які етапи створення сайту? У чому вони полягають?
- Які програми можна використати для створення HTML-коду вебсторінок?
- Із чого складається та як створюється HTML-код вебсторінок?

Як ви знаєте, створення сторінок сайту розпочинається після визначення його структури та створення дизайн-макета. Для створення сайту можна самостійно вводити HTML-код його сторінок у текстовому редакторі або використати автоматизовані засоби створення та публікації вебсайтів. У таких засобів існують шаблони дизайнерського оформлення, а також варіанти структури сайтів.

**Спеціалізовані вебредактори** – це програми, що призначені для візуального конструювання вебсторінок, наприклад **Adobe Dreamweaver**, **Google Web Designer**, **WYSIWYG Web Builder** та інші. Розробник розміщує в окремому документі шаблони окремих елементів сторінки – заголовків, зображень, таблиць, списків, гіперпосилань та інших, вибираючи їх з бібліотеки об'єктів, заповнює шаблони потрібним умістом, а HTML-код сторінки генерується автоматично. Такі програми називають **WYSIWYG-редакторами** (англ. *What You See Is What You Get* – що ви бачите, то ви й отримуєте).

Іншим засобом створення сайтів є **системи управління вебконтентом WCMS** (англ. *Web Content Management System* – система управління вебконтентом) – програми, що призначені для створення, редагу-

вання та керування вмістом вебсайту. Популярними **WCMS** є системи **Joomla**, **WordPress**, **MediaWiki** та інші. Вони містять набори готових шаблонів вебсторінок і модулів, що роблять сайт динамічним: форумів, стрічок новин, каталогів файлів тощо. Контент сторінок, створених у цих системах, зберігається в базі даних, а HTML-код генерується автоматично на основі вибраного шаблону.

Спеціалізовані вебредактори та системи керування вебконтентом можна встановити на локальному комп'ютері. Створені в їх середовищі файли потрібно самостійно розміщувати в Інтернеті.

Існують також **онлайн-сервіси конструювання сайтів**, які призначені для розробки сайтів безпосередньо в Інтернеті. Такими є сервіси **Google Сайти** (*sites.google.com*), **WIX** (*uk.wix.com*), **Weblium** (*ua.weblium.com*) та інші. Створені в них на основі готових шаблонів сайти автоматично розміщуються на серверах Інтернету, але розробник зазвичай не може безпосередньо редагувати HTML-код вебсторінок.

Створення вебсайту засобами онлайн-сервісів конструювання сайтів відбувається в кілька кроків:

1. Реєстрація облікового запису на сервері онлайн-сервісу конструювання сайтів.
2. Вибір шаблону сайту.
3. Створення та редагування сторінок сайту, заповнення їх контентом.
4. Публікація сайту в Інтернеті.

## СТВОРЕННЯ САЙТУ В СЕРВІСІ GOOGLE САЙТИ





### Пригадайте

● Які сервіси **Google** ви використовували? З якою метою? ● Як і на якому носії даних відбувається збереження електронних документів у сервісах **Google**? ● Які переваги надає використання сервісів **Google** порівняно з використанням локальних програмних засобів? Які є недоліки їх використання?

Розглянемо, як відбувається створення сайту в сервісі **Google Сайти**. Для його використання потрібно мати обліковий запис **Google**.

Щоб створити сайт, слід:

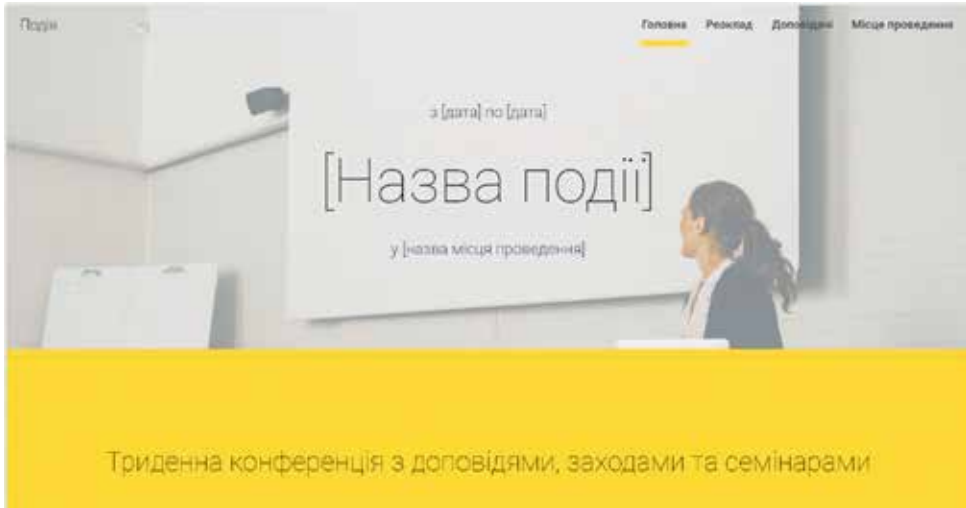
1. Відкрити у вікні браузера сторінку **Google** (*google.com*) та ввести дані власного облікового запису.
2. Вибрати у списку кнопки **Додатки Google**  додаток **Сайти**  або відкрити сторінку з адресою *https://sites.google.com*.
3. У вікні **Сайти** вибрати шаблон, на основі якого буде створено новий сайт (мал. 5.13).



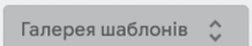
Мал. 5.13. Список шаблонів сайтів на сторінці **Сайти**



Усі шаблони, за виключенням шаблону **Порожній сайт**, містять кілька сторінок, на яких вже розміщено панелі навігації, елементи оформлення та рекомендації щодо заповнення сторінок контентом. На малюнку 5.14 наведено вигляд головної сторінки шаблону *Подія*.




Мал. 5.14. Головна сторінка шаблону *Подія*

Більше шаблонів можна побачити, якщо відкрити **Галерею шаблонів**, вибравши відповідну кнопку  у вікні **Сайти**.

Після вибору шаблону подальша робота буде полягати в заповненні його контентом відповідно до рекомендацій, за потребою – змінненні вмісту та оформлення, додаванні або вилученні сторінок, публікації готового сайту.

Якщо ви не знайшли шаблон, який вас задовольняє, то ви можете створити сайт на основі шаблону **Порожній сайт**. Для цього можна скористатися одним із способів:

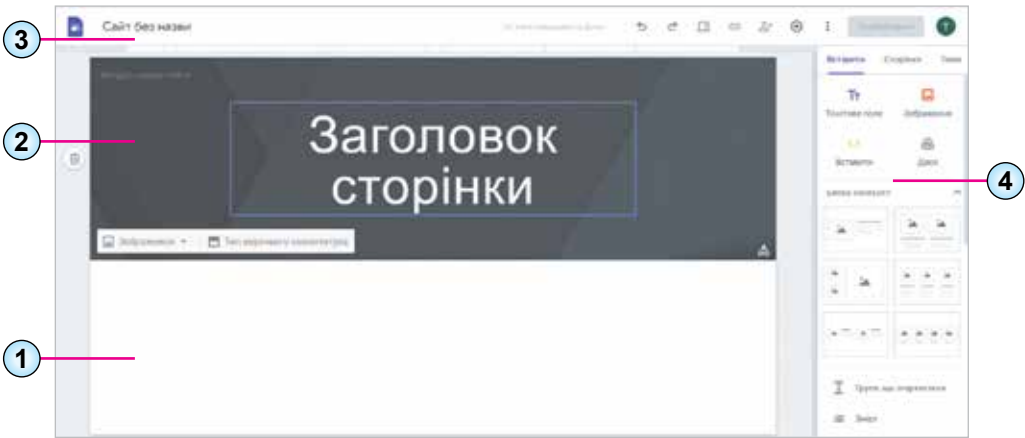
- вибрати у вікні **Сайти** шаблон **Порожній сайт**;
- вибрати в нижньому правому куті вікна **Сайти** кнопку **Створити новий сайт** ;
- відкрити вікно **Google Диска** та виконати **Створити** ⇒ **Більше** ⇒ **Google Сайти**.

Незалежно від обраного способу у вікні браузера буде відкрито редактор сайтів **Google** з однією головною сторінкою нового сайту в режимі редагування (мал. 5.15).

**Верхня панель** редактора містить назву сайту та інструменти для роботи із сайтом.

Основну частину вікна займає **робоча область**, у якій розміщено:

- **верхній колонтитул** вебсторінки з фоновим зображенням, назвою (заголовком) сторінки та назвою сайту;
- **робоче поле** сторінки для розміщення текстових і нетекстових елементів.



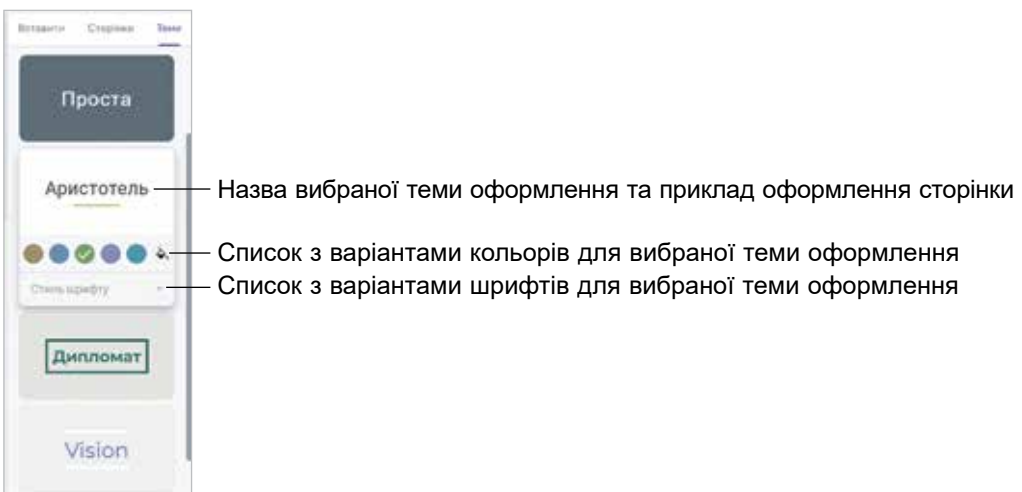
Мал. 5.15. Вікно редактора сайтів **Google** з головною сторінкою нового сайту: 1 – робоче поле вебсторінки; 2 – верхній колонтитул вебсторінки; 3 – верхня панель редактора сайтів; 4 – бічна панель редактора сайтів

Справа від робочої області сторінки розміщена **бічна панель** редактора сайтів з вкладками **Вставити**, **Сторінки**, **Теми**. Інструменти цієї панелі використовують для додавання, видалення та редагування сторінок, вставлення об'єктів на сторінки, вибору та змінення оформлення тощо.

Кожен сайт має назву, яка відображає його вміст або призначення. Щоб задати назву сайту, її потрібно ввести в поле **Введіть назву сайту** у верхньому колонтитулі головної сторінки.

Для змінення теми оформлення сайту слід:

1. Вибрати вкладку **Теми** на бічній панелі.
2. Вибрати назву теми оформлення.
3. Вибрати кольорову гаму та стиль шрифту з тих, що запропоновані для вибраної теми (мал. 5.16).




Мал. 5.16. Вкладка **Теми** редактора сайтів **Google**



Вигляд верхнього колонтитула сторінки можна змінювати. Після наведення вказівника на верхній колонтитул з'являється панель з командами **Зображення** та **Тип верхнього колонтитула**. Після вибору першої команди можна додати зображення для верхнього колонтитула з носіїв даних комп'ютера або вибрати з розміщених в Інтернеті. Вибір другої команди дає можливість змінити тип колонтитула – *Обкладинка*, *Великий банер*, *Банер* або *Лише заголовок*. Верхні колонтитули на різних сторінках сайту можуть відрізнятися.

Також на всіх сторінках можна розмістити нижній колонтитул. Для цього потрібно підвести вказівник до нижньої межі сторінки та вибрати кнопку **Додати нижній колонтитул**. У поле, що з'явиться, потрібно ввести дані, наприклад, рік створення або оновлення сайту, відомості про його розробника та інші. Нижній колонтитул буде автоматично повторюватися на всіх сторінках сайту.

У процесі створення файл з проектом сайту автоматично зберігається на **Google Диску**. Ім'я файлу збігається з назвою сайту, значок файлу має вигляд . Цей файл можна відкривати в подальшому для редагування сайту. Також у вікні **Сайти** з'являється ескіз цього сайту, використовуючи який також можна відкрити сайт у режимі редагування.

## СТВОРЕННЯ ВЕБСТОРОІНОК У РЕДАКТОРІ САЙТІВ GOOGLE





### Поміркуйте

- Від чого залежить кількість вебсторінок, з яких складається сайт?
- Чи можуть бути однаковими назви кількох вебсторінок одного сайту?
- Для чого вебсторінки сайту розподіляють за окремими розділами?

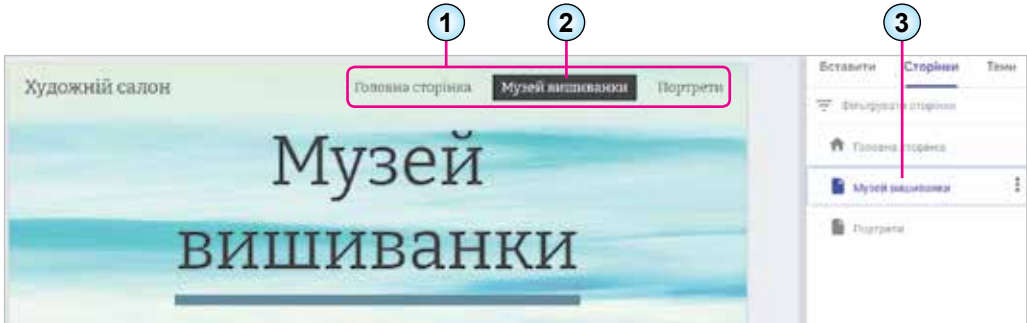
Створення вебсторінок сайту та його тематичних розділів здійснюється відповідно до мапи сайту, яка розроблена на етапі визначення структури сайту.

Одразу сайт, після створення його на основі шаблону **Порожній сайт**, містить лише одну сторінку, яка є *головною сторінкою сайту*. Назву головної сторінки потрібно ввести в поле **Заголовок сторінки** у верхньому колонтитулі. Часто назва головної сторінки збігається з назвою сайту.

Для створення нової сторінки на сайті слід:



1. Відкрити на бічній панелі вкладку **Сторінки**.
2. Навести вказівник на кнопку  в нижній частині бічної панелі.
3. Вибрати кнопку **Створити сторінку** .
4. Увести назву сторінки в поле **Назва** у вікні **Створити сторінку**.
5. Вибрати посилання **Готово**.

Після цього на вкладці **Сторінки** бічної панелі з'явиться блок з назвою створеної сторінки, а на панелі навігації сайту буде утворено посилання на цю сторінку (мал. 5.17).



Мал. 5.17. Панель навігації та вкладка **Сторінки** бічної панелі:  
 1 – панель навігації сайту; 2 – посилання на створену сторінку;  
 3 – блок створеної сторінки на бічній панелі


У структурі сайту можуть бути тематичні розділи, у яких згруповано всі сторінки певної теми. Для створення тематичного розділу слід:

1. Відкрити на бічній панелі вкладку **Сторінки**.
2. Навести вказівник на кнопку  в нижній частині бічної панелі.
3. Вибрати кнопку **Розділ нового меню** .
4. Увести назву розділу в поле **Назва** у вікні **Розділ нового меню**.
5. Вибрати посилання **Готово**.

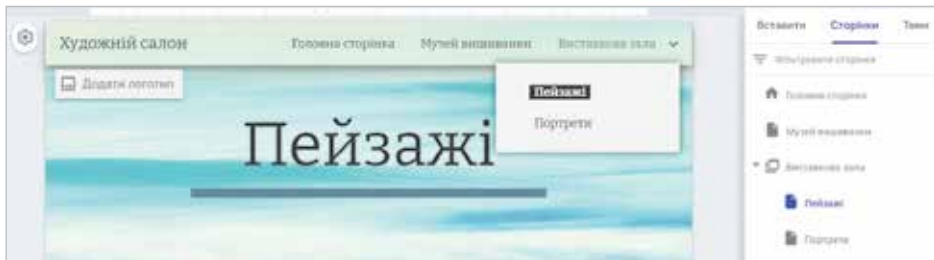
На бічній панелі блок тематичного розділу має вигляд



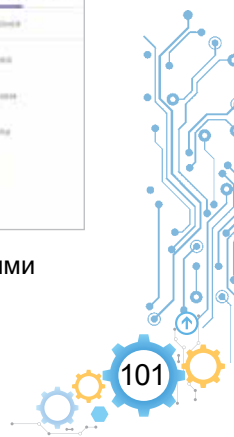
На панелі навігації посилання на розділ буде активним, якщо до розділу входить хоча б одна сторінка. Щоб створити в розділі сторінку, слід:

1. Вибрати кнопку  у блоці розділу.
2. Вибрати в меню, що відкриється, команду **Додати підсторінку**.
3. Увести назву сторінки в поле **Назва** у вікні **Нова підсторінка**.
4. Вибрати посилання **Готово**.



На малюнку 5.18 наведено вигляд вкладки **Сторінки** та панелі навігації сайту з посиланнями на сторінки тематичного розділу.

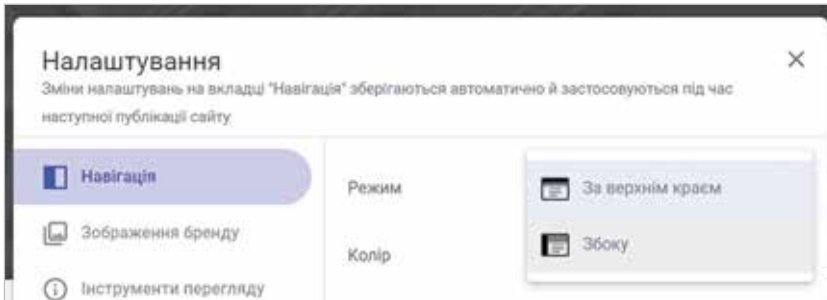


Мал. 5.18. Вкладка **Сторінки** та панель навігації сайту з посиланнями на сторінки тематичного розділу



Розміщення посилань на панелі навігації можна змінити, перетягнувши на вкладці **Сторінки** бічної панелі блок з назвою сторінки в потрібне місце.

У деяких шаблонах сайтів панель навігації згорнута у значок так званого **бургер-меню** , після вибору якого список сторінок відкривається в лівій бічній панелі. Змінити вигляд і положення панелі навігації можна, вибравши кнопку **Налаштування навігації**  ліворуч від верхнього колонтитула та у вікні **Налаштування** в розділі **Навігація** – відповідний режим: **За верхнім краєм** або **Збоку** (мал. 5.19).



Мал. 5.19. Вибір місця розташування панелі навігації

## ВСТАВЛЕННЯ ОБ'ЄКТІВ НА ВЕБСТОРІНКУ



### Пригадайте



- Які об'єкти можуть бути вставлені на вебсторінку?
- Що таке *макет слайда комп'ютерної презентації*?
- Як уставити об'єкт на слайд комп'ютерної презентації, використовуючи покажчик місця заповнення?

На вебсторінку можна вставити різні об'єкти: текстові поля, зображення, відео, документи, які розміщено на **Google Диску**, та інше. Для цього призначено вкладку **Вставити** на правій бічній панелі редактора сайтів.








Призначення деяких елементів керування вкладки **Вставити** наведено в таблиці 5.2.

Таблиця 5.2

### Призначення деяких елементів керування вкладки **Вставити**

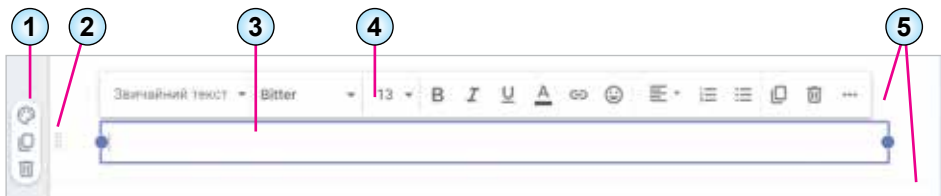
Елемент керування	Призначення
 Текстове поле	Уставлення на сторінку текстового поля
 Зображення	Уставлення на сторінку зображення з <b>Google Диска</b> , за URL-адресою з Інтернету, з носіїв даних комп'ютера тощо



Елемент керування	Призначення
	Уставлення об'єктів різних типів з Інтернету за їх URL-адресою або вставлення фрагментів HTML-коду
	Уставлення документів різних типів з <b>Google Диска</b>
	Вибір макета розміщення об'єктів на сторінці: у два, три, чотири стовпці та інші
	Уставлення змісту для швидкого переходу до вибраного заголовка на поточній сторінці, який автоматично формується із заголовків у текстових полях
	Уставлення слайд-шоу з вибраних зображень
	Уставлення відео з сервісу <b>YouTube</b>
	Уставлення <b>Google Форми</b>

Також на вебсторінку можна вставити **Google Карти**, **Google Календарі**, кнопки з посиланнями на вебсторінки, посилання на сторінки соціальних мереж у вигляді відповідних значків, діаграми з електронних таблиць та інші об'єкти.




Під час вставлення об'єкта на сторінці утворюється окремий її розділ завширшки на всю сторінку. Уставлений об'єкт розміщується всередині цього розділу сторінки у прямокутному блоці з маркерами на межі блока. На малюнку 5.20 наведено вигляд розділу сторінки, у якому міститься блок текстового поля.



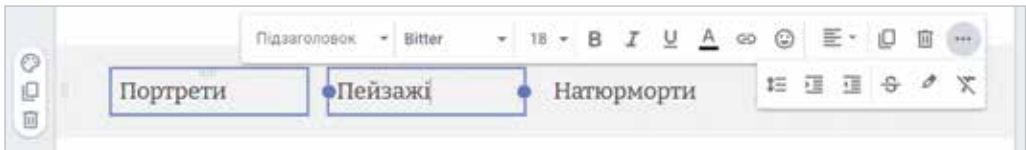
Мал. 5.20. Розділ сторінки з блоком текстового поля:

- 1 – панель інструментів розділу сторінки; 2 – кнопка для переміщення розділу сторінки; 3 – блок текстового поля; 4 – панель інструментів блока текстового поля; 5 – верхня та нижня межа розділу сторінки

Кожний розділ сторінки має панель інструментів, призначенням яких є:

-  – вибір кольору фону для розділу сторінки;
-  – створення копії розділу сторінки;
-  – видалення розділу сторінки.


Блок з об'єктом можна перемістити всередині розділу сторінки або в інший її розділ, змінити розміри блока, використовуючи маркери на межах, або видалити. На малюнку 5.21 наведено вигляд розділу сторінки з трьома блоками текстових полів.



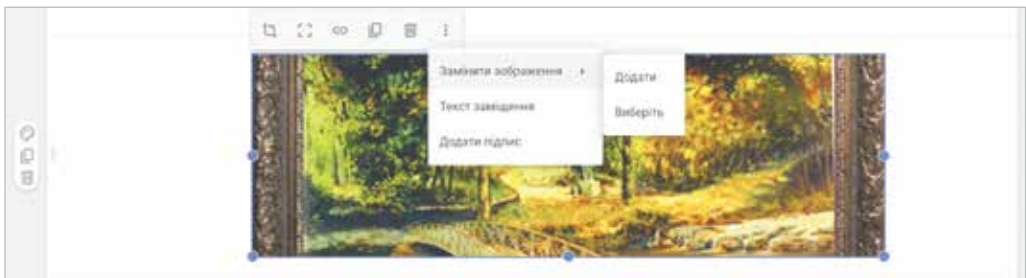
Мал. 5.21. Розділ сторінки з трьома блоками текстових полів

Для кожного блока, якщо його вибрати, відкривається окрема панель інструментів.



На панелі інструментів текстового поля (мал. 5.21) розміщено елементи керування, подібні до тих, що використовуються в текстовому процесорі. До тексту, уведеного в текстовому полі, може бути застосований один зі стандартних стилів – *Назва*, *Заголовок*, *Підзаголовок*, *Звичайний текст* і *Малий текст*. Редагування та форматування тексту в текстовому полі відбувається так само, як у текстовому процесорі.


Блок зображення створюється після вибору кнопки  на бічній панелі та вибору файлу, зображення з якого буде відображатися на сторінці. Файл можна додати з носіїв даних вашого комп'ютера, вибрати з вашого **Google Диска** або сервісу **Google Фотографії**, устави за відомою URL-адресою або знайти в **Google Зображеннях**.

Вигляд панелі інструментів блока зображення наведено на малюнку 5.22.

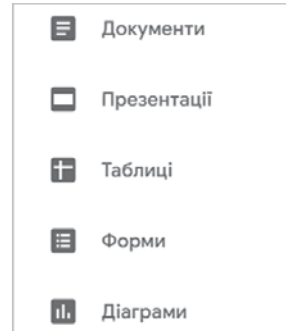


Мал. 5.22. Панель інструментів блока зображення

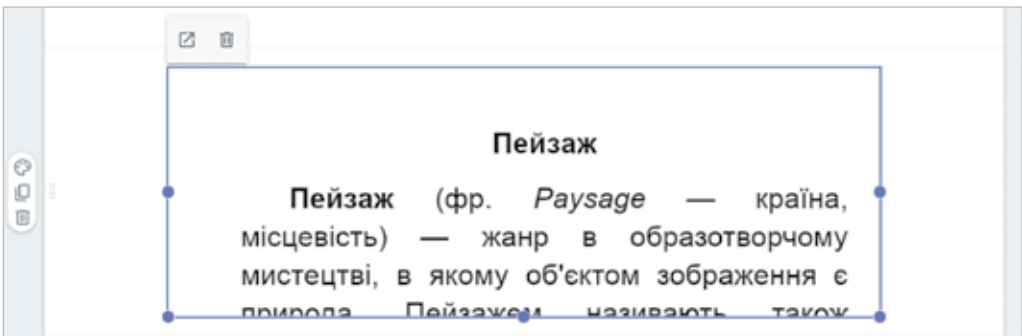
За замовчуванням ширина блока зображення складає третину ширини сторінки. Розміри блока можна змінювати. Використовуючи кнопку панелі інструментів **Обрізати (Вести)** , можна вибрати, яку частину зображення буде відображено у блоці. Вибір кнопки **Скасувати обрізування**  приводить до відображення у блоці всього зображення.

На вебсторінку можна вставляти файли різних типів, які зберігаються на вашому **Google Диску**. Якщо вибрати кнопку  **Диск**, то право-

руч від робочої області сторінки відкриється панель **Google Диск**, на якій можна побачити вміст вашого **Google Диска**, знайти та вибрати файл, який потрібно додати на сторінку. Якщо потрібно вставити файл, створений з використанням одного з сервісів **Google**, то можна скористатися кнопками вкладки **Вставити** для вибору та вставлення відповідного файлу (мал. 5.23). Після вибору будь-якої з цих кнопок відкривається панель з переліком файлів лише вибраного типу на вашому **Google Диску**. Вибраний на цій панелі файл буде вставлений на сторінку в окремому блоці. За потреби буде запитаний ваш дозвіл на перегляд цього файлу користувачами Інтернету. На малюнку 5.24 наведено вигляд блоку зі вставленим Google-документом. Побачити документ у повному обсязі можна буде після публікації сайту.




Мал. 5.23. Кнопки для вставлення файлів, створених у різних сервісах **Google**



Мал. 5.24. Блок Google-документа на сторінці

Якщо на вкладці **Вставити** вибрати будь-який елемент з групи **Блоки контенту**, то на сторінці буде створено розділ відповідного макета. У цьому розділі на визначених місцях можна вводити заголовки та абзаци тексту, вставляти зображення, документи різних типів, відео з сервісу **YouTube**, **Google Календарі** та **Google Карти**.



Для видалення вставленого об'єкта потрібно вибрати кнопку **Видалити**  на панелі інструментів блока цього об'єкта.



## ПОПЕРЕДНІЙ ПЕРЕГЛЯД І ПУБЛІКАЦІЯ САЙТУ

**Пригадайте**

- Який дизайн називають *адаптивним*?
- Які зміни в розміщенні та вигляді об'єктів вебсторінки можуть відбуватися під час змінення розміру області перегляду?
- Що таке *URL-адреса* сторінки?


Вебсторінки, що створюються в редакторі сайтів **Google**, мають адаптивний дизайн. Щоб побачити, як виглядатиме сайт для користувачів Інтернету під час перегляду його на екрані смартфона, планшетного комп'ютера, ноутбука чи стаціонарного комп'ютера, можна вибрати кнопку **Попередній вигляд**  на верхній панелі редактора сайтів і відповідну кнопку (*Телефон, Планшет, Великий екран*) на панелі керування попереднім переглядом (мал. 5.25), яка розміщується в нижньому правому куті екрана. Вийти з режиму попереднього перегляду можна вибором кнопки .



Мал. 5.25. Панель керування попереднім переглядом сайту

Створений вами сайт буде залишатися недоступним для користувачів Інтернету, поки ви його не опублікуєте. Для публікації потрібно:

1. Вибрати кнопку **Опублікувати** на верхній панелі редактора сайтів.
2. Увести, використовуючи малі літери англійського алфавіту, цифри та дефіс, останню частину URL-адреси сайту. Перша частина адреси для всіх сайтів буде однаковою: *sites.google.com/view/*. Остання частина повинна бути унікальною для кожного сайту. Якщо введена вами адреса вже існує, то про це буде повідомлено та її потрібно змінити.
3. Вибрати кнопку **Опублікувати**.

Опублікований сайт можна переглянути, вибравши команду **Переглянути опублікований сайт** у списку кнопки **Опублікувати**. URL-адреса, яку потрібно повідомити користувачам Інтернету для перегляду вашого сайту, міститься в рядку адреси у вікні перегляду опублікованого сайту. Також цю адресу можна скопіювати, вибравши кнопку **Копіювати посилання на опублікований сайт**  на панелі інструментів редактора сайтів.

Після внесення будь-яких змін до вашого сайту його потрібно повторно публікувати. Уведення URL-адреси сайту буде вимагатися лише під час першої публікації сайту.

Як і до інших електронних документів, створених у різних сервісах **Google**, до сайту можна надавати дозвіл іншим людям для спільного редагування. Для цього призначена кнопка **Надати доступ іншим**



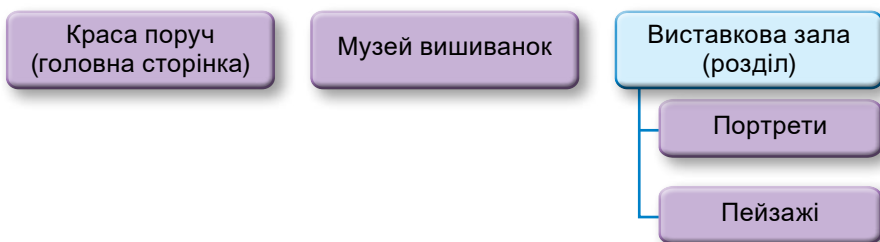
, яка розміщена на верхній панелі редактора сайтів.




## Працюємо з комп'ютером

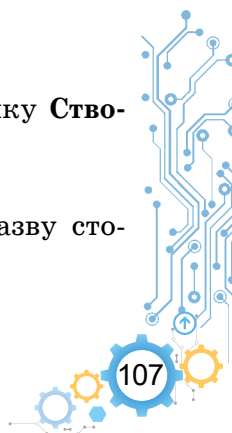
**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.




1. Створіть у сервісі **Google Сайти** сайт *Художній салон* на основі шаблону **Порожній сайт**. Для цього:
  1. Відкрийте свій **Google Диск**, указавши дані вашого облікового запису **Google**.
  2. Виконайте **Створити** ⇒ **Більше** ⇒ **Google Сайти**.
  3. Уведіть у поле **Введіть назву сайту** назву сайту *Художній салон*.
  4. Уведіть у поле **Заголовок сторінки** назву головної сторінки *Краса поруч*.
2. Виберіть для сайту тему оформлення *Арістотель*, змініть зображення у верхньому колонтитулі, створіть нижній колонтитул. Для цього:
  1. Виберіть вкладку **Теми** на бічній панелі редактора сайтів.
  2. Виберіть тему *Арістотель*, колір – *сірий*, стиль шрифту – *Класичний*.
  3. Наведіть вказівник на верхній колонтитул і виконайте **Зображення** ⇒ **Виберіть**.
  4. Виберіть одне з зображень галереї та кнопку **Додати**.
  5. Підведіть вказівник до нижньої межі сторінки та виберіть кнопку **Додати нижній колонтитул**.
  6. Уведіть у поле поточний рік, ваше прізвище та ім'я.
  7. Виберіть будь-яку точку на сторінці вище нижнього колонтитула.
3. Створіть сторінки та розділ сайту відповідно до наведеної мапи сайту (мал. 1). Для цього:

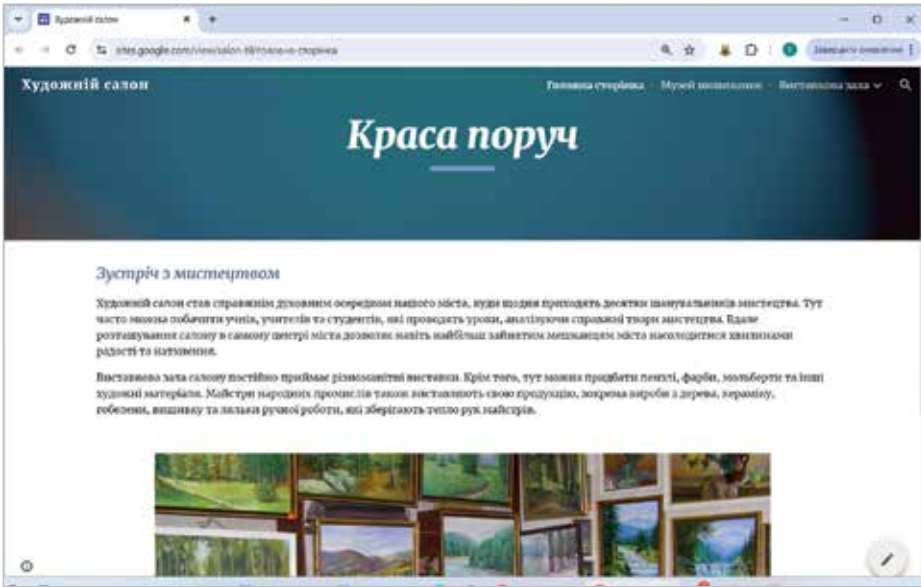


Мал. 1. Мапа сайту *Художній салон*

1. Виберіть вкладку **Сторінки** на бічній панелі.
2. Створіть сторінку *Музей вишиванок*. Для цього:
  1. Наведіть вказівник на кнопку **+**, виберіть кнопку **Створити сторінку** .
  2. Уведіть в поле **Назва вікна** **Створити сторінку** назву сторінки *Музей вишиванок*.
  3. Виберіть посилання **Готово**.



3. Створіть розділ *Виставкова зала*. Для цього:
  1. Наведіть вказівник на кнопку , виберіть кнопку **Розділ нового меню** .
  2. Уведіть у поле **Назва** вікна **Розділ нового меню** назву розділу *Виставкова зала*.
  3. Виберіть посилання **Готово**.
4. Створіть у розділі *Виставкова зала* сторінку з назвою *Портрети*. Для цього:
  1. Виберіть кнопку  у блоці розділу *Виставкова зала* на бічній панелі.
  2. Виберіть у меню, що відкриється, команду **Додати підсторінку**.
  3. Уведіть назву сторінки *Портрети* в поле **Назва** у вікні **Нова підсторінка**.
  4. Виберіть посилання **Готово**.
5. Створіть у розділі *Виставкова зала* сторінку з назвою *Пейзажі*.
4. Заповніть сторінки матеріалами з папки **Розділ 5\Пункт 5.3\Вправа 5.3**. Для цього:
  1. Виберіть головну сторінку *Краса поруч*.
  2. Відкрийте на бічній панелі вкладку **Вставити**.
  3. Виберіть кнопку **Текстове поле**.
  4. Скопіюйте з файлу **Розділ 5\Пункт 5.3\Вправа 5.3\вправа 5.3.txt** текст до головної сторінки та вставте в текстове поле на сторінці *Краса поруч*.
  5. Застосуйте до слів *Зустріч з мистецтвом* стиль *Заголовок*. Для цього установіть курсор у рядку з текстом *Зустріч з мистецтвом* і виберіть на панелі інструментів текстового поля у списку **Стилі** стиль *Заголовок*.
  6. Виберіть на вкладці **Вставити** кнопку **Зображення**.
  7. Виберіть команду **Додати** та відкрийте файл **вернісаж.jpg** з папки **Розділ 5\Пункт 5.3\Вправа 5.3**.
  8. Виберіть кнопку **Відкрити**.
  9. Перетягніть блок зображення, розмістивши його по центру в нижній частині сторінки. Зверніть увагу на появу сітки для розміщення блока. На малюнку 2 наведено вигляд головної сторінки сайту після вставлення та заповнення текстового поля та зображення.
  10. Розмістіть фрагменти тексту з файлу **Розділ 5\Пункт 5.3\Вправа 5.3\вправа 5.3.txt** та зображення з папки **Розділ 5\Пункт 5.3\Вправа 5.3** на інших сторінках сайту відповідно до назв сторінок.
5. Опублікуйте сайт. Для цього:
  1. Виберіть кнопку **Опублікувати** у верхній частині вікна браузера.
  2. Уведіть частину адреси *salon-прізвище*, де *прізвище* – ваше прізвище, записане англійською.
  3. Виберіть кнопку **Опублікувати**.
6. Перегляньте опублікований сайт, вибравши команду **Переглянути опублікований сайт** у списку кнопки **Опублікувати**.



Мал. 2. Вигляд головної сторінки сайту

7. Скопіюйте URL-адресу сайту з рядка адреси та надішліть учителю/вчительці інформатики.
8. Закрийте вікно браузера.



### Найважливіше в цьому пункті

Створення сайту може виконуватися з використанням автоматизованих засобів: спеціалізованих вебредакторів, систем управління вебконтентом **WCMS**, онлайн-систем конструювання сайтів.

Створення вебсайту засобами онлайн-систем конструювання сайтів відбувається у кілька кроків: реєстрація облікового запису на сервері онлайн-системи конструювання сайтів, вибір шаблону сайту, створення та редагування сторінок сайту, заповнення їх контентом, публікація сайту в Інтернеті.

Для створення сайту в сервісі **Google Сайти** можна використати один із запропонованих шаблонів з рекомендаціями до вмісту або шаблон **Порожній сайт**. На вебсторінки можна вставляти різні об'єкти: текстові поля, зображення, відео, документи, які розміщено на **Google Диску**, та інше.

Вебсторінки, що створюються в редакторі сайтів **Google**, мають адаптивний дизайн.



### Дайте відповіді на запитання

1. Які існують засоби автоматизованого створення сайтів? Коротко охарактеризуйте їх.
2. Які етапи створення вебсайтів засобами сервісу **Google Сайти**? Поясніть їх.
3. Як змінити вигляд верхнього та створити нижній колонтитули на сторінці?
4. Як вибрати тему оформлення для сайту? Значення яких властивостей можуть бути вибрані разом з темою оформлення?



