



Наталія Морзе, Ольга Барна



УДК 004*кл7(075.3)
М79

*Рекомендовано Міністерством освіти і науки України
(наказ Міністерства освіти і науки України від 25.03.2020 № 449)*

ВИДАНО ЗА РАХУНОК ДЕРЖАВНИХ КОШТІВ. ПРОДАЖ ЗАБОРОНЕНО

Навчальне видання
МОРЗЕ Наталія Вікторівна
БАРНА Ольга Василівна

ІНФОРМАТИКА

Підручник для 7 класу закладів
загальної середньої освіти

Головний редактор *І. В. Красуцька*
Редактор *В. М. Ліченко*
Головний художник *І. П. Медведовська*
Технічний редактор *Е. А. Авраменко*
Коректор *С. В. Войтенко*

Малюнки *Олени Харченко*

Використано ілюстрації з ресурсів
shutterstock.com і depositphotos.com

Формат 70x100 ¹/₁₆.
Ум. друк. арк. 14,256 + 0,324 форзац.
Обл.-вид. арк. 13,60 + 0,55 форзац.
Зам. №
Наклад 131 973 пр.

ТОВ «УОВЦ «ОРІОН»

Свідоцтво про внесення суб'єкта видавничої
справи до Державного реєстру видавців,
виготовлювачів і розповсюджувачів
видавничої продукції

Серія ДК № 4918 від 17.06.2015 р.
Адреса видавництва: 03061,
м. Київ, вул. Миколи Шепелева, 2

Віддруковано
ТОВ «ПЕТ»

61024, м. Харків, вул. Максиміліанівська, 17

Свідоцтво про внесення
суб'єкта видавничої справи
до Державного реєстру видавців,
виготівників і розповсюджувачів видавничої
продукції серія ДК № 6847 від 19.07.2019 р.

Н. В. Морзе

М79 Інформатика. Підручник для 7 кл. закладів загальної середньої освіти /
Н. В. Морзе, О. В. Барна. — Київ : УОВЦ «Оріон», 2020. — 176 с. : іл.

ISBN 978-966-991-037-0.

УДК 004*кл7(075.3)

ISBN 978-966-991-037-0

© Н. Морзе, О. Барна, 2020
© УОВЦ «Оріон», 2020



Дорогі друзі!

Цей підручник допоможе вам продовжити вивчення інформатики. Він складається із трьох розділів. Ви навчитесь використовувати мережу інтернет для електронного листування та спільної роботи з документами. А ще дізнаєтеся, що не тільки люди працюють в інтернеті, а й речі можуть використовувати мережу, навіть без участі людини! Другий розділ переведе вас на новий рівень вивчення алгоритмів: ви складатимете програми сучасною мовою програмування. За матеріалами третього розділу ви опануєте програму для опрацювання табличних даних. По його завершенні ви будете будувати моделі й розв'язувати практичні задачі.

Усі уроки згруповано в тематичні блоки. Ви будете вивчати їх протягом одного або декількох уроків. Кожний блок завершується рубрикою **Повторюємо**, яка допоможе систематизувати свої знання та підготуватись до розділу **Узагальнення**.

Щоб досягти очікуваних результатів навчання та бути компетентними в цифровому світі, радимо:

- напередодні уроку читати матеріали рубрики **Вивчаємо**, щоб разом з учнями та вчителем або вчителькою обговорити незрозуміле й поділитись вивченим;
- виконати всі завдання з рубрики **Діємо** й бути готовими надати допомогу тим, хто не може оволодіти базовими вміннями без сторонньої допомоги;
- постійно експериментувати під час виконання завдань рубрики **Досліджуємо**;
- готуватися до дискусій, уміти відстояти свою думку та вислухати інших під час обговорення питань з рубрики **Обговорюємо**;
- постійно співпрацювати з іншими, навчати їх і навчатись у них під час роботи в парах і малих групах рубрики **Працюємо в парах**;
- планувати свою навчальну діяльність, мислити логічно, шукати незвичні розв'язки завдань, робити висновки та самостійно приймати рішення, виконуючи завдання з рубрики **Працюємо самостійно**;
- обирати не тільки завдання базового рівня, а й ті, що потребують креативного мислення, здатності аналізувати, оцінювати,

створити, — саме вони позначені  та .