5. Як створити нову вебсторінку на сайті? Як створити сторінку в тематичному розділі?

6. Які об'єкти можна вставити на вебсторінки сайту, створеного у сервісі **Google Сайти**?



### Обговоріть і зробіть висновки

1. Чому, на вашу думку, потрібно виконувати публікацію сайту після внесення змін до нього?

2. Які, на ваш погляд, переваги та недоліки створення сайтів шляхом написання HTML-коду їх сторінок і шляхом використання онлайн-систем конструювання сайтів?



### Виконайте завдання

1. Створіть у сервісі **Google Сайти** сайт відповідно до мапи, наведеної на малюнку 5.1 (див. с. 113). Надайте сайту назву *Україна*. Виберіть тему оформлення *Дипломат*. Уведіть на кожній сторінці розділу *Загальні відомості* по одному абзацу тексту. Вставте на кожній сторінці розділу *Символіка* по два зображення. Опублікуйте сайт. Повідомте вчителю/вчительці URL-адресу створеного вами сайту.

2. Доберіть назви сторінок і створіть у текстовому процесорі мапу сайту з теми *Рекордсмени живої природи*. Збережіть її у вашій папці у файлі з іменем завдання **5.3.2.docx**. Створіть у сервісі **Google Сайти** сторінки сайту відповідно до розробленої вами мапи. Уведіть на кожній сторінці по одному абзацу тексту, вставте по одному зображенню. Опублікуйте сайт. Повідомте вчителю/вчительці URL-адресу створеного вами сайту.

3. Створіть у сервісі **Google Сайти** сайт *Наші космічні сусіди*. Мапа сайту, текст та зображення для розміщення на сторінках містяться в папці **Розділ 5\Пункт 5.3\Завдання 5.3.3**. Виберіть для сайту тему оформлення *Вражаюча*. Опублікуйте сайт. Повідомте вчителю/вчительці URL-адресу створеного вами сайту.



4. Створіть у сервісі **Google Сайти** сайт про себе на основі шаблону *Портфоліо учня*. Виберіть шаблон у **Галереї шаблонів**

Галерея шаблонів 

, якщо він не відображається на сторінці **Сайти**.

Уведіть текстові дані відповідно до рекомендацій шаблону, заповніть блоки-заповнювачі та сторінки предметів і позашкільних занять відповідно до вашого класу та ваших інтересів. Замініть зображення на власні. Видаліть блоки шаблону, які залишилися не заповненими, або додайте та заповніть нові блоки за потреби. Опублікуйте сайт. Повідомте вчителю/вчительці URL-адресу створеного вами сайту.



5. Об'єднайтесь у групу з трьох осіб. Доберіть матеріали та створіть спільно сайт *Домашні улюбленці* у сервісі **Google Сайти**. Розробіть і розмістіть на сторінках сайту Google-форму для опитування відвідувачів сайту про їх домашніх улюбленців. Уставте відео з сервісу **YouTube** про домашніх тварин. Опублікуйте сайт. Повідомте вчителю/вчительці URL-адресу створеного вами сайту.





### Готуємось до вивчення нового матеріалу

1. Які типи даних ви знаєте? Дані яких типів можуть бути опрацьовані в електронних таблицях?
2. Як утворюються адреси даних в електронних таблицях?
3. Як записують формули в електронних таблицях?



### Оцініть свої знання та вміння

Оцініть свої навчальні досягнення з розділу «Створення та публікація вебресурсів» (початковий, середній, достатній, високий рівень).

- Я розумію та вмію пояснити особливості нелінійного тексту.
- Я знаю та враховую критерії оформлення вебсторінок, їх відповідність сучасним трендам вебдизайну, можу запропонувати власні критерії.
- Я вмію використовувати гіпертекстові документи та створювати гіпертекстові посилання на ресурси Інтернету.
- Я знаю можливості програмних засобів для створення гіпертекстових документів.
- Я вмію використовувати програмні засоби та технології для створення й редагування вебсторінок у різних режимах (онлайн, офлайн).
- Я вмію обирати та комбінувати тексти, зображення, мультимедійні дані для проектування, створення, публікації вебсторінок.
- Я можу визначити відповідність вмісту та вигляду вебсторінки цільовій аудиторії, отримати відгуки користувачів для вдосконалення продукту.

Повторіть той матеріал, який ви знаєте недостатньо.



### Практична робота № 7.

#### «Створення сайту з використанням онлайн-системи конструювання сайтів»

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

1. Створіть у сервісі **Google Сайти** сайт *Винаходи українців у IT-галузі*.
2. Виберіть тему оформлення *Vision*.
3. Створіть і розмістіть вебсторінки на основі такої мапи сайту:



4. Уставте на головній сторінці відео **Історії українського IT** з сервісу **YouTube** (<https://www.youtube.com/watch?v=PG8QoaTG1kc>).
5. Заповніть сторінки текстовими матеріалами, наведеними у файлі **Розділ 5\Пункт 5.3\практична 7.docx**, та зображеннями з Інтернету з наданими URL-адресами файлів. Створіть гіперпосилання на інтернет-ресурси, з яких отримано відомості та фотографії.
6. Створіть нижній колонтитул і уведіть ваші дані як розробника сайту.
7. Опублікуйте сайт з адресою *discovery-прізвище*, де *прізвище* – ваше прізвище, записане англійською.
8. Повідомте вчителю/вчительці URL-адресу створеного вами сайту.

# ОПРАЦЮВАННЯ ДАНИХ В ЕЛЕКТРОННИХ ТАБЛИЦЯХ

У цьому розділі ви отримаєте нові, а також поглибите та розширите свої знання й удосконалисте навички з таких тем:

- ▶ об'єкти електронної таблиці, їх властивості;
- ▶ формати даних в електронних таблицях;
- ▶ абсолютні та мішані посилання, їх модифікація та використання в електронних таблицях;
- ▶ призначення та використання логічних, математичних і статистичних функцій у табличному процесорі;
- ▶ умовне форматування даних в електронних таблицях;
- ▶ сортування та фільтрування даних в електронних таблицях;
- ▶ установлення параметрів сторінки та друкування електронної таблиці;
- ▶ розв'язування задач з різних дисциплін засобами табличного процесора.

## 6.1. ОБ'ЄКТИ ЕЛЕКТРОННОЇ ТАБЛИЦІ, ЇХ ВЛАСТИВОСТІ. ФОРМАТИ ДАНИХ В ЕЛЕКТРОННИХ ТАБЛИЦЯХ. АДРЕСАЦІЯ В ЕЛЕКТРОННИХ ТАБЛИЦЯХ

У цьому пункті йтиметься про:

- ▶ об'єкти електронної таблиці, їх властивості;
- ▶ формати числових даних в електронних таблицях;
- ▶ відносні, абсолютні та мішані посилання у формулах.


### ОБ'ЄКТИ ЕЛЕКТРОННОЇ ТАБЛИЦІ, ЇХ ВЛАСТИВОСТІ



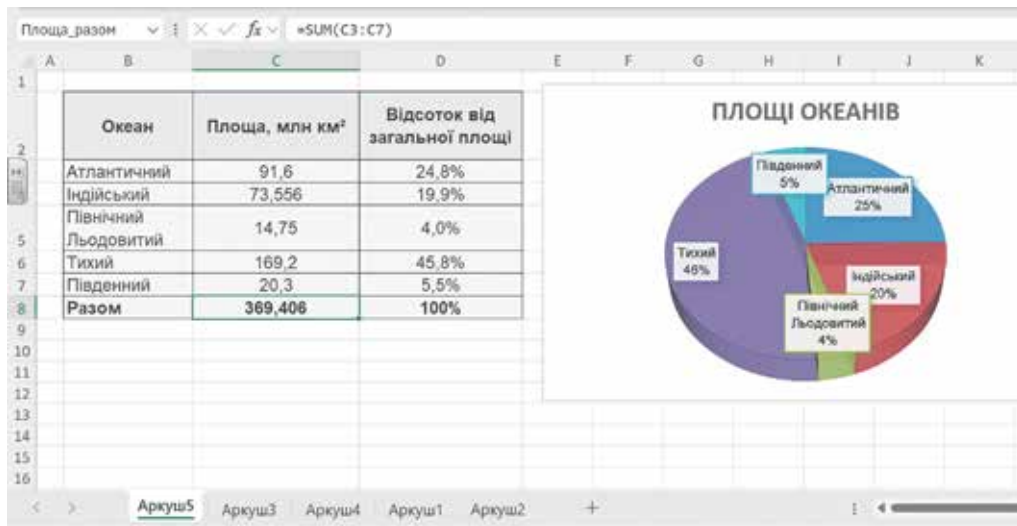
#### Пригадайте

● Що таке *табличний процесор*? Яке його призначення? ● Що таке *електронна таблиця*? Із чого вона складається? Які дані можуть міститися у клітинках електронної таблиці? ● Із чого складається адреса клітинки? Що таке *діапазон клітинок*? Як задається його адреса?

У 6 класі ви почали вивчати та використовувати електронні таблиці для виконання математичних обчислень у табличному процесорі **Microsoft Excel**. У 8 класі ви продовжите вивчення цього програмного забезпечення на прикладі програми **Excel** з пакета програм **Microsoft Office 365**.

Як вам уже відомо, основним об'єктом опрацювання табличного процесора **Microsoft Excel** є **електронна книга**, яка має ім'я, за замовчуванням це ім'я *Книга1*. Користувач може змінити ім'я книги під час її збереження у файлі. Стандартним типом файлу електронної книги є тип **Книга Excel**, а стандартним розширенням імені файлу є **xlsx** (значок .

Складається книга з **аркушів** з іменами **Аркуш1**, **Аркуш2** і так далі (мал. 6.1), які зазначено на **ярликах** аркушів. Користувач за



Мал. 6.1. Електронна книга та її об'єкти

бажання може видаляти аркуші або додавати нові, змінювати їх імена на інші.

На аркуші електронної книги автоматично створюється **електронна таблиця**, яка складається з **рядків** і **стовпців**, на перетині яких утворюється **клітинка**. Кожна клітинка електронної таблиці має **адресу**, яка складається з номера стовпця та рядка, на перетині яких вона розміщена (наприклад, **C5, D2**). У клітинках електронної таблиці можуть зберігатися числа, тексти та формули.

Клітинці можна надати **ім'я**, наприклад, клітинка **C8** на малюнку 6.1 має ім'я *Площа\_разом*. Імена клітинок легше запам'ятовувати, ніж їх адреси. Також їх зручніше використовувати у формулах, оскільки ім'я клітинки не змінюється у формулах під час їх копіювання. Якщо клітинку, яка має ім'я, зробити поточною, то в полі **Ім'я** буде відображатися не адреса клітинки, а її ім'я.

Щоб надати клітинці ім'я, потрібно виділити клітинку, увести її ім'я в поле **Ім'я** і натиснути **Enter**. Водночас використовувати це ім'я клітинки можна буде в усій електронній книзі. *Слід пам'ятати*, що в імені клітинки не можна використовувати пропуски, воно не може розпочинатися із цифри, не може збігатися з адресами клітинок і не може бути клітинок з однаковими іменами.

Для отримання списку всіх імен клітинок електронної таблиці можна виконати **Формули** ⇒ **Визначені імена** ⇒ **Диспетчер імен**.

Деяка сукупність клітинок аркуша електронної книги утворює **діапазон клітинок**, який також має адресу. Адреса діапазону клітинок задається адресами двох клітинок, розташованих у його протилежних кутах і розділених двокрапкою (наприклад, **B2:D2, C3:D7**). Діапазону, як і окремій клітинці, можна надати ім'я.

Увесь рядок та увесь стовпець електронної таблиці також є діапазонами клітинок. Наприклад, адресою діапазону клітинок, до якого входять усі клітинки десятого рядка, є **10:10**, а адресою діапазону

клітинок, до якого входять усі клітинки стовпця **B**, є **B:B**. Відповідно **6:8** – це адреса діапазону клітинок, що включає всі клітинки рядків з номерами **6, 7, 8**, а **H:L** – адреса діапазону клітинок, до якого входять усі клітинки стовпців **H, I, J, K, L**.

На аркуші, крім електронної таблиці, можуть розміщуватися й інші об'єкти: діаграми, малюнки тощо (мал. 6.1). *Звертаємо увагу*, що ці об'єкти не розміщуються у клітинках електронної таблиці, а знаходяться на аркуші. Деякі аркуші можуть містити тільки діаграми, без електронних таблиць. Такі аркуші за замовчуванням мають імена **Діаграма1, Діаграма2** і т. д.

У таблиці 6.1 наведено перелік основних об'єктів табличного процесора та їх властивості.

Таблиця 6.1

### Властивості об'єктів табличного процесора

Об'єкт	Властивості об'єкта
Електронна книга	Ім'я, кількість аркушів, порядок розташування аркушів та ін.
Аркуш	Ім'я, кількість розміщених об'єктів та їх вид, колір ярлика, орієнтація аркуша, колір тла, наявність сітки та ін.
Електронна таблиця	Загальна кількість рядків, стовпців і клітинок; кількість рядків, стовпців і клітинок, що містять дані, та ін.
Рядок	Номер, висота, кількість заповнених даними клітинок та ін.
Стовпець	Номер, ширина, кількість заповнених даними клітинок та ін.
Клітинка	Адреса/ім'я, уміст, тип даних у клітинці, межі, заливка, шрифт символів та ін.
Діапазон клітинок	Адреса/ім'я, кількість клітинок та ін.
Діаграма	Тип, вид, назва, розмір області діаграми, колір заливки та ін.

Значення властивостей об'єктів електронної таблиці можна змінювати командами меню, кнопками **Стрічки**, командами контекстного меню.



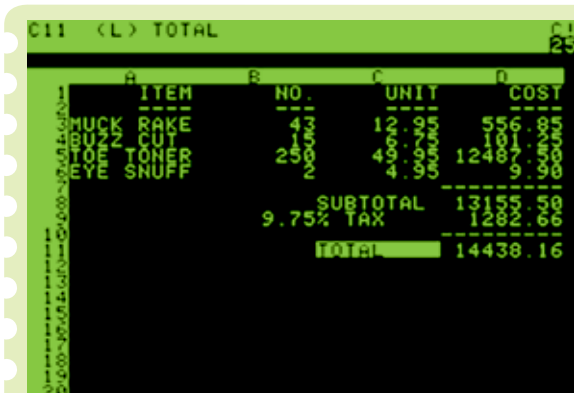
#### Цікаві факти з історії

Перший табличний процесор створили в 1979 році студент Гарвардського університету (США) **Ден Бріклін** і його товариш – програміст **Боб Френкстон** (мал. 6.2). Він був розроблений для комп'ютера **Apple II** і одержав назву **VisiCalc** (англ. *Visible Calculator* – наочний калькулятор). Електронна таблиця мала 5 стовпців і 20 рядків, а програма виконувала 4 арифметичні дії. Переобчислення формул у 100 клітинках таблиці після змінення даних відбувалося за 20 с.

Після **VisiCalc** з'явилися аналогічні програми від інших розробників і для інших моделей комп'ютерів, такі як **SuperCalc, Lotus 1-2-3, Multiplan** та інші.

Перша версія табличного процесора **Excel** з'явилася в 1985 році. Вона також була створена для комп'ютера **Apple**. Його розробники – американські програмісти **Дуг Кландер** і **Філіп Флоренце**.

Серед сучасних найбільш поширених табличних процесорів можна назвати **Microsoft Office Excel, LibreOffice Calc, Google Таблиці**.



Мал. 6.2. Вікно першого табличного процесора VisiCalc та його автори **Ден Бріклін** (праворуч) і **Боб Френкстон**

### Для тих, хто хоче знати більше

Стовпці електронної таблиці можуть також нумеруватися натуральними числами від 1 до 16 384. У цьому разі адреси клітинок записуються, наприклад, так: **R1C1**, **R5C2**, **R17C4**, **R27C26**, – де після літери **R** (англ. *Row* – рядок) указується номер рядка, а після **C** (англ. *Column* – стовпець) – номер стовпця. Тобто адресу **R5C2** потрібно розуміти так: **рядок п'ятий, стовпець другий**. Для змінення способу нумерації стовпців потрібно виконати **Файл** ⇒ **Параметри** ⇒ **Формули** та встановити позначку прапорця **Стиль посилань R1C1**.

## ФОРМАТИ ЧИСЛОВИХ ДАНИХ В ЕЛЕКТРОННИХ ТАБЛИЦЯХ

### Пригадайте

- Які операції форматування можна виконати над даними у клітинках електронної таблиці?
- У яких форматах може бути подано числові дані в електронній таблиці?
- Що таке *стандартний вигляд числа*?


В 6 класі ви вже використовували для обчислення в табличному процесорі *числовий* формат подання даних. Але в табличному процесорі Excel числові дані можуть бути подані й у інших форматах (мал. 6.3)

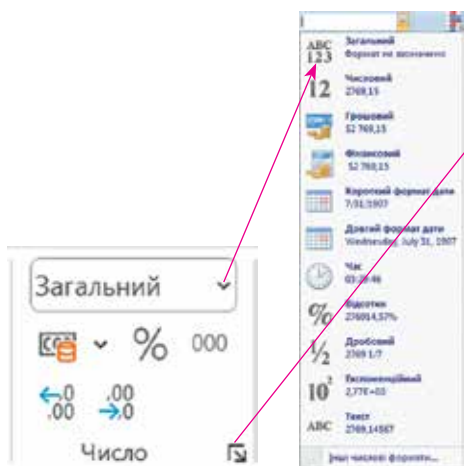
	A	B	C	D	E	F	G	H	I	J	K	L
1												
2												
3		Загальний	Числовий	Грошовий	Фінансовий	Дата	Час	Відсотковий	Дробовий	Експоненційний	Текстовий	Додатковий
4		12789,2879	0,28	315,008	315,008	25.12.10	22:12:45	56%	1/8	2,80E-01 12E+04		04050
5		-356,78	237	€ 1 125,60	€ 1 125,60	02/15/01	14:15	24%	2 3/5	-2,87E+00 15,4		482-0115
6		2,79E+00	2 789,15	\$27 879,23	\$ (27 878,23)	10 червня 2012 р.	2:43 AM	237%	-56 13/37	2,77E+03 -6793		{38067} 501-2348
7												

Мал. 6.3. Приклади числових форматів даних

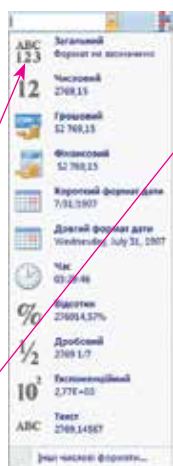
*Нагадаємо*, що змінення формату подання даних не змінює значення цих даних у пам'яті комп'ютера, а лише встановлює певний вигляд їх подання у клітинці. Реальне значення даних можна побачити в **Рядку формул**, зробивши відповідну клітинку поточною.

Установлення формату даних для поточної клітинки або для виділеного діапазону клітинок здійснюється з використанням елементів керування групи **Число** на вкладці **Основне** (мал. 6.4), списку поля **Числовий формат** (мал. 6.5), а також діалогового вікна **Формат клітинок** (мал. 6.6). Стрілками на малюнках показано спосіб відкриття зазначених елементів керування.

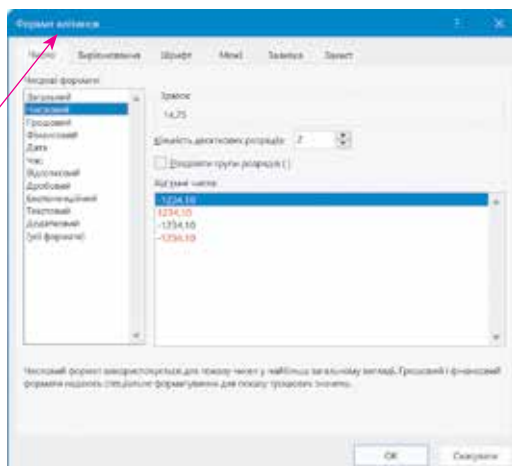
**Формат Загальний**  **123** є форматом даних у клітинках за замовчуванням. Він використовується для подання чисел здебільшого так, як їх було введено (клітинки **V4:V6** на мал. 6.3). Під час введення певних даних загальний формат може автоматично змінитися на інший.



Мал. 6.4. Група **Число**





Мал. 6.5. Список поля **Числовий формат**





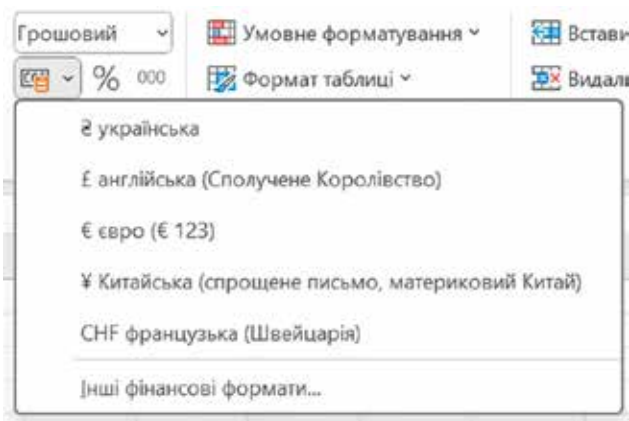
Мал. 6.6. Вікно **Формат клітинок**

## ЧИСЛОВИЙ ФОРМАТ


Формат **Числовий 12** (клітинки **S4:S6** на мал. 6.3) використовується для подання чисел у вигляді десяткового дробу із заданою кількістю десяткових розрядів. Ця кількість установлюється на лічильнику **Кількість десяткових розрядів** діалогового вікна **Формат клітинок** (мал. 6.6) і визначає, до якого десяткового розряду буде округлене число. Якщо число в клітинці має менше цифр після коми, ніж передбачено форматом, то під час виведення на екран воно буде доповнено нулями праворуч. Розрядність числа можна змінити, вибравши кнопки **Збільшити розрядність**  чи **Зменшити розрядність**  у групі **Число** (мал. 6.4) на **Стрічці**.


Також можна встановити розділювач груп розрядів (класів) у вигляді пропуску між групами по три цифри у цілій частині числа, установивши прапорець **Розділяти групи розрядів** у діалоговому вікні **Формат клітинок** (мал. 6.6). Те саме відбудеться і за вибору кнопки **Формат з роздільниками 000** у групі **Число** (мал. 6.4) на **Стрічці**.

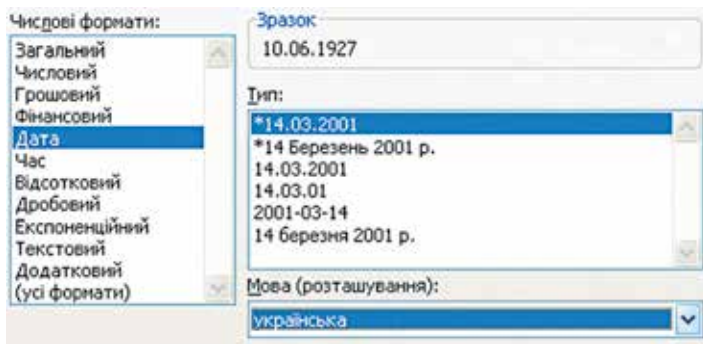
**Формат Грошовий**  (клітинки **D4:D6** на мал. 6.3) використовується для встановлення значень тих самих властивостей, що і для формату **Числовий**, з додаванням до числа позначення грошової одиниці, яке вибирається зі списку **Позначення** вікна **Формат клітинок**. Розділення груп розрядів (класів) у числі встановлюється автоматично. Вибір грошової одиниці можна зробити й у списку кнопки **Фінансовий формат чисел**  у групі **Число** (мал. 6.7) на **Стрічці**.



Мал. 6.7. Список кнопки **Фінансовий формат чисел**

**Формат Фінансовий**  (клітинки **E4:E6** на мал. 6.3) відрізняється від формату **Грошовий** тим, що значення у клітинках вирівнюються і за десятковою комою, і за знаком грошової одиниці.

**Формат Дата**  (клітинки **F4:F6** на мал. 6.3) використовується для подання числових даних у вигляді дати певного типу, які представлено на малюнку 6.8. Дати зберігаються в **Excel** як натуральні числа. Відлік у **Excel** починається з 01.01.1900 і цій даті відповідає число 1. Кожній наступній даті відповідає наступне натуральне число:




Мал. 6.8. Формат **Дата**



02.01.1900 – 2,  
 03.01.1900 – 3, ... ,  
 06.06.2006 – 38 874, ... ,  
 01.09.2010 – 40 422.

Таке подання дат дає змогу виконувати обчислення над ними. Так, кількість днів між двома датами визначається різницею чисел, що відповідають цим датам. Наприклад, різниця 01.09.2010 – 01.01.2010 буде обчислюватися так:  $40\,422 - 40\,179 = 243$ .

Формат **Час**  (клітинки **G4:G6** на мал. 6.3) використовується для відображення даних як часу з певним виглядом подання. Час у **Excel** зберігається у вигляді десяткового дробу, який визначає частину доби. Наприклад,

24:00	24 год = 1
01:00	1 год = $1/24 = 0,0416666666666667$
00:01:00	1 хв = $1/(24*60) = 0,0006944444444444444$
00:00:01	1 с = $1/(24*3600) = 0,0000115740740740741$
13:35:00	відповідає число 0,5659722222222222

У форматі **Відсотковий** % (клітинки **H4:H6** на мал. 6.3) дані подаються у вигляді числа, яке отримано автоматичним множенням умісту клітинки на 100, зі знаком % у кінці.

Формат **Дробовий**  $\frac{1}{2}$  використовують для подання числа у вигляді звичайного дробу. Наприклад, число **0,125** у клітинці **I4** подано як  $\frac{1}{8}$ , а число **2,6** у клітинці **I5** – дробовим числом  $2\frac{3}{5}$  (мал. 6.3). Слід урахувувати, що більшість чисел у цьому форматі буде подано наближено. Наприклад, число **0,65** у форматі звичайного дробу буде подано як  $\frac{2}{3}$  (хоча  $\frac{2}{3} = 0,666666\dots$ ).

Формат **Експоненційний**  $10^2$  використовують для подання числа у стандартному вигляді. Так, запис **2,80E-01** відповідає числу  $2,80 \cdot 10^{-1}$ , а **2,77E+03** відповідає числу  $2,77 \cdot 10^3$  (клітинки **J4:J6** на мал. 6.3).

Формат **Текстовий** **ab** використовують для подання чисел у клітинках як текст (клітинки **K4:K6** на мал. 6.3). Значення відображаються так само, як і вводяться.

**Стандартним виглядом додатного числа** називають його представлення у вигляді  $a \cdot 10^m$ , де  $1 \leq a \leq 10$ , а  $m$  – натуральне число. Число  $m$  називають *порядком числа*  $a$ , при цьому  $a$  називають *значущою частиною числа*.

## ВІДНОСНІ, АБСОЛЮТНІ ТА МІШАНІ ПОСИЛАННЯ



### Пригадайте

● Як у формулах у табличному процесорі записують адреси клітинки? Для чого їх використовують? ● Що таке *модифікація формул*? Коли вона відбувається? За якими правилами здійснюється?

Як ви знаєте, в електронних таблицях для обчислень у формулах можна використовувати адреси клітинок чи діапазонів клітинок.



У такому разі під час змінення даних у цих клітинках відбувається автоматичне переобчислення значень за всіма формулами, які містять такі адреси. Адреси клітинок, які використовують у формулах, також називають *посиланнями на клітинки*.

Під час копіювання формул посилання в них автоматично змінюються – **модифікуються**. Це відбувається за таким правилом: номери стовпців (рядків) у посиланнях змінюються на різницю номерів кінцевого і початкового стовпців (рядків). Під час переміщення формули не модифікуються.

Але інколи потрібно, щоб під час копіювання формул певні посилання не модифікувалися, тобто залишались незмінними.

Посилання, яке модифікується під час копіювання формули, називається **відносним**. Посилання, яке не модифікується під час копіювання формули, називається **абсолютним**. Щоб посилання було абсолютним, потрібно перед номером стовпця та перед номером рядка ввести символ \$. Наприклад, посилання **B10** та **A1:A7** є відносними, а посилання **\$B\$10** та **\$A\$1:\$A\$7** – абсолютними.

Розглянемо приклад, який пояснює доцільність використання абсолютних посилань.

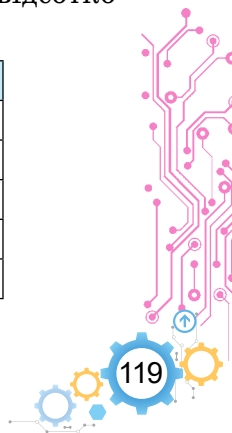
Нехай у клітинках діапазону **C3:C7** (мал. 6.9) міститься площа п'яти океанів, а у клітинці **C8** обчислено їхню загальну площу. У клітинках діапазону **D3:D7** потрібно обчислити, який відсоток загальної площі займає кожен океан.

	A	B	C	D
1				
2		Океан	Площа, млн км <sup>2</sup>	Відсоток
3		Атлантичний	91,6	24,8%
4		Індійський	73,556	19,9%
5		Північний Льодовитий	14,75	4,0%
6		Тихий	169,2	45,8%
7		Південний	20,3	5,5%
8		Разом	369,406	100%

Мал. 6.9. Використання відносних і абсолютних посилань

Для обчислення цих відсотків у клітинках діапазону **D3:D7** повинні розміщуватися такі формули з урахуванням установлення відсоткового формату чисел у клітинках:

Клітинка	Формула у клітинці
D3	=C3/C8
D4	=C4/C8
D5	=C5/C8
D6	=C6/C8
D7	=C7/C8



Якщо проаналізувати всі ці формули, то можна зробити висновок, що посилання в чисельнику змінюються, а посилання у знаменнику – ні. Якщо ми введемо у клітинку **D3** формулу  $=C3/C8$  і скопіюємо її в клітинки діапазону **D4:D7**, то всі посилання у формулі модифікуються. Тобто під час копіювання цієї формули отримуємо у відповідних клітинках такі формули:

Клітинка	Формула у клітинці
D3	$=C3/C8$
D4	$=C4/C9$
D5	$=C5/C10$
D6	$=C6/C11$
D7	$=C7/C12$

Щоб уникнути такої модифікації, потрібно посилання на клітинку **C8** у формулі  $=C3/C8$  зробити абсолютним, увівши у клітинку **D3** формулу  $=C3/\$C\$8$ . Посилання на клітинку **C3** залишається у формулі відносним і буде змінюватися під час копіювання відповідним чином.

Якщо в посиланні символ **\$** додано тільки перед номером стовпця або тільки перед номером рядка, наприклад, **\$B10** або **B\$10**, то під час копіювання такі посилання модифікуються частково: змінюється лише номер рядка або лише номер стовпця, біля якого не стоїть символ **\$**. Такі посилання називаються **мішаними**.

Змінити у формулі вид посилання (абсолютне, відносне чи мішане) можна під час його введення або редагування послідовним натисненням клавіші **F4**. Тобто якщо у клітинку, наприклад, увести відносне посилання **C8**, то після першого натиснення клавіші **F4** посилання перетвориться на абсолютне посилання **\$C\$8**, після другого натиснення **F4** це посилання перетвориться на мішане посилання **C\$8**, після третього натискання – на мішане посилання **\$C8**, а після четвертого – знов стане відносним **C8**.

Якщо у формулі замість посилання на клітинку використати її ім'я, то під час копіювання цієї формули таке посилання модифікуватись не буде. Отже, ім'я клітинки у формулі є абсолютним посиланням на цю клітинку. Тому в розглянутому вище прикладі можна було клітинці **C8** надати ім'я, наприклад **Площа\_разом**, і тоді формула у клітинці **D3** виглядала би так:  $=C3/Площа\_разом$  і під час копіювання посилання **Площа\_разом** не змінюватиметься.

Приклади абсолютних, відносних і мішаних посилань наведено в таблиці 6.2.

Таблиця 6.2

### Типи посилань

Посилання	Тип	Пояснення
<b>C3</b>	Відносне	Під час копіювання формули може змінюватись і номер стовпця, і номер рядка

Продовження таблиці 6.2

Посилання	Тип	Пояснення
\$A\$1	Абсолютне	Під час копіювання формули номер стовпця і номер рядка незмінні
курс_валюти	Абсолютне	Під час копіювання формули номер стовпця і номер рядка незмінні
C\$1	Мішане	Під час копіювання формули номер стовпця може змінюватись, а номер рядка незмінний
\$A3	Мішане	Під час копіювання формули номер стовпця незмінний, а номер рядка може змінюватись

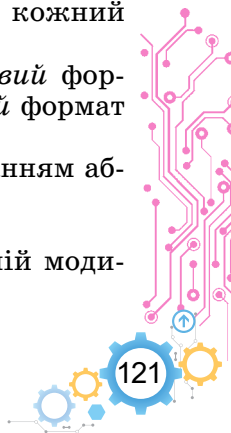


### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Завдання.** Виконайте обчислення в табличному процесорі.

- Відкрийте вказаний учителем/учителькою файл (наприклад, **Розділ 6\Пункт 6.1\вправа 6.1.xlsx**).
- На аркуші **Аркуш1** наведено результати роботи кількох фірм за I квартал року. У клітинці **C2** вказано, яку долю від квартального прибутку складає преміальний фонд фірм за I квартал. Обчисліть суму преміального фонду за I квартал для кожної фірми. Для цього:
  - Установіть для клітинки **C2** **відсотковий** формат даних (кнопка **Відсотковий** формат у групі **Число** на **Стрічці**), для клітинок діапазону **V4:C11** – **числовий** з розділювачем групи розрядів (кнопка **Формат з роздільниками** **000** у групі **Число** на **Стрічці**).
  - У клітинку **C4** введіть формулу  $=V4*\$C\$2$  з використанням абсолютного посилання на клітинку **C2**. Для зміни типу посилань можна використати клавішу **F4**.
  - Скопіюйте цю формулу в клітинки діапазону **C5:C10**.
  - Перегляньте формули та переконайтеся в їх правильній модифікації.
  - Змініть значення відсотка преміального фонду на інші. Переконайтеся у правильності переобчислень у клітинках.
- На аркуші **Аркуш2** обчисліть прибуток однієї фірми за кожний квартал відповідно до зазначених відсотків. Для цього:
  - Установіть для клітинок діапазону **V4:F4** – **відсотковий** формат даних, для клітинок діапазону **V5:F5** – **фінансовий** формат з розділювачем групи розрядів.
  - У клітинку **C5** введіть формулу  $=B\$5*C4$  з використанням абсолютного посилання на клітинку **V5**.
  - Скопіюйте цю формулу на діапазон клітинок **D5:F5**.
  - Перегляньте формули та переконайтеся в їх правильній модифікації.



5. Змініть прибуток фірми у клітинці **B5** на інше значення, переконайтесь у правильності переобчислень у клітинках.
4. На аркуші **Аркуш3** обчисліть прибуток кожної фірми за кожний квартал відповідно до зазначених відсотків. Для цього:
  1. Установіть для клітинок рядка **2** – *відсотковий* формат даних, для клітинок діапазону **B4:F10** – *грошовий* формат.
  2. Уведіть у клітинку **C4** формулу  $=B4*C\$3$  з використанням мішаних посилань на клітинки **B4** та **C3**.
  3. Скопіюйте цю формулу на діапазон клітинок **C4:F10**.
  4. Перегляньте формули та переконайтесь в їх правильній модифікації.
5. Змініть відсоток прибутку фірм по кварталах у клітинках **C3:F3** на інші значення, переконайтесь у правильності переобчислень у клітинках.
6. Збережіть книгу у вашій папці у файлі з іменем **вправа 6.1.xlsx**.
7. Закрийте вікно **Excel**.



### Найважливіше в цьому пункті

Основними об'єктами табличного процесора **Excel** є **електронна книга, аркуш, електронна таблиця, рядок, стовпець, клітинка, діапазон клітинок**.

Кожна клітинка електронної таблиці має **адресу**, яка складається з номера стовпця та рядка, на перетині яких вона розміщена. Деяка сукупність клітинок електронної таблиці утворює **діапазон клітинок**. Адреса діапазону клітинок задається адресами двох клітинок, розташованих у його протилежних кутах і розділених двокрапкою. На аркуші електронної книги можуть розміщуватися електронні таблиці, діаграми, малюнки та інші об'єкти. У клітинках електронної таблиці можуть зберігатися числа, тексти та формули.

Клітинці та діапазону клітинок можна надати **ім'я**. Його зручніше використовувати у формулах, оскільки ім'я клітинки не змінюється у формулах під час їх копіювання. Якщо клітинка має ім'я та її зробити поточною, то в полі **Ім'я** буде відображатися не адреса клітинки, а її ім'я.

Числові дані в **Excel** можна подати в кількох форматах: **Загальний, Числовий, Грошовий, Дата, Відсотковий, Дробовий, Текстовий** та інші. Форматування не змінює дані у клітинці таблиці, а лише встановлює певний вигляд їх подання. Реальне значення даних можна побачити в **Рядку формул**, зробивши відповідну клітинку поточною.

Установлення формату даних для поточної клітинки або для виділеного діапазону клітинок здійснюється з використанням елементів керування групи **Число** на вкладці **Основне**, списку поля **Числовий формат**, а також діалогового вікна **Формат клітинок**.

У формулах електронної таблиці можна використовувати різні види посилань клітинки: **відносні** (модифікуються під час копіювання формули), **абсолютні** (не модифікуються під час копіювання формули), **мішані** (модифікуються частково).

Щоб посилання було абсолютним, потрібно перед номером стовпця та номером рядка ввести символ **\$**. У мішаних посиланнях символ **\$** уводиться або перед номером стовпця, або перед номером рядка.

Під час копіювання формул адреси клітинок у них автоматично **модифікуються**. Це відбувається за таким правилом: номери стовпців (рядків) у посиланнях змінюються на різницю номерів кінцевого і початкового стовпців (рядків). Під час переміщення формули не модифікуються.

Якщо у формулі замість посилання на клітинку використати її **ім'я**, то під час копіювання цієї формули це посилання модифікуватися не буде, воно вважається абсолютним.



### Дайте відповіді на запитання

1. Які об'єкти електронної таблиці ви знаєте? Назвіть їх властивості та наведіть приклади значень.
2. Які дії можна виконувати над об'єктами електронної таблиці?
3. Які формати числових даних можна встановити? Дайте їм стислу характеристику.
4. Які типи посилань (адрес) можна використовувати у формулах в електронних таблицях? Як вони записуються?
5. Чим відрізняється використання різних типів посилань на клітинки у формулах?
6. До якого типу посилань належить ім'я клітинки, яке надає користувач?



### Обговоріть і зробіть висновки

1. Який тип посилання на клітинку **C5** потрібно використати у формулі, щоб під час її копіювання по рядку це посилання модифікувалось, а по стовпцю – ні? Не модифікувалося під час копіювання в будь-якому напрямку?
2. Чи однаковий результат буде під час копіювання формул, у яких використано мішані та абсолютні посилання?
3. З'ясуйте, які задачі з різних шкільних предметів під час їх розв'язання в **Excel** потребували б використання абсолютних посилань. Що спільного у знайдених прикладах?
4. Що спільного та відмінного у використанні абсолютних посилань та імен клітинок у формулах?