Ви вже дорослі. Тому маєте розуміти, що саме від вас залежить, чого ви можете досягти у своєму житті. Використовуйте цей підручник та різноманітні навчальні ресурси, щоб пізнавати нове, навчатися завжди і скрізь.

Успіхів вам у пізнанні цікавих загадок цього світу!

Розділ 1

Електронне листування

1



Поштові служби інтернету. Створення електронної скриньки

ВИ ДІВНАЄТЕСЬ:

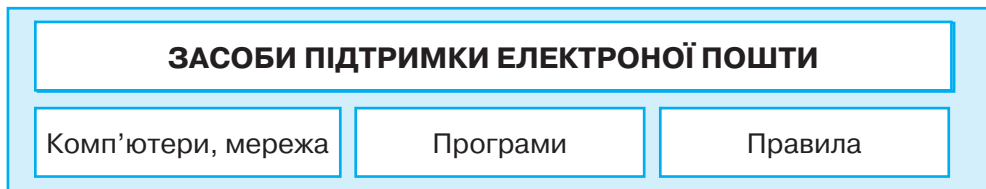
- ▶ що таке електронна пошта;
- ▶ як створюється адреса електронної пошти;
- ▶ як зареєструвати електронну скриньку на поштовому сервері.



Вивчаємо

1. Що таке електронна пошта?

Однією з найпопулярніших служб інтернету є електронна пошта. Для користування електронною поштою необхідні різні засоби: комп'ютерні пристрої та мережі, спеціальні програми й відповідні правила (протоколи), за якими здійснюватиметься обмін електронними листами (мал. 1.1).



Мал. 1.1

Електронна пошта (e-mail) — це сукупність засобів, призначених для обміну електронними листами між користувачами комп'ютерної мережі.

Для електронного листування потрібно мати електронну скриньку для листів. Електронна скринька має електронну адресу — так само, як ви маєте поштову адресу свого місця проживання. Електронний лист відправляють на поштовий сервер, що забезпечує надсилання всіх листів за відповідною адресою, він приймається на той поштовий сервер, де адресат/адресатка має свій обліковий запис. Для

доступу до електронної поштової скриньки необхідно мати логін — ім'я користувача та пароль. Лист пишеться мовою, зрозумілою адресату. До нього можна приєднати різні файли: фотографії, листівки, малюнки, інші документи.

Поштовий сервер — це комп'ютер і відповідне програмне забезпечення, за допомогою якого підтримується робота електронної пошти.

Найпростішим способом отримання власної електронної поштової скриньки є використання одного з безкоштовних поштових серверів, наприклад:

- українських: *mail.ukr.net*, *webmail.meta.ua*, *mail.oboz.ua*, *mail.i.ua*, *mail.bigmir.net*;
- міжнародних: *mail.yahoo.com*, *gmail.com*.

Електронне листування має свої переваги й недоліки порівняно зі звичайною поштою. Його перевагами є висока швидкість передавання повідомлень — з материка на материк лист доходить за секунди, відносно невисока вартість послуг, обмін повідомленнями з кореспондентами у зручний час тощо. Недоліками є неможливість пересилання оригіналів офіційних документів з підписом і печаткою, можливість перехоплення поштового повідомлення іншою особою, ймовірність отримання шкідливих програм разом з електронним листом тощо. Щоб листуватися електронною поштою, потрібно мати доступ до комп'ютерної мережі, доступ до поштового сервера та знати адресу електронної скриньки адресата.

2. Як створюється адреса електронної пошти?

Адреса електронної пошти складається з двох частин, розділених символом @. Цей символ часто називають «ет», «а комерційне», «собака» тощо.