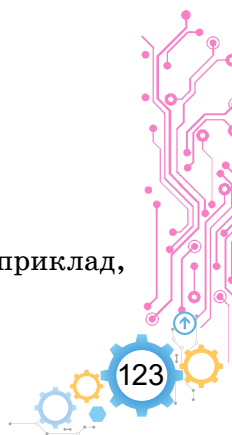
### Виконайте завдання

1. У клітинці **E3** розміщено формулу  $=\$B\$3+C4$ . Запишіть, як виглядатиме ця формула, якщо її скопіювати у клітинку:
  - а) **E7**;
  - б) **I8**;
  - в) **C3**;
  - г) **A2**.
2. У клітинці **K8** розміщено формулу  $=E5+\$C3$ . Запишіть, як виглядатиме ця формула, якщо її скопіювати у клітинку:
  - а) **C9**;
  - б) **B4**;
  - в) **E9**;
  - г) **D8**;
  - д) **I2**;
  - е) **F5**.
3. Запишіть значення у клітинках **C2** і **D2**, якщо в них скопіювати формулу з клітинки **B2** (мал. 6.10).

	A	B	C	D
1	1	10	20	30
2		$=B1+A\$1$		
3		100		

Мал. 6.10

4. Відкрийте вказаний учителем/учителькою файл (наприклад, Розділ 6\пункт 6.1\зразок 6.1.4.xlsx).



1. На аркуші **Аркуш1** обчисліть, який шлях пройшов кожний турист за день і залишок шляху, якщо весь маршрут змагань складає 400 км (це значення записано у клітинці **C10**). Які типи даних використано в таблиці?
2. На аркуші **Аркуш2** обчисліть ціни товарів у гривнях (курс євро розміщено у клітинці **B3**). Які типи даних використано в таблиці?
3. На аркуші **Аркуш3** обчисліть кількість учнів та учениць у кожному 8 класі та на всій паралелі 8-х класів, використавши для клітинки **D10** ім'я **Учні\_8**. Використайте це ім'я у формулі для обчислення відсотків кількості учнів та учениць кожного 8-го класу відносно загальної кількості учнів та учениць 8-х класів.

Збережіть книгу у вашій папці у файлі з іменем **завдання 6.1.4.xlsx**.

5. Відкрийте вказаний учителем/учителькою файл (наприклад, **Розділ 6\пункт 6.1\зразок 6.1.5.xlsx**).

1. На аркуші **Аркуш1** обчисліть прибуток кожної фірми та всього виробництва за рік, а також відсоток вкладу кожної фірми в річний прибуток.
2. На аркуші **Аркуш2** обчисліть щомісячний розмір депозитного вкладу з капіталізацією відсотків. Які типи даних використано в таблиці?
3. На аркуші **Аркуш3** обчисліть кількість змінних носіїв різних видів, потрібних для збереження даних, що містяться в 1 г молекули ДНК, – 455 млрд Гб (за дослідженнями американських учених). Використайте для клітинки з указаним значенням ім'я **ДНК**. Відшукайте самостійно потрібні відомості про сучасні види змінних носіїв та їх ємності. Які типи даних використали в таблиці? Які типи посилань на клітинки таблиці ви використали під час проведення обчислень?

**Капіталізація відсотків** – накопичені за певні проміжки часу відсотки додаються до основної суми вкладу.

Збережіть книгу у вашій папці у файлі з іменем **завдання 6.1.5.xlsx**.

6. Створіть за наведеним зразком (мал. 6.11) електронну таблицю для обчислення значень 10 перших членів числових послідовностей різних видів:

Тип послідовності	Різниця/знаменник	Номер члена послідовності									
		1	2	3	4	5	6	7	8	9	10
Арифметична прогресія	2	1									
Геометрична прогресія	2	1									
Числа Фібоначчі		1	1								

Мал. 6.11

- Кожен наступний член послідовності, починаючи з другого, дорівнює попередньому, до якого додано одне й те саме число (*різницю* послідовності). Така послідовність чисел називається *арифметичною прогресією*.
- Кожен наступний член послідовності, починаючи з другого, дорівнює попередньому, помноженому на одне й те саме число, відмінне від 0 (*знаменник* послідовності). Така послідовність чисел називається *геометричною прогресією*.
- Кожен наступний член послідовності, починаючи з третього, дорівнює сумі двох попередніх членів цієї послідовності. Така послідовність чисел називається *числа Фібоначчі*.



### Цікаві факти з історії

Леонардо Фібоначчі, справжнє ім'я якого Леонардо Пізанський, був видатним математиком середньовічної Європи. Він народився приблизно 1170 року в Пізі, Італія. Вивчав математику в арабських учителів, познайомив європейців з арабською системою числення, яка тоді була невідомою у Європі. У 1202 році Леонардо Фібоначчі опублікував книгу «Liber Abaci» («Книга абака»), де пояснив нову систему числення на основі десяткової системи. Це значно полегшило обчислення і стало справжньою революцією в математиці того часу.

Виконайте обчислення. Які формули ви для цього використовували? Які типи посилань у них використовувались? Які типи даних використано в таблиці? Поясніть їх вибір. Збережіть книгу у вашій папці у файлі з іменем завдання 6.1.6.xlsx.

7. Створіть таблицю для визначення вартості різних українських грошових банкнот у доларах, євро та фунтах стерлінгів. Курси валют з'ясуйте на сайті Національного банку України. Передбачте зберігання значень курсів долара, євро та фунта стерлінгів в окремих клітинках. Збережіть книгу у вашій папці у файлі з іменем завдання 6.1.7.xlsx.

8. Створіть електронну таблицю з обчислення елементів таблиці множення (мал. 6.12), використавши одну формулу з мішаними посиланнями. Збережіть книгу у вашій папці у файлі з іменем завдання 6.1.8.xlsx. Надрукуйте створену таблицю множення.

	A	B	C	D	E	F	G	H	I	J
1	Таблиця множення									
2		1	2	3	4	5	6	7	8	9
3	1	1	2	3	4	5	6	7	8	9
4	2	2	4	6	8	10	12	14	16	18
5	3	3	6	9	12	15	18	21	24	27
6	4	4	8	12	16	20	24	28	32	36
7	5	5	10	15	20	25	30	35	40	45
8	6	6	12	18	24	30	36	42	48	54
9	7	7	14	21	28	35	42	49	56	63
10	8	8	16	24	32	40	48	56	64	72
11	9	9	18	27	36	45	54	63	72	81

Мал 6.12





9. Заповніть електронну таблицю натуральними числами від 1 до 90 по спіралі (мал. 6.13). Поміркуйте, яким чином можна автоматизувати цей процес. Які формули для цього ви використаєте? Який тип адресації буде присутній у формулах і чому? Збережіть книгу у вашій папці у файлі з іменем **завдання 6.1.9.xlsx**. Надрукуйте створену таблицю.

1	2	3	4	5	6	7	8	9
10	9	8	7	6	5	4	3	2
11	10	9	8	7	6	5	4	3
12	11	10	9	8	7	6	5	4
13	12	11	10	9	8	7	6	5
14	13	12	11	10	9	8	7	6
15	14	13	12	11	10	9	8	7
16	15	14	13	12	11	10	9	8
17	16	15	14	13	12	11	10	9
18	17	16	15	14	13	12	11	10

Мал. 6.13



### Готуємось до вивчення нового матеріалу

1. Що таке *функція*? Які математичні функції ви знаєте? Наведіть приклади.
2. Які вбудовані функції Excel ви використовували? Наведіть приклади.
3. Які правила запису функцій у формулах?

## 6.2. МАТЕМАТИЧНІ, СТАТИСТИЧНІ ТА ЛОГІЧНІ ФУНКЦІЇ

У цьому пункті йтиметься про:

- ▶ функції в табличному процесорі та їх вставлення у формули;
- ▶ використання математичних, статистичних і логічних функцій для обчислень.

### ФУНКЦІЇ В ТАБЛИЧНОМУ ПРОЦЕСОРІ



#### Пригадайте

• Що таке *функція* в математиці? Що таке *аргумент функції*? Що таке *результат функції*? • Для чого використовують у табличному процесорі вбудовані функції **SUM**, **MIN**, **MAX**, **AVERAGE**?

Ви вже знаєте, що у формулах у табличному процесорі Excel можна використовувати **вбудовані функції**. Табличний процесор Excel має бібліотеку таких функцій, до якої входить понад 500 різноманітних функцій. Усі вони для зручності розподілено на **категорії**: *математичні, статистичні, текстові, логічні, фінансові* та інші.

У 6 класі ви вже вивчали вбудовані функції в табличному процесорі та використовували такі з них: **SUM** – для знаходження суми



чисел, **AVERAGE** – для знаходження середнього арифметичного, **MAX** та **MIN** – для визначення максимального та мінімального значення серед указаних чисел. Перша з них належить до категорії математичних функцій, інші – до категорії статистичних.

Функція в Excel має ім'я, аргументи та значення. Під час запису функції у формулі спочатку вказується її ім'я, а потім у дужках аргументи функції. Якщо аргументів кілька, то вони відділяються крапкою з комою.

Так, у наведеній на малюнку 6.14 формулі **=SUM(B3:B13;K3:K13)** використано функцію з іменем **SUM**, аргументом якої є два діапазони клітинок **B3:B13** та **K3:K13**, а значенням є сума чисел з указаних діапазонів клітинок.

**=SUM(B3:B13;K3:K13)**

Мал. 6.14. Приклад запису функції **SUM**

Кількість аргументів функції та їх тип залежить від конкретної функції. Аргументами функції можуть бути числа, тексти (їх потрібно брати в подвійні лапки), посилання на клітинки та діапазони, вирази та інші функції.

В Excel є функції з різною кількістю аргументів: з одним чи кількома, без аргументів. Приклади функцій наведено в таблицях 6.3 та 6.4 цього пункту підручника.

Результат однієї функції можна використати як аргумент іншої функції. Такі функції називають **вкладеними**. Наприклад, подана на малюнку 6.15 функція **SQRT** використовує вкладені функції **MIN** і **MAX**. Можна використовувати до семи рівнів вкладення функцій.

Мал. 6.15. Вкладені функції

Призначення кожної функції, наявність аргументів та їх кількість, типи аргументів можна подивитися в **Довідці** або в коментарях у ході вставлення функції у формулу, як це продемонстровано на малюнках 6.16 та 6.18.

Слід пам'ятати, що під час копіювання формул з функціями аргументи будуть модифікуватися.

## ВСТАВЛЕННЯ ФУНКЦІЙ У ФОРМУЛУ

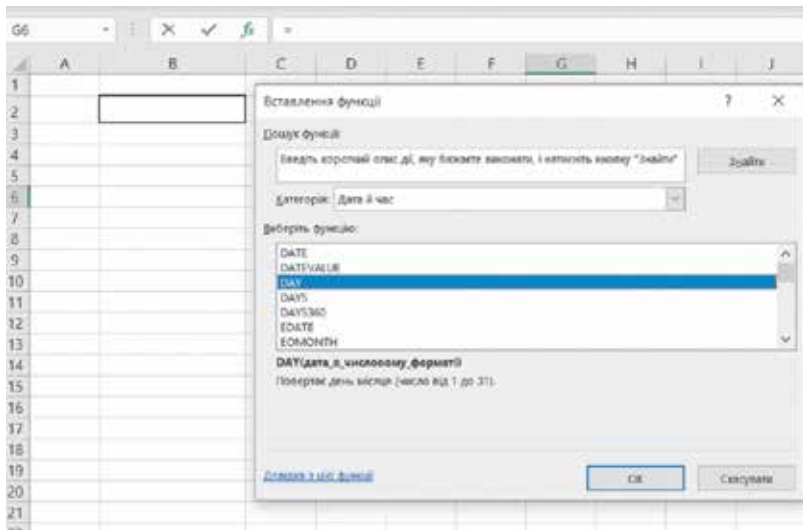


## Пригадайте

- Як вставити в формулу вбудовані функції **SUM**, **MIN**, **MAX**, **AVERAGE**?
- Що може бути аргументом функції в табличному процесорі?
- Як редагувати формулу в табличному процесорі?

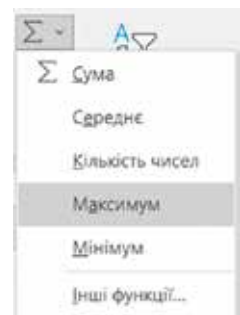
Уставити функцію у формулу можна кількома способами:

- Вибравши кнопку **Вставлення функції**  $fx$  ліворуч Рядка формул або виконавши **Формули**  $\Rightarrow$  **Бібліотека функцій**  $\Rightarrow$  **Вставити функцію**  $fx$  (мал. 6.16). Ці дії відкривають діалогове вікно **Вставлення функцій**, де всі функції розподілено за **Категоріями** та вміщено стислий опис призначення функцій.



Мал. 6.16. Вставлення функції у формулу з використанням кнопки **Вставлення функцій**

- Вибравши потрібну функцію зі списку **Сума** у групі **Редагування** вкладки **Основне** (мал. 6.17). Цим способом ви користувалися в попередніх класах під час обчислення суми та середньоарифметичних, мінімальних і максимальних значень у діапазоні клітинок. Команда **Інші функції** відкриває діалогове вікно **Вставлення функцій**, де можна вибрати будь-яку іншу функцію табличного процесора.
- Увівши функцію вручну безпосередньо у клітинку або в **Рядок формул** (мал. 6.18). Під час уведення з клавіатури перших літер імені функції поруч з клітинкою з'являються підказки, у яких послідовно демонструється список мож-



Мал. 6.17. Список кнопки **Автосума**

ливих функцій, зразок запису, порядок запису аргументів та їх типи. Вибравши потрібну функцію зі списку подвійним клацанням, далі можна вводити аргументи. Уводити з клавіатури імена функції та посилання на клітинки можна як малими, так і великими англійськими літерами. Цим способом зручніше користуватися під час введення складних формул з кількома функціями.



Мал. 6.18. Введення функції у формулу в клітинку з клавіатури

Розглянемо детальніше алгоритм уставки функції у формулу з використанням кнопки **Вставка функції**  $fx$ . Наприклад, потрібно у клітинці **B1** обчислити добуток (функція **PRODUCT**) всіх чисел діапазону клітинок **A1:A10**.

Для цього слід:

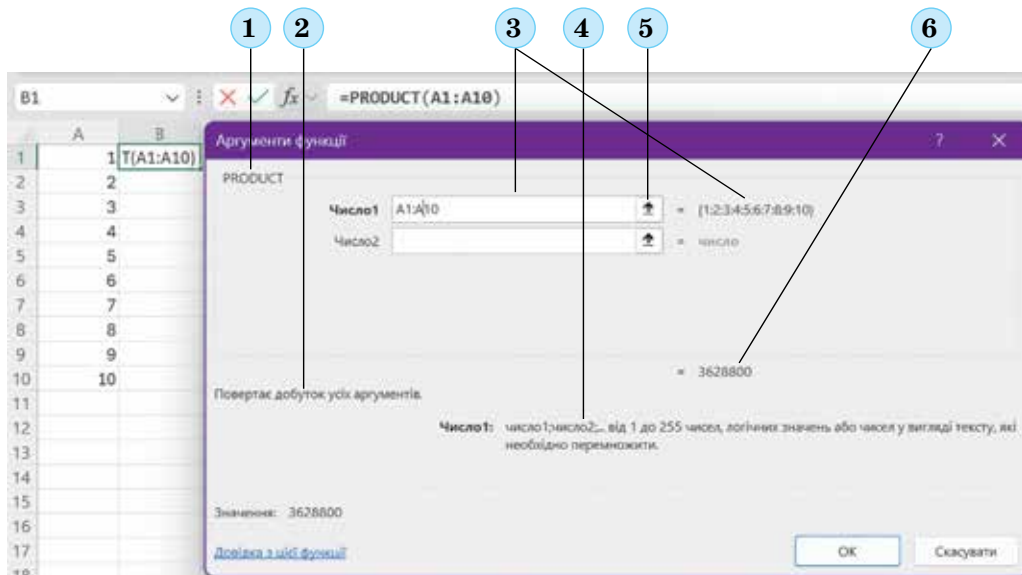
1. Вибрати клітинку електронної таблиці, у яку потрібно вставити формулу. Наприклад, **B1**.
2. Вибрати кнопку **Вставка функції**  $fx$  ліворуч від **Рядка формул**.
3. Вибрати у вікні **Вставка функції** у списку **Категорія** потрібну категорію функцій. Наприклад, **Математичні**.
4. Вибрати у списку потрібну функцію. Наприклад, **PRODUCT**.
5. Вибрати кнопку **ОК**.

Після вибору виконання цих дій у поточну клітинку автоматично буде вставлено знак **=** (якщо в цій клітинці введення формули ще не розпочиналося), ім'я функції та пара круглих дужок, а також відкривається вікно **Аргументи функції** з полями для введення аргументів цієї функції.

6. Увести у вікні **Аргументи функції** в поле **Число1** і за потреби в інших полях аргументи функції – сталі значення, адреси клітинок і діапазонів (мал. 6.19). Наприклад, **A1:A10**.
7. Вибрати кнопку **ОК**.

Після виконаних дій треба перевірити правильність введення формули у **Рядку формул** та отриманий результат (у клітинці **B1**). І далі за потреби продовжити введення формули, якщо вона складається з кількох компонентів.

Вікно **Аргументи функції** (мал. 6.19) містить коментарі про призначення даної функції та її аргументи. Крім того, під час введення аргументів праворуч від полів з'являються значення відповідних аргументів і нижче відображається поточний результат обчислення. На все це варто звертати увагу! Щоб детальніше дізнатися щодо використання функції, можна вибрати посилання **Довідка з цієї функції**.






Мал. 6.19. Введення даних у вікні **Аргументи функції**:

- 1 – ім'я функції; 2 – призначення функції; 3 – аргументи функції та їх значення;  
4 – опис аргументів функції; 5 – кнопка **Розгорнути**; 6 – значення функції

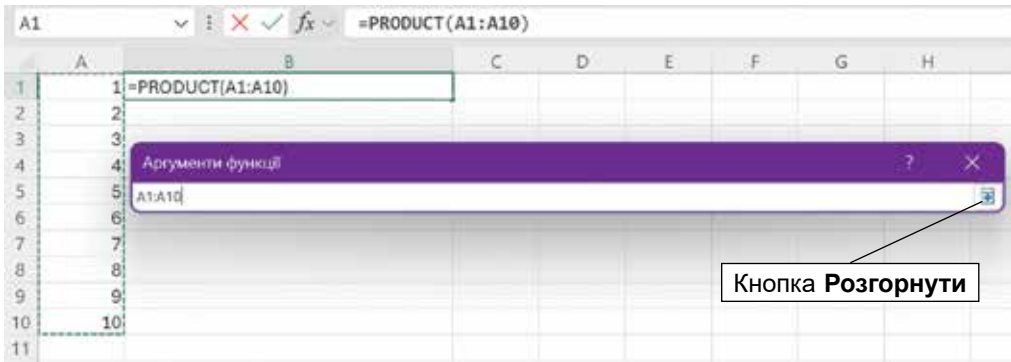
Якщо функція має фіксовану кількість аргументів, то вікно **Аргументи функції** одразу містить відповідну кількість полів для їх введення. Якщо функція має змінну кількість аргументів, то у вікні спочатку з'являється кілька полів, а потім, у процесі введення аргументів, з'являються наступні поля.

Якщо аргументом є число або текст, то його можна вводити в поле з клавіатури або вставляти з **Буфера обміну**. Якщо аргументом є посилання на клітинки, то його можна або вводити з клавіатури, або виділити відповідні клітинки з використанням миші.

Для введення посилань на клітинки з використанням миші потрібно:

1. Вибрати кнопку **Згорнути**  (мал. 6.19, 5) відповідного поля для введення аргументів (після цього вікно **Аргументи функції** змінює вигляд: у ньому, крім рядка заголовка, залишається тільки це поле, а кнопка **Згорнути** змінюється на кнопку **Розгорнути** ).
2. Виділити потрібні клітинки (посилання на виділені клітинки автоматично вставляються у відповідне поле і в формулу (мал. 6.20)). Наприклад, клітинки **A1:A10**.
3. Вибрати кнопку **Розгорнути**  (після цього вікно **Аргументи функції** відновлює свій попередній вигляд).

Для деяких функцій Excel автоматично пропонує перший аргумент. Наприклад, для функції **SUM**, **PRODUCT** пропонується знайти суму чи добуток чисел з діапазону клітинок, заповнених числовими



Мал. 6.20. Вставлення посилань на клітинки у функцію

даними, що знаходяться безпосередньо над клітинкою з формулою або безпосередньо зліва від неї, якщо верхній діапазон порожній. Користувач може прийняти цю пропозицію, натиснувши клавішу **Enter**, або ввести замість цього аргументу інший.

Коли формула введена, то її можна редагувати. Це можна зробити безпосередньо в **Рядку формул** або в самій клітинці, клацнувши на неї двічі.

## МАТЕМАТИЧНІ ФУНКЦІЇ



### Пригадайте

- Які функції ви вивчали на уроках математики?
- Як записуються арифметичні вирази в **Excel**?

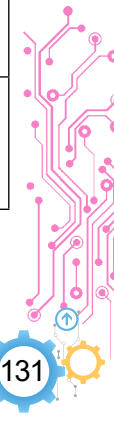
До категорії математичних функцій належать усі функції, які ви вивчали наразі в математиці, а також функції **SUM** і **PRODUCT**, які ви вже знаєте.

Приклади деяких інших математичних функцій наведено в таблиці 6.3.

Таблиця 6.3

### Деякі математичні функції

Функція	Кількість аргументів	Результат	Приклад використання
<b>ABS(число)</b> (англ. <i>absolute</i> – абсолютний)	1	Модуль (абсолютна величина) <b>числа</b>	<b>ABS(C10)</b>
<b>INT(число)</b> (англ. <i>integer</i> – цілий)	1	<b>Число</b> , округлене до найближчого меншого цілого (ціла частина числа)	<b>INT(B2)</b>



Функція	Кількість аргументів	Результат	Приклад використання
PI()	0	Значення числа $\pi = 3,14159265358979$ з точністю до 15-го десяткового розряду	PI()
POWER(число; степінь) (англ. <i>power</i> – потужність, степінь)	2	Число, піднесене до степеня з показником степінь (показник степеня – будь-яке число)	POWER(C5;5)
ROUND(число; кількість_розрядів) (англ. <i>round</i> – округлювати)	2	Округлене число. Якщо кількість_розрядів > 0, то в поданні числа залишається вказана кількість розрядів із заокругленням останнього після коми. Якщо кількість_розрядів < 0, то число округлюється до вказаної кількості розрядів перед комою. Якщо кількість_розрядів = 0, то число округлюється до цілого	ROUND(C1;3)
SIN(число) (англ. <i>sin</i> – синус)	1	Синус числа (кута, заданого в радіанах)	SIN(B8)
SQRT(число) (англ. <i>square root</i> – квадратний корінь)	1	Арифметичний квадратний корінь із числа	SQRT(B10)
DEGREES(кут) (англ. <i>degrees</i> – градуси)	1	Градусна міра кута, поданого в радіанах	DEGREES(A2)
RADIANS(кут) (англ. <i>radians</i> – радіани)	1	Радіанна міра кута, поданого в градусах	RADIANS(B2)
RANDBETWEEN (нижн.межа; верх.межа) (англ. <i>random between</i> – випадкове між)	2	Випадкове ціле число з відрізка [нижн.межа; верх.межа]	RANDBETWEEN(A5;10)

Звертаємо вашу увагу! Якщо у клітинці, наприклад **A1**, міститься число – градусна міра кута, то для знаходження його синуса потрібно спочатку перевести градуси в *радіани*, а потім обчислити значення функції **SIN**. Тобто формула обчислення буде така: **=SIN(RADIANS(A1))**.

**Радіан** – одиниця вимірювання величини кута (позначається рад).

$$1 \text{ рад} = \left(\frac{180}{\pi}\right)^\circ \approx 57^\circ$$

$$1^\circ = \frac{\pi}{180} \text{ рад} \approx 0,017 \text{ рад}$$

## СТАТИСТИЧНІ ФУНКЦІЇ



### Пригадайте

- Чи знаєте ви, що таке *статистика*? Де вам траплявся цей термін?
- Які статистичні функції ви вже використовували під час розв'язування завдань?

Ви вже використовували для обчислення статистичні функції **AVERAGE**, **MAX**, **MIN**. Інші функції цієї групи наведено в таблиці 6.4.

Таблиця 6.4

### Деякі статистичні функції

Функція	Кількість аргументів	Результат	Приклад використання
<b>COUNT</b> (число1; [число2];...) (англ. <i>count</i> – підрахунок)	Від 1 до 255, усі, крім першого, необов'язкові	Кількість чисел у вказаних клітинках. <b>Число1</b> – обов'язковий аргумент: посилання на клітинку або діапазон клітинок. <b>Число2</b> та інші – необов'язкові аргументи	<b>COUNT(B2:C5;E1)</b>
<b>MODE.SNGL</b> (число1; [число2];...)	Від 1 до 255, усі, крім першого, необов'язкові	Найчастіше повторюване значення в вказаних клітинках. <b>Число1</b> – обов'язковий аргумент: посилання на клітинку або діапазон клітинок. <b>Число2</b> і т. д. – необов'язковий аргумент. Для отримання результату потрібно, щоб хоч одне значення повторювалось хоча б 1 раз. В іншому разі результат обчислення буде відсутній	<b>MODE.SNGL (B1:A13)</b>



Функція	Кількість аргументів	Результат	Приклад використання
<b>RANK</b> (число; посилання; [порядок]) (англ. <i>ranking</i> – рейтинг)	3	Ранг, або Рейтинг (місце за величиною), числа серед інших значень діапазону. <b>Число</b> – обов'язковий аргумент: число, ранг якого потрібно визначити. <b>Посилання</b> – обов'язковий аргумент: діапазон чисел, серед яких визначається ранг указанного числа. <b>Порядок</b> – необов'язковий аргумент: число, що визначає, як розподіляються порядкові номери	<b>RANK(B5;B1:B10)</b>

## ЛОГІЧНІ ФУНКЦІЇ



### Пригадайте

- Що таке *логічні вирази*, *логічні операції*? Які значення вони можуть мати?
- Що таке *розгалуження*? Які види розгалуження ви знаєте? Як вони виконуються?
- Як записуються команди розгалуження в мові програмування, яку ви вивчаєте?

Крім формул, що містять числа, знаки дій, функції, в Excel можуть використовуватися формули, які містять **логічні вирази** зі знаками порівняння: > (більше), < (менше), = (дорівнює), <> (не дорівнює), >= (більше або дорівнює), <= (менше або дорівнює). Наприклад, =A2+15=B4-1; =SUM(A2:C10)>100 та інші.

Ви вже знаєте, що такі логічні вирази можуть набувати значення **True** (істина) або **False** (хиба, хибність).

В Excel використовують і логічні функції. Функція, результат якої дорівнює **True** або **False**, називається **логічною**. До **логічних функцій** в Excel належать функції **IF** (англ. *if* – якщо), **AND** (англ. *and* – і), **OR** (англ. *or* – або), **NOT** (англ. *not* – не) та інші.

Їх використовують у формулах тоді, коли потрібно виконувати різні обчислення залежно від істинності або хибності певного логічного виразу. Такі задачі вам уже траплялися, зокрема у програмуванні. Наприклад, потрібно нарахувати працівнику премію у розмірі 20 % від окладу, якщо він перевиконав план, і не нараховувати премію у протилежному випадку. Або, наприклад, якщо термін продажу товару в магазині завершується, то його уціняють на 50 %.

Загальний вигляд логічної функції **IF** такий:

**IF**(логічний\_вираз;значення\_якщо\_істина;значення\_якщо\_хибність).

Значення цієї функції визначається так:



1. Визначається значення логічного виразу (**True** чи **False**).
2. Якщо логічний вираз має значення **True**, то значення функції дорівнює значенню виразу **значення\_якщо\_істина**.
3. Якщо ж логічний вираз має значення **False**, то значення функції дорівнює значенню виразу **значення\_якщо\_хибність**.

Для прикладу розглянемо таку задачу. *Обчислити премію продавцю-консультанту магазину. Якщо продавець за попередні два місяці обслуговував більше ніж 100 покупців, то йому нараховують премію 15 % від вартості відповідних покупок, у іншому разі премія складатиме 5 %.* Для розрахунків можна використати таку формулу:  $=IF(A1+B1>100;C1*15\%;C1*5\%)$ , обчислення значення за якою відбуватиметься так: спочатку Excel обчислить суму чисел з клітинок **A1** і **B1** та визначить значення логічного виразу **A1+B1>100**; якщо це значення **True**, то значенням функції буде значення виразу **C1\*15%**; якщо значення логічного виразу **A1+B1>100** буде дорівнювати **False**, то значенням функції буде значення виразу **C1\*5%**.

Розглянемо ще один приклад задачі, у розв'язку якої використано логічну функцію **IF**. Кожен з робітників бригади має щоденну норму і реальну кількість виготовлених деталей за день. Потрібно з'ясувати для кожного робітника, чи виконав він свою денну норму. На малюнку 6.21 наведено розв'язок цієї задачі.

	A	B	C	D	E	F
1		Робітник	Норма	Виготовлено	Виконання норми	
2		Величко	100	95	не виконано	
3		Іванов	120	112	не виконано	
4		Петренко	120	122	виконано	
5		Хіміч	130	143	виконано	

Аргументи функції

IF

Лог\_вираз: D4>=C4 = TRUE

Значення\_якщо\_істина: "виконано" = "виконано"

Значення\_якщо\_хибність: "не виконано" = "не виконано"

Перевіряє, чи виконується умова, і повертає одне значення, якщо вона виконується, та інше значення, якщо ні.

Лог\_вираз: будь-яке значення або вираз, який можна визначити як значення TRUE або FALSE.

Значення: виконано

Довідка з цієї функції

OK Скасувати

Мал. 6.21. Приклад використання логічної функції **IF**

Як бачимо, правило обчислення значення логічної функції **IF** у табличному процесорі аналогічно до правила виконання команди повного розгалуження у програмуванні.

Аргумент значення\_якщо\_хибність – це необов’язковий аргумент логічної функції **IF**, тому в **Excel** можна використовувати і скорочений запис цієї функції, який відповідає команді неповного розгалуження:

**IF(логічний\_вираз;значення\_якщо\_істина).**

Розглянемо логічні функції **AND**, **OR**, **NOT** (табл. 6.5).

Таблиця 6.5

### Логічні функції **AND**, **OR**, **NOT**

Функція	Кількість аргументів	Результат	Приклад запису функції
<b>AND</b> (логіч1; [логіч2];...)	Від 1 до 255, усі, крім першого, необов’язкові	<ul style="list-style-type: none"> <li>● <b>True</b>, якщо всі її аргументи мають значення <b>True</b>;</li> <li>● <b>False</b>, якщо хоча б один аргумент має значення <b>False</b></li> </ul>	<b>AND(A1&gt;2;B1&gt;10; B1&lt;20;C1=5)</b>
<b>OR</b> (логіч1; [логіч2];...)	Від 1 до 255, усі, крім першого, необов’язкові	<ul style="list-style-type: none"> <li>● <b>True</b>, якщо хоча б один з аргументів має значення <b>True</b>;</li> <li>● <b>False</b>, якщо всі аргументи мають значення <b>False</b></li> </ul>	<b>OR(A1&gt;2;B1&gt;10; B1&lt;20;C1=5)</b>
<b>NOT</b> (логічне)	1	<ul style="list-style-type: none"> <li>● <b>True</b>, якщо аргумент має значення <b>False</b>;</li> <li>● <b>False</b>, якщо аргумент має значення <b>True</b></li> </ul>	<b>NOT(F1&gt;25)</b>

*Звертаємо увагу*, результати наведених функцій в **Excel** збігаються з результатами аналогічних логічних операцій кон’юнкція, диз’юнкція, заперечення, які були розглянуті в п. 4.5.

Логічні функції **AND**, **OR**, **NOT** найчастіше використовують для запису складних логічних виразів як аргумент функції **IF**.

Наприклад, у функції **IF** не можна записувати подвійну нерівність. Тому замість подвійної нерівності  $10 < A3 < 20$  потрібно використовувати логічний вираз **AND(A3>10;A3<20)**. Або, наприклад, функцію задано так:

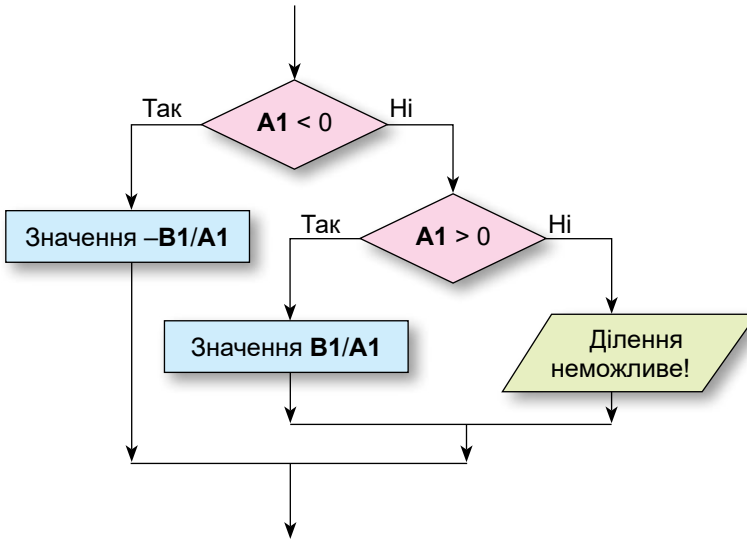
$$y = \begin{cases} 2x - 5, & \text{якщо } x < -2 \text{ або } x > 10, \\ \text{функція не визначена за інших значень } x. \end{cases}$$

Якщо, припустимо, значення  $x$  міститься у клітинці **A3**, то формулу для обчислення значення функції  $y$  записують так:

**=IF(OR(A3<-2;A3>10);2\*A3-5;"функція не визначена")**

У логічній функції **IF** може бути до 7 рівнів вкладеності. Тобто другий і третій аргументи функції **IF** можуть містити вкладені арифметичні операції та функції, у тому числі й функцію **IF**. Наприклад, для обчислення значення за наведеною блок-схемою (мал. 6.22) слід використати таку функцію:

**=IF(A1<0;-B1/A1; IF(A1>0;B1/A1;"Ділення неможливе!"))**.



Мал. 6.22. Блок-схема для обчислень

### Для тих, хто хоче знати більше

#### Математичні й статистичні функції з умовами

Excel має функції, які обчислюють суму, середнє арифметичне, кількість не всіх значень з діапазонів клітинок, а лише тих, які задовольняють певній умові, – **SUMIF**, **AVERAGEIF**, **COUNTIF**.

Наведемо приклад використання цих функцій. Нехай у клітинках діапазону **A1:A25** розміщено кількості деталей, які виготовив робітник за кожний з 22 робочих днів місяця. Його щоденна норма зберігається у клітинці **C1** і, наприклад, дорівнює 15. За умовою контракту, якщо робітник перевиконує щоденну норму, то за ці деталі він одержує додаткову винагороду. Щоб обчислити, за скільки деталей робітнику потрібно заплатити винагороду, слід скористатися функцією **SUMIF**: **SUMIF(A1:A22;">15")**.

#### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

**Задача.** Дано три цілих випадкових числа в діапазоні від 1 до 10. Визначте, чи існує трикутник з такими сторонами. Умовою існування трикутника є таке правило: довжина більшої сторони повинна бути менше, ніж сума довжин двох інших сторін. Результатом обчислення повинна бути відповідь «трикутник існує» або «трикутник не існує». Якщо трикутник існує, то обчисліть його площу, інакше виведіть повідомлення «помилка».

**Довідка з математики.** Площа трикутника за відомими довжинами його сторін  $a$ ,  $b$ ,  $c$  обчислюється за формулою Герона:

$$S = \sqrt{p(p-a)(p-b)(p-c)}, \text{ де } p = \frac{a+b+c}{2},$$

$p$  – половина периметра трикутника (півпериметр).



Створимо математичну модель для цієї задачі:

**вхідні дані:** три довільних числа  $a$ ,  $b$ ,  $c$ ;

**кінцеві результати:** значення площі трикутника  $S$ .

**формули:**

*значення сторін трикутника:* випадкові числа в діапазоні  $[1;10]$ ;

*обчислення півпериметра:*  $p = \frac{a + b + c}{2}$ ;

*обчислення площі:*  $S = \sqrt{p(p - a)(p - b)(p - c)}$ .

Уведення формул у клітинки таблиці наведено на малюнку 1.

	A	B
1	<b>Існування трикутника</b>	
2	сторона А	=RANDBETWEEN(1;10)
3	сторона В	=RANDBETWEEN(1;10)
4	сторона С	=RANDBETWEEN(1;10)
5	найбільше число	=MAX(B2:B4)
6	сума двох інших	=B2+B3+B4-B5
7	трикутник	=IF(B5<B6;"існує";"не існує")
8	півпериметр	=IF(B7="існує";(B2+B3+B4)/2;"помилка")
9	площа трикутника	=IF(B7="існує";SQRT(B8*(B8-B2)*(B8-B3)*(B8-B4));"помилка")

Мал. 1. Вигляд електронної таблиці для обчислення

Для виконання обчислень у задачі:

- Відкрийте вказаний учителем/учительською файл (наприклад, **Розділ 6\Пункт 6.2\права 6.2.xlsx**).
- На аркуші **Аркуш1** у клітинки **B2**, **B3**, **B4** введіть формули для обчислення трьох значень: **=RANDBETWEEN(1;10)**, використавши математичну функцію **RANDBETWEEN** для генерування випадкових чисел у діапазоні від 0 до 10.
- Обчисліть у клітинці **B5** значення найбільшого з трьох чисел, увівши формулу **=MAX(B2:B4)**, використавши статистичну функцію **MAX**.
- Уведіть у клітинці **B6** формулу для обчислення суми двох інших чисел, увівши формулу **=B2+B3+B4-B5**.
- Перевірте у клітинці **B7** умову існування трикутника, увівши формулу **=IF(B5<B6;"існує";"не існує")** з використанням логічної функції.
- Якщо трикутник існує, то обчисліть у клітинці **B8** півпериметр трикутника, увівши відповідну формулу **=IF(B7="існує";(B2+B3+B4)/2;"помилка")**.
- Якщо трикутник існує, обчисліть у клітинці **B9** площу трикутника, увівши таку формулу: **=IF(B7="існує";SQRT(B8\*(B8-B2)\*(B8-B3)\*(B8-B4));"помилка")**.
- Установіть у клітинках **B8** та **B9** числовий формат з двома десятковими розрядами.
- Перегляньте, як змінення вхідних даних впливає на результати обчислень. Для цього перезапустіть генератор випадкових чисел, натиснувши клавішу **F9** або виконавши **Дані ⇒ Оновити все**.

10. Збережіть книгу у вашій папці у файлі з іменем **вправа 6.2.xlsx**.
11. Закрийте вікно Excel.



### Найважливіше в цьому пункті

Табличний процесор **Excel** має бібліотеку вбудованих функцій, до якої входить понад 500 різноманітних функцій. Усі вони для зручності розподілені на категорії: *математичні, статистичні, текстові, логічні, фінансові* та інші.

Функція в **Excel** має **ім'я**, **аргументи** і **значення**. Під час запису функції у формулі спочатку вказується її ім'я, а потім у дужках **аргументи**, розділені крапкою з комою. Кількість аргументів функції та їх тип залежить від конкретної функції, це можуть бути числа, текст (його потрібно брати в подвійні лапки), посилання на клітинки та діапазони, математичні та логічні вирази, інші функції. Результат однієї функції можна використати як аргумент іншої функції. Такі функції називають **вкладеними**.

Уставити функцію у формулу можна кількома способами:

- виконавши **Формули** ⇒ **Бібліотека функцій** ⇒ **Вставити функцію** або вибравши кнопку **Вставлення функції** ліворуч у **Рядку формул**;
- відкривши список кнопки **Сума** у групі **Редагування** вкладки **Основне**;
- увівши ім'я функції та її аргументи безпосередньо у клітинку з клавіатури або в **Рядок формул**.

Математичні та статистичні функції призначено для автоматизації рутинних обчислень, їх використання підвищує точність обчислень і дає змогу ефективніше аналізувати великі обсяги даних. Приклади математичних функцій: модуль числа **ABS**, арифметичний квадратний корінь із числа **SQRT**, піднесення числа до степеня **POWER**, сума чисел **SUM**. Приклади статистичних функцій: середнє арифметичне **AVERAGE**, максимальне число **MAX**, мінімальне число **MIN**, кількість чисел **COUNT**.

До логічних функцій в **Excel** належать функції **IF**, **AND**, **OR**, **NOT** та інші. Їх використовують у формулах тоді, коли табличний процесор повинен виконувати різні операції залежно від істинності або хибності певного логічного виразу.

Загальний вигляд логічної функції **IF**:

**IF(логічний\_вираз;значення\_якщо\_істина;значення\_якщо\_хибність).**

Значення цієї функції визначається так само, як і виконання команди розгалуження у програмуванні.



### Дайте відповіді на запитання

1. Чим характеризуються функції в табличному процесорі **Excel**? Наведіть кілька прикладів.
2. Які ви знаєте категорії функцій в **Excel**? Наведіть кілька прикладів.
3. Що таке *логічні вирази* та *функції* в **Excel**? Яких значень вони можуть набувати?
4. Як визначається значення логічної функції **IF**?
5. Як визначається значення логічних функцій **AND**, **OR**, **NOT**?
6. Що таке *вкладеність функцій*? Як обчислюється значення функції з вкладенням?



### Обговоріть і зробіть висновки

1. Які типи даних можуть мати аргументи функцій? Поясніть це на прикладах.
2. Перегляньте список функцій у табличному процесорі **Excel** і знайдіть функції, у яких немає аргументів. Яке їх призначення? Як ці функції записують?



3. Перегляньте підручник з алгебри та виберіть, які функції там розглянуто та які задачі можна розв'язати в табличному процесорі Excel. Запропонуйте кілька прикладів.

4. Ви проводили опитування учнів/учениць вашої школи щодо дотримання ними правильного розпорядку дня, де були питання про вік респондентів/респонденток, їх стать, клас навчання, кількість калорій споживання на день, тривалість нічного сну, фізичні навантаження протягом дня. Запропонуйте функції Excel, які ви можете використати для аналізу отриманих даних в електронних таблицях.



### Виконайте завдання

1. Відкрийте табличний процесор Excel. Створіть таблицю для обчислення довжин діагоналей 5 прямокутників за даними довжинами їх сторін. Значення довжин сторін визначити випадковими значеннями в діапазоні від 5 до 25. Установіть формат результатів обчислення з двома десятковими розрядами. Збережіть книгу у вашій папці у файлі з іменем завдання 6.2.1.xlsx.

2. Відкрийте вказаний учителем/учителькою файл (наприклад, Розділ 6\Пункт 6.2\зразок 6.2.2.xlsx). На аркуші Аркуш1 наведено таблицю очок, набраних кожним з учасників змагань з п'ятиборства в кожному виді змагань. Додайте до таблиці формули для обчислення суми очок і максимальної кількості очок, набраних кожним з учасників, а також середньої кількості очок, набраних учасниками в кожному виді змагань. Середню кількість очок округліть до цілих, установивши відповідний формат клітинок. Збережіть книгу у вашій папці у файлі з іменем завдання 6.2.2.xlsx.


3. У клітинці A5 міститься число 10, у клітинці B5 – число -7, а в клітинці C5 – число 0. Визначте та запишіть у зошит значення наведених логічних функцій:

- а)  $\text{AND}(A5>5;A5<20)$ ;
- б)  $\text{OR}(C5<10;C5\geq 20)$ ;
- в)  $\text{OR}(C5<10;A5>20)$ ;
- г)  $\text{AND}(A5>-2;B5>0)$ ;
- д)  $\text{NOT}(B5<20)$ ;
- е)  $\text{AND}(\text{OR}(B5>5;B5<-5);\text{NOT}(B5<>10))$ ;
- ж)  $\text{OR}(\text{AND}(A5>2;A5\leq 10);B5<0)$ ;
- з)  $\text{NOT}(\text{AND}(A5<100;C5=0))$ .

4. Відкрийте вказаний учителем/учителькою файл (наприклад, Розділ 6\Пункт 6.2\зразок 6.2.4.xlsx). На аркуші Аркуш1 наведено кількості деталей, зроблених кожним з робітників протягом дня. У клітинці A1 міститься денна норма. Додайте до таблиці формули для визначення, виконав кожний з робітників норму чи ні. Результат подайте у вигляді True – False. Збережіть книгу у вашій папці у файлі з іменем завдання 6.2.4.xlsx.


5. Відкрийте вказаний учителем/учителькою файл (наприклад, Розділ 3\Пункт 6.2\зразок 6.2.5.xlsx). На аркуші Аркуш1 наведено результати виступів учасників/учасниць олімпіади у кожному з двох турів. За умовами проведення олімпіади, Дипломом I ступеня нагороджуються учасники, що набрали не менше ніж 85 % очок, Дипло-


мом II ступеня – учасники, що набрали не менше ніж 75 %, але менше від 85 % очок, **Дипломом III ступеня** – учасники, що набрали не менше ніж 50 %, але менше від 75 % очок, усі інші одержують **Диплом учасника**. Додайте до таблиці формули для визначення, який **Диплом** буде вручено кожному/кожній з учасників/учасниць. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.2.5.xlsx**.


 **6.** Відкрийте вказаний учителем/вчителькою файл (наприклад, **Розділ 6\Пункт 6.2\зразок 6.2.6.xlsx**). На аркуші **Аркуш1** наведено таблицю значень змінної  $x$ . Додайте до таблиці формули для визначення відповідних значень функції  $y$ :


$$y = \begin{cases} 2x - 5, & \text{якщо } x < -2 \text{ або } x > 10, \\ 3x + 1, & \text{якщо } 2 \leq x \leq 3, \\ 5 - 4x^3 & \text{за інших значень } x. \end{cases}$$


Збережіть книгу у вашій папці у файлі з іменем **завдання 6.2.6.xlsx**.

 **7.** Складіть електронну таблицю, за якою визначте, які умови вкладу в банк вигідніші клієнту: 9 % річних з нарахуванням відсотків щомісячно чи 10 % річних з нарахуванням відсотків кожного півріччя. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.2.7.xlsx**.

 **8.** Складіть електронну таблицю для визначення коренів квадратного рівняння, значення коефіцієнтів обчисліть як випадкові цілі числа в діапазоні від  $-5$  до  $5$ . Збережіть книгу у вашій папці у файлі з іменем **завдання 6.2.8.xlsx**.

 **9.** Знайдіть в Інтернеті дані про кількість населення в 10 країнах Європи. Складіть електронну таблицю і визначте місце кожної із цих країн за кількістю населення в їх загальному списку, використавши функцію **RANK**. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.2.9.xlsx**. *Порада:* щоб функція дала правильні результати для всіх країн, потрібно діапазон, у якому визначається рейтинг, зробити з абсолютною адресацією.

 **10.** Використовуючи дані з Інтернету, складіть електронну таблицю для визначення індексу маси тіла людини та порівняння її з нормою. Перевірте виконання обчислень для себе, а вдома – для членів вашої родини. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.2.10.xlsx**.

 **11.** Відомі дати народження 5 ваших друзів. Визначте, у який день тижня вони народилися і в який день тижня святкуватимуть свій день народження цього року. Самостійно знайдіть відповідні функції в переліку вбудованих функцій у категорії **Дата та час**. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.2.11.xlsx**.



### Практична робота № 8.

#### «Використання функцій у табличному процесорі»

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

1. Відкрийте вказаний учителем/учителькою файл (наприклад, **Розділ 6\Пункт 6.2\практична 8.xlsx**).
2. На аркуші **Аркуш1** наведено таблицю показників лічильника електроенергії за кожний місяць минулого року. У клітинці **A1** розміщено вартість  $1 \text{ кВт} \cdot \text{год}$ . Додайте до таблиці формули для обчислення щомісячної плати за електроенергію, сумарної плати за рік, середніх щомісячних витрат електроенергії.
3. На аркуші **Аркуш2** наведено прізвища учнів/учениць класу та їх зріст. Додайте до таблиці формули для визначення рейтингу учнів/учениць у класі за спаданням зросту та визначення для кожного/кожної учня/учениці, нижче чи вище його/її зріст від середнього значення.
4. На аркуші **Аркуш3** побудуйте таблицю значень функції  $y = \sqrt{x + 9} + x^2$  на відрізку  $[-3; 3]$ .
5. Збережіть книгу у вашій папці у файлі з іменем **практична 8.xlsx**.



#### Готуємось до вивчення нового матеріалу

1. Як відсортувати елементи списку в текстовому документі?
2. Як швидко в таблиці в текстовому документі знайти потрібний запис?

## 6.3. УПОРЯДКУВАННЯ ТА ФІЛЬТРУВАННЯ ДАНИХ У ТАБЛИЧНОМУ ПРОЦЕСОРІ

У цьому пункті йтиметься про:

- ▶ упорядкування даних у електронній таблиці;
- ▶ фільтрування даних у електронній таблиці.

### УПОРЯДКУВАННЯ ДАНИХ У ЕЛЕКТРОННІЙ ТАБЛИЦІ



#### Пригадайте

- Що таке *таблиці кодів символів*? Для чого їх використовують? Які таблиці кодів символів ви знаєте?
- Яким чином здійснюється сортування елементів списку в текстовому процесорі?

Дані в електронній таблиці можна впорядковувати (*сортувати*) – змінювати порядок розташування даних у рядках або стовпцях за їх значеннями.

У табличному процесорі можна виконувати сортування за **зростанням** або за **спаданням**. Дані різних типів у таблиці впорядковуються за різними правилами.

В Excel сортування за зростанням відбувається за такими правилами:

- числа упорядковуються від найменшого значення до найбільшого, тобто так, як вони розміщуються на числовій прямій;






- дати упорядковуються від старішої до новішої, тобто у хронологічній послідовності;
- тексти впорядковуються в алфавітному порядку, в порядку розміщення символів тексту в кодовій таблиці **Unicode**; тобто так, як це відбувається у словниках;
- спочатку розташовуються прописні літери, потім великі літери;
- спочатку розташовуються тексти, написані латиницею, потім – кирилицею;
- логічне значення **FALSE** розміщується перед значенням **TRUE**;
- порожні клітинки розташовуються останніми;
- якщо у стовпці таблиці містяться дані різних типів, то після сортування вони будуть розміщуватися в такому порядку: числові, дата/час, текстові, логічні.

Під час сортування за спаданням порядок розташування буде зворотний. Приклади розміщення значень різних типів після сортування за зростанням і за спаданням наведено на малюнку 6.23.

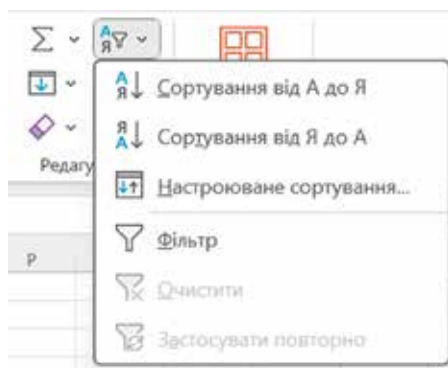
Уведені дані	Тип даних	Сортування за зростанням	Тип даних	Сортування за спаданням	Тип даних
09.03.1964	Дата	-23	Числові	TRUE	Логічні
Тетяна	Текстові	10%	Числові	FALSE	Логічні
FALSE	Логічні	0,78	Числові	Тетяна	Текстові
134	Числові	134	Числові	Людмила	Текстові
Victor	Текстові	09.03.1964	Дата	Victor	Текстові
06.07.1985	Дата	06.07.1985	Дата	Oleg	Текстові
10%	Числові	Oleg	Текстові	06.07.1985	Дата
Oleg	Текстові	Victor	Текстові	09.03.1964	Дата
		Людмила	Текстові	134	Числові
0,78	Числові	Тетяна	Текстові	0,78	Числові
TRUE	Логічні	FALSE	Логічні	10%	Числові
-23	Числові	TRUE	Логічні	-23	Числові
Людмила	Текстові				

Мал. 6.23. Приклади сортування даних

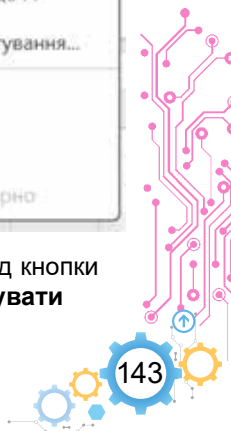
Щоб швидко виконати сортування даних в одному зі стовпців таблиці, потрібно:

1. Виділити відповідний стовпець.
2. Виконати **Основне** ⇒ **Редагування** ⇒ **Сортувати й фільтрувати**  ⇒ **Сортування від найменшого до найбільшого**  або **Сортування від найбільшого до найменшого**  (мал. 6.24).

У результаті цих дій дані у стовпці змінять своє розташування відповідно до вибраного порядку.



Мал. 6.24. Список команд кнопки **Сортувати й фільтрувати**

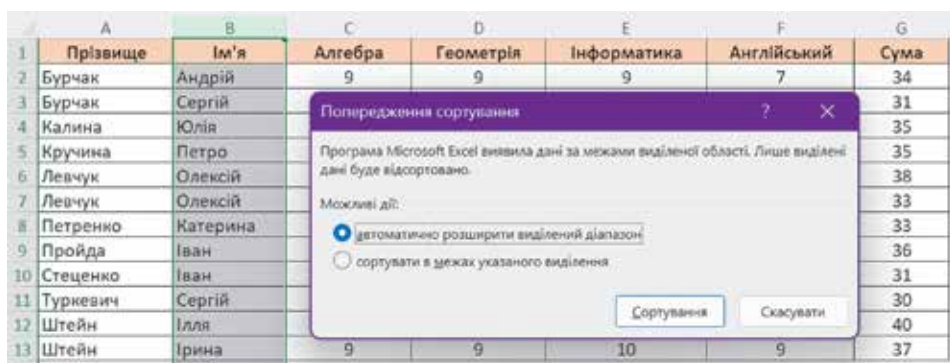


*Звертаємо вашу увагу!* Назва команди **Сортування від найменшого до найбільшого** і **Сортування від найбільшого до найменшого** автоматично змінюється залежно від типу даних у перших клітинках виділеного діапазону, які потрібно відсортувати:

- для текстових даних – **Сортування від А до Я** та **Сортування від Я до А**;
- для дат і часу – **Сортування від найстаршого до наймолодшого** та **Сортування від наймолодшого до найстаршого**.

Якщо під час сортування виділити діапазон клітинок у кількох стовпцях, то значення в першому стовпці цього діапазону будуть впорядковані у вибраному порядку, а дані в інших стовпцях будуть переміщені разом з відповідним рядком.

Якщо виконувати сортування в таблиці з кількома стовпцями, виділивши дані тільки в одному стовпці, відкривається вікно **Попередження сортування** (мал. 6.25), у якому пропонується вибрати: розширити виділений діапазон клітинок на всю заповнену даними таблицю або сортувати в межах виділення.



Мал. 6.25. Вікно **Попередження сортування**

Сортування у виділеному діапазоні можна здійснювати і за значеннями в кількох стовпцях (мал. 6.26). Це відбувається так:

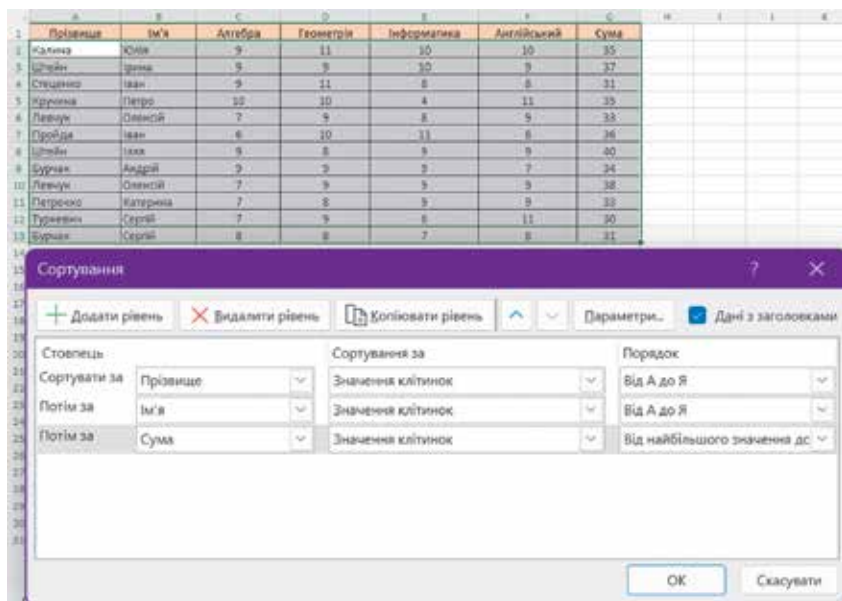
- спочатку рядки переставляються за значеннями в першому з указаних стовпців;
- потім сортування в кожному наступному з вибраних стовпців відбувається лише для тих рядків, у яких значення в усіх попередніх указаних для сортування стовпцях збігаються.

Кожний з указаних для сортування стовпців називається *рівнем сортування*.

Щоб відсортувати дані у виділеному діапазоні клітинок за значеннями в кількох стовпцях, потрібно:

1. Виділити потрібний діапазон клітинок (бажано, щоб до нього увійшли заголовки стовпців, але до нього не повинні входити об'єднані клітинки).
2. Виконати **Основне** ⇒ **Редагування** ⇒ **Сортувати й фільтрувати**

⇒ **Настроюване сортування** (або **Дані** ⇒ **Сортування й фільтр** ⇒ **Сортувати**).



Мал. 6.26. Електронна таблиця та вікно Сортування

- Установити або змінити у вікні **Сортування** значення параметрів сортування: у списку поля **Стовпець** вибрати заголовок першого стовпця, за значеннями в якому потрібно відсортувати дані у стовпці, у списку поля **Сортування за** – *Значення клітинок*, а у списку поля **Порядок** – вид сортування (мал. 6.26). Зверніть увагу в таблиці на однофамільців.
- За потреби сортування за значеннями іншого стовпця, вибрати кнопку **Додати рівень** і встановити умови сортування за цим стовпцем, вибравши в полях рядка **Потім за** значення потрібних параметрів сортування.
- За потреби повторити попередній крок кілька разів для інших стовпців, за якими слід провести сортування.
- Якщо у виділений діапазон увійшов рядок заголовка з підписами стовпців, а позначка прапорця **Дані з заголовками** не встановлена, потрібно встановити її (тоді рядок заголовка не буде враховуватися під час сортування і залишиться на своєму місці).
- Вибрати кнопку **ОК**.

Пояснимо таке сортування на конкретному прикладі. На малюнку 6.26 наведено приклад невідсортованої таблиці річних оцінок учнів/учениць, а на малюнку 6.27 цю саму таблицю відсортовано за значеннями у трьох стовпцях: **Прізвище** (від А до Я), **Ім'я** (від А до Я), **Сума** (від найбільшого значення до найменшого).

Процес упорядкування відбувався так. Спочатку рядки переставлялися так, щоб у стовпці **Прізвище** значення були розташовані в алфавітному порядку (**від А до Я**). Подальше сортування відбувалося вже за значеннями у другому з указаних стовпців (**Ім'я**). Оскільки таблиця містить три групи однофамільців: Бурчак, Левчук, Штейн, – тому за другим стовпцем сортування відбувалося тільки в межах кожної

	A	B	C	D	E	F	G
1	Прізвище	Ім'я	Алгебра	Геометрія	Інформатика	Англійська	Сума
2	Бурчак	Андрій	9	9	9	7	34
3	Бурчак	Сергій	8	8	7	8	31
4	Калина	Юлія	9	11	10	10	35
5	Кручина	Петро	10	10	4	11	35
6	Левчук	Олексій	7	9	9	9	38
7	Левчук	Олексій	7	9	8	9	33
8	Петренко	Катерина	7	8	9	9	33
9	Пройда	Іван	6	10	11	8	36
10	Стеценко	Іван	9	11	8	8	31
11	Туркевич	Сергій	7	9	6	11	30
12	Штейн	Ілля	9	8	9	9	40
13	Штейн	Ірина	9	9	10	9	37

Мал. 6.27. Електронна таблиця, відсортована за умовами, наведеними на малюнку 6.26

із цих груп рядків, по кожній окремо. У результаті цього кроку сортування рядки переставилися так, щоб імена були розташовані в алфавітному порядку (від А до Я).

Сортування за значеннями у третьому з вибраних стовпців (Сума) відбувалося тільки в межах рядків, у яких у перших з двох вибраних стовпців збіглися значення (Левчук Олексій). Рядки переставилися так, щоб у стовпці Сума значення були відсортовані за спаданням (38 та 33).



#### Для тих, хто хоче знати більше

Сортувати дані в таблиці можна не тільки за значеннями у стовпцях, а й за значеннями в рядках. Для змінення режиму сортування потрібно у вікні **Сортування** вибрати кнопку **Параметри**, а потім у вікні вибрати відповідний перемикач. Там само можна визначити, чи враховувати регістр літер під час сортування, тобто чи розрізняти великі й малі літери.

## ФІЛЬТРУВАННЯ ДАНИХ



#### Пригадайте

● Що таке *фільтрування*? Де вам траплялася така дія? ● Яких значень можуть набувати логічні функції **AND** та **OR** залежно від значень аргументів?

**Фільтрування** – це автоматизований відбір значень у клітинках електронної таблиці, що відповідають певним умовам.


Після виконання фільтрування в таблиці відображаються лише ті рядки, дані в яких відповідають умовам **фільтрування**. Усі інші рядки — тимчасово приховуються.

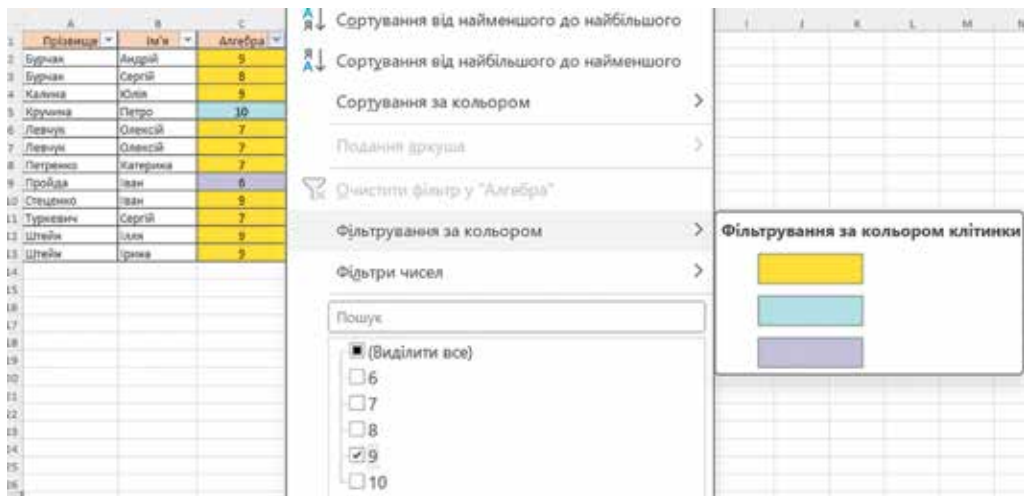
У **Excel** є два режими фільтрування: простий та розширений фільтри.

Щоб установити режим простого фільтрування даних, потрібно:

1. Установити табличний курсор у довільну клітинку заповненої даними електронної таблиці або виділити потрібний діапазон клітинок.
2. Виконати **Основне** ⇒ **Редагування** ⇒ **Сортувати й фільтрувати**

▲▼ ⇒ **Фільтр** (або **Дані** ⇒ **Сортування й фільтрування** ⇒ **Фільтр**).


Після виконання цих дій біля правої межі заголовків стовпців усієї заповненої даними таблиці або виділеного діапазону клітинок з'являється кнопка  – відкриття списку команд фільтрування. У цьому списку (мал. 6.28) містяться команди сортування та фільтрування за значеннями (числовими, текстовими або інших типів), а також за кольором.



Мал. 6.28. Застосування фільтра за значенням у стовпці **Алгебра**

Розглянемо деякі приклади виконання фільтрування.

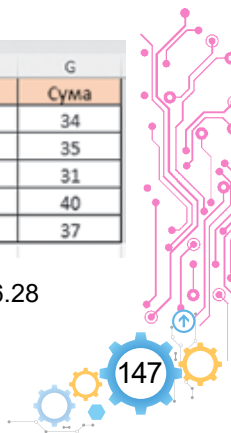
Найчастіше виконують фільтрування за значеннями в одному або кількох стовпцях даних. Наприклад, потрібно за наведеною на малюнку 6.28 таблицею оцінок відобразити список учнів/учениць, які мають з *алгебри* оцінку **9**. Для цього потрібно:


1. Застосувати фільтр до діапазону з даними, виконавши **Основне** ⇒ ⇒ **Редагування** ⇒ **Сортувати й фільтрувати**  ⇒ **Фільтр**.
2. Відкрити список команд фільтрування у стовпці **Алгебра**.
3. Зняти позначку прапорця **Виділити все**.
4. Установити позначку прапорця біля значення **9**.
5. Вибрати кнопку **ОК**.

Після виконання цих дій у таблиці будуть відображені лише ті рядки, у яких значення у стовпці **Алгебра** дорівнюють **9**. Результат описаного прикладу наведено на малюнку 6.29.

	A	B	C	D	E	F	G
1	Прізвище	Ім'я	Алгебра	Геометрія	Інформатика	Англійська	Сума
2	Бурчак	Андрій	9	9	9	7	34
4	Калина	Юлія	9	11	10	10	35
10	Стеценко	Іван	9	11	8	8	31
12	Штейн	Ілля	9	8	9	9	40
13	Штейн	Ірина	9	9	10	9	37

Мал. 6.29. Результат застосування фільтра до таблиці на малюнку 6.28

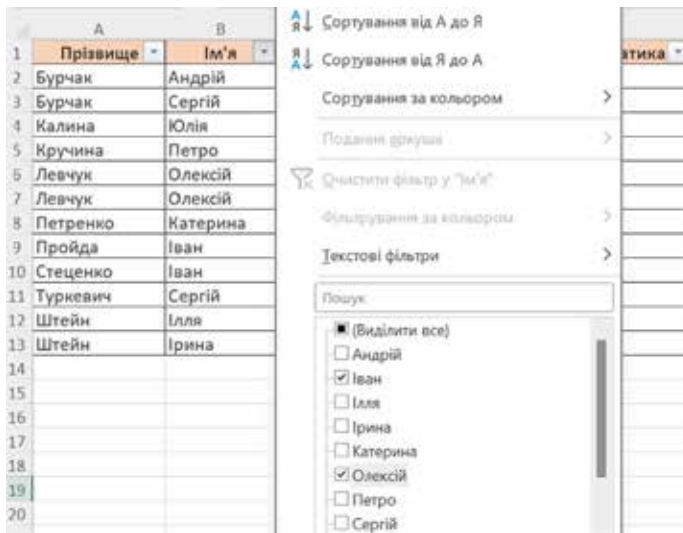


Зверніть увагу, вигляд кнопки фільтра у стовпці **Алгебра** на малюнку 6.29 змінився на такий: , це означає наявність установленого фільтра для даних у цьому стовпці таблиці. Також колір номерів рядків та їх нумерація з пропусками говорить про те, що фільтр застосовано.

Фільтрування можна здійснювати і за кількома значеннями. Наприклад, потрібно відобразити дані про учнів з іменами *Іван* або *Олексій*. Для цього потрібно:

1. Застосувати фільтр до діапазону з даними.
2. Відкрити список команд фільтрування у стовпці **Ім'я**.
3. Зняти позначку прапорця **Виділити все**.
4. Установити позначку прапорців біля значень *Іван* та *Олексій* (мал. 6.30).
5. Вибрати кнопку **ОК**.

У результаті отримаємо таблицю, яку наведено на малюнку 6.31.



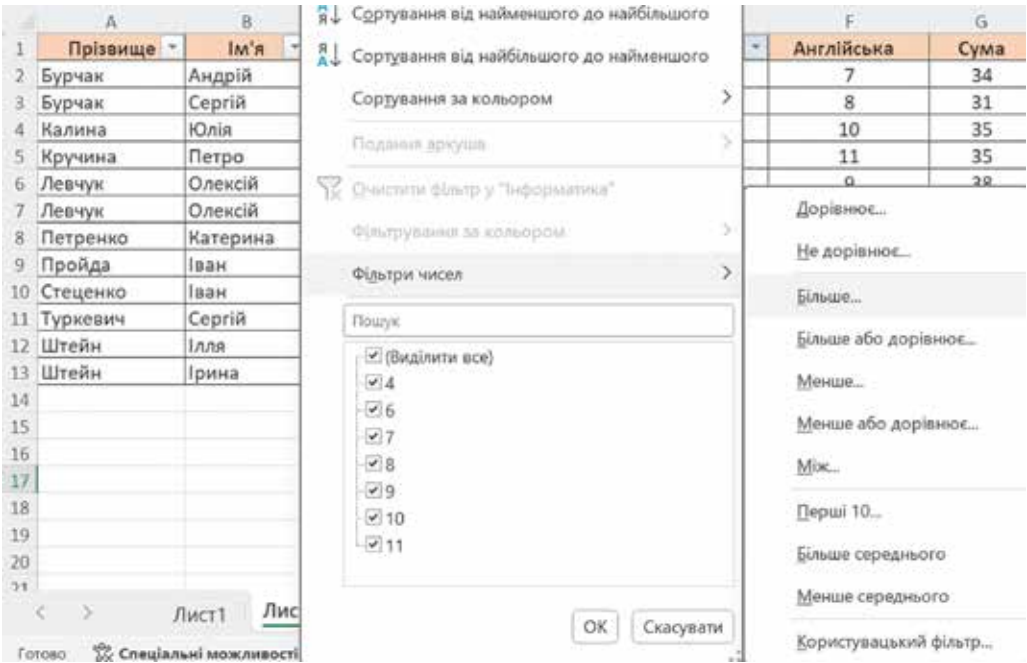
Мал. 6.30. Фільтрування за двома значеннями стовпця **Ім'я**

	A	B	C	D	E	F	G
1	Прізвище	Ім'я	Алгебра	Геометрія	Інформатика	Англійська	Сума
6	Левчук	Олексій	7	9	9	9	38
7	Левчук	Олексій	7	9	8	9	33
9	Пройда	Іван	6	10	11	8	36
10	Стеценко	Іван	9	11	8	8	31

Мал. 6.31. Результат застосування фільтра, наведеного на малюнку 6.30

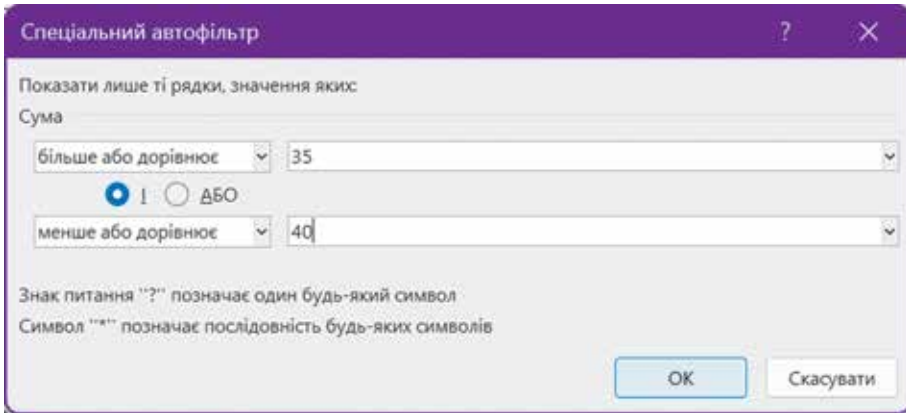
Якщо у стовпці міститься більше **числових даних**, то за даними цього стовпця можна виконувати фільтрування з використанням **Фільтрів чисел**, застосовуючи команди **Дорівнює**, **Не дорівнює**, **Більше**, **Менше**, **Між** тощо (мал. 6.32).

Наприклад, потрібно відобразити дані тих учнів, які мають сумарний бал за предметами  $\geq 35$  і  $\leq 40$ . Для цього слід виконати:



Мал. 6.32. Команди встановлення умов фільтрування числових даних

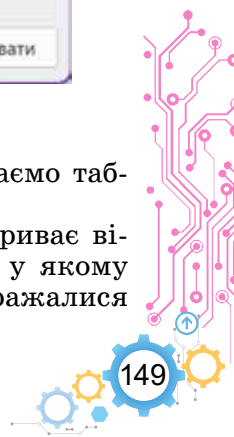
1. Застосувати фільтр до діапазону з даними.
2. Відкрити список команд фільтрування у стовпці Сума.
3. Вибрати **Фільтри чисел** ⇒ **Між**.
4. Увести у відповідні поля вікна потрібні дані **35** та **40** (мал. 6.33).
5. Вибрати кнопку **ОК**.



Мал. 6.33. Установлення числового фільтра

Застосувавши фільтр відповідно до малюнка 6.33, отримаємо таблицю, яку наведено на малюнку 6.34.

Команда **Перші 10** зі списку команд **Фільтри чисел** відкриває вікно **Автофільтр** для добору найкращої десятки (мал. 6.35), у якому можна встановити умову фільтрування, щоб у таблиці відображалися

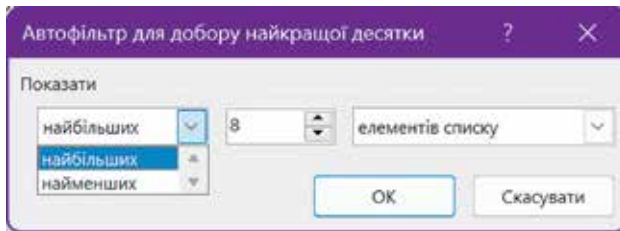


	A	B	C	D	E	F	G
1	Прізвище	Ім'я	Алгебра	Геометрія	Інформатика	Англійська	Сума
4	Калина	Юлія	9	11	10	10	35
5	Кручина	Петро	10	10	4	11	35
6	Левчук	Олексій	7	9	9	9	38
9	Пройда	Іван	6	10	11	8	36
12	Штейн	Ілля	9	8	9	9	40
13	Штейн	Ірина	9	9	10	9	37

Мал. 6.34. Результат застосування фільтра для таблиці з малюнка 6.33

тільки ті рядки таблиці, які містять у даному стовпці 10 найбільших/найменших (або іншу встановлену кількість) значень.

Команда **Перші 10** зі списку команд **Фільтри чисел** відкриває вікно **Автофільтр для добору найкращої десятки** (мал. 6.35), у якому можна встановити умову фільтрування, щоб у таблиці відображалися тільки ті рядки таблиці, які містять у даному стовпці 10 найбільших/найменших (або іншу встановлену кількість) значень.



Мал. 6.35. Вікно **Автофільтр для добору найкращої десятки**

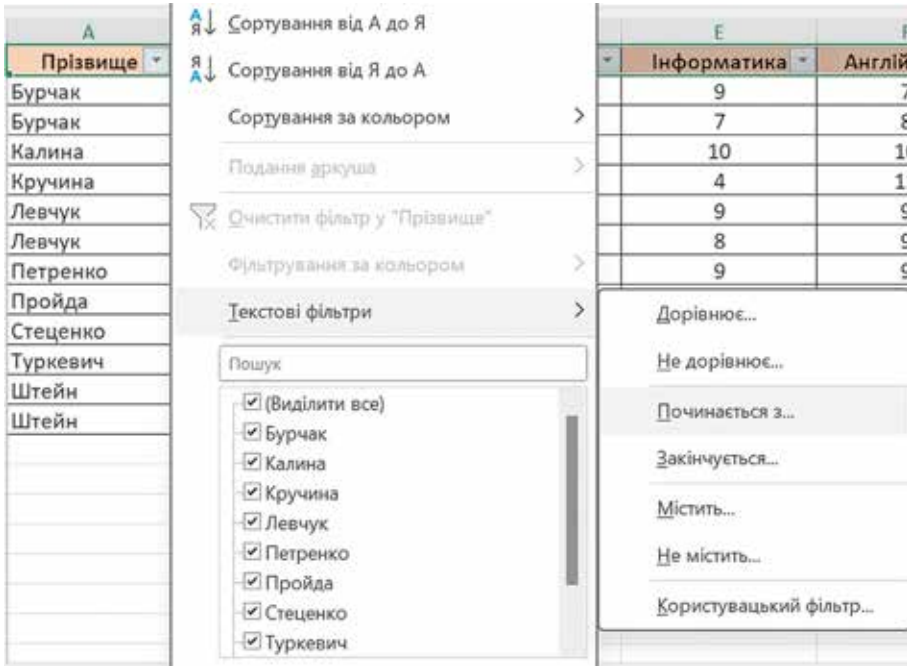
Команда **Більше середнього (Менше середнього)** виконує фільтрування, за яким відображаються тільки ті рядки таблиці, які містять у даному стовпці значення більші (менші), ніж середнє арифметичне значення в даному стовпці.