Перша частина адреси — це ім'я поштової скриньки користувача, а друга — ім'я поштового сервера, на якому розташована поштова скринька.

<ім'я_поштової_скриньки_користувача>@<ім'я_поштового_сервера>

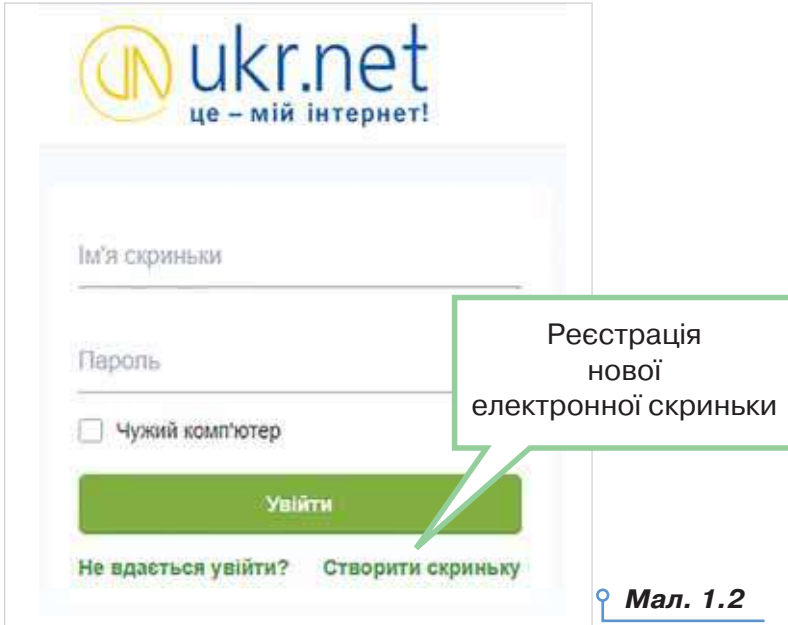
Наприклад:

AntonyGood@i.ua, *petrova.d@ukr.net*, *dik_2008@gmail.com*

Ім'я поштової скриньки має бути унікальним, тобто не повторюватися на обраному поштовому сервері. Використання пропусків, розділових знаків і символів кирилиці в електронній адресі не допускається. Якщо необхідно відокремити в адресі поштової скриньки прізвище від імені або ініціалів, можна використовувати символ підкреслювання «_» або крапку «.». Останнім часом на кожному поштовому сервері встановлено свої правила формування адреси поштової скриньки.

3. Як зареєструвати електронну скриньку на поштовому сервері?

На головній сторінці будь-якого поштового сервера (мал. 1.2) надається можливість доступу до скриньки, створеної на цьому сервері, та реєстрації нової поштової скриньки. Для реєстрації поштові сервери пропонують прийняти угоду про використання поштової скриньки, у якій користувач/користувачка має підтвердити свої наміри надати правдиві відомості про себе, не використовувати поштову скриньку для здійснення протизаконних дій, а також дій, що суперечать мережевому етикету.



Мал. 1.2

Якщо користувач/користувачка погоджується із запропонованими умовами, на екрані відобразиться сторінка, що містить анкету, яку потрібно заповнити. Її потрібно заповнити лише один раз під час створення скриньки. Деякі поля анкети є обов'язковими для заповнення. Як правило, їх позначено зірочкою. Одним з обов'язкових полів є **адреса поштової скриньки**. Вона обирається за бажанням. Ліва частина адреси поштової скриньки відображає ім'я користувача/користувачки, його/її прізвище або вибирається довільно. Її називають **логіном**. Фактично цим ім'ям називають папку, у якій зберігається на сервері вся кореспонденція користувача/користувачки.

При виборі логіна необхідно враховувати, що, можливо, папку з таким іменем уже хтось має, тому бажано обирати логін, який був би унікальним, наприклад, ініціали та прізвище. Під час реєстрації нової поштової скриньки можна перевірити, чи доступним є обраний логін, чи він уже був зареєстрований кимось раніше.

При введенні **пароля**, як і в інших програмах, введені символи на екрані для конфіденційності не відображаються, замість них відображаються зірочки або кружечки. Пароль призначено для захисту поштової скриньки від доступу сторонніх осіб, тому його слід тримати в таємниці. Сукупність логіна й пароля складає **обліковий запис**, який використовують для розпізнавання користувача/користувачки під час звертання до ресурсів сервера. Обліковий запис також називають **акаунтом** (англ. *account* — обліковий запис).

Для захисту своєї електронної скриньки потрібно дотримуватись таких правил:

- 1) при реєстрації створювати надійний пароль. Наприклад, ненадійними паролями є такі, які використовують малу кількість символів, дублюють частину логіна, вказують на ім'я чи інші персональні дані — номер телефону, дату народження тощо;
- 2) нікому не повідомляти свій пароль;
- 3) при реєстрації інших скриньок використовувати нові паролі.

Один із кроків реєстрації передбачає введення символів, наведених на зображенні. Як правило, це спотворені літери або цифри. Введення таких символів підтверджує, що реєстрацію виконує людина, а не комп'ютерна програма.

За успішного проходження реєстрації буде створено поштову скриньку із зазначеною адресою, доступ до якої можна отримати шляхом введення відповідного логіна й пароля, які були обрані при реєстрації.

На нову скриньку з поштового сервера може бути надіслано першого листа з повідомленням про успішне створення скриньки та деякі поради щодо роботи з електронною поштою.



ДІЄМО

Вправа 1. Реєстрація власної електронної скриньки.

Завдання. Зареєструвати власну електронну скриньку на поштовому сервері *mail.ukr.net*.

1. Відкрийте браузер і до рядка адреси введіть адресу поштового сервера *mail.ukr.net*.
2. На головній сторінці поштового сервера оберіть посилання *Створити скриньку*. Через деякий час за наявності зв'язку з інтернетом на екрані відобразиться анкета для заповнювання.
3. Введіть бажане ім'я користувача (наприклад, *s.ivanov*). Якщо користувача/користувачку з такою адресою вже зареєстровано, змініть адресу та знову перевірте доступність обраного імені.
4. Заповніть інші обов'язкові поля анкети й оберіть за власним бажанням пароль.

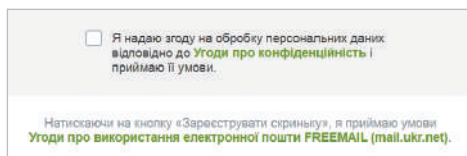
5. Ознайомтеся з *Умовами використання скриньки та Угодою про конфіденційність*.
6. Натисніть кнопку *Отримати скриньку*.
7. Адресу й пароль своєї нової поштової скриньки занотуйте у свій зошит або блокнот.



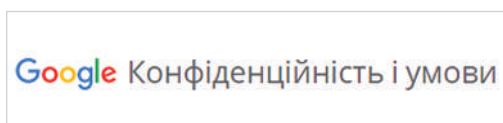
Досліджуємо

Вправа 2. Умови використання поштової скриньки.

Завдання. Дослідіть умови використання поштової скриньки, що надаються поштовими серверами *mail.ukr.net* і *gmail.com*. Для цього ознайомтеся з умовами користування скринькою й угодою конфіденційності на *ukr.net* (мал.1.3), загальними положеннями та умовами й політикою конфіденційності *Google* (мал.1.4), що пропонуються при реєстрації нової електронної скриньки та збережені в однойменних файлах у папці *Електронна пошта*. Визначте спільне й відмінне в умовах використання електронної скриньки на цих поштових серверах.



Мал. 1.3



Мал. 1.4



Обговорюємо

1. Яке призначення має електронна пошта?
2. Які переваги й недоліки має електронна пошта в порівнянні зі звичайною поштою?
3. Скільки електронних поштових скриньок може мати користувач?
4. Як записується адреса електронної пошти? Наведіть приклади.
5. У чому полягає сутність реєстрації користувача на поштовому сервері?