Для **текстових даних** застосовують дещо інші команди фільтрування (мал. 6.36): **Дорівнює**, **Не дорівнює**, **Починається з**, **Закінчується**, **Містить** тощо. Наприклад, потрібно відобразити дані про тих учнів/учениць, прізвища яких починаються з літери **П**. Для цього слід:

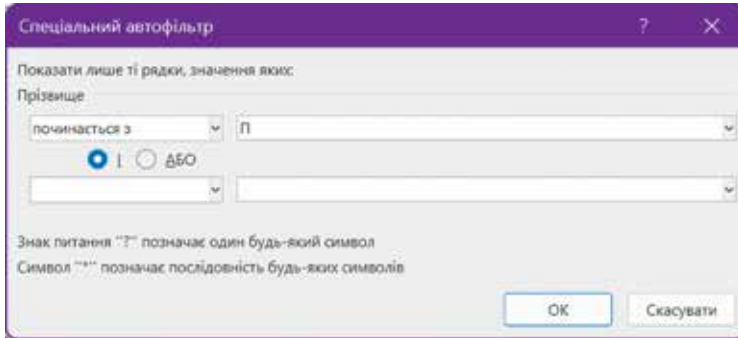
1. Застосувати фільтр до діапазону з даними.
2. Відкрити список команд фільтрування у стовпці **Прізвище**.
3. Вибрати **Текстові фільтри**  $\Rightarrow$  **Починається з...** (мал. 6.36).
4. Увести у відповідне поле літеру **П** (мал. 6.37).
5. Вибрати кнопку **ОК**.

Фільтрування також можна здійснювати послідовно для кількох стовпців таблиці. Тоді кожний наступний фільтр буде застосований вже тільки для тих рядків таблиці, які відображаються після застосування всіх попередніх фільтрів. У таких випадках загальна умова фільтрування – це умови фільтрування для кожного стовпця, об'єднані логічною операцією **І**. Наприклад, якщо провести фільтрування за даними стовпця **Сума** з умовою **більша за 35**, а потім за даними стовпця **Інформатика** з умовою **більший за 8**, то загальне фільтрування буде виконано за умовою **(сума балів більша за 35) І (бал з інформатики більший за 8)**. Результат такого фільтрування наведено на малюнку 6.38.





Мал. 6.36. Команди встановлення умов фільтрування текстових даних



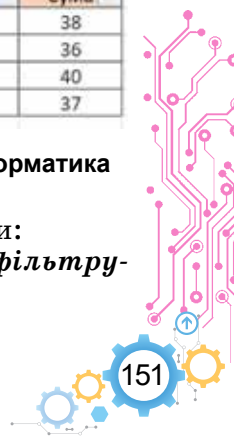
Мал. 6.37. Умова для фільтрування текстових даних

	А	В	С	Д	Е	Ф	Г
1	Прізвище	Ім'я	Алгебра	Геометрія	Інформатика	Англійська	Сума
6	Левчук	Олексій	7	9	9	9	38
9	Пройда	Іван	6	10	11	8	36
12	Штейн	Ілля	9	8	9	9	40
13	Штейн	Ірина	9	9	10	9	37



Мал. 6.38. Результат послідовного фільтрування у стовпцях **Сума** та **Інформатика**

Скасувати фільтрування даних можна кількома способами:

- виконати **Основне** ⇒ **Редагування** ⇒ **Сортувати й фільтрувати** ⇒ **Очистити**;



- виконати команду **Очистити фільтр** у списку кнопки фільтрування стовпця, за даними якого було проведено фільтрування;
- установити позначку прапорця **Виділити все** у списку кнопки фільтрування стовпця, за даними якого було проведено фільтрування, після чого вибрати кнопку **ОК**.

Щоб узагалі скасувати режим фільтрування, потрібно повторно виконати одну з команд, яка встановлює цей режим. Наприклад, виконати *Основне* ⇒ *Редагування* ⇒ *Сортувати й фільтрувати*  ⇒ *Фільтр* .



### Для тих, хто хоче знати більше

#### Розширений фільтр


Використання простих фільтрів дає змогу виконати фільтрування не в усіх випадках. Так, наприклад, розглянутими способами не можна виконати фільтрування за умовою, яка є об'єднанням умов фільтрування двох стовпців логічною операцією **АБО**, наприклад, **(сума балів більша за 35) АБО (бал з інформатики більший за 8)**.

Виконати таке фільтрування в табличному процесорі **Excel** можна з використанням **розширених фільтрів**. Для цього слід:

1. Скопіювати у вільні клітинки електронної таблиці (зручніше це зробити під заповненою таблицею) назви тих стовпців, за даними яких буде здійснюватися фільтрування (наприклад, на малюнку 6.39 назви стовпців *Сума*, *Інформатика*).

	A	B	C	D	E	F	G
1	Прізвище	Ім'я	Алгебра	Геометрія	Інформатика	Англійська	Сума
2	Бурчак	Андрій	5	9	9	7	34
3	Бурчак	Сергій	8	8	7	8	31
4	Налича	Юлія	9	11	10	10	35
5	Ірчука	Петро	10	10	4	11	35
6	Левчук	Олександр	7	9	9	9	38
7	Левчук	Олександр	7	9	8	9	33
8	Петренко	Надія	7	8	9	9	33
9	Пройда	Іван	6	10	11	8	36
10	Степанко	Іван	9	11	8	8	31
11	Турецький	Сергій	7	9	6	11	30
12	Штайн	Ілля	9	8	9	9	40
13	Штайн	Ірина	9	9	10	9	37
14							
15			Сума	Інформатика			
16			>35	>8			
17							

Мал. 6.39. Умови фільтрування та вікно **Розширений фільтр**

2. Увести у клітинки під назвами стовпців умови фільтрування. Якщо ці умови повинні об'єднуватися логічною операцією **І**, то вони мають розташовуватися в одному рядку. Якщо логічною операцією **АБО** – то в різних (на малюнку 6.39 умова *Сума > 35* або *Інформатика > 8*).
3. Виконати *Дані* ⇒ *Сортування й фільтр* ⇒ *Додатково* .
4. У вікні **Розширений фільтр**:
  1. Вибрати один з перемикачів для визначення області розташування результату фільтрування (на малюнку 6.39 – *фільтрувати список на місці*).
  2. Увести в поле **Вихідний діапазон** діапазон клітинок, дані в яких повинні фільтруватися (на малюнку 6.39 – **A1:G13**).
  3. Увести в поле **Діапазон умов** діапазон клітинок, у яких розташовано назви стовпців і умови фільтрування (наприклад, **D15:E17**).

4. Якщо був вибраний перемикач **скопіювати результат до іншого розташування**, то ввести в поле **Діапазон для результатів** діапазон клітинок, де має розміститися результат фільтрування.
5. Вибрати кнопку **ОК**.

На малюнку 6.40 наведено результат фільтрування, виконаного за умовами, наведеними на малюнку 6.39.



	A	B	C	D	E	F	G
1	Прізвище	Ім'я	Алгебра	Геометрія	Інформатика	Англійська	Сума
2	Бурчак	Андрій	9	9	9	7	34
4	Калина	Юлія	9	11	10	10	35
6	Левчук	Олексій	7	9	9	9	38
8	Петренко	Катерина	7	8	9	9	33
9	Пройда	Іван	6	10	11	8	36
12	Штейн	Ілля	9	8	9	9	40
13	Штейн	Ірина	9	9	10	9	37

Мал. 6.40. Результат фільтрування за умовами, наведеними на малюнку 6.39





### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

1. Відкрийте вказаний учителем/учителькою файл (наприклад, **Розділ 6\Пункт 6.3\вправа 6.3.xlsx**). На аркушах електронної книги наведено список учасників/учасниць підсумкових змагань з легкої атлетики.
2. На аркуші **Аркуш1** виконайте сортування таблиці за алфавітом прізвищ учасників/учасниць. Для цього:
  1. Виділіть заповнений даними діапазон клітинок **B3:G24**.
  2. Виконайте **Основне** ⇒ **Редагування** ⇒ **Сортувати й фільтрувати**  ⇒ **Настроюване сортування** .
  3. Установіть у вікні **Сортування** в першому рядку умов сортування такі значення параметрів: **Стовпець** – **Прізвище**, **Сортувати за** – **Значення клітинок**, **Порядок** – **від А до Я**.
  4. Виберіть кнопку **ОК**.
3. На аркуші **Аркуш2** виконайте сортування таблиці за значеннями в кількох стовпцях в указаному порядку: **Рік народження** (за спаданням), **Прізвище** (за зростанням), **Ім'я** (за спаданням), **Зріст** (за зростанням). Для цього:
  1. Виділіть заповнений даними діапазон клітинок **B3:G24**.
  2. Виконайте **Настроюване сортування**.
  3. Установіть у вікні **Сортування** в рядках умов сортування такі значення параметрів:
    - У першому рядку: **Стовпець** – **Рік народження**, **Сортувати за** – **Значення клітинок**, **Порядок** – **від найбільшого значення до найменшого**.
    - Додайте ще один рівень сортування, вибравши кнопку **Додати рівень**.



- У другому рядку: *Стовпець – Прізвище*, Сортувати за – *Значення клітинок*, Порядок – *від А до Я*.
  - Додайте ще один рівень сортування, вибравши кнопку **Додати рівень**.
  - У третьому рядку: *Стовпець – Ім'я*, Сортувати за – *Значення клітинок*, Порядок – *від Я до А*.
  - Додайте ще один рівень сортування, вибравши кнопку **Додати рівень**.
  - У четвертому рядку: *Стовпець – Зріст*, Сортувати за – *Значення клітинок*, Порядок – *від найменшого значення до найбільшого*.
4. Виберіть кнопку **ОК**.
  4. На аркуші **Аркуш3** відфільтруйте дані про учнів/учениць, зріст яких *більше за 1,8 м*. Для цього:
    1. Застосуйте фільтр до діапазону з даними, виконавши *Основне* ⇒ *Редагування* ⇒ *Сортувати й фільтрувати*  ⇒ *Фільтр*.
    2. Відкрийте список команд фільтрування у стовпці **Зріст**, вибравши кнопку  у стовпці **Зріст**.
    3. Виберіть *Фільтри чисел* ⇒ *Більше...*
    4. Уведіть у відповідне поле число 1,8.
    5. Виберіть кнопку **ОК**.
  5. На аркуші **Аркуш4** відфільтруйте дані про учнів/учениць, які народилися *у 2011 році в місті Києві*. Для цього:
    1. Застосуйте фільтр до діапазону з даними.
    2. Відкрийте список команд фільтрування у стовпці **Рік народження**.
    3. Зніміть усі прапорці в переліку значень, крім значення *2011*.
    4. Відкрийте список команд фільтрування у стовпці **Місце народження**.
    5. Зніміть усі прапорці в переліку значень, крім значення *Київ*.
    6. Виберіть кнопку **ОК**.
  6. На аркуші **Аркуш5** відфільтруйте дані про учнів, які мають масу, більшу за середню серед учасників/учасниць змагань. Для цього:
    1. Застосуйте фільтр до діапазону з даними.
    2. Відкрийте список команд фільтрування у стовпці **Маса**.
    3. Виберіть *Фільтри чисел* ⇒ *Більше середнього...*
  7. Збережіть книгу у вашій папці у файлі з іменем **вправа 6.3.xlsx**.
  8. Закрийте вікно **Excel**.



#### Найважливіше в цьому пункті

Дані в електронній таблиці можна впорядковувати (*сортувати*) – змінювати порядок розташування даних у рядках або стовпцях за їх значеннями.


У табличному процесорі можна виконувати сортування **за зростанням** або **за спаданням**. Дані різних типів у таблиці впорядковуються за різними правилами.

Щоб виконати сортування виділеного діапазону клітинок за значеннями одного стовпця цього діапазону, потрібно виконати *Основне* ⇒ *Редагування* ⇒ *Сортувати й фільтрувати* ⇒ *Сортування від найменшого до найбільшого* або *Сортування від найбільшого до найменшого*.

Сортування у виділеному діапазоні можна здійснювати і за значеннями кількох стовпців. Для цього слід виконати **Дані** ⇒ **Сортування й фільтр** ⇒ **Сортувати** та встановити в діалоговому вікні параметри та умови сортування. Кожний з указаних для сортування стовпців називається *рівнем сортування*. При цьому спочатку рядки сортуються за першою умовою, а потім на наступному рівні сортування відбувається лише для тих рядків, у яких значення в усіх попередніх стовпцях сортування збігаються.

**Фільтрування** – це автоматизований відбір значень у клітинках електронної таблиці, що відповідають певним умовам. Після виконання фільтрування в таблиці відображаються лише ті рядки, дані в яких відповідають умовам фільтрування. Усі інші — тимчасово приховуються.

Для встановлення режиму *простого* фільтрування потрібно встановити табличний курсор у довільну клітинку заповненої даними електронної таблиці або виділити потрібний діапазон клітинок і виконати **Основне** ⇒ **Редагування** ⇒ **Сортувати й фільтрувати** ⇒ **Фільтр** або **Дані** ⇒ **Сортування й фільтр** ⇒ **Фільтр**.

Після виконання цих дій біля правої межі заголовків стовпців усієї заповненої даними таблиці або виділеного діапазону клітинок з'являться кнопки  – відкриття списку команд фільтрування. У цьому списку містяться команди сортування та фільтрування за значеннями (числовими, текстовими або інших типів), а також за кольором.

Фільтрування можна здійснювати послідовно для кількох стовпців таблиці. Тоді кожний наступний фільтр буде застосовано тільки для тих рядків таблиці, які відображаються після застосування всіх попередніх фільтрів. У таких випадках загальна умова фільтрування – це умови фільтрування для кожного стовпця, об'єднані логічною операцією І. Для виконання фільтрування за умовою, яка є об'єднанням умов фільтрування двох стовпців логічною операцією **АБО**, використовують *розширені фільтри*.

Для скасування фільтрування потрібно виконати **Дані** ⇒ **Сортувати й фільтрувати** ⇒ **Очистити**.



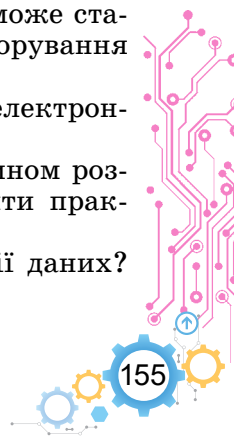
### Дайте відповіді на запитання

1. У чому полягає сортування даних у Excel?
2. Які правила сортування даних за зростанням у Excel?
3. Як здійснюється сортування даних у таблицях за значеннями в кількох стовпцях?
4. Коли команда **Сортування від найменшого до найбільшого** змінюється на команду **Сортування від А до Я**?
5. Що таке *фільтрування даних*? Які види фільтрування можна здійснювати для числових даних? Для текстових даних?
6. Для чого використовують простий фільтр?



### Обговоріть і зробіть висновки

1. Під час сортування таблиці на екрані з'явилося повідомлення про розширення діапазону сортування. У якому випадку це може статися? Які дії користувача в такому разі? Які наслідки ігнорування цього повідомлення?
2. Що спільного та відмінного в операціях сортування в електронних таблицях і текстових документах?
3. Якщо виконувати сортування за кольором, то яким чином розмістяться кольори у відсортованій таблиці? Як це перевірити практично?
4. Як використовувати кольорові фільтри для організації даних? Наведіть кілька прикладів.





### Виконайте завдання

1. Відкрийте вказаний учителем/учительською файл, наприклад **Розділ 6\Пункт 6.3\зразок 6.3.1.xlsx**. На аркуші **Аркуш1** відсортуйте таблицю за зростанням часу відправлення поїздів. На аркуші **Аркуш2** відсортуйте таблицю так, щоб дані про поїзди, що відправляються з одного пункту, йшли за алфавітом, дані про кінцевий пункт слідування розміщувались у порядку від **Я до А**, а час відправлення – зростав. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.3.1.xlsx**.

2. Відкрийте вказаний учителем/учительською файл, наприклад **Розділ 6\Пункт 6.3\зразок 6.3.2.xlsx**. На аркуші **Аркуш1** відсортуйте таблицю за значеннями у трьох стовпцях: **Бали** (за спаданням), **Час** (за зростанням), **Висота** (за спаданням). На аркуші **Аркуш2** впорядкуйте дані у стовпці *Прізвище* за зростанням, а у стовпці *Продано* – за спаданням. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.3.2.xlsx**.

3. Відкрийте вказаний учителем/учительською файл, наприклад **Розділ 6\Пункт 6.3\зразок 6.3.3.xlsx**. На аркуші **Аркуш1** відсортуйте дані про пристрої в порядку спадання ціни та зростання ємності пристроїв. На аркуші **Аркуш2** відсортуйте дані про пристрої в алфавітному порядку фірми-виробника, за спаданням діагоналі та спаданням ціни. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.3.3.xlsx**.

4. Відкрийте вказаний учителем/учительською файл, наприклад **Розділ 6\Пункт 6.3\зразок 6.3.4.xlsx**. На аркуші **Аркуш1** відобразіть дані лише про поїзди, які слідують до Львова. На аркуші **Аркуш2** відобразіть 5 найкоротших за часом поїздки маршрутів. На аркуші **Аркуш3** відобразіть дані про поїзди, які відправляються до Харкова з 9 год до 15 год. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.3.4.xlsx**.

5. Відкрийте вказаний учителем/учительською файл, наприклад **Розділ 6\Пункт 6.3\зразок 6.3.5.xlsx**. На аркуші **Аркуш1** відобразіть дані про учасників/учасниць, які набрали у фехтуванні не менше ніж 40 очок і стрибнули у висоту від 2,20 м до 2,25 м. На аркуші **Аркуш2** виберіть книги, прибуток від продажу яких більший за 1000 грн та ім'я авторів яких *Дмитро* або *Микола*. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.3.5.xlsx**.

6. Відкрийте вказаний учителем/учительською файл, наприклад **Розділ 6\Пункт 6.3\зразок 6.3.6.xlsx**. На аркуші **Аркуш1** задайте умову фільтрування для добору даних про пристрої, у яких ємність більша ніж 20 000 мА і ціна менша ніж 3000 грн. На аркуші **Аркуш2** задайте умову фільтрування для добору даних про пристрої, ціна яких менша ніж середня. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.3.6.xlsx**.



### Готуємось до вивчення нового матеріалу

1. Що розуміють під форматуванням текстових об'єктів? Яким чином це можна виконати?
2. Як відформатувати символи у клітинках електронної таблиці?
3. Як надрукувати текстовий документ? Які властивості сторінки документа потрібно визначити?

## 6.4. УМОВНЕ ФОРМАТУВАННЯ. ДРУКУВАННЯ ЕЛЕКТРОННОЇ ТАБЛИЦІ

У цьому пункті йтиметься про:

- ▶ умовне форматування даних в електронній таблиці;
- ▶ друкування електронної таблиці.

### УМОВНЕ ФОРМАТУВАННЯ



#### Пригадайте

• Що таке *сортування даних електронної таблиці*? Якими способами його можна виконати? • Що розуміють під форматуванням клітин і даних у електронній таблиці? Як це можна виконати?

Ще одним способом виділити в таблиці дані, які задовольняють певним умовам, є так зване **умовне форматування**.

**Умовне форматування змінює формат клітинки або її вмісту на заданий, якщо дані в цій клітинці задовольняють певній умові.**

Умовне форматування застосовується до виділеного діапазону клітинок і виконується на основі правил, які складаються з **умови**, що стосується значення у клітинці, та **значень властивостей** форматування вмісту клітинки.

Наприклад, у електронному журналі класу зручно позначати різними кольорами дані про учнів, які мають оцінки низького, середнього, достатнього та високого рівнів. Тобто клітинки, значення яких знаходяться в діапазоні від 1 до 3, наприклад, виділити червоним кольором, у діапазоні від 4 до 6 – зеленим, у діапазоні від 7 до 9 – фіолетовим, у діапазоні від 10 до 12 – синім кольором (мал. 6.41).

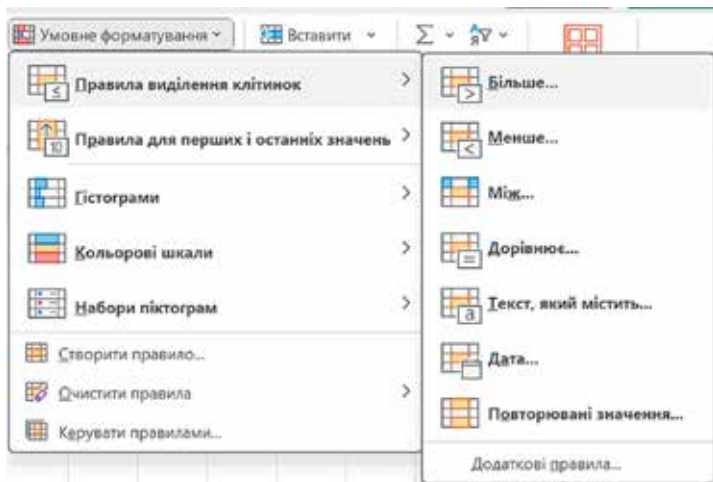
	А	В	С
1	Прізвище	Ім'я	Інформатика
2	Бурчак	Андрій	9
3	Бурчак	Сергій	7
4	Калина	Юлія	10
5	Кручина	Петро	4
6	Левчук	Олексій	3
7	Левчук	Олексій	8
8	Петренко	Катерина	3
9	Пройда	Іван	11
10	Стеценко	Іван	8
11	Туркевич	Сергій	6
12	Штейн	Ілля	5
13	Штейн	Ірина	10

Мал. 6.41. Форматування вмісту клітинок відповідно до умов

В Excel існує 5 типів правил виділення клітинок під час умовного форматування (мал. 6.42):


- **Правила виділення клітинок;**
- **Правила для перших і останніх значень;**
- **Гістограми;**
- **Кольорові шкали;**
- **Набори піктограм.**

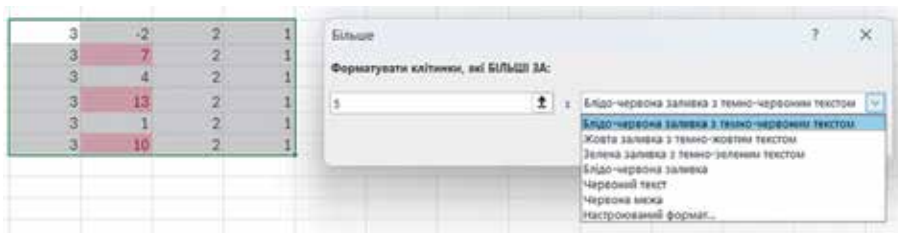




Мал. 6.42. Кнопка **Умовне форматування**, список типів правил, а також список правил типу **Правила виділення клітинок**

Для встановлення перших двох правил умовного форматування, а саме **Правила виділення клітинок** і **Правила для перших і останніх значень**, слід:

1. Виділити потрібний діапазон клітинок, у якому буде застосовуватись умовне форматування.
2. Виконати *Основне* ⇒ *Стилі* ⇒ *Умовне форматування* .
3. Вибрати у списках кнопки **Умовне форматування** потрібний тип правил (мал. 6.42), наприклад **Правила виділення клітинок**.
4. Вибрати у списку обраного типу потрібне правило форматування, наприклад **Більше...**
5. Задати у вікні, що відкриється, умову форматування та формат, який буде встановлений, якщо умова буде виконуватися, а саме значення властивостей фону і символів тексту (мал. 6.43). Формат можна вибрати з наведених або, вибравши в кінці списку команду **Настроюваний формат**, установити власний.
6. Вибрати кнопку **ОК**.

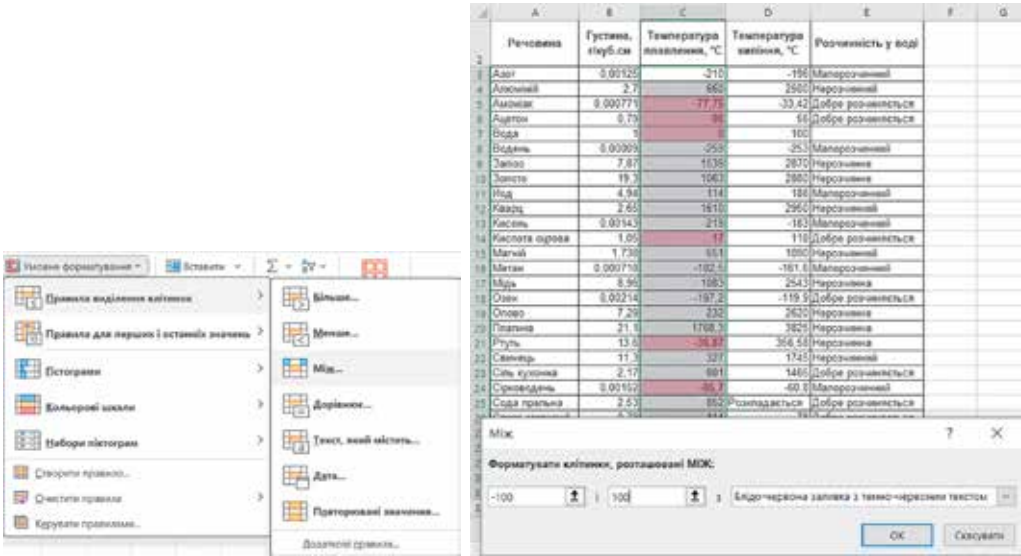


Мал. 6.43. Установлення умови та формату для умовного форматування

На малюнку 6.44 наведено, як приклад, вікно умови **Між**, у якому встановлено умову форматування **Між –100 і 100 градусів Цельсія**

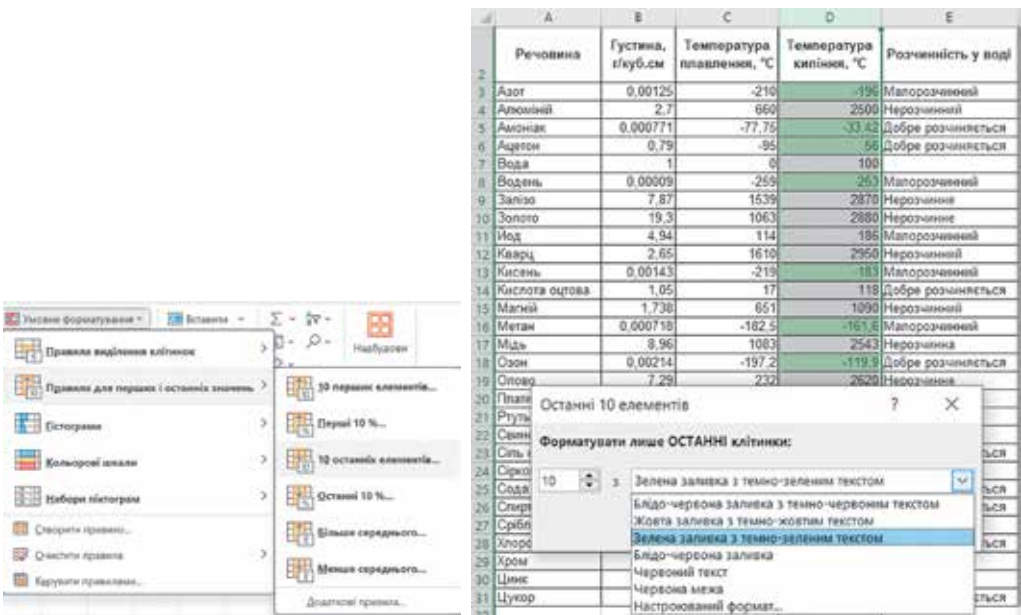


як значення температури плавлення речовин з вибраним стандартним форматом *Блідо-червона заливка з темно-червоним текстом*, а також попередній перегляд результатів вибраного умовного форматування.

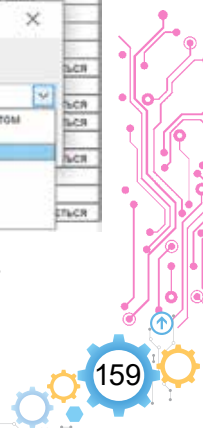


Мал. 6.44. Виділений діапазон клітинок і вікно **Між** з умовою форматування

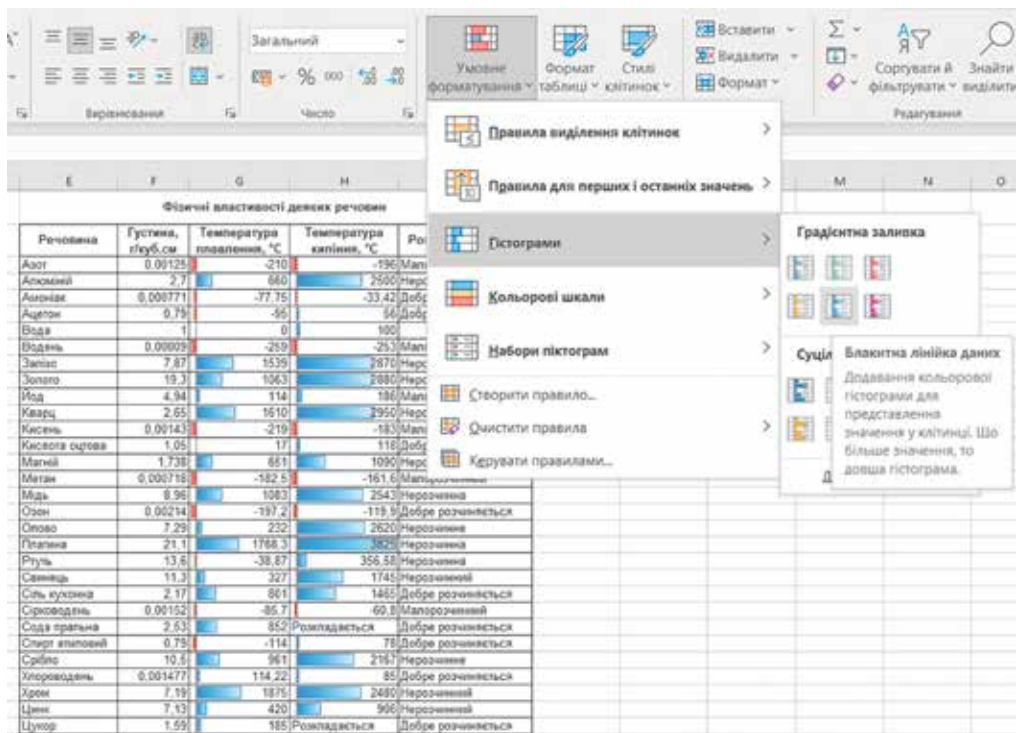
А на малюнку 6.45 наведено установлення та попередній перегляд умовного форматування **10 останніх елементів** (10 найменших значень) з вибраним форматом *зелена заливка з темно-зеленим текстом*.



Мал. 6.45. Виділений діапазон клітинок, вікно **Останні 10 елементів** з умовою і списком стандартних форматів



Використання іншого правила умовного форматування **Гістограми** приводить до вставлення у клітинки виділеного діапазону гістограм, довжина кольорового рядка яких пропорційна значенню у клітинці, а колір позначає додатні та від'ємні значення (мал. 6.46).

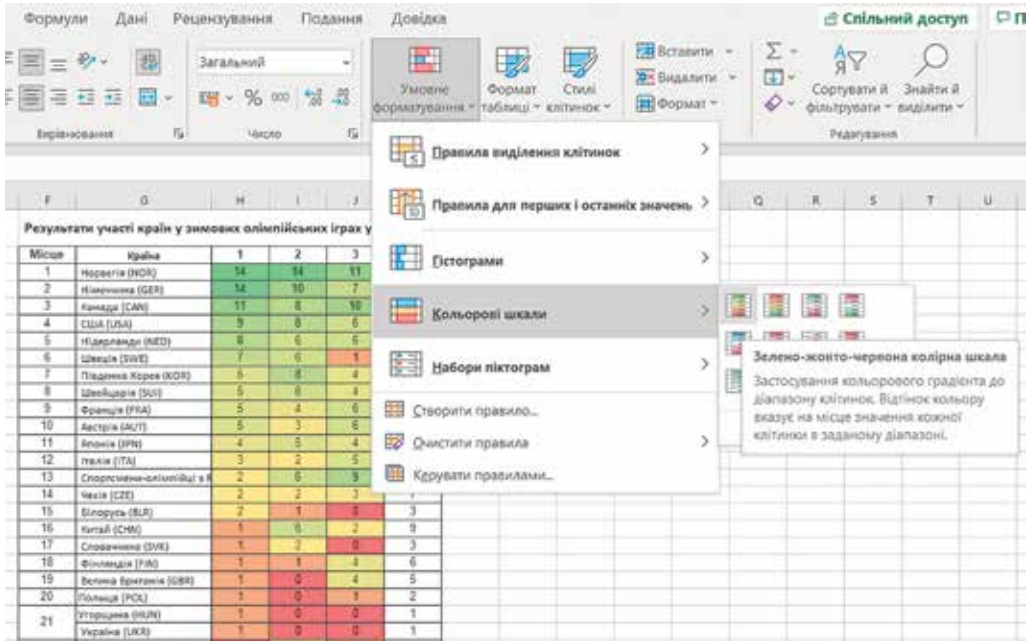


Мал. 6.46. Умове форматування з уставленням у клітинки гістограм

Використання умовного форматування **Кольорові шкали** приводить до градієнтної заливки клітинок виділеного діапазону встановленими кольорами. Клітинки з однаковими значеннями мають один і той самий колір, а насиченість чи відтінок кольорів для клітинок з різними значеннями змінюється в межах кольорів вибраного формату (мал. 6.47).

Можна також вибрати правило умовного форматування зі списку **Набори піктограм**. За такого форматування у клітинках виділеного діапазону з'являтимуться піктограми з вибраного набору. Поява конкретної піктограми з набору в клітинці означає, що значення в цій клітинці задовольняє умові, встановленій для кожної піктограми набору.

*Звертаємо вашу увагу!* До одного діапазону клітинок може бути застосовано кілька правил умовного форматування, установлених *по-слідовно*. У такому разі умови цих правил перевіряються по черзі. І, якщо вони не суперечать одна одній, то умовне форматування відбувається *по-слідовно*. Наприклад, установити червоний колір символів для чисел, які *більше за 10*, та застосувати для того самого діапазону форматування у вигляді *гістограм*.



Мал. 6.47. Умове форматування зелено-жовто-червоною колірною шкалою

Для видалення умовного форматування потрібно виконати **Умове форматування** ⇒ **Очистити правила** ⇒ **Очистити правила у виділених клітинках** або **Очистити правила у всьому аркуші**.

## ДРУКУВАННЯ ЕЛЕКТРОННОЇ ТАБЛИЦІ



### Пригадайте

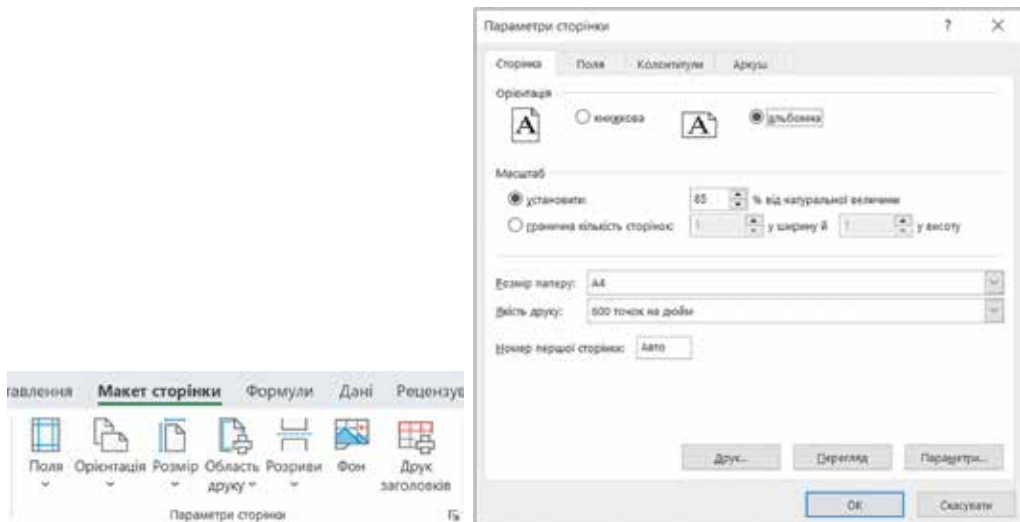
- Які властивості має сторінка текстового документа? Які значення цих властивостей ви можете навести?
- Як надрукувати текстовий документ у текстовому процесорі **Word**?

Ви вже вмієте друкувати текстові документи і знаєте, що перед тим, як надрукувати підготовлений документ, потрібно встановити значення властивостей його сторінок (розмір аркуша, орієнтацію сторінки, ширину полів тощо), потім у режимі **Попереднього перегляду** переглянути зовнішній вигляд документа і надрукувати його.

Схожим є й алгоритм друкування електронної таблиці, але з певними відмінностями.

Установлення значень властивостей сторінки документа, який планується друкувати, здійснюється з використанням елементів керування групи **Параметри сторінки** вкладки **Макет сторінки** або на вкладках діалогового вікна **Параметри сторінки** (мал. 6.48), яке відкривається вибором кнопки **Параметри сторінки** у нижньому правому куті групи **Параметри сторінки**.

На вкладці **Сторінка** можна встановити значення виду орієнтації аркуша паперу під час друкування (*альбомна* або *книжкова*); масштаб



Мал. 6.48. Група **Параметри сторінки** та діалогове вікно **Параметри сторінки**

таблиці у відсотках від натуральної величини, розмір аркуша паперу для друкування; якість друку, номер першої сторінки в надрукованому документі та інші.

На вкладці **Поля** можна встановити значення таких властивостей друкування: розміри полів, висота колонтитулів, спосіб центрування таблиці на аркуші та інші.

На вкладці **Колонтитули** можна створити верхній і нижній колонтитули, вибравши відповідну кнопку, або вибрати потрібний колонтитул у полях зі списком **Верхній колонтитул** або **Нижній колонтитул** тощо.

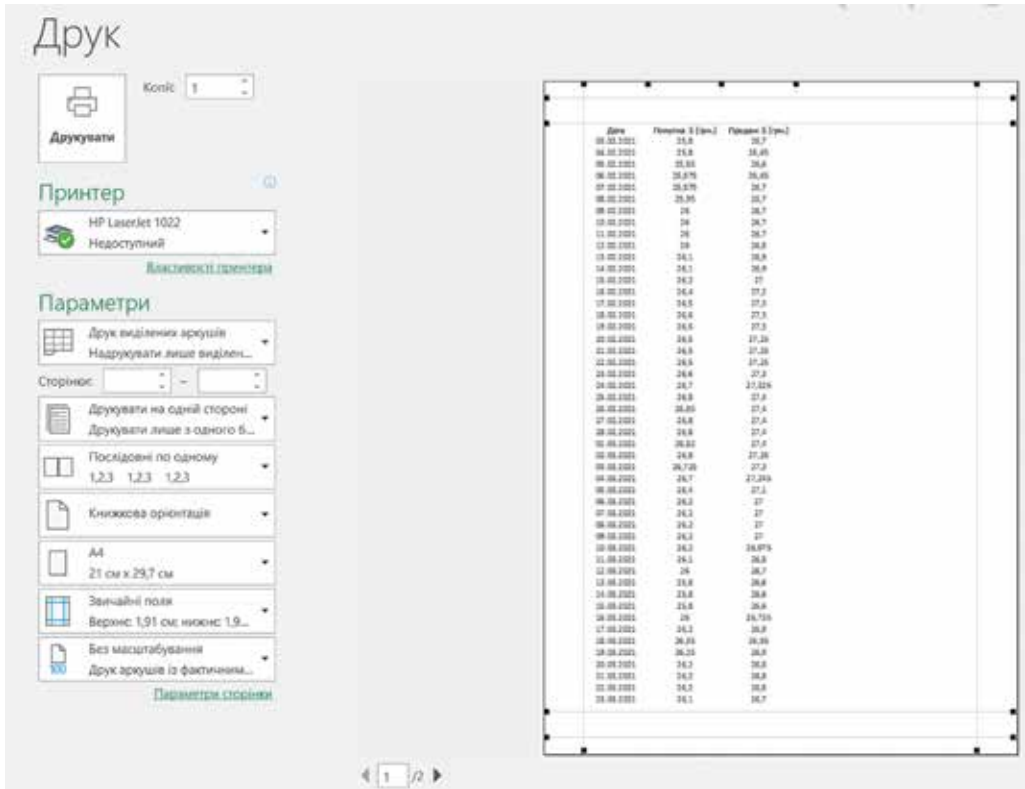
На вкладці **Аркуш** можна встановити значення таких властивостей друкування:

- область друкування – діапазон клітинок аркуша електронної таблиці, який буде виводитися на друк;
- друк наскрізних рядків і стовпців на кожній сторінці – діапазон клітинок, уміст якого друкуватиметься як підписи рядків і стовпців на кожній сторінці, коли надрукована таблиця на папері займає більше ніж одну сторінку;
- якість друку (чернетка, чорно-білий, сітка тощо);
- порядок друку сторінок – вниз, потім управо чи вправо, потім униз тощо.