Працюємо в парах

6. По черзі ставте одне одному запитання щодо реєстрації власної електронної скриньки. Речення розпочинайте зі слів: *як, який, навіщо, що відбуватиметься* тощо.
7. З переліку запропонованих записів оберіть ті, які можуть бути адресою електронної пошти:



- 1) *olena2020@i.ua*
- 2) *олена2020@i.ua*
- 3) *olena2020@Д.ua*
- 4) *olena2020@ukr.net*
- 5) *olena:/&@i.ua*
- 6) *OIEA2020@ukr.net*
- 7) *olena 2020@mail.ukr.net*

Обговоріть причини, чому не всі ці записи можуть бути адресою електронної пошти.



8. Обговоріть, як подати спільне й відмінне між роботою зі звичайною поштою та електронним листуванням у вигляді таблиці. Доберіть критерії порівняння. Заповніть таблицю, розподіливши між собою обов'язки.



9. Обговоріть алгоритм створення поштової скриньки на поштово-му сервері. Називайте кожен команду алгоритму по черзі. Подайте створений алгоритм у вигляді схеми.



Працюємо самостійно

10. Знайдіть в інтернеті список безкоштовних поштових серверів і відомості про їхні особливості.



11. Запропонуйте три можливі адреси електронної пошти для своєї бабусі на одному з відомих вам поштових серверів. Поясніть свій вибір.



12. На поштово-му сервері є правила створення паролю: кількість символів не менша 8, обов'язково має бути велика й малі літери, цифри та спеціальні знаки. Серед запропонованих паролів обери-ть такі, що відповідають правилам і будуть надійними й легкими для введення:

- 1) *17071967*
- 2) *Data17071967\$*
- 3) *Antkiv17071967*
- 4) *1767Fynrsd!*
- 5) *lfgrfyfgjdfhf1!*

Повторюємо



Розгляньте схему та складіть розповідь, як пов'язані її складові.



Електронна поштова скринька

Адреса електронної пошти

Поштовий сервер

Електронна пошта

Логін

Пароль



Використання електронної пошти

ВИ ДІЗНАЄТЕСЬ:

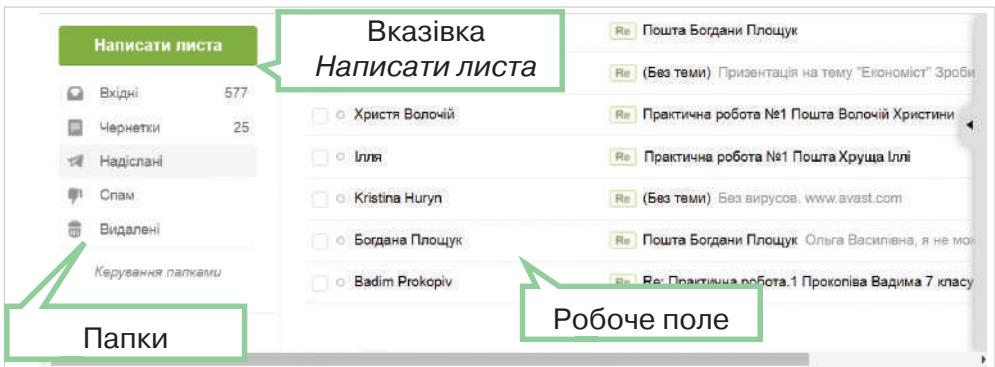
- ▶ як працювати з електронною скринькою;
- ▶ як створити та надіслати електронного листа;
- ▶ які дії можна виконувати з отриманими листами;
- ▶ як можна впорядковувати листи в електронній скриньці;
- ▶ як користуватись адресною книгою та створювати списки розсилання.



Вивчаємо

1. Як працювати з електронною скринькою?

Для листування з друзями потрібно дізнатися їхні електронні адреси й написати їм листа. Спочатку слід під'єднатися до інтернету та зайти на головну сторінку поштового сервера. Після введення логіна й пароля і натискання кнопки *Увійти* буде відображено вміст вашої електронної скриньки (мал. 2.1).



Мал. 2.1

У вікні, що відкриється, відобразяться адреса електронної скриньки й вказівки для роботи з електронними листами. Зовнішній вигляд та місце розміщення деяких вказівок може час від часу змінюватися, коли відбуваються оновлення на поштовому сервері. Усі листи, що зберігаються у скриньці, упорядковуються у стандартних папках *Вхідні*, *Надіслані*, *Чернетки*, *Видалені* тощо.

У робочому полі вікна міститься список заголовків листів, що зберігаються в обраній папці. Заголовок листа включає ім'я тих, хто

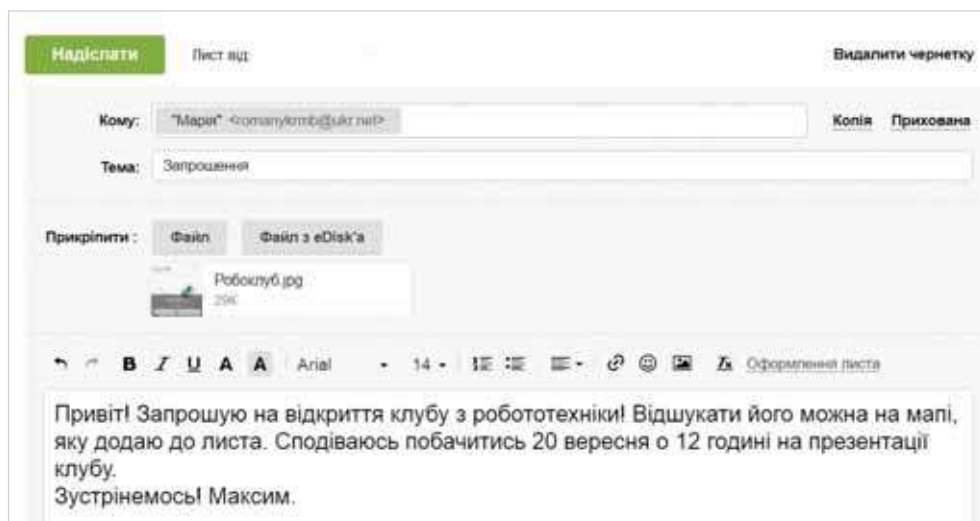
відправляє, тему, дату й час відправлення. Ці відомості дають змогу одержувачу/одержувачці відразу дізнатися про зміст електронного листа, не відкриваючи його. Заголовки нових непрочитаних листів відображаються жирним шрифтом.

Ліворуч від заголовків кожного листа розташовані прапорці . За їх допомогою можна позначати листи, з якими слід виконати певні дії — видалити, позначити як прочитані чи непрочитані тощо.

2. Як створити та надіслати електронного листа?

Щоб створити та надіслати нове повідомлення, застосовують вказівку *Написати листа*. При цьому відкривається вікно поштового повідомлення, що складається зі службової частини й робочого поля.

Службова частина містить кілька полів, заповнення яких нагадує підписування конверта (мал. 2.2).



 **Мал. 2.2**

Поле *Кому* — для введення адреси електронної пошти одержувача/одержувачки — є обов'язковим для заповнення.


На відміну від поштового листування, електронного листа можна відправити кільком адресатам. Якщо лист призначено кільком особам, то їхні адреси слід ввести через кому в полі *Кому*. Також можна надіслати копії листа. Наприклад, лист-запрошення до участі у спортивних змаганнях надсилається учням 7 класу, і доцільно надіслати *Копію* листа тренеру/тренерці команди, а *Приховану копію* — учасникам групи підтримки команди. Щоб додати такі поля до службової частини листа, слід обрати *Копія* або *Прихована* зліва від поля *Кому*.