Вибором кнопки **Перегляд** у вікні **Параметри сторінки** можна встановити режим **Попереднього перегляду**, щоб побачити, як виглядатиме надрукована електронна таблиця на папері. Це можна зробити також, виконавши **Файл** ⇒ **Друк**.


У цьому режимі буде відображено таблицю, підготовлену до друку, яка розділена на сторінки. Якщо область друкування не вміщується на одному аркуші паперу заданого розміру, табличний процесор **Excel** автоматично розділяє цю область на кілька аркушів (такий випадок наве-


дено на мал. 6.49). Уміст на окремих аркушах паперу можна переглянути, обираючи потрібну сторінку в Області перегляду ◀ 1 /2 ▶.

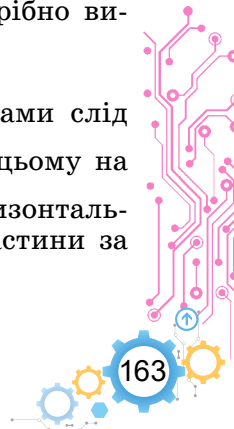


Мал. 6.49. Вкладка **Друк** режиму **Попередній перегляд**

На вкладці **Друк** також можна змінити значення різних властивостей друкування сторінок електронної таблиці – розміри полів, розмір аркуша паперу, орієнтацію аркуша тощо, а також налаштувати параметри друкування – кількість копій, поточний принтер, об’єкт для друкування (виділений діапазон, або виділені аркуші чи всю книгу, або окремі сторінки з указаними номерами), послідовність друку частин таблиці електронної книги, масштаб тощо.

Після того, як всі параметри друкування електронної таблиці визначені та попередній вигляд задовольняє користувача, потрібно звернути на вкладці **Друк** кнопку **Друкують** .

Для повернення з вкладки **Друк** в основне вікно програми слід вибрати кнопку  у верхній частині бічного меню. При цьому на аркуші електронної таблиці з’являються вертикальні та горизонтальні штрихові лінії, які демонструють розбиття таблиці на частини за розмірами аркушів для друку (мал. 6.50).



D	E	F	G	H
Дата	Покупка \$ (грн)	Продаж \$ (грн)		
03.02.2021	25,8	26,7		
04.02.2021	25,8	26,65		
05.02.2021	25,85	26,6		
06.02.2021	25,875	26,65		
07.02.2021	25,875	26,7		

Мал. 6.50. Розбиття електронної таблиці на сторінки



### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

- Запустіть табличний процесор Excel.
- Відкрийте вказаний учителем/вчителькою файл, наприклад **Розділ 6\Пункт 6.4\вправа 6.4.xlsx**, у таблиці якого наведено відомості про найважливіші річки України.
- Установіть у стовпці **Річки** умовне форматування, щоб клітинки з назвами річок, які містять літеру **Д**, зафарбувалися *жовтим кольором з темно-жовтим текстом*. Для цього:
  - Виділіть діапазон клітинок **A3:A23**.
  - Виконайте **Основне** ⇒ **Стилі** ⇒ **Умовне форматування**




- Виберіть у списку кнопки **Умовне форматування** тип правил **Правила виділення клітинок**.
- Виберіть у списку правило форматування **Текст, який містить...**
- Задайте у вікні, що відкрилося, у відповідному полі літеру **Д**; у списку форматів – *жовта заливка з темно-жовтим текстом*.
- Виберіть кнопку **ОК**.
- Установіть у стовпці **Річний стік** умовне форматування, щоб виділити червоним кольором клітинки **5 найбільших** повноводних річок. Для цього:
  - Виділіть діапазон клітинок **D3:D23**.
  - Виконайте **Основне** ⇒ **Стилі** ⇒ **Умовне форматування**



- Виберіть у списку кнопки **Умовне форматування** тип правил **Правила для перших і останніх значень**.
- Виберіть у списку правило форматування **10 перших елементів**.
- Задайте у вікні, що відкрилося, на лічильнику значення **5**; у списку форматів – *червоний текст*.
- Виберіть кнопку **ОК**.

5. Установіть у стовпці **Довжина** умовне форматування **Кольорові шкали**, зелено-жовта-червона колірна шкала. Для цього:
  1. Виділіть діапазон клітинок **B3:B23**.
  2. Виконайте **Оснoвне** ⇒ **Стилi** ⇒ **Умовне форматування**



3. Виберіть у списку кнопки **Умовне форматування** тип правил **Кольорові шкали**.
4. Виберіть у списку правило форматування *Зелено-жовто-червона колірна шкала*.
5. Задайте у вікні, що відкрилось, на лічильнику значення **5**; у списку форматів – *червоний текст*.
6. Виберіть кнопку **ОК**.
6. Виконайте попередній перегляд електронної книги. Для цього:
  1. Виконайте **Файл** ⇒ **Друк**.
  2. Перегляньте електронну таблицю в режимі **Попередній перегляд**.
  3. Ознайомтеся з інтерфейсом вікна **Друк**.
7. Надрукуйте таблицю на принтері, який укаже вчитель/вчителька, на папері формату А4 альбомної орієнтації, розмістивши таблицю на одному аркуші. Для цього:
  1. Установіть властивості сторінок електронної таблиці:
    - Принтер – *вказаний учителем/учителькою*;
    - Номери сторінок для друку – *1-2*;
    - Орієнтація аркуша паперу – *Альбомна*;
    - Розмір аркуша паперу – *А4*;
    - Поля – *Широкі*;
    - Настроюване масштабування – *Розмістити аркуш на одній сторінці*.
  2. Виберіть кнопку **Друкувати**.
8. Поверніться в основне вікно програми, вибравши кнопку  у верхній частині бічної панелі.
9. Збережіть книгу у вашій папці у файлі з іменем **вправа 6.4.xlsx**.
10. Закрийте вікно **Excel**.



### Найважливіше в цьому пункті

Одним зі способів виділити в таблиці дані, які задовольняють певним умовам, є так зване **умовне форматування**. Умовне форматування змінює формат клітинки або її вмісту на заданий, якщо дані в цій клітинці задовольняють певній умові.

Умовне форматування застосовується до виділеного діапазону клітинок і виконується на основі правил, які складаються з **умови**, що стосується значення у клітинці, та **значень властивостей** форматування вмісту клітинки.

В **Excel** існує **5 типів** виділення клітинок під час умовного форматування: *виділення клітинок за певними правилами, виділення перших або останніх значень діапазону, форматування клітинок у вигляді гістограм, кольорових шкал чи наборів піктограм*.

Для встановлення цих видів правил форматування для виділеного фрагмента електронної таблиці використовуються відповідні команди з меню кнопки **Умовне**

**форматування**

До одного діапазону клітинок може бути застосовано кілька

правил умовного форматування, заданих послідовно.

Установлення значень параметрів сторінки електронної таблиці для друкування здійснюється на **Стрічці** елементами керування групи **Параметри сторінки** на вкладці **Макет сторінки** або на вкладках діалогового вікна **Параметри сторінки** цієї групи.

Щоб переглянути, як буде виглядати електронна таблиця на папері, потрібно перейти до режиму **Попередній перегляд**, який встановлюється **Файл** ⇒ **Друк**. У цьому режимі на вкладці **Друк** можна додатково налаштувати параметри друкування – кількість копій, вибір принтера, об'єкт для друкування (виділений діапазон, або виділені аркуші чи всю книгу, або окремі сторінки з указаними номерами), послідовність друку сторінок електронної книги, масштаб тощо.

Щоб надрукувати електронну таблицю після узгодження параметрів друку, потрібно вибрати кнопку **Друкувати** на вкладці **Друк**.

**Дайте відповіді на запитання**

1. Що таке *умовне форматування*? Із чого складаються правила умовного форматування?
2. Які типи умовного форматування існують в Excel?
3. Як виконати умовне форматування? Як очистити умовне форматування на всьому аркуші?
4. Як надрукувати електронну таблицю?
5. Для чого використовують попередній перегляд сторінок електронної таблиці? Які дії можна виконувати в цьому режимі?
6. Які властивості друкування електронної таблиці можна встановити?

**Обговоріть і зробіть висновки**

1. Що спільне та відмінне між умовним форматуванням та фільтрацією?
2. Як буде відформатовано діапазон клітинок, якщо до нього застосовують два різних правила, які між собою несумісні? Перевірте це на практиці і сформулюйте загальне правило виконання умовного форматування.
3. Як ви вважаєте, що потрібно зробити, щоб надрукувати всю електронну таблицю на одному аркуші паперу?
4. Що спільного та відмінного у друці текстового документа та електронної таблиці?

**Виконайте завдання**

1. Відкрийте вказаний учителем/учителькою файл, наприклад Розділ 6\Пункт 6.4\зразок 6.4.1.xlsx. На аркуші **Аркуш1** наведено таблицю з даними про монітори. Установіть рожевий колір тла для клітинок, у яких діагональ монітора менша ніж 21". Установіть зелену заливку з темно-зеленим текстом у клітинках 7 найдорожчих моніторів. Збережіть книгу у вашій папці у файлі з іменем **завдання 6.4.1.xlsx**.
2. Відкрийте вказаний учителем/учителькою файл, наприклад Розділ 6\Пункт 6.4\зразок 6.4.2.xlsx. На аркуші **Аркуш1** наведено



таблицю з даними про зовнішні накопичувачі. Установіть *синій* колір фону для клітинок, у яких ціна більша за 2000 грн. Установіть *блідо-рожеву* заливку клітинок, ємність яких менша від середньої. Збережіть книгу у вашій папці у файлі з іменем завдання 6.4.2.xlsx.

3. Знайдіть в Інтернеті прогноз погоди на 10 днів для м. Києва. Створіть електронну таблицю та вставте знайдені дані на **Аркуш1**. Установіть умовне форматування (*жовта заливка з темно-жовтим текстом*) для *повторювальних* значень температури вдень. Установіть умовне форматування **Гістограми** (*фіолетова лінійка даних*) для значень температури вночі. Збережіть книгу у вашій папці у файлі з іменем завдання 6.4.3.xlsx.

4. Знайдіть в Інтернеті дані про 10 найбільших гір України. Створіть електронну таблицю та вставте знайдені дані на **Аркуш1**. Установіть умовне форматування з використанням **Наборів піктограм** (*4 світлофори*) для числових даних таблиці. Виділіть *червоним кольором* назви гір, у яких міститься літера Г. Збережіть книгу у вашій папці у файлі з іменем завдання 6.4.4.xlsx.

5. Відкрийте вказаний учителем/учителькою файл, наприклад **Розділ 6\Пункт 6.4\зразок 6.4.5.xlsx**. Виконайте попередній перегляд сторінок перед друкуванням електронної таблиці з аркуша **Аркуш1**. Виконайте друкування першої сторінки цієї електронної таблиці з такими значеннями властивостей друкування: кількість копій – 2, орієнтація аркуша – книжкова, поля – по 1 см, сторінки за замовчуванням. Збережіть книгу у вашій папці у файлі з іменем завдання 6.4.5.xlsx.



### Практична робота № 9.

#### «Вибірання даних в електронних таблицях»

1. Відкрийте вказаний учителем/учителькою файл, наприклад **Розділ 6\Пункт 6.4\зразок практична 9.xlsx**.
2. На аркуші **Аркуш1** виконайте сортування таблиці за зростанням значень у стовпці **Фірма, що взяла кредит**.
3. На аркуші **Аркуш2** виконайте сортування таблиці за значеннями в кількох стовпцях в указаному порядку: **Банк-кредитор** (за зростанням), **Сума кредиту** (за зростанням), **Річний відсоток** (за спаданням).
4. На аркуші **Аркуш3**, використавши фільтрування, відобразіть дані про фірми, сума кредиту яких більша за 750 000, а річний відсоток менший ніж 15 %.
5. На аркуші **Аркуш4**, використавши фільтрування, відобразіть дані про фірми, які взяли кредит на 6 місяців або рік в банку «Ажіо».
6. На аркуші **Аркуш5** установіть умовне форматування для клітинок стовпця **Сума кредиту**, якщо їх значення більше за середнє у стовпці, вибравши для них зелений колір заливки.
7. Збережіть електронну книгу у вашій папці у файлі з іменем **практична 9.xlsx**.



#### Готуємось до вивчення нового матеріалу

1. Що таке *комп'ютерна мережа*? Які її складові? Для чого вони призначені?



2. Скільки пристроїв підключено до Інтернету у вас удома? Поясніть, навіщо це зроблено.
3. Чи чули ви термін «розумний» дім? Як ви його розумієте?



### Оцініть свої знання та вміння

Оцініть свої навчальні досягнення з розділу «Опрацювання даних в електронних таблицях» (початковий, середній, достатній, високий рівень).

- Я розумію призначення та функції табличного процесора для опрацювання числових даних.
- Я знаю типи об'єктів електронної таблиці, їх властивості та можливі операції над ними.
- Я знаю формати числових даних і використовую їх під час обчислень.
- Я вмію добирати та аргументувати вибір засобів табличного процесора для розв'язування певного кола навчальних і життєвих задач.
- Я знаю різні типи посилань на клітинки та використовую їх для обчислень.
- Я вмію обрати спосіб візуального подання даних засобами табличного процесора та застосовувати його.
- Я вмію створювати електронні таблиці для реалізації різних моделей об'єктів, явищ і процесів.
- Я вмію використовувати табличний процесор для опрацювання числових даних, отримання результатів з використанням вбудованих функцій (математичних, статистичних, логічних тощо).
- Я вмію виконувати сортування та фільтрацію даних.
- Я вмію застосовувати умовне форматування.
- Я знаю значення властивостей друкування та вмію їх налаштувати під час друку.

# SMART – ТЕХНОЛОГІЇ. ШТУЧНИЙ ІНТЕЛЕКТ

У цьому розділі ви отримаєте нові, а також поглибите та розширите свої знання й удосконалили навички з таких тем:

- ▶ «розумні» речі та Smart-технології;
- ▶ Інтернет речей та основні принципи його функціонування;
- ▶ штучний інтелект;
- ▶ використання штучного інтелекту в різних галузях.

## 7.1. ІНТЕРНЕТ РЕЧЕЙ І ЙОГО ВИКОРИСТАННЯ. SMART-ТЕХНОЛОГІЇ

У цьому пункті йтиметься про:

- ▶ «розумні» речі та Smart-технології;
- ▶ Інтернет речей;
- ▶ основні принципи функціонування мережі «Інтернет речей».

### «РОЗУМНІ» РЕЧІ ТА SMART-ТЕХНОЛОГІЇ

#### Пригадайте

- Що ви знаєте про «розумні» пристрої, «розумний» дім? Як ви розумієте ці терміни?
- Чи є у вас «розумні» пристрої? Для чого ви їх використовуєте?

Раніше на уроках інформатики ви вже ознайомилися з поняттям «розумні» пристрої, або **Smart-речі** (англ. *Smart* – розумний, енергійний, кмітливий). Наприклад, «розумний» годинник з підтримкою функції відслідковування серцевого ритму, фітнес-трекер, Smart-окуляри, окуляри віртуальної реальності тощо (мал. 7.1).



Мал. 7.1. Приклади «розумних» речей

Ці гаджети мають невеликі розміри та малу вагу, їх зручно носити із собою. Вони можуть мати камеру, термометр, барометр, компас, GPS-навігатор. Також їх можна під'єднати до локальної мережі чи Інтернету для збирання й передавання даних, які вони отримують. Через Інтернет за допомогою спеціальних програм ними можна керувати відділено (дистанційно), що сприяє прийнят-

тю оперативних рішень, організації особистого життя та спілкування тощо.

Наприклад, «розумна» годівниця призначена для віддаленого керування годуванням тварин і керується через додаток у смартфоні. Годівниця видає встановлену їжу в певний час і повідомляє хазяїна про її закінчення. Пластир-градусник, який через певні інтервали часу визначає температуру тіла людини, передає дані на смартфон для аналізу лікарем або пацієнтом у спеціальній програмі.

З використанням «розумних» пристроїв створюють «розумні» будинки, оснащуючи оселю різноманітними датчиками та сенсорами. Пристрої збирають дані та передають їх через Інтернет у спеціальну програму керування «розумним» домом на сервері, смартфоні господарів або пульті керування (мал. 7.2).



Мал. 7.2. Функціонування системи «розумний» дім

Ця система, наприклад, аналізує дані навколишнього середовища і, залежно від показників, регулює температуру в приміщенні. У зимовий період регулює інтенсивність опалення, а в разі спекотної погоди – вмикає кондиціонер. За допомогою розумних пристроїв у домі може працювати охоронна та пожежна сигналізація, налаштовуватися використання електроенергії із сонячних батарей.

Також власник «розумного» будинку може з мобільного телефона через Інтернет керувати різними пристроями у своєму домі. Наприклад, включити полив трави, відчинити/зачинити двері, включити/виключити пральну машинку тощо.

Водночас украй важливим у світі «розумних» пристроїв постає питання **інтернет-безпеки**. Якщо елементи такої системи не будуть належним чином захищені від несанкціонованого втручання, замість

користі вони можуть принести шкоду. Оскільки такі пристрої можуть зберігати багато інформації про їх власника, зокрема можуть знати його точне місцезнаходження. Доступ кіберзлочинців до такої інформації і до комп'ютерної системи керування може призвести до порушення інформаційної безпеки та скоєння злочину, а в гірших випадках і нести загрозу життю та здоров'ю людей.

Усі такі технології, які забезпечують взаємодію цифрових пристроїв з оточуючим середовищем і наділяють цю систему здатністю до автоматичного прийняття рішень, віддаленого керування задля ефективного виконання щоденних справ, побутових і виробничих завдань, називають **Smart-технологіями**.

На основі розвитку цих технологій виникають нові поняття: *Smart-міста*, *Smart-країни*, *Smart-освіта*, *Smart-економіка*, *Smart-суспільство* (мал. 7.3). В основі цього «розумного» суспільства лежить розвиток «суспільства знань», цифрових технологій, усього того, що приведе до цифрової ери розвитку нашої цивілізації.



Мал. 7.3. Smart-місто

## ІНТЕРНЕТ РЕЧЕЙ

### Пригадайте

● Що таке *комп'ютерна мережа*? Які пристрої можуть бути приєднані до комп'ютерної мережі? ● Що таке *Інтернет*? Яким чином комп'ютерні пристрої підключаються до Інтернету?



«Розумні» пристрої, «розумні» будинки є прикладами реалізації найпопулярнішої наукової ідеї сучасної інформатики, яка зараз активно втілюється в життя, – **Інтернету речей**. Ці технології кардинально впливають на розвиток сучасного суспільства, оскільки забезпечують можливість багатьом процесам відбуватися автоматично, без безпосередньої участі людини, але під її контролем.

**Інтернет речей** (англ. *Internet of Things*, скорочено *IoT*) – це сукупність пристроїв, оснащених датчиками, сенсорами, засобами опрацювання, передавання та прийому сигналів, підключених до глобальної мережі «Інтернет», з метою передавання власних даних та отримання команд.

У такому цифровому середовищі створюються нові умови для бізнесу, охорони здоров'я, забезпечення екологічної безпеки, змінюються особисті та соціальні аспекти життя (мал. 7.4).



Мал. 7.4. Інтернет речей і його використання

Так, уже зараз лікарі з використанням переносних датчиків, які встановлено на пацієнті, можуть віддалено відслідковувати його стан

здоров'я та реагувати на зміни в режимі реального часу. Автодорожні служби встановлюють спеціальні датчики на автомобільних шляхах для відслідковування швидкості автомобілів, автоматичної фіксації порушень і відправлення інформації про цей факт у патрульну службу та водієві.

Для запобігання лісовим пожежам використовують мережу «Лісовий дозор». У лісах встановлюють систему датчиків температури та відеокамери, дані з яких передаються у програму для аналізу ситуації та сповіщають чергових у разі пожежі.

Наразі широкої популярності набули моніторинг стану водного середовища, моніторинг атмосферного повітря, а також сейсмічної активності. Використовуються датчики, які вимірюють уміст вуглекислого газу, вологість повітря, кількість радіоактивних елементів і передають ці дані на обробку в центри керування. Такі пристрої можуть знаходитися в різних куточках планети, у тому числі у важкодоступних місцях, і працювати через Wi-Fi, Bluetooth, GPS.

Підключення пристроїв до Інтернету дає людям більше можливостей для раціонального управління ресурсами: витрачання газу, води, світла, видобутку різних видів енергії тощо. Наприклад, стало холодніше – вмикається опалення в усіх будинках міста із центральним опаленням, стало тепліше – вимикається. Увечері зменшився рівень освітленості на вулиці, то відповідно вмикається вуличне освітлення; світлофори переходять у режим «увага» в певний час доби, коли зменшується потік автомобілів на дорозі.

Автоматизація вирішення в такий спосіб багатьох життєвих потреб і виробничих завдань, оперативного ухвалення рішень надає можливість людині більше часу приділяти сім'ї, творчості, хобі, професійному та особистому розвитку.

### Цікаві факти з історії

Термін *Інтернет речей* уперше використав американський фахівець з телекомунікацій Пітер Льюїс у 1985 році. Він описав **IoT** як «інтеграцію людей, процесів і технологій з пристроями для підключення і датчиками для забезпечення віддаленого моніторингу, стану, маніпулювання та оцінки тенденцій таких пристроїв».

Пізніше, у 1999 році, термін *Інтернет речей* підтримав і засновник дослідницького центру **Auto-ID Center** у Масачусетському технологічному інституті Кевін Ештон (нар. 1968 р.) (мал. 7.5). Він висловив припущення, що згодом у кожній з речей реального фізичного світу в **IoT** буде цифровий двійник, її віртуальна копія.

Напряму **IoT** став активно розвиватися, коли на початку 2000-х років кількість пристроїв, підключених до мережі «Інтернет», перевищила кількість користувачів Інтернету. Тобто *Інтернет речей* перевищив *Інтернет людей*.

Період з 2008 по 2009 роки аналітики корпорації **Cisco** вважають «справжнім народженням Інтернету речей», тому що саме в цей період кількість пристроїв, підключених до глобальної мережі, перевищила чисельність населення Землі. А за даними компанії **Ericsson**, у 2018 році кількість таких пристроїв перевищила кількість мобільних телефонів у світі. У 2021 році кількість **IoT**-пристроїв перевищила 12 млрд одиниць, а до 2025 року може досягти 41 млрд. І надалі їх кількість буде зростати.



Мал. 7.5. Кевін Ештон



У зв'язку з такими глобальними масштабами використання IoT, набуває поширення новий термін – **Всеохопний Інтернет, Інтернет всього** (англ. *Internet of Everything, IoE*), який із часом може прийти на зміну IoT.

## ОСНОВНІ ПРИНЦИПИ ФУНКЦІОНУВАННЯ МЕРЕЖІ «ІНТЕРНЕТ РЕЧЕЙ»

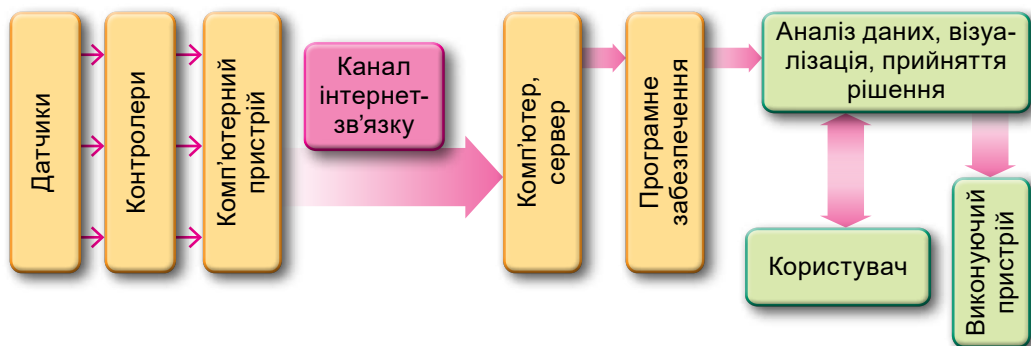
### Пригадайте

- Що таке *IP-адреса комп'ютера*? Із чого вона складається?
- Що таке *інформаційна система*? Які її складові?

Інтернет речей ґрунтується на таких базових **принципах**:

- наявність глобальної комп'ютерної мережі з високою швидкістю;
- до мережі під'єднуються різноманітні пристрої, які мають конкретну IP-адресу в мережі;
- пристрої мають можливість отримувати дані та відправляти їх по мережі;
- робота пристроїв у автоматизованому режимі відбувається за спеціальними програмами.

На малюнку 7.6 наведено загальну схему функціонування мережі «Інтернет речей».



Мал. 7.6. Функціонування мережі «Інтернет речей»

Для відслідковування стану навколишнього середовища пристрої оснащують **датчиками (сенсорами)**, які отримують інформацію про зовнішні умови в реальному часі. Наприклад, датчики для вимірювання температури, тиску, швидкості руху, місця розташування, вологості та інші. У більшості випадків вони мають автономні джерела живлення.

Датчик вимірює фізичні параметри деякого об'єкта/явища та перетворює їх у цифровий сигнал, який може бути переданий і прийнятий відповідним комп'ютерним пристроєм. Для опрацювання та накопичення даних з датчиків використовують **вбудовані комп'ютерні пристрої**, до яких датчики приєднуються через **контролери**. **Контролери** – це спеціальні пристрої, що з'єднують елементи системи один з одним і зв'язують систему із зовнішнім світом.

Якщо ж опрацювання отриманих даних планується на локальному комп'ютері, смартфоні, сервері, то для обміну даними між пристроями використовують різні **канали передавання даних (бездротові**



або *дротові мережі*). Водночас важливим є забезпечення надійності та конфіденційності передавання даних.

Для аналізу даних і прийняття рішення комп'ютерні пристрої використовують відповідне **програмне забезпечення**, яке за певними алгоритмами опрацьовує отримані дані. Ці програми можуть установлюватися на мобільні пристрої, у вбудовані комп'ютери або використовувати хмарні технології.

Результати обробки отриманих даних у вигляді відповідних сигналів поступають на так звані **виконуючі пристрої** (*актуатори, кінцеві пристрої*). Це цифрові пристрої, які безпосередньо виконують команди, наприклад, автоматичні вимикачі, підйомники, розетки, сирени, клапани, двигуни тощо.

Більшість операцій в Інтернеті речей відбувається без участі людини, але при цьому **користувач** може взаємодіяти з будь-яким пристроєм: віддавати команди, інструкції, отримувати доступ до інформації.

Зручність, економічність, різноманітність варіантів використання **IoT** робить Інтернет речей найважливішою технологією XXI століття, яка буде продовжувати набирати обертів. У міру того, як дедалі більше компаній усвідомлюють потенціал Інтернету речей, збільшуватиметься і кількість підключених пристроїв і продовжиться розвиток функціональних можливостей.



### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

1. Перегляньте відеоролик про проектування та побудову «розумної» теплиці за адресою <https://cutt.ly/LeLdEokb> або QR-кодом. Запишіть, які датчики використовують у відео та для яких цілей. Які показники ними вимірюються? Які результати планується отримати? Які пристрої використано у проекті? Відповівши на запитання, створіть у зошиті схему функціонування «розумної» теплиці.
2. Відкрийте на смартфоні додаток **Play Market** або аналогічний. Знайдіть додаток *Arduino Science Journal* і завантажте його. Установіть автоматичний переклад сторінки українською мовою. Ознайомтеся з дослідженням «Яскравість». Виконайте це дослідження. Поясніть, які датчики смартфона було використано в цьому дослідженні та для яких цілей. Відповіді у вигляді звіту про дослідження оформіть у текстовому документі та збережіть у вашій папці у файлі з іменем **вправа 7.1.2**.



### Найважливіше в цьому пункті

«**Розумні**» пристрої, або **Smart-печі**, – це цифрові пристрої, які під'єднують до локальної мережі чи Інтернету для збирання й передавання даних з використанням різноманітних датчиків. Вони мають невеликі розміри й незначну масу, їх зручно носити із собою.

З використанням розумних пристроїв створюють «**розумні**» будинки. Датчики та сенсори цих пристроїв збирають дані та передають через Інтернет у спеціальну



програму керування «розумним» домом на сервері, смартфоні господарів або пульті керування.

Під час використання «розумних» пристроїв важливим є питання **інтернет-безпеки**. Якщо елементи такої системи не будуть належним чином захищені від несанкціонованого втручання, замість користі вони принесуть шкоду.

Технології, які забезпечують взаємодію цифрових пристроїв з навколишнім середовищем і наділяють цю систему здатністю до автоматичного ухвалення рішень, віддаленого керування задля ефективного виконання щоденних справ, побутових і виробничих завдань, називаються **Smart-технологіями**.

На основі розвитку цих технологій виникають нові поняття: *Smart-міста*, *Smart-країни*, *Smart-освіта*, *Smart-економіка*, *Smart-суспільство*. В основі цього «розумного» суспільства лежить розвиток «суспільства знань», цифрових технологій, усього того, що приведе до цифрової ери розвитку нашої цивілізації.

**Інтернет речей** (англ. *Internet of Things*, скорочено **IoT**) – це сукупність пристроїв, оснащених датчиками, сенсорами, засобами опрацювання, передавання та прийому сигналів, підключених до глобальної мережі «Інтернет» з метою передавання власних даних і отримання команд.

У такому цифровому середовищі створюються нові умови для бізнесу, охорони здоров'я, забезпечення екологічної безпеки, змінюються особисті та соціальні аспекти життя. Виникають можливості автоматизації вирішення багатьох життєвих потреб і виробничих завдань, оперативного ухвалення рішень.

Інтернет речей ґрунтується на таких базових **принципах**:

- наявність глобальної комп'ютерної мережі;
- до мережі під'єднуються різноманітні пристрої (датчики, контролери, актуатори та інші), які мають конкретну IP-адресу в мережі;
- пристрої мають можливість отримувати дані та відправляти їх по мережі;
- робота пристроїв у автоматизованому режимі відбувається за спеціальними програмами.

Зручність, економічність, різноманітність варіантів використання **IoT** робить Інтернет речей найважливішою технологією XXI століття, яка буде продовжувати набирати обертів.



### Дайте відповіді на запитання

1. Які ви можете навести приклади використання «розумних» речей? Поясніть їх функціонування.
2. Що розуміють під *Smart-технологіями*? Чи використовують такі технології у вашому населеному пункті?
3. У чому сутність питання *інтернет-безпеки* під час використання «розумних» пристроїв?
4. Що таке *Інтернет речей*? Із чого він складається?
5. Які базові принципи функціонування Інтернету речей?
6. Як функціонує мережа «Інтернет речей»?



### Обговоріть і зробіть висновки

1. Яке значення має Інтернет речей у житті людей?
2. Які негативні аспекти Інтернету речей ви можете назвати? Чим це може нашкодити суспільству; людям; природі?
3. Яким чином Інтернет речей створює новий життєвий простір?
4. Чи можна вважати Інтернет речей, «розумний» будинок інформаційними системами? Відповідь поясніть.



**Виконайте завдання**

1. Перегляньте в Інтернеті відеоролик про Інтернет речей за адресою <https://cutt.ly/aeLdRwVE> або QR-кодом. Підготуйте на основі відео презентацію про використання Інтернету речей і «розумних» пристроїв. Збережіть результат роботи у вашій папці у файлі з іменем **завдання 7.1.1.**



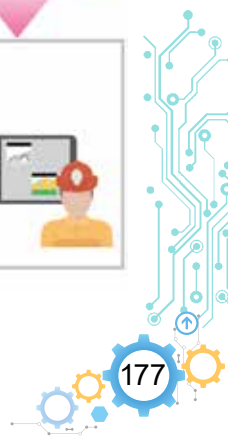
2. Знайдіть в Інтернеті інформацію про використання Інтернету речей у різних сферах економіки. Підготуйте групове повідомлення у вигляді відеоролика. Збережіть результат роботи у вашій папці у файлі з іменем **завдання 7.1.2.**

3. Опишіть функціонування «розумних» систем на наведених прикладах у текстовому документі. Збережіть результат роботи у вашій папці у файлі з іменем **завдання 7.1.3.**

**а) Фітнес-браслет**



**б) «Розумний» завод**





4. Знайдіть в Інтернеті інформацію про використання Smart-технологій в освіті та навчанні. Підготуйте колективну презентацію на тему «Smart-освіта сьогодні», використовуючи онлайн-редактор презентацій **Google**. Збережіть результат роботи у вашій папці у файлі з іменем **завдання 7.1.4**.

5. Знайдіть в Інтернеті інформацію про створення в різних країнах світу сучасних Smart-міст. Оформіть результати роботи в текстовому документі у вигляді буклета та збережіть його у вашій папці у файлі з іменем **завдання 7.1.5**.



6. Ознайомтесь з відео щодо підключення в домі «розумної» розетки за посиланням <https://cutt.ly/EeLdTKU3>. Спробуйте встановити її у вашому домі. Зніміть відео, як ви це робили. Які зручності та переваги принесе цей пристрій вашій сім'ї? Поясніть це у своєму відеозвіті.



7. Створіть реальну «розумну» теплицю (завдання № 1 у рубриці «Працюємо з комп'ютером»), використавши свої знання з фізики, технологій, інформатики тощо.



8. Виконайте проект «Наша Smart-школа», у якому передбачте підключення різноманітних датчиків для автоматизації функціонування школи. Складіть схему реалізації вашого проекту. Розрахуйте орієнтовний кошторис реалізації такого проекту, використавши ціни на обладнання з інтернет-магазинів. Збережіть результати роботи у вашій папці у файлі з іменем **завдання 7.1.8**.

9. Виконайте проект «Мій “розумний” дім», у якому передбачте підключення різноманітних датчиків для автоматизації функціонування будинку/квартири. Складіть схему реалізації вашого проекту. Розрахуйте орієнтовний кошторис реалізації такого проекту. Збережіть результати роботи у вашій папці у файлі з іменем **завдання 7.1.9**. Виконайте завдання разом з батьками чи рідними. Під час виконання проекту скористайтесь інформацією із сайту за адресою <https://220lviv.top/rozumnyj-budynok/>.



10. Перегляньте статтю за посиланням <http://surl.li/jiqily> та ознайомтесь з прикладами використання Інтернету речей у майбутньому. Визначте, які перспективи розвитку суспільства та життя людей у новому цифровому світі. Створіть презентацію. Збережіть результат роботи у вашій папці у файлі з іменем **завдання 7.1.10**.



11. Відкрийте сторінку онлайн-курсу Вступ до IoT та цифрової трансформації від мережної академії Cisco за адресою <http://surl.li/usbmjf>. Ознайомтесь зі змістом курсу, зареєструйтеся в ньому та пройдіть навчання. Після завершення покажіть свій сертифікат учителю/вчительці.



### Готуємось до вивчення нового матеріалу

1. Що таке, на вашу думку, *штучний інтелект*? Чи використовували ви його? Для яких цілей?
2. Які ви бачите переваги та недоліки використання штучного інтелекту в школі?

## 7.2. ПОНЯТТЯ ШТУЧНОГО ІНТЕЛЕКТУ. ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В РІЗНИХ ГАЛУЗЯХ

У цьому пункті йтиметься про:

- ▶ штучний інтелект і напрями розробок систем штучного інтелекту;
- ▶ використання штучного інтелекту в різних галузях;
- ▶ переваги та недоліки використання штучного інтелекту;
- ▶ штучний інтелект у навчанні.

### ШТУЧНИЙ ІНТЕЛЕКТ



#### Поміркуйте

- Що, на вашу думку, означає слово *розум*? Яку людину вважають *розумною*?
- Що таке *навчання*? Як навчаєтеся ви? ● Що таке *алгоритм*; *математична модель*? Де й коли ви використовуєте ці терміни?

Одним із найсучасніших, найпрогресивніших і найнеоднозначніших напрямів розвитку інформатики як науки є штучний інтелект (ШІ).

Про це свідчить хоча б той факт, що у 2024 році дві з трьох наукових нагород Нобелівської премії отримали за відкриття в галузі штучного інтелекту – у галузі фізики та хімії.

Перші згадування про штучний інтелект можна знайти ще у стародавньому світі в міфах про створення штучних істот, яких майстри наділили розумом чи свідомістю.

Сьогодні використання ШІ стало невід'ємною частиною життя сучасної людини. Сканер відбитків пальців, розпізнавання обличчя в телефоні, друкування тексту за допомогою автонабору тексту, спілкування із чат-ботом у онлайн-магазині, голосові помічники у смартфоні, система «Розумний будинок», автопілоти в машинах, сервіс **Google Maps**, електронний додаток таксі **Uber** – усе це функціонує на основі штучного інтелекту.

**Штучний інтелект** (англ. *artificial intelligence*) – це область інформатики, яка здійснює розробку інтелектуальних комп'ютерних систем і програм, що імітують роботу людського мозку.

Такі інтелектуальні системи та програми запрограмовані виконувати певні дії, як люди: розуміти мову, навчатися, міркувати, робити висновки та передбачати, вирішувати проблеми, ухвалювати рішення тощо. Інакше кажучи, у рамках штучного інтелекту вивчають способи розв'язування комп'ютером задач, що не мають явного алгоритмічного розв'язку.

Сьогодні створення систем штучного інтелекту ведеться за такими напрямками:

1. **Створення комп'ютерних систем, що імітують діяльність людини** (наприклад, емоції, мовлення, жести, відчуття, творчість тощо). Це так звані програмно-апаратні асистенти, помічники, що заміняють людей під час виконання деяких робіт.

**Інтелект** (лат. *intellectus* – відчуття, сприйняття, розуміння, розум) – здатність до пізнання і вирішення труднощів, яка об'єднує всі пізнавальні здібності людини: відчуття, сприйняття, пам'ять, уявлення, мислення.