До поля *Тема* слід ввести зміст листа в стислій формі, зазначаючи його мету й використовуючи основні ключові слова. За правилами етикету електронного листування тема листа є обов'язковою. Поле *Від*

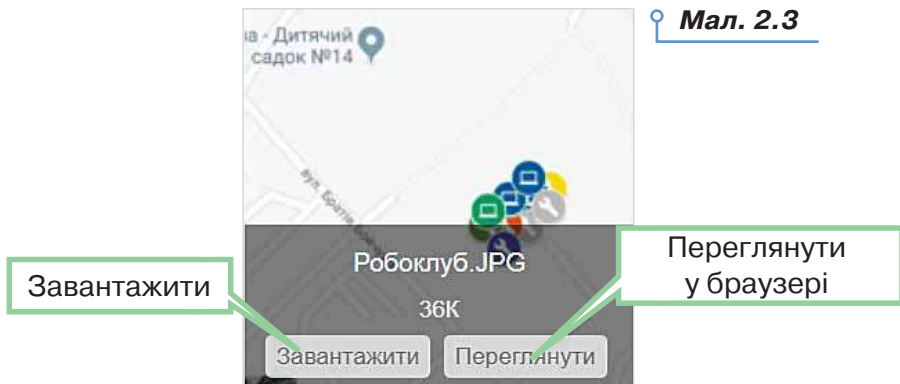
заповнюється автоматично: вписується ім'я власника/власниці відкритої поштової скриньки та його/її адреса. Якщо до листа необхідно приєднати файл, слід у розділі *Прикріпити* натиснути кнопку *Файл* та обрати потрібний файл. До робочого поля вводиться текст листа. Поки лист не готовий, він зберігається в папці *Чернетки*. Це дає змогу повернутися до роботи з ним у будь-який час. Для відправлення листа необхідно натиснути кнопку *Надіслати*. Якщо дію виконано успішно, то на екрані з'явиться відповідне повідомлення.

3. Які дії можна виконувати з отриманими листами?

Щоб прочитати листа, що надійшов, треба відкрити папку *Вхідні* та клацнути мишею на заголовку листа — текст повідомлення відкриється в новому вікні. При цьому відображається заголовок листа й саме повідомлення.

Якщо лист містить вкладені файли, то їхній список відображається після тексту листа. Приєднані файли позначаються значком у заголовку листа .

Для перегляду вмісту вкладених файлів їх слід відкрити у вікні браузера. Потрібні вам файли можна зберігати на своєму комп'ютері (мал. 2.3).



Мал. 2.3

Важливі листи дозволяється відмічати спеціальними позначками:



Непотрібні листи можна *Видалити*. При цьому вони потрапляють до папки *Видалені*, так само, як і видалені файли та папки переміщуються до папки *Кошик*.

Якщо листи було видалено випадково, їх можна відновити. Для цього слід виконати вказівку *Відновити* й вказати потрібну папку та файл. Якщо папка *Видалені* містить лише неважливі листи, доцільно *Очистити* папку, при цьому листи з неї будуть видалені остаточно.

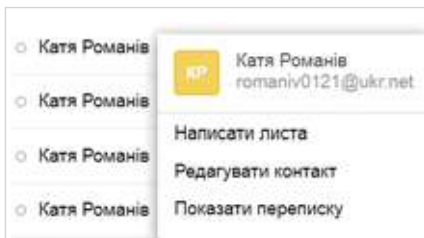
На одержаний лист можна відповісти або переслати його іншому адресату за допомогою відповідної вказівки. У робочому полі слід

ввести відповідь або коментарі до листа, що пересилається, та натиснути кнопку *Надіслати*.

При використанні вказівки *Відповісти* не потрібно заповнювати службову частину у вікні поштового повідомлення — вона заповнюється автоматично:

- у полі *Кому* вставляється адреса отримувача/отримувачки, оскільки це відповідь на отриманий лист;
- перед назвою теми з'являється слово *Re:*, яке вказує на те, що це відповідь (від англ. *to Replay* — відповідати). Текст листа, на який дається відповідь, дозволяється не видаляти або видалити частково й написати відповідь перед або під ним.