Наприклад, людиноподібного робота Софію (мал. 7.7) було створено в 2015 році гонконгською компанією Hanson Robotics у вигляді жінки. Вона наділена штучним інтелектом, здатна імітувати 62 вирази обличчя, установлювати зоровий контакт, розпізнавати обличчя, запам'ятовувати людей і підтримувати з ними діалог. Вона постійно вдосконалюється, стаючи «розумнішою».



Мал. 7.7. Людиноподібний робот Софія

**2. Створення комп'ютерних систем на основі використання біологічних елементів** (наприклад, *нейрокомп'ютер, біокомп'ютер*). Нейрокомп'ютер розробляється для прямого зв'язку між мозком людини та зовнішніми пристроями.

Наприклад, керування пристроями силою думки через уживлені в мозок датчики можна використати для людей з обмеженими фізичними можливостями. Датчики, які при цьому використовуються, реагують на сигнали мозку, декодують їх і перетворюють у спеціальні команди для комп'ютерного пристрою (мал. 7.8).



Мал. 7.8. Елементи нейрокомп'ютера

**3. Створення комп'ютерних систем, які імітують логічне мислення людини на основі використання систем логічного програмування** (наприклад, мов програмування *Пролог, Лісп* та інші). Такі системи ухвалення рішень (*експертні системи*) використовують для опрацювання великих неструктурованих даних з урахуванням багатьох факторів (діагностування).

Однією з перших логічних мов програмування був *Пролог* (1972 рік), і залишається на сьогодні найпопулярнішою серед таких мов для розробки так званих експертних систем. Програма на *Пролозі* не є такою у традиційному розумінні, оскільки не містить базових алгоритмічних конструкцій. Основне призначення програми – не виконати послідовність дій і отримати потрібний результат, а знайти рішення на основі аналізу набору фактів за заданими правилами побудови логічного висновку (мал. 7.9).

**4. Створення комп'ютерних систем, які будуть так званими інтелектуальними агентами**, що сприймають навколишній світ за допомогою датчиків і впливають на об'єкти в навколишньому середовищі за допомогою деяких механізмів.

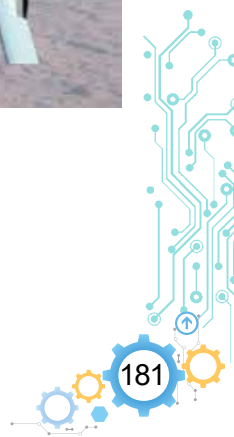


Мал. 7.9. Схема отримання висновку на мові Пролог

Наприклад, автоматизовані блоки (мал. 7.10) керування в літаку, автомобілі чи БпЛА, які можуть в автономному режимі без участі людини керувати рухомих об'єктом, виконувати певні дії, аналізуючи при цьому швидкість, траєкторію, перешкоди та інші зовнішні фактори. Або це може бути пошуковий робот, який збирає для користувача інформацію за потрібним запитом, аналізуючи дані в мережі «Інтернет» або на визначених носіях, обирає найбільш підходящі та зберігає відібрані дані у вказаному місці або повідомляє користувачу.



Мал. 7.10. Автоматизовані блоки керування та пристрої – автопілот метро, літальний апарат



## Для тих, хто хоче знати більше

Існує кілька основних типів алгоритмів, за якими працює штучний інтелект: *машинне навчання*, *глибинне навчання*, *генетичні алгоритми* тощо. Одним з ключових алгоритмів є **машинне навчання**, яке надає комп'ютерній системі алгоритми, як на основі великих обсягів даних робити висновки, ймовірнісні умовиводи, знаходити закономірності, без їх явного програмування на такий вид завдань. Це так званий *етап навчання ШІ*. І ці ж отримані результати в подальшому система ШІ використовує вже як свої знання та отриманий досвід. Отримуючи нові вхідні дані, система застосовує до них свої знання та досвід і може сформулювати на цій основі свої висновки, свої рішення. Це другий етап – *використання ШІ*. Таким чином, така інтелектуальна система отримує нові знання, тобто вчиться так, як це робить людина. Від цього система стає продуктивнішою, ефективнішою та точніше опрацьовує дані (мал. 7.11).

### Етап навчання штучного інтелекту («накопичення знань»)



### Етап застосування штучного інтелекту



Мал. 7.11. Як працює машинне навчання

Основою систем штучного інтелекту є **нейронні мережі**, які імітують роботу людського мозку в цифровому вигляді. Вони складаються із цифрових нейронів, які обробляють інформацію та передають її далі через свої зв'язки по своїй мережі.

У реалізації алгоритмів роботи систем штучного інтелекту лежать певні математичні моделі, кожна з яких має свої особливості та застосування:

- *мовні моделі* – використовують для обробки та генерації тексту;
- *генеративні моделі* – здатні створювати новий контент (зображення, текст або музику);
- *дискримінативні моделі* – використовують для класифікації та розпізнавання об'єктів.

Для більшого розуміння цих питань пропонуємо переглянути відеоролик та ознайомитися з основними поняттями ШІ більш детально за посиланням <http://surl.li/fjlops>.





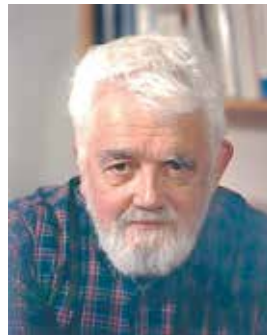


### Цікаві факти з історії

Активні розробки зі створення ШІ розпочалися в середині 50-х років ХХ ст. Влітку 1956 року в Університеті Дартмута в США пройшла перша робоча конференція науковців з проблем штучного інтелекту. Саме тоді і з'явився сам термін «штучний інтелект», який увів Джон Маккарті (американський інформатик, 1927–2011) на тій самій конференції.

У цей період почали створювати перші інтелектуальні системи, розробляти алгоритми їх роботи, але відсутність потужних комп'ютерів не давала можливості отримати вагомих результатів. Потім фінансування розробок у галузі ШІ було призупинено на довгий період часу, настала «Зима ШІ».

Інвестиції та інтерес до штучного інтелекту відродилися на початку ХХ ст., коли для опрацювання величезних обсягів даних у багатьох задачах науки та промисловості стали успішно застосовувати потужні та надпотужні комп'ютери. Це, своєю чергою, забезпечувало розробку методології роботи інтелектуальних систем, створювалися, удосконалювалися та розповсюджувалися моделі нейронних мереж для різних сфер застосування, накопичувалися датасети для навчання мереж, розроблялись нові алгоритми ШІ.



Мал. 7.12.  
Джон Маккарті

## ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В РІЗНИХ ГАЛУЗЯХ ДІЯЛЬНОСТІ ЛЮДИНИ



### Поміркуйте

- Чи використовували ви онлайн-перекладачі для перекладів тексту? Онлайн-карти для побудови маршруту переміщення? • Скільки разів ви сьогодні використовували «розумні» пристрої, штучний інтелект? Наведіть приклади.

2 грудня 2020 року в Україні Розпорядженням Кабінету Міністрів України № 1556-р було схвалено Концепцію розвитку штучного інтелекту в Україні <http://surl.li/orqejn>, яка передбачає визначення основних напрямів і пріоритетних завдань розвитку технологій штучного інтелекту з метою забезпечення конкурентоспроможності національної економіки та захисту технологічних інформаційно-комунікаційних систем.

Штучний інтелект знаходить широке застосування в різних галузях – на виробництві, у побуті, науці, медицині, освіті, банківській сфері та торгівлі, управлінні та менеджменті, логістиці, на транспорті, у сільському господарстві тощо. Його використовують для автоматизації виконання рутинних завдань, оптимізації бізнес-процесів, аналізу великих обсягів даних, розпізнавання образів, управління системами та створення інтелектуальних помічників.

В Україні розробками в галузі штучного інтелекту активно займаються співробітники Інституту проблем штучного інтелекту, Інституту кібернетики імені В. М. Глушкова НАН України, інших наукових установ та ІТ-компаній.



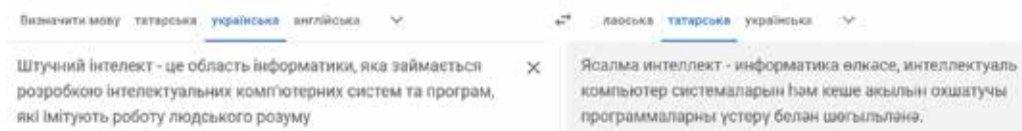
Більш детально з прикладами використання ШІ в різних сферах діяльності людини можна ознайомитися, переглянувши відеоролик за посиланням <http://surl.li/mmloo0>.

Узагальнюючи, можна виокремити, що сьогодні штучний інтелект у світі використовується в таких напрямках:

- **Розпізнавання об'єктів** (текстів, мови, графічних об'єктів на зображеннях, емоцій, запахів, шумів тощо): розпізнавання обличчя людей на смартфоні і з камер відеоспостережень (мал. 7.13), аналіз аерокосмічних фотографій і геодезичних карт, перетворення графічних зображень сканованих текстів у текстовий документ тощо. Наприклад, розпізнавання тексту успішно виконують програми *ABBY FineReader*, *Office Lens* для смартфонів, *Online OCR*.
- **Машинний переклад** текстів різними мовами, наприклад з використанням комп'ютерних програм *Amazon Translate*, *DeepL Pro*, *SYSTRAN Translate Pro* *Google Перекладач* (мал. 7.14) тощо.



Мал. 7.13.  
Розпізнавання обличчя



Мал. 7.14. Переклад тексту в **Google Перекладач**

- **Аналітична діяльність, експертні системи** – наприклад, підбір квитків на транспорт з пересадками, прокладання оптимального маршруту по карті (мал. 7.15, діагностика захворювань, створювання нових ліків, автопілот літака та автотранспорту, управління ядерним реактором тощо).
- **Інтелектуальні системи інформаційної безпеки** – наприклад, розпізнавання та захист від комп'ютерних вірусів (мал. 7.16), кібератак, програми інтелектуального захисту банківських систем тощо.
- **Робототехніка** – створення та застосування *роботів* (технічних пристроїв для автоматизації людської праці), наприклад, на конвеєрних лініях автомобілів, у важкодоступних місцях вугільних шахт, у небезпечних для лю-



Мал. 7.15. Маршрут, побудований у **Google Карти**



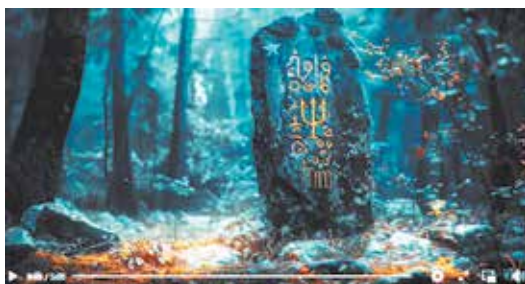
Мал. 7.16. Антивірусні засоби захисту інформаційних систем

дини місцях атомного виробництва, військовій справі (мал. 7.17) тощо.

- **Творчість та ігри** – наприклад, створення комп’ютерної музики та малювання картин (мал. 7.18), комп’ютерних програм гри у шахи, розробка інтелектуальних пристроїв-іграшок.



Мал. 7.17. Робот для дистанційного розмінування



Мал. 7.18. **Ukraine Heroica** – фільм-лауреат премії MEGOGO AI Film Festival 2024 року <http://surl.li/acexks>



### Чи знаєте ви, що...

Значний внесок у розвиток теорії штучного інтелекту зробив український учений Глушков Віктор Михайлович (1923–1982) – математик і кібернетик, засновник Інституту кібернетики АН України, академік і віцепрезидент АН України (мал. 7.19). Суть поглядів Глушкова полягала в тому, що він бачив у машині не заміник людського мозку, а спеціальний інструмент, який його посилює, як молоток підсилює руку, а мікроскоп – очі. Він уважав, що машина – це не конкурент людини, а його знаряддя, яке багаторазово збільшує можливості людини.



Мал. 7.19. Глушков В. М.

## ПЕРЕВАГИ ТА НЕДОЛІКИ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ



### Поміркуйте

● Як ви розумієте такі поняття, як *авторське право*, *академічна доброчесність*, *плагіат*? ● З якого віку можна самостійно створювати власні акаунти в різних онлайн-сервісах? ● Які правила безпечного використання Інтернету ви можете назвати?

Дедалі частіше в новинах з’являються повідомлення про те, як людина і робот змагаються в різних сферах діяльності. І не завжди результати виявляються на користь людини. Так, наприклад, у січні 2018 року компанії **Alibaba Group** і **Microsoft** провели порівняння рівня розуміння прочитаного тексту людьми та інтелектуальними системами цих компаній. Виявилося, що показник штучного інтелекту склав 82,4 бала, а найбільш високий показник серед людей – 82,3 бала. Людина, наприклад, програє системам штучному інтелекту в грі у шахи. У довготривалому керуванні літаком штучний інтелект «не втомлюється», на відміну від людини, і точно слідкує за показниками на автопілоті. А влітку 2024 року штучний інтелект отримав срібну медаль на міжнародній учнівській олімпіаді з математики.

У зв'язку із цим у багатьох виникає етичне питання – чи зможе штучний інтелект повністю замінити людину? Із цього приводу дискусії ведуться вже майже сто років. І однозначної відповіді на сьогодні немає.

Одним з головних викликів використання систем ШІ сьогодні є те, чи завжди ШІ ухвалює правильні рішення на основі отриманих даних. Наприклад, у таких критичних сферах, як охорона здоров'я або правосуддя, використання ШІ може призвести до непередбачуваних наслідків або упереджених рішень через недосконалість алгоритмів та обмеженість бази знань ШІ.

Також ШІ уже сьогодні автоматизує багато виробничих процесів, через що певні професії можуть стати непотрібними. Це викликає занепокоєння щодо майбутнього ринку праці та того, як забезпечити нові можливості для працівників, чиї професії зникають.

Оскільки ШІ вимагає великих обсягів даних для своєї ефективної роботи, виникає питання захисту конфіденційної інформації. Хакери можуть використовувати вразливості в алгоритмах опрацювання даних для зловживань, що загрожує як приватним особам, так і компаніям.

Крім того, системи ШІ чудово справляються з аналітичними задачами, але їм все ж бракує емоційної складової, яка є важливою в таких сферах, як обслуговування клієнтів або ухвалення рішень, що вимагають людської інтуїції.

Важливим і складним на сьогодні є і питання плагіату з використанням штучного інтелекту. Це відбувається через те, що генеративні моделі ШІ навчаються на великих наборах даних, які можуть бути взяті з Інтернету, в тому числі включати захищений авторським правом матеріал. І спираючись на них, інтелектуальні системи можуть ненавмисно відтворювати великі уривки тексту із цих матеріалів без посилання на джерела, що призводить до порушення авторських прав і плагіату.

Але існує і зворотна ситуація – створено інструменти на основі ШІ, які допомагають виявляти плагіат у текстах. Вони порівнюють наданий текст з великою базою даних вебсторінок, наукових робіт, книжок та інших джерел, щоб знайти збіги або схожість. Це допомагає забезпечити академічну доброчесність і захистити авторські права.

21 травня 2024 року Рада ЄС схвалила закон про штучний інтелект, установивши обмеження у використанні цих сучасних технологій (<http://surl.li/porhhe>). Нові правила забороняють певні інструменти ШІ, які порушують права громадян: використання біометрії, збирання світлин людей з мережі або записів камер відеоспостереження для створення баз даних розпізнавання облич, створення фейків і пропагандистської інформації тощо.

Крім того, існують вікові обмеження для використання деяких систем штучного інтелекту. Наприклад, для використання ChatGPT користувачам повинно бути щонайменше 13 років, а для користувачів віком від 13 до 18 років потрібна згода батьків або законних представників для використання ШІ.

Проте у будь-якому разі використання штучного інтелекту має свої переваги, зокрема:



- **Ефективність та автоматизація.** Рутинні завдання, які раніше займали години, тепер можуть виконуватися автоматично. ШІ звільняє наш час для більш важливих і креативних речей.
- **Швидкість обробки даних.** Завдяки складним алгоритмам і моделям машинного навчання штучний інтелект може швидко аналізувати величезні масиви даних, виявляти певні закономірності та тенденції, отримувати оперативні висновки. Це особливо важливо у сферах, де від швидкості залежать критичні рішення.
- **Персоналізація інформації.** ШІ аналізує ваші останні запити та пропонує за результатами пошуку те, що вас найбільше цікавило. Усі чат-боти та голосові помічники, які завжди готові допомогти, також працюють на основі ШІ. Вони відповідають на наші запити швидко та точно.
- **Витривалість.** Штучному інтелекту не потрібен сон і перерва на обід, він не допускає помилок через перевтому.
- **Інновації та творчість.** Штучний інтелект є каталізатором інновацій і творчості в різних галузях. Автоматизуючи рутинні та повсякденні завдання, ШІ вивільняє людський талент, щоб зосередитися на більш творчих починаннях. Спільна взаємодія між ШІ і людською творчістю сприяє інноваціям, розширюючи межі можливого в різних сферах – від дизайну продуктів до вирішення проблем і генерування ідей.
- **Скорочення витрат і оптимізація ресурсів.** Завдяки автоматизації процесів та оптимізації розподілу ресурсів, як визначають системи ШІ на виробництві, бізнес може значно скоротити витрати на робочу силу, пов'язані з виконанням рутинних завдань, мінімізувати втрати та досягти кращих фінансових результатів.
- **Посилення безпеки та виявлення шахрайства.** У сфері кібербезпеки штучний інтелект є потужним союзником у виявленні потенційних загроз і реагуванні на них як у сфері персональних даних, так і на рівні установ, держави та світу.
- **Мовний переклад і комунікація.** Інструменти машинного перекладу на основі штучного інтелекту сприяють безперешкодному спілкуванню людей у глобальному масштабі, це значно розширює культурні, освітні та бізнес-можливості.

Більш детально про переваги та недоліки використання штучного інтелекту ознайомтесь у відеороліку за посиланням <http://surl.li/fjlops>.



### Для тих, хто хоче знати більше

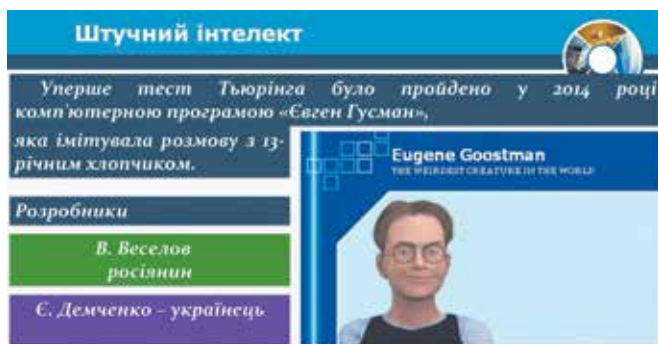
Уважається, що штучний інтелект здатний проявляти поведінку, яка не відрізняється від людської. Так, один з основоположників-розробників теорії штучного інтелекту Алан Тьюрінг (1912–1954) (мал. 7.20) у своїй книзі «Чи може машина думати?» у 1950 році висловив думку, що машина може вважатися розумною тоді, коли буде здатна підтримувати спілкування з людиною, і та не зможе зрозуміти, що спілкується з машиною. Такий своєрідний тест (так званий *тест Тьюрінга*) вважається пройденим штучним інтелектом, якщо 30 % експертів, які братимуть участь у розмові, не розпізнають штучний інтелект.



Мал. 7.20.  
Алан Тьюрінг

Однією з перших комп'ютерних програм (2014 рік), що пройшла тест Тьюрінга, була програма «Eugene Goostman» (мал. 7.21). Вона вела розмову з людиною від імені 13-річного хлопчика з Одеси Євгена Гусмана (один з розробників програми Є. Демченко родом з України).

У 2024 році дослідники з Каліфорнійського університету провели експеримент, у якому 500 експертів по 5 хвилин спілкувалися в чаті з трьома моделями штучного інтелекту (ChatGPT-4, ChatGPT-3.5 і Eliza 1966 рік) та однією людиною. За результатами, у 54 % бесід з програмою ChatGPT-4 експерт уважав, що спілкується з людиною; з ChatGPT-3.5 – у 50 % випадків. Ці результати показують, що штучна система успішно пройшла тест Тьюрінга, але водночас дослідники наголошують, що тест може відображати здатність штучного інтелекту наслідувати або обманювати людей, а не демонструвати справжній інтелект.



Мал. 7.21. Заставка програми Eugene Goostman

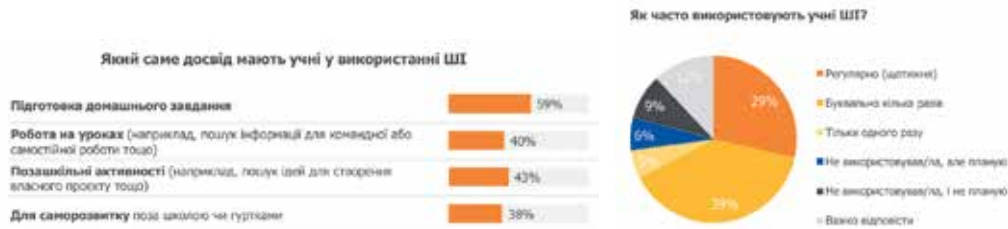
## ШТУЧНИЙ ІНТЕЛЕКТ У НАВЧАННІ



### Поміркуйте

● Як здійснюється пошук потрібної інформації в пошукових системах? ● Чи використовували ви на уроках системи штучного інтелекту? Якщо так, то для вирішення яких навчальних завдань?

У вересні–жовтні 2024 року в закладах освіти України проводилося дослідження щодо використання систем штучного інтелекту в освіті, у якому взяли участь 3190 учителів/вчительок та учні/учениці 8–11 класів. Виявилося, що 76 % опитаних учителів/вчительок мають успішний досвід використання ШІ у своїй діяльності та 54 % вважають, що ШІ змінить шкільне навчання в найближчі роки. І 85 % учнів/учениць уже використовували ШІ в навчанні (мал. 7.22).



Мал. 7.22. Використання учнями/ученицями систем штучного інтелекту

Штучний інтелект відкриває перед учнівством і вчителями безліч нових можливостей і значно полегшує виконання рутинних завдань.

Наведемо кілька напрямів використання систем ШІ в навчанні:

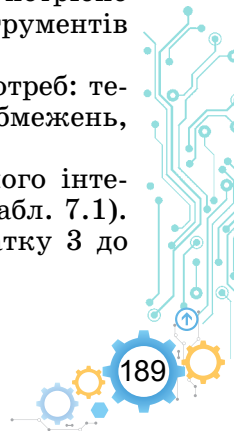
- Швидко знаходити потрібну інформацію, відповіді на різноманітні запитання, вести розмови на цікаві для вас теми, уточнюючи та деталізуючи зміст відповідей.
- Отримувати від систем ШІ візуальне подання інформації для кращого розуміння складних навчальних тем (графіки, схеми, діаграми, інфографіка).
- Перевірити грамотність написання тексту будь-якою мовою, поліпшити стиль написання тексту за допомогою інструментів ШІ.
- Отримувати за запитом згенеровані ШІ ідеї для виконання власних проєктів, конкурсів, творів, сценаріїв тощо.
- Створювати за допомогою систем ШІ заготовки презентацій, текстів, зображень, резюме, аудіо- та відеоматеріалів до уроків, позакласних заходів, життєвих потреб.
- Досліджувати за допомогою різних віртуальних симуляторів складні питання шкільних предметів, проводити віртуальні дослідження з використанням ШІ.
- Генерувати з використанням ШІ різні однотипні навчальні завдання з різних предметів і тем, розв'язувати їх з автоматичною перевіркою ШІ, використовуючи як навчальний тренажер, або тренуватися в розмові будь-якою іноземною мовою зі штучним інтелектом.
- Використовувати ШІ-інструменти не тільки для розв'язування завдань, а й для пояснення кроків для отримання результатів, інструкцій для виконання такого завдання.
- Повноцінно навчатися на онлайн-курсах з різних предметів, де ШІ може спланувати персональну траєкторію навчання, залежно від рівня успішності, потреб і бажаного темпу навчання.

Сьогодні існує велика кількість різноманітних інструментів, які працюють на основі штучного інтелекту, вони постійно вдосконалюються, змінюються умови їх використання, деякі з часом зникають і з'являються нові. У зв'язку із цим якихось виключних списків таких систем не існує, їх приклади та способи використання можна знайти в Інтернеті самостійно.

Деякі системи ШІ безкоштовні, але більшість платні або з безкоштовною пробною версією. Майже всі ці сервіси потребують попередньої реєстрації на цьому ресурсі, деякі можуть використовувати раніше створені акаунти в **Google** чи **Microsoft**. При цьому потрібно пам'ятати, що самостійне використання більшості ШІ-інструментів дозволено з 13 років, але за погодження батьків.

Вибір системи штучного інтелекту залежить від ваших потреб: тематики запиту, цільової аудиторії, фінансових і вікових обмежень, технічних можливостей тощо.

Наведемо кілька прикладів безкоштовних систем штучного інтелекту, які можна використовувати в навчальній діяльності (табл. 7.1). Більш розширений список ШІ-інструментів наведено в Додатку 3 до підручника.



Назва ШІ-інструмента, тип моделі, посилання на ресурс	Приклади використання
<b>Gemini Google</b> (мовна модель) <a href="https://gemini.google.com/app?hl=uk">https://gemini.google.com/app?hl=uk</a>	Пошук інформації за запитом, переклад текстів різними мовами, створення різних текстів за сюжетом, генерування ідей за тематикою, складання планів, розв'язування задач з математики, логіки, мов; аналіз текстів, надання рекомендацій, ведення розмов та інше
<b>Copilot Дизайнер</b> (генеративна модель) <a href="https://www.bing.com/create">https://www.bing.com/create</a>	Генератор графічних зображень, які створюються за текстовим описом, покращення готових зображень, створення макетів і шаблонів
<b>Prezo</b> (генеративна модель) <a href="https://prezo.ai/">https://prezo.ai/</a>	Генератор і редактор презентацій за текстовим описом, скорочує довгі документи до стилю презентацій, додає до створених слайдів згенеровані графічні зображення, надає можливість поділитися презентацією

Ефективність використання систем ШІ залежить не тільки від розуміння їх можливостей і вміння користуватися їх інструментами, а й від того, як правильно сформульовано до них **запити (промпти, англ. *prompt* – підказка)**. Точні та зрозумілі запити до систем забезпечують отримання корисних і релевантних відповідей.

Рекомендуємо дотримуватися таких правил:

- Сформулюйте чітко дію, яку повинен зробити ШІ (обговорити, сформулювати, запропонувати, класифікувати тощо).
- Чітко формулюйте тематику запити, використовуйте ключові слова, уникайте двозначних слів, сленгу, загальних фраз.
- Укажіть результат, який бажаєте отримати, і його вигляд.
- Укажіть роль, від імені якої потрібно отримати результат, або цільову аудиторію, для якої буде використовуватися.
- Уникайте зайвих слів і складних речень, складне запитання краще розбити на кілька простих.
- Пишіть граматично правильно, дотримуйтеся правил граматики.
- Указуйте максимальні розміри потрібного результату в кількісних чи якісних показниках.
- Надайте за потреби ШІ приклад виконання, зразок, шаблон оформлення.
- Отримавши результат виконання завдання, визначте його відповідність вашим потребам.
- Якщо результат з першого разу вас не задовольнив, змініть запит, надавши додаткову інформацію, використавши інші ключові слова.
- Системи ШІ постійно вчаться на ваших запитах, тому тему одного завдання бажано продовжувати в тому самому чаті, щоб ШІ зміг отримати з вашого діалогу більш системні знання.

Але ж, звертаємо увагу, що не слід на 100 % довіряти результатам роботи ШІ. Можливість технічних збоїв, нерозуміння контексту,



відсутність емоційного сприйняття, обмеженість даних – усе це може призвести до неправильних висновків і некоректних результатів.



Тому, користуючись ШІ, слід завжди критично оцінювати отриману інформацію, перевіряти її з інших джерел. І пам'ятати про обмеженість можливостей ШІ, що він усього лише потужний інструмент, який може допомогти вам у навчанні. Використовуйте його, щоб генерувати ідеї, створювати чернетки контенту або долати творчий ступор. Але пам'ятайте, що останній штрих завжди має залишатися за людиною – додавайте ваше особисте бачення, творчі ідеї, власні висновки.



### Працюємо з комп'ютером


**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

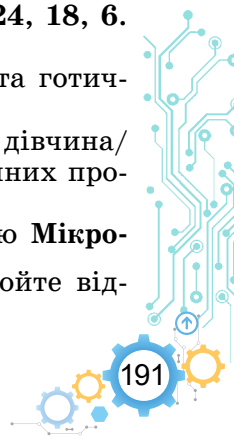
**Завдання.** Ознайомитися з роботою сервісу зі штучним інтелектом Gemini з текстовими та графічними запитамі.

1. Увійдіть у свій обліковий запис **Google**, використавши браузер **Google Chrome**.
2. Відкрийте меню сервісів **Google** вибором кнопки **Додатки Google** .
3. Виберіть у списку сервісів кнопку **Gemini** .
4. Якщо ви ще ніколи не використовували сервіс **Gemini**, то пройдіть додаткові кроки реєстрації:
  - розпочніть роботу із сервісом, вибравши кнопку **Чат з Gemini**;
  - ознайомтеся з правилами користування сервісом, прийміть запропоновані умови, вибираючи послідовно кнопки **Більше** ⇒ ⇒ **Використовувати Gemini** ⇒ **Продовжити**.
5. Ознайомтеся з інтерфейсом вікна сервісу **Gemini**. У нижній частині вікна розміщено текстове поле для введення запитів, на які користувач хоче отримати відповіді.
6. Уведіть у текстове поле по черзі кілька запитів. Перегляньте отриманий результат і проаналізуйте, чи відповідає він вашим запитам. За потреби надайте додаткові відомості для отримання більш точної відповіді або переформулюйте запит.

#### I. Запити з текстовими повідомленнями:

- Які цікаві події відзначають в Україні у травні?
- Запропонуй, яке дослідження про електризацію тіл можна провести вдома.
- Знайди, яке число пропущено в послідовності **36, 30, 24, 18, 6**. Поясни свою відповідь.
- Порівняй, чим відрізняються мистецькі стилі бароко та готичний стиль. Підтвердь це двома парами фотографій.
- Склади оптимальне меню на день для мене. Мої дані – дівчина/хлопець, 13 років, відвідую ввечері басейн, не їм молочних продуктів, люблю фрукти.

Уведіть запит голосом. Для цього скористайтеся кнопкою **Мікрофон**  у правій частині поля для введення запитів. Скопіюйте від-




повідь на запит у текстовий документ і збережіть його у вашій папці з іменем **вправа 7.2-1**.

## II. Запити з графічними зображеннями:

- Намалюйте малюнок: сонце, море з хвилями, пляж і яхта в морі з червоними вітрилами.
- Скопіюйте в **Буфер обміну** опис одного з дослідів про електризацію тіл, який було наведено в попередніх запитах. Для швидкого перегляду скористайтеся списком запитів на лівій бічній панелі. Уведіть запит *намалювати зображення за наданим описом* і вставте опис з **Буфера**. Збережіть малюнок у вашій папці з іменем **вправа 7.2-2**.

- Знайдіть в Інтернеті фотографію картини І. Айвазовського «Дев'ятий вал». Уведіть запит *зробити текстовий опис картини*. Додайте до запиту зображення, скориставшись кнопкою



у правій частині поля для введення запитів. Прослухайте відповідь у голос, скориставшись кнопкою **Прослухати** , яку розміщено праворуч у заголовку відповіді.

## 7. Вийдіть зі свого облікового запису **Google**.



### Найважливіше в цьому пункті

**Штучний інтелект** – це область інформатики, яка здійснює розробку інтелектуальних комп'ютерних систем і програм, що імітують роботу людського мозку: розуміти мову, навчатися, міркувати, робити висновки та передбачати, вирішувати проблеми, ухвалювати рішення тощо.

Сьогодні створення систем штучного інтелекту ведеться за напрямками створення комп'ютерних систем, що імітують діяльність людини; імітують логічне мислення людини; імітують функціонування людини; будуть інтелектуальними агентами.

Штучний інтелект набуває широкого застосування в різних галузях діяльності. Його використовують для розпізнавання різних об'єктів, машинного перекладу текстів, аналізу даних, створення експертних систем та інтелектуальних систем інформаційної безпеки, у робототехніці, для творчості та ігор тощо.

Основними недоліками використання результатів роботи систем ШІ сьогодні є: можливість отримання недостовірних даних і неправильних рішень, слабкий захист персональних даних і збереження конфіденційності використовуваної інформації, відсутність емоційної складової у спілкуванні та інтуїції, недотримання академічної доброчесності та допущення плагіату користувачами ШІ. Існують і певні обмеження його використання, зокрема, заборонено використовувати інструменти ШІ, які порушують права громадян, або вікові обмеження.

У використанні штучного інтелекту є й свої переваги: ефективність та автоматизація деяких рутинних видів діяльності, швидкість обробки даних, персоналізація отримуваної інформації, витривалість і стійкість системи протягом тривалого часу, скорочення витрат і оптимізація ресурсів, посилення безпеки та виявлення шахрайства, скасування мовних бар'єрів тощо.

Штучний інтелект відкриває нові можливості й перед учнями/ученицями та вчителями/вчительками, значно полегшує виконання рутинних завдань, таких як пошук потрібної інформації, візуалізація текстового матеріалу, перевірка правопису, генерування ідей, створення інформаційних продуктів різного типу та інше.

Існує велика кількість різноманітних інструментів, які працюють на основі штучного інтелекту, вони постійно вдосконалюються, змінюються умови їх використання,

деякі з часом зникають і з'являються нові. Вибір системи штучного інтелекту залежить від ваших потреб: тематики запиту, цільової аудиторії, фінансових і вікових обмежень, технічних можливостей тощо.

Значною мірою ефективність використання систем ШІ залежить від точності та зрозумілості сформульованих **запитів (промптів)**.

Користуючись ШІ, слід завжди критично оцінювати отриману інформацію, перевіряти її з інших джерел. Усвідомлювати, що це лише потужний інструмент, який може допомогти вам у навчанні, на роботі, у житті.



### Дайте відповіді на запитання

1. Що таке *штучний інтелект*?
2. У яких напрямках ведеться створення систем штучного інтелекту? Наведіть приклади.
3. Які напрями використання систем штучного інтелекту в різних галузях діяльності людини? Наведіть приклади.
4. Що таке *промпт*? Які правила формулювання промптів?
5. Які існують етичні проблеми створення систем штучного інтелекту?



### Обговоріть і зробіть висновки

1. Що спільного між машинним і людським інтелектом?
2. Як ви розумієте слова Норберта Вінера (засновника кібернетики): «Обчислювальна машина цінна рівно на стільки, на скільки цінна людина, що її використовує»?
3. Які переваги та недоліки використання систем штучного інтелекту ви можете назвати в таких прикладах:
  - а) штучний інтелект визначає діагноз хворій людині;
  - б) штучний інтелект створює казку для маленьких дітей;
  - в) штучний інтелект веде урок у школі;
  - г) штучний інтелект проектує робота.
4. Які ризики несуть системи штучного інтелекту під час їх використання? Поспілкуйтеся про це із самим штучним інтелектом. Чи розділяєте ви висловлені зауваження? Проранжуйте висловлені ризики за ступенем важливості для вас.
5. Як ви вважаєте, чого не може зробити штучний інтелект? Які професії він не зможе замінити?
6. Чи може отримати штучний інтелект авторське право на згенеровані ним матеріали? Як потрібно діяти авторам, щоб уникнути плагіату під час використання систем ШІ? Проаналізуйте та обговоріть із цього приводу статті Закону України «Про авторське право та суміжні права» та проекту Закону України «Про академічну доброчесність».




### Виконайте завдання


1. Відкрийте інтернет-майданчик для інтелектуального малювання **AutoDraw** <https://www.autodraw.com>. Ви можете намалювати будь-яку криву лінію, і програма домалює за вас малюнок, який можна обрати з певного переліку зображень. Ознайомтеся з роботою програми. Намалюйте кілька зображень з використанням можливостей цього ресурсу. Збережіть результат роботи у вашій папці у файлі з іменем

**завдання 7.2.1.** Чи можна вважати цю програму прикладом системи штучного інтелекту?

**2.** Відкрийте сайт **Акінатор** <https://www.silvergames.com/uk/akinator>. Налаштуйте переклад інтерфейсу українською і пограйте у віртуальну гру «Вгадай слово». Чи вдалося програмі у вас виграти? Чи можна вважати цю програму прикладом системи штучного інтелекту?

**3.** Відкрийте онлайн-додаток для озвучування тексту <https://qudata.com/uk/text-to-speech/> та ознайомтеся з його роботою. Уведіть кілька текстових фраз і прослухайте їх озвучування. Налаштуйте різні режими роботи програми, збережіть отримані результати у вашій папці. Поясніть, чи можна вважати цю програму прикладом штучного інтелекту. Запишіть свої аргументи в зошит.

 **4.** Відкрийте сайт **Quick,Draw!** <https://quickdraw.withgoogle.com/?locale=ua#>. Ознайомтеся з роботою нейронної мережі **Google**, для якої потрібно намалювати зображення за вказаною темою. Опишіть у зошиті, як ця система штучного інтелекту вчиться.

 **5.** Ознайомтеся зі статтею «Забута «Машина логічного мислення» професора Щукарьова» за посиланням <https://uacomputing.com/stories/professor-shchukarevs-forgotten-logical-thinking-machine/>. Визначте та запишіть у зошит кілька ідей, які були сформульовані вченим та які знайшли відображення в сучасних розробках штучного інтелекту.

**6.** Перегляньте освітній телесеріал для учнів/учениць *Штучний інтелект для школярів* <https://osvita.dii.gov.ua/courses/artificial-intelligence-for-schoolchildren>. Запишіть у зошит по три приклади: які відомості ви вже знали; що нового ви дізналися; про що хотіли б дізнатися більше?



### Готуємось до вивчення нового матеріалу

1. Що таке *авторське право*? Які вимоги до його дотримання?
2. Як цифрові технології використовуються в освіті?
3. Які ресурси зі штучним інтелектом ви вже використовували?



### Оцініть свої знання та вміння

Оцініть свої навчальні досягнення з розділу «**Smart-технології. Штучний інтелект**» (*початковий, середній, достатній, високий рівень*).

- Я можу пояснити, що таке Smart-пристрій і чим він відрізняється від традиційних.
- Я розумію, які переваги та недоліки пов'язані з використанням Smart-пристроїв та Інтернету речей.
- Я можу навести приклади пристроїв, які належать до Інтернету речей.
- Я розумію, як пристрої Інтернету речей під'єднуються до мережі.
- Я можу описати, як Smart-технології, Інтернет речей змінюють наше життя.
- Я можу пояснити, що таке штучний інтелект і які його види існують.
- Я знаю основні напрями розробок систем ШІ та їх використання в різних сферах діяльності людини.
- Я вмію створювати запити до систем ШІ та передбачати результат їх виконання.
- Я можу оцінити переваги та недоліки використання штучного інтелекту.
- Я розумію етичні аспекти застосування штучного інтелекту.

# ПРАКТИКУМ З ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

У цьому розділі ви отримаєте нові, а також поглибите та розширите свої знання й удосконалисте навички з таких тем:

- ▶ навчальний проєкт та етапи його виконання;
- ▶ використання систем штучного інтелекту для виконання проєкту.

## 8.1. НАВЧАЛЬНИЙ ПРОЄКТ. ВИКОРИСТАННЯ СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ВИКОНАННЯ ПРОЄКТУ

### НАВЧАЛЬНИЙ ПРОЄКТ ТА ЕТАПИ ЙОГО ВИКОНАННЯ

#### Пригадайте

- Що таке *навчальний проєкт*? Наведіть приклади проєктів, учасником виконання яких ви були.
- Що таке п'ять «П» проєкту? У чому їх сутність?

З навчальними проєктами ви починали працювати ще з початкової школи і продовжували брати в них участь і в 5–7 класах.

Проєкти – це не просто навчальні завдання, а захоплива подорож у світ знань! Під час роботи над проєктом учасники самостійно досліджують цікаві теми, шукають відповіді на складні запитання, вчаться працювати в команді та презентувати результати своєї роботи.

Пригадаємо етапи реалізації навчального проєкту (мал. 8.1):



Мал. 8.1. Етапи виконання проєкту

1. **Постановка завдання.** Спочатку потрібно чітко формулювати, чого хочемо досягти. Яка проблема нас цікавить? Які цілі та завдання ми ставимо перед собою?
2. **Планування роботи.** Далі створюємо детальний план роботи та встановлюємо терміни виконання. Розподіляємо завдання між учасниками проєкту, визначаємо ролі та завдання кожного.

3. **Збір інформації.** Шукаємо потрібні матеріали в різних джерелах: книжках, журналах, Інтернеті, проводимо інтерв'ю та опитування.
4. **Обробка даних.** Аналізуємо та опрацьовуємо зібрану інформацію, систематизуємо її та робимо висновки.
5. **Оформлення результатів.** Оформляємо результати проєкту у вигляді презентації, відеоролика, вебсторінки або іншого інформаційного продукту.
6. **Захист проєкту.** Представляємо свою роботу однокласникам/однокласницям, учителю/вчительці або батькам/рідним.



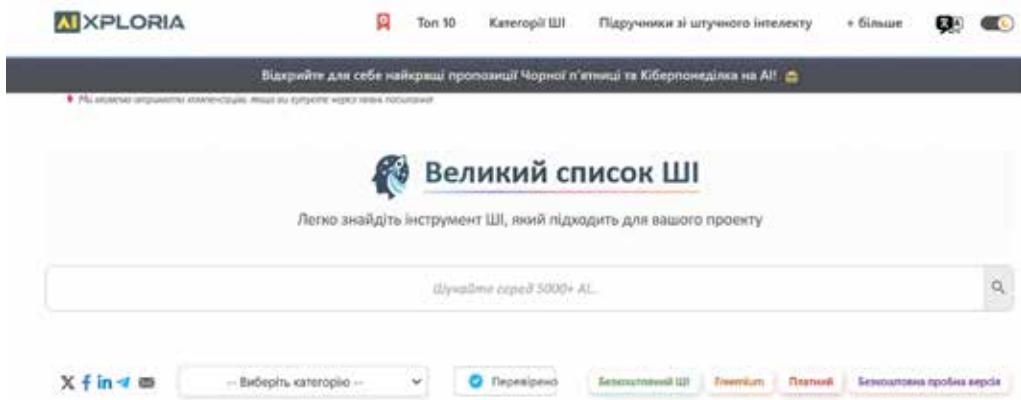
Мал. 8.2. Учні виконують груповий проєкт (згенеровано ШІ)

Участь у виконанні проєктів допомагає розвинути творчі, дослідницькі та комунікативні навички. Завдяки проєктам можна детально вивчити навчальну тему, над якою працюєте. Робота в команді вчить взаємодіяти з іншими людьми, ділитися ідеями та досягати спільних цілей. Проєкти дають змогу використовувати різноманітні інструменти та сервіси, які роблять роботу цікавішою та ефективнішою.

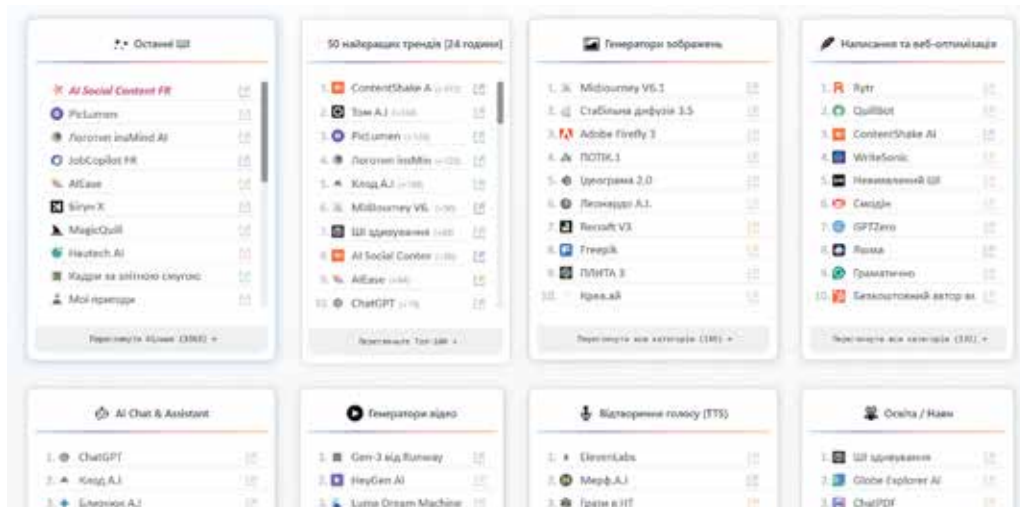
Цього року пропонуємо вам виконати проєкти з використанням систем штучного інтелекту. Ви зможете обрати тему, яка вас найбільше цікавить, і створити проєкт разом з вашими друзями/подругами та однокласниками/однокласницями. Це може бути вебсайт, презентація,

збірка фотоальбомів, колаж малюнків, презентація, відеоролик або щось інше, за вашим бажанням. А для більш ефективної роботи скористайтеся ШІ-інструментами, які допоможуть вам на всіх етапах реалізації.

Для вибору інструментів ви можете скористатися Додатком 3 підручника. Або переглянути майже повний перелік актуальних на сьогодні систем ШІ на сайті **Aixploria** <https://www.aixploria.com/> (за потреби виберіть переклад сторінки українською мовою) (мал. 8.3). На ньому представлено добірку ШІ-інструментів, згрупованих за категоріями, щоб легко знайти ресурс, який відповідає вашим потребам. До речі, сам перелік створено також штучним інтелектом.



Мал. 8.3 (а)



Мал. 8.3 (б)

За потреби під час реалізації проекту рекомендуємо скористатися текстовими та відеоінструкціями для роботи з різними ШІ-інструментами <http://surl.li/qdjwup>.



## ВИКОРИСТАННЯ СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ВИКОНАННЯ ПРОЄКТУ



### Пригадайте

- Які комп'ютерні програми можуть бути використані для оформлення результатів проекту?
- Які можливості використання систем штучного інтелекту в навчанні?

Розглянемо виконання проекту з використанням систем штучного інтелекту на прикладі проекту з інформатики *Штучний інтелект у побутових пристроях*.

На кожному етапі реалізації проекту використовуватимемо відповідні ШІ-інструменти для прискорення нашої роботи. Їх перелік необмежений, і ви можете вибрати для реалізації своїх задумів інші ресурси.

### 1. Постановка завдання.

Виконання цього етапу рекомендуємо провести вашою групою спільно, для того щоб оперативніше проаналізувати отриману інформацію та обрати тему, цікаву для всіх. Для формулювання мети та завдань проекту скористаємося системою штучного інтелекту **Gemini** (<https://gemini.google.com/app>).

Надамо системі такий запит: «*Ми, учні 8 класу, плануємо брати участь у навчальному проекті з інформатики за темою “Штучний інтелект у побутових пристроях”*». Тривалість проекту 2–3 уроки. *Придумай 10 назв для проектів за цією темою та їх мету*».

У результаті ШІ нам запропонував 10 ідей, наприклад «Штучний інтелект у кіно», «Штучний інтелект і здоров'я», «Штучний інтелект у смартфоні», «“Розумний” дім: зручно чи небезпечно?» тощо.

Далі спільним обговоренням у групі потрібно проаналізувати запропоновані варіанти, обрати цікаву для всіх тему, спільно уточнити назву. Для визначення інших складових проекту можна сформулювати додаткові запити ШІ, орієнтовно такі: «Наша команда обрала тему проекту “Штучний інтелект у смартфоні”, Сформулюй завдання такого проекту. Які ресурси для цього потрібні? Що може бути результатом такої роботи?». Результати роботи ШІ потрібно скопіювати в текстовий документ, відредагувати та зберегти у спільній папці групи у файлі з іменем *Складові проекту*.

У результаті «мозкового штурму» в групі учасників/учасниць визначаємо складові нашого проекту:

*Тема проекту:* «Штучний інтелект у моєму смартфоні».

*Мета проекту:* розглянути, які функції та пристрої смартфона використовують штучний інтелект і для чого ми їх використовуємо.

*Завдання проекту:*

- Створити список усіх додатків на своїх смартфонах, які, на вашу думку, використовують штучний інтелект (наприклад, камера, голосовий помічник, GPS-навігатор). *Результат* – текстовий документ.
- Знайти інформацію про те, як працюють голосові помічники (Siri, Google Assistant, Alexa тощо). *Результат* – збірка текстових і графічних матеріалів у спільній папці на **Google Диску**.
- Дослідити, як штучний інтелект використовується в камерах смартфонів (розпізнавання облич, поліпшення якості зображень). *Результат* – відеоролик проведення дослідів.
- Створити опитувальник для однолітків, щоб з'ясувати, які функції смартфона, що використовують штучний інтелект, вони використовують найчастіше і чому. *Результат* – **Google Форм** з 5 питаннями.
- Створити презентацію, яка демонструє результати роботи групи у проекті. *Результат* – презентація на 10 слайдів.
- *Час виконання проекту* – 10 днів чи два тижні.

Результати роботи потрібно скопіювати в текстовий документ, відредагувати та зберегти у спільній папці групи.

## 2. Планування роботи.

Для подальшої роботи спільно придумуємо назву нашої групи, наприклад «Мобільні дослідники». Запросимо ідеї логотипу нашої групи в системи штучного інтелекту **Copilot** <https://copilot.microsoft.com/>, надавши запит «Намалюй логотип для нашої групи “Мобільні дослідники”, які виконують навчальний проект “Штучний інтелект у смартфоні”. Це повинен бути схематичний кольоровий малюнок невеликого розміру». Отриманий логотип (мал. 8.4) збережемо у спільній папці проекту у файлі з іменем *Логотип*.



Мал. 8.4. Логотип групи

Складемо детальний план роботи, указавши хід виконання завдань, виконавців, терміни роботи, використавши систему штучного інтелекту **Gemeni**, указавши в запиті перелік наших завдань,



терміни роботи та подання плану у вигляді таблиці. Отриманий текст **Експортувати в Таблиці**, проаналізувати у групі та відредагувати за потреби, зазначити прізвища виконавців і конкретні дати виконання, зберегти у спільній папці групи (мал. 8.5).

Дні	Дати	Завдання	Деталі	Відповідальний
1-2	05-06.05.2025	Підготовка та планування	Розподіл завдань, створення Google Диску, розробка плану досліджень	всі члени групи
3-4	07-08.05.2025	Збір інформації	Пошук в Інтернеті, створення списків додатків у мобільних телефонах учасників груп та їх батьків	Іван
			Дослідження голосових помічників, пошук в Інтернеті принципів їх роботи	Катя
			Дослідження камер, запис відео проведення досліджень	Петро
			Складання опитувальника, створення Гугл-форми	Олена
			Керівництво проєктом, допомога учасникам	Тарас
5-7	09-11.05.2025	Аналіз та створення матеріалів	Впорядкування списків, їх узагальнення, створення текстового документ з описом додатків	Іван
			Створення порівняльного опису роботи голосових помічників	Катя
			Обробка відео, монтаж	Петро
			Проведення опитування, створення діаграм та їх опису	Олена
			Керівництво проєктом, допомога учасникам	Тарас
8-9	12-13.05.2024	Об'єднання результатів та підготовка презентації	Обговорення результатів, створення презентації (кожен по 2 слайди за своєю роботою)	всі члени групи
10	14.05.2025	Презентація проєкту	Презентація результатів перед класом	Тарас

Мал. 8.5

### 3. Збір інформації.

Далі робота групи виконується кожним учасником/учасницею за своїм індивідуальним завданням. Для визначення ресурсів, за допомогою яких здійснювати пошук інформації та виконувати завдання, а також змістові питання щодо виконання окремих дій, можна знову ж таки використати системи штучного інтелекту, наприклад **Gemeni**, **Copilot**, **Perplexity AI** (<https://www.perplexity.ai/>). Наприклад:

- Напиши список 10 додатків у мобільних телефонах, які використовують штучний інтелект.
- Напиши список 10 пристроїв у мобільних телефонах, які використовують штучний інтелект.
- Які функції може виконувати камера мобільного телефона? Які з них працюють з використанням ШІ?
- Склади список з 10 питань для опитування однокласників/однокласниць щодо користування ними функціями та пристроями смартфона, які використовують ШІ.

Далі отримана інформація аналізується й обов'язково перевіряється іншими джерелами. Після чого вже використовується учасниками/учасницями групи для практичного дослідження у смартфонах (аудіозапис спілкування з голосовими помічниками, відео по роботі з камерами, скріншотами функцій смартфонів) і створення опитувальника в **Google Форм**. Усі матеріали зберігаються у спільній папці групи.

### 4. Обробка даних.

На цьому етапі виконання проєкту кожен учасник групи аналізує отримані дані та створює узагальнений документ за результатами роботи над своїм завданням: список функцій і пристроїв мобільного телефона, порівняння голосових помічників та їх можливостей, порів-

няння можливостей камер, проводиться опитування однокласників/однокласниць і візуалізація результатів.

Усі це можна зробити в текстовому документі, електронній таблиці, презентації. А можна доручити узагальнення цих даних системам ШІ, наприклад, **Gemini**, **Copilot**, **Perplexity AI** або іншим. Для цього потрібно ввести в поле для запитів дані, які потрібно узагальнити, та сформулювати завдання – *Створити порівняльну таблицю, узагальнений список тощо.*

### 5. Оформлення результатів.

На наступному етапі за результатами кожен/кожна учасник/учасниця створює свої 2 слайди у спільній презентації з використанням хмарного сервісу **Google Презентації**, а керівник/керівниця проекту – загальні слайди. Для підготовки окремих компонентів презентації можна використати системи штучного інтелекту. Наприклад, створити аватарки учасників/учасниць групи та інфографіку за результатами з використанням ресурсів **Copilot**, **DALLE-3** <https://www.bing.com/images/create>, **DreamStudio** <https://dreamstudio.com/start/>. Або створити всю підсумкову презентацію проекту з використанням сервісу для створення електронних презентацій **Gamma** <https://gamma.app/>.

### 6. захист проекту.

На заключному етапі проекту керівник/керівниця групи представляє на спільній конференції вашого класу мету та завдання проекту, хід його реалізації, вклад кожного члена групи, отримані результати, створені інформаційні продукти, основні висновки.



### Працюємо з комп'ютером

**Увага!** Під час роботи з комп'ютером дотримуйтеся вимог безпеки життєдіяльності та санітарно-гігієнічних норм.

#### Завдання.

1. Сформууйте у класі кілька груп учасників/учасниць і виконайте проєкт «Штучний інтелект у побутових пристроях» за алгоритмом, який наведено в тексті параграфа.
2. Презентуйте результати роботи груп на підсумковому занятті.



### Найважливіше в цьому пункті

Реалізація навчальних проєктів передбачає реалізацію певних етапів, а саме: постановка завдання, планування роботи, збір інформації, обробка даних, оформлення результатів. На кожному етапі виконання проєкту для більш ефективної та творчої його реалізації можна використовувати системи штучного інтелекту: для формулювання мети та завдань проєкту, планування проєкту та його візуалізації, генерування зображень, діаграм, інфографіки, відеороликів та інших інформаційних продуктів. Для кожної з таких задач сьогодні існує значна кількість ШІ-інструментів.

Виконання проєктів з використання ШІ дають змогу ознайомитися із сучасними ІТ, освоїти нові інструменти для опрацювання даних різних типів і створювати за їх допомогою різноманітні інформаційні продукти. А головне, розвиває критичне мислення, уміння аналізувати надану інформацію та її творчо використовувати.

Важливим моментом використання таких систем під час виконання проєкту є питання доброчесності та творчої адаптації ідей, запропонованих ШІ. У всіх випадках штучний інтелект використовується для полегшення роботи у проєкті, вирішення рутинних завдань.



### Дайте відповіді на запитання

1. Які ресурси штучного інтелекту можуть бути використані для збору інформації в навчальних проєктах? Наведіть приклади.
2. Як технології штучного інтелекту можуть підтримати учнів у пошуку джерел інформації? Наведіть приклади.
3. Як штучний інтелект може допомогти в аналізі даних, зібраних під час дослідження? Наведіть приклади.
4. Які програми штучного інтелекту можна використовувати для створення інформаційних продуктів під час реалізації навчальних проєктів? Наведіть приклади.
5. Які основні переваги та ризики використання штучного інтелекту в навчальних проєктах? Наведіть приклади.
6. Які етичні питання виникають під час використання штучного інтелекту в проєктах? Наведіть приклади.



### Обговоріть і зробіть висновки

1. Порівняйте отримані результати та ШІ-інструменти, які використовувалися різними групами під час виконання проєктів. Чим ви пояснюєте різницю отриманих результатів?
2. Обговоріть найбільш вдалі моменти роботи в навчальних проєктах і складності, з якими зіштовхнулися в ході роботи. Чи сприяло цьому використання систем ШІ? Яким чином?
3. Поясніть, які нові знання та навички ви отримали в ході виконання проєкту з використанням ШІ.
4. Чи може штучний інтелект замінити вчителя/вчительку? Поясніть вашу думку.

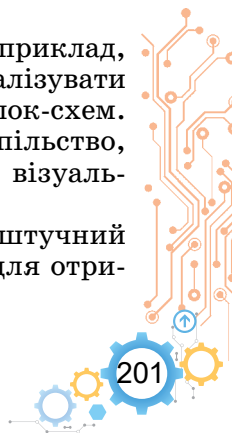


### Виконайте завдання

1. Виконайте мініпроєкт, створивши інформаційний продукт з використанням системи штучного інтелекту, за одним з наведених завдань. Збережіть результат роботи у вашій папці у файлі з іменем **завдання 8.1**.

Орієнтовна тематика завдань:

- 1) Створити інфографіку або анімоване пояснення складного фізичного явища (наприклад, рух планет, будова атома) за допомогою інструмента для генерації зображень.
- 2) Створити навчальне відео, де штучний інтелект озвучує розв'язування математичної задачі або пояснює складну теорему.
- 3) Написати короткий історичний огляд певної події чи постаті, використовуючи чат-бот зі штучним інтелектом як помічника в дослідженні.
- 4) Створити алгоритм для розв'язування певної задачі (наприклад, сортування чисел, пошук елемента в масиві) і візуалізувати його роботу за допомогою інструмента для створення блок-схем.
- 5) Створити презентацію про вплив технологій на суспільство, використовуючи генератор зображень для створення візуальних елементів.
- 6) Написати короткий науково-фантастичний твір, де штучний інтелект відіграє ключову роль. Використати чат-бот для отримання ідей.



- 7) Проаналізувати достовірність інформації, згенерованої штучним інтелектом, порівнявши її з даними з інших джерел.
- 8) Створити казку для молодших учнів про «розумні» пристрої в нашому житті.

**2.** Реалізуйте навчальний проєкт у групі однокласників/однокласниць за однією з наведених тем. Виберіть об'єкти дослідження відповідно до мети, засоби опрацювання даних і форму подання результатів. Використайте в реалізації проєкту системи штучного інтелекту на різних етапах його виконання. Збережіть підсумковий результат роботи вашої групи у вашій папці у файлі з іменем завдання **8.2**.

Орієнтовна тематика проєктів:

- 1) *Професія – IT-шник. Мета:* скласти довідник професій, які можна набути в галузі інформаційних технологій; з'ясувати, навички роботи з якими програмними засобами повинні опанувати представники цих професій.
- 2) *Історія штучного інтелекту. Мета:* дослідити етапи створення штучного інтелекту в світі та в Україні в тому числі.
- 3) *Людина чи машина? Мета:* дослідити ставлення оточуючих до етичних аспектів використання штучного інтелекту.
- 4) *Штучний інтелект на службі людини. Мета:* дослідити використання ШІ в різних галузях діяльності людини.
- 5) *Школа майбутнього. Мета:* спрогнозувати бачення учнів/учениць, учителів/учительок, батьків/рідних щодо використання інформаційних технологій і штучного інтелекту в навчанні.



### Оцініть свої знання та вміння

Оцініть свої навчальні досягнення з розділу «Практикум з використання інформаційних технологій. Навчальні проєкти» (початковий, середній, достатній, високий рівень).

- Я можу формулювати мету та завдання виконання проєкту з використанням ІТ для вирішення навчальної чи життєвої ситуації.
- Я можу визначити свої дії під час виконання групового проєкту, під час взаємодії з іншими, відповідально ставлюсь до виконання своїх завдань.
- Я знаю та застосовую різні методи збору інформації для проєкту, використовую їх та можу оцінити надійність і доцільність джерел.
- Я вмію добирати різні типи ресурсів до виконання проєкту, у тому числі з використанням систем ШІ.
- Я можу обирати оптимальний спосіб представлення результатів проєкту та обґрунтовувати свій вибір.
- Я вмію створювати інформаційні продукти різних типів подання як результатів виконання проєктів, у тому числі з використанням ШІ-інструментів.
- Я можу зберігати результати виконання проєкту у файлах різних типів на різних носіях даних, у тому числі в мережі та хмарних середовищах.
- Я вмію працювати у групі з іншими учасниками в ході реалізації спільного проєкту, враховувати погляди й емоційний стан членів групи.

Додатки до посібника можна знайти за адресою <https://cutt.ly/benwo0jJ> або QR-кодом.



**Абсолютне посилання** – посилання на клітинку електронної таблиці, яке не модифікується під час копіювання формули.

**Адреса клітинки** – це унікальне позначення, яке вказує на точне розташування цієї клітинки в електронній таблиці. Вона складається з номера стовпця та рядка, на перетині яких вона розміщена.

**Аргументи функції** – дані, над якими виконується обчислення у функціях (*числа, тексти, вирази, посилання, інша функція*).

**Арифметично-логічний пристрій** – складова процесора, яка забезпечує здійснення операцій над даними.

**Аркуш електронної книги** – об'єкт електронної книги, на якому розміщується електронна таблиця, діаграма тощо.

**Архів** (комп'ютерний) – копії даних, як правило, упакованих в один файл.

**Вбудовані функції** – функції, які наявні в бібліотеці функцій табличного процесора (*математичні, статистичні, текстові, логічні, фінансові та ін.*).

**Виконавчі пристрої** – цифрові пристрої, які здатні виконувати певні дії у відповідь на отримані команди або дані, забезпечують взаємодію інформаційної системи з навколишнім середовищем.

**Висловлювання** – речення, яке містить твердження про певний об'єкт або про зв'язки між об'єктами і про яке можна однозначно сказати, істинне воно чи хибне.

**Відносне посилання** – посилання на клітинку електронної таблиці, яке модифікується під час копіювання формули.

**Відсотковий формат** – формат даних в електронній таблиці, який використовується для подання числових даних в перерахунку на відсотки (%).

**Вкладені функції** – функції табличного процесора, у яких результат однієї функції використовується як аргумент іншої.

**Генеративна модель** – алгоритм машинного навчання, здатний створю-

вати нові дані, імітуючи стиль та структуру наявних даних.

**Грошовий формат** – формат даних в електронній таблиці, який використовується для подання числових даних з додаванням позначення грошової одиниці.

**Датчики (сенсори)** – об'єкти для відслідковування стану навколишнього середовища, реалізують з'єднання фізичного і віртуального світів, забезпечуючи збирання й опрацювання даних в реальному часі.

**Диз'юнкція** двох логічних виразів  $x$  і  $y$  – логічна операція, результат якої дорівнює **True**, якщо значення хоча б одного з логічних виразів  $x$  або  $y$  дорівнює **True**, і дорівнює **False**, якщо значення кожного з логічних виразів  $x$  і  $y$  дорівнює **False**.

**Діапазон клітинок** – деяка сукупність клітинок електронної таблиці.

**Довжина двійкового коду повідомлення (обсяг даних)** – це кількість бітів у двійковому коді цього повідомлення.

**Дробовий формат** – формат даних в електронній таблиці, який використовується для подання числа у вигляді звичайного дробу.

**Друкування таблиці** – перенесення електронної таблиці, створеної в табличному процесорі, на папір за допомогою принтера.

**Експоненційний формат** – формат даних в електронній таблиці, який використовується для подання числа в стандартному вигляді.

**Електронна книга** – документ табличного процесора, який є сукупністю кількох електронних таблиць, розміщених на окремих аркушах.

**Електронна таблиця** – основний об'єкт табличного процесора, який складається з рядків і стовпців.

**Загальний формат** – формат даних в електронній таблиці за замовчуванням.

**Заперечення логічного виразу  $x$**  – логічна операція, результат якої дорівнює **True**, якщо значення ло-

гічного виразу  $x$  дорівнює **False**, і дорівнює **False**, якщо значення логічного виразу  $x$  дорівнює **True**.

**Запит до системи штучного інтелекту (prompt)** – текстове формулювання, яке задає конкретне завдання моделі штучного інтелекту.

**Захист проєкту** – публічне представлення результатів проєкту.

**Збір інформації в проєкті** – пошук потрібних матеріалів у різних джерелах.

**Змінна логічного типу** – змінна, яка може набувати одного з двох значень: **True** або **False**.

**Зовнішня пам'ять** – складова пам'яті комп'ютера, що призначена для тривалого зберігання даних. До її складу входять пристрої на магнітних та оптичних дисках, флешнакопичувачі тощо.

**Інтернет речей (IoT)** – концепція, яка передбачає об'єднання різних фізичних пристроїв через Інтернет, забезпечуючи їм взаємодію та обмін даними.

**Інтернет-безпека** – система захисту онлайн-даних від несанкціонованого втручання.

**Ім'я клітинки** – персоналізована адреса клітинки в електронній таблиці, яка не модифікується під час копіювання у формулі.

**Ім'я функції** – унікальне словосполучення, яке ідентифікує функцію в бібліотеці функцій табличного процесора.

**Карта (мапа) сайту** – структура, яка містить перелік тематичних розділів сайту та сторінок у них, визначає зв'язки між сторінками, пояснює логічну структуру сайту.

**Контент** – текстові, графічні, мультимедійні та інтерактивні елементи на сторінках сайту.

**Кон'юнкція** двох логічних виразів  $x$  і  $y$  – логічна операція, результат якої дорівнює **True**, якщо значення кожного з логічних виразів  $x$  і  $y$  дорівнюють **True**, і дорівнює **False**, якщо значення хоча б одного з логічних виразів  $x$  або  $y$  дорівнює **False**.

**Клітинка** – об'єкт електронної таблиці, який утворився на перетині рядків і стовпців таблиці.

**Контролери** – спеціальні пристрої, що призначені для управління зовнішніми пристроями комп'ютера або датчиками.

**Контрольні точки відновлення системи** – копії (архіви) системних налаштувань.

**Колективна навчальна діяльність** – це форма навчальної діяльності в малих групах учнів, об'єднаних спільною навчальною метою, з розподіленим завданням між учасниками.

**Кратні одиниці вимірювання довжини двійкового коду** – утворюються з використанням префіксів *кіло-, мега-, гіга-, тера-* та інших.

$1 \text{ кБ (кілобайт)} = 2^{10} \text{ Б} = 1024 \text{ Б}$

$1 \text{ МБ (мегабайт)} = 2^{10} \text{ кБ} = 2^{20} \text{ Б} = 1\,048\,576 \text{ Б}$

$1 \text{ ГБ (гігабайт)} = 2^{10} \text{ МБ} = 2^{20} \text{ кБ} = 2^{30} \text{ Б}$

$1 \text{ ТБ (терабайт)} = 2^{10} \text{ ГБ} = 2^{20} \text{ МБ} = 2^{30} \text{ кБ} = 2^{40} \text{ Б}$

**Логічна функція (IF, AND, OR, NOT)** – функція, результат якої дорівнює **True** або **False**.

**Логічна помилка** – причина неспівпадіння результатів виконання проєкту з очікуваними результатами.

**Логічний вираз** – вираз, який може набувати одного з двох значень: **True** або **False**.

**Мапа сайту** – див. Карта сайту.

**Материнська, або системна, плата** – основна складова сучасних персональних комп'ютерів, яка забезпечує передавання даних між пристроями комп'ютера.

**Машинне навчання** – підгалузь штучного інтелекту, яка визначає спосіб самостійного виконання комп'ютерними системами завдань та ухвалення рішень, аналізуючи великі обсяги даних і виявляючи в них закономірності, без явного їх програмування.

**Методи збирання даних** – спостереження, опитування, опрацювання друкованих джерел та історичних знахідок, пошук відомостей в Інтернеті, фіксація даних з датчиків, результатів експериментів та інші.

**Мішане посилання** – посилання на клітинку електронної таблиці, яке

частково модифікується під час копіювання формули (або номер рядка, або номер стовпця).

**Модифікація формули** – автоматична зміна адрес клітинок у формулах під час їх копіювання в табличному процесорі.

**Навчальний проєкт** – це вид та результат діяльності, яка спрямована на досягнення певної навчальної мети, розв'язування деякої проблемної задачі.

**Напис** – елемент керування, який використовується для виведення у вікні текстових повідомлень.

**Нейронні мережі** – математичні моделі, натхненні біологічними нейронними мережами, які складаються із взаємопов'язаних вузлів, що обробляють інформацію та адаптуються до даних, імітуючи роботу біологічних нейронів мозку людини.

**Обсяг даних** – див. Довжина двійкового коду.

**Обробка даних у проєкті** – аналіз та опрацювання зібраної інформації, її систематизація та формулювання висновків.

**Оперативна пам'ять** – основна пам'ять комп'ютера, у якій розміщуються програми та дані, що в подальшому опрацьовуються процесором.

**Оформлення результатів проєкту** – створення інформаційних продуктів за підсумками виконання проєкту.

**Перемикачі** – елементи керування, які використовують для вибору тільки одного з кількох можливих варіантів. З кількох перемикачів однієї групи може бути вибраний тільки один.

**Планування роботи в проєкті** – складання детального плану виконання проєкту, встановлення термінів, розподіл завдань та ролей між учасниками проєкту.

**Поле** – елемент керування, який використовується для введення значень вхідних даних і виведення значень результатів.

**Постановка завдання проєкту** – чітке формулювання проблеми, цілі та завдань проєкту.

**Постійна пам'ять** – пам'ять комп'ютера, яка містить програми та дані, які

потрібні для виконання дій, пов'язаних з початком роботи комп'ютера.

**Прапорці** – елементи керування, які використовують для вибору одного або кількох варіантів з набору можливих чи не вибору жодного. З кількох прапорців може бути вибраний або один, або кілька, або всі, або жодний.

**Пристрій керування** – складова процесора, яка забезпечує виконання команд комп'ютерних програм.

**Пристрої керування роботою програм** – пристрої, що використовуються для передавання команд від користувача комп'ютерним програмам.

**Простий фільтр** – це інструмент у табличному процесорі, який дозволяє швидко відфільтрувати дані в таблиці за одним критерієм в одному стовпці.

**Режим попереднього перегляду** – це відображення в табличному процесорі електронної таблиці в тому вигляді, як вона буде надрукована на папері.

**Результат функції** – значення, яке повертається функцією після виконання обчислень.

**Розгалуження** – фрагмент алгоритму, який розпочинається з команди визначення значення логічного виразу і в якому виконуються або не виконуються певні команди залежно від значення цього виразу.

**Резервне копіювання** – процес створення копій даних та налаштувань операційної системи.

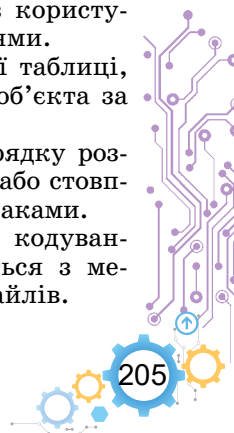
**«Розумний» (Smart) будинок** – будинок, у якому використовуються цифрові технології для автоматизації побутових операцій.

**«Розумний» пристрій** – електронний пристрій, здатний самостійно виконувати завдання, підключатися до Інтернету та взаємодіяти з користувачем або іншими пристроями.

**Рядок** – об'єкт електронної таблиці, який містить опис одного об'єкта за всіма його властивостями.

**Сортування** – змінення порядку розташування даних в рядках або стовпцях таблиці за деякими ознаками.

**Стиснення даних** – процес кодування даних, який здійснюється з метою зменшення розмірів файлів.



**Стовпець електронної таблиці** – об'єкт електронної таблиці, який містить опис однієї властивості для всіх об'єктів таблиці.

**Табличний процесор** – це прикладна програма, яка призначена для опрацювання даних, поданих в електронних таблицях.

**Таблиця істинності** – таблиця, у якій обчислюється значення логічного виразу за всіх можливих значень змінних, які входять у цей вираз.

**Таблиця кодів символів** – таблиця для кодування текстів, у якій кожному символу, що може бути використаний у текстовому повідомленні, поставлено у відповідність деяке число.

**Тег** – елемент мови **HTML**, що вказує браузеру, який об'єкт розмістити на сторінці та як його відобразити.

**Текстовий формат** – формат даних у електронній таблиці, який використовується для подання чисел як тексту.

**Тестові набори вхідних даних** – набори значень вхідних даних, які використовуються для визначення правильності роботи проекту.

**Типи даних** в електронних таблицях – числа, тексти і формули.

**Умови фільтрування** – це набір критеріїв у табличному процесорі, що визначають, які саме дані з таблиці будуть відображені, а які – ні.

**Умовне форматування** – вид форматування даних, який автоматично змінює формат клітинки електронної таблиці на заданий, якщо виконується певна умова.

**Фільтрування** – це автоматизований відбір значень у клітинках електронної таблиці, що відповідають певним умовам.

**Формат даних** – спосіб відображення даних у клітинках електронної таблиці.

**Формат Дата** – формат даних у електронній таблиці, який використовується для подання числових даних у вигляді дати певного типу.

**Формат Час** – формат даних у електронній таблиці, який використовується для подання числових даних у вигляді часу певного типу.

**Формула в електронній таблиці** – це вираз, який задає операції над даними в клітинках електронної таблиці та порядок їх виконання.

**Функція в табличному процесорі** – це попередньо визначена формула, яка виконує певні обчислення над заданими даними (аргументами) і занесена в бібліотеку функцій табличного процесора.

**Цикл в алгоритмі** – фрагмент алгоритму, який може повторюватися поспіль більше ніж один раз.

**Числовий формат** – формат даних у електронній таблиці, який використовується для подання чисел у вигляді десяткового дробу із заданою кількістю десяткових розрядів.

**Штучний інтелект** – область інформатики, яка здійснює розробку інтелектуальних комп'ютерних систем та програм, що імітують роботу людського мозку.

**СМУК** – колірна модель, у якій використовують чотири базові компоненти: блакитний, пурпурний, жовтий, чорний кольори.

**EXCEL** – табличний процесор пакета прикладних програм **Microsoft Office**.

**False** – один з можливих результатів логічної функції.

**HSV (HSB, HSL)** – колірна модель, у якій використовують три базові компоненти: відтінок, насиченість і значення (яскравість, світлість).

**HTML** – мова, якою описують структуру та контент вебсторінки для відображення цієї сторінки у вікні браузера.

**RGB** – колірна модель, у якій базовими компонентами є три кольори спектра – червоний, зелений і синій.

**True** – один з можливих результатів логічної функції.

**Smart-технології** – технології, що базуються на використанні датчиків, Інтернету речей та штучного інтелекту для створення «розумних» систем, які можуть самонавчатися та адаптуватися до змінних умов.

**XLSX** – стандартне розширення імені файлу електронної книги в **Microsoft Office Excel**.



## ЗМІСТ

Дорогі восьмикласниці та восьмикласники!	
Шановні вчительки та вчителі! . . . . .	3

### Розділ 4. Алгоритми та програми

4.1. Етапи розв'язування задач з використанням алгоритмів і проектів . . . . .	4
4.2. Кнопка . . . . .	11
4.3. Напис . . . . .	17
4.4. Поле. Проекти з вхідними даними та результатами . . . . .	23
<i>Практична робота № 4. «Проекти з полями, написами і кнопками, з уведенням даних і виведенням результатів» . . . . .</i>	34
4.5. Логічні вирази. Змінні логічного типу. Логічні операції . . . . .	35
4.6. Логічні вирази в розгалуженнях. Віконні проекти з розгалуженнями . . . . .	43
4.7. Прапорці та перемикачі у віконних проектах . . . . .	54
<i>Практична робота № 5. «Проекти з розгалуженнями» . . . . .</i>	62
4.8. Цикли з лічильником . . . . .	62
4.9. Цикли з передумовою . . . . .	70
<i>Практична робота № 6. «Проекти із циклами» . . . . .</i>	77

### Розділ 5. Створення та публікація вебресурсів

5.1. Етапи створення сайту. Дизайн вебсторінок. Тренди вебдизайну . . . . .	78
5.2. Поняття про мову HTML . . . . .	87
5.3. Автоматизовані засоби створення та публікації вебсайтів . . . . .	96
<i>Практична робота № 7. «Створення сайту з використанням онлайн-системи конструювання сайтів» . . . . .</i>	111

### Розділ 6. Опрацювання даних в електронних таблицях

6.1. Об'єкти електронної таблиці, їх властивості. Формати даних в електронних таблицях. Адресація в електронних таблицях . . . . .	112
6.2. Математичні, статистичні та логічні функції . . . . .	126
<i>Практична робота № 8. «Використання функцій у табличному процесорі» . . . . .</i>	142
6.3. Упорядкування та фільтрування даних у табличному процесорі . . . . .	142
6.4. Умовне форматування. Друкування електронної таблиці . . . . .	157
<i>Практична робота № 9. «Вибір даних в електронних таблицях» . . . . .</i>	167

### Розділ 7. Smart-технології. Штучний інтелект

7.1. Інтернет речей і його використання. Smart-технології . . . . .	169
7.2. Поняття штучного інтелекту. Використання штучного інтелекту в різних галузях . . . . .	179

### Розділ 8. Практикум з використання інформаційних технологій

8.1. Навчальний проект. Використання систем штучного інтелекту для виконання проекту . . . . .	195
Словник . . . . .	203