Якщо затримати вказівник миші над назвою адресата, то можна переглянути деталі контакту, написати листа, редагувати контакт або отримати список усіх електронних листів від адресата (мал. 2.4).



Мал. 2.4



ДІЄМО

Вправа 1. Створення та відправлення електронного листа із вкладенням.

Завдання. Створіть електронний лист — запрошення на відкриття нового клубу з робототехніки, приєднайте до листа файл *Робоклуб.jpg* та відправте лист.

1. Відкрийте вікно браузера, введіть: адресу поштового сервера *mail.ukr.net*, свій логін і пароль доступу до поштової скриньки — та натисніть кнопку *Увійти*.
2. У вікні поштової скриньки оберіть вказівку *Написати листа*.
3. Підготуйте лист для вашого сусіда чи сусідки, що працює поруч, — запитайте електронну адресу та введіть її в поле *Кому*. До поля *Тема* введіть слово *Запрошення*. У робочому полі введіть текст:

*Привіт! Запрошую на відкриття клубу з робототехніки! Відшукати його можна на мапі, яку додаю до листа. Сподіваюсь побачитись 20 вересня о 12.00 на презентації клубу!
До зустрічі!*

4. У розділі *Прикріпити* натисніть кнопку *Файл* та оберіть файл *Робоклуб.jpg* з папки *Електронна пошта*.
5. Натисніть кнопку *Надіслати*. Якщо з поштового сервера прийде повідомлення про неможливість відправлення листа, переконайтеся у правильності адреси, записаної в полі *Кому*, та спробуйте ще раз відправити лист.

Вправа 2. Дії з отриманим електронним листом, що містить вкладений файл.

Завдання. Перевірте пошту, перегляньте вміст отриманого листа й збережіть у своїй папці прикріплений до листа файл. Дайте відповідь на лист і відправте цей лист сусіду чи сусідці, що працює поруч.

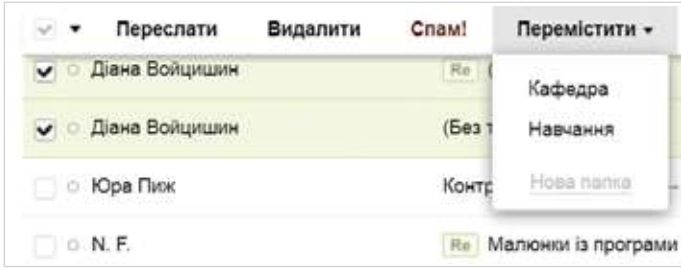
1. У вікні своєї поштової скриньки в переліку листів, що містяться в папці *Вхідні*, знайдіть повідомлення з темою *Запрошення* та клацніть мишею на його заголовок.
2. Перегляньте лист, що надійшов. Збережіть у своїй папці прикріплений до листа файл, для цього виконайте вказівку *Завантажити* для файла *Робоклуб.jpg* (див. мал. 2.3).
3. Виберіть вказівку *Відповісти*.
4. Створіть лист-відповідь, у якому подякуйте за уважність і висловіть сподівання на зустріч на презентації. Натисніть кнопку *Надіслати*.
5. Відкрийте папку *Вхідні* та лист із темою *Запрошення*.
6. Виконайте вказівку *Переслати*. У поле *Кому* введіть адресу сусіда чи сусідки, що працює поруч. Залишіть текст листа, який пересилається, і на початку листа додайте повідомлення від себе, що запрошення отримали й пропонуєте приєднатися. Зверніть увагу на те, що при пересиланні повідомлень прикріплені файли також додаються до нового листа. Натисніть кнопку *Надіслати*.
7. У правій верхній частині вікна поштової скриньки клацніть на адресі власної поштової скриньки та оберіть вказівку *Вихід*.
8. Закрийте вікно браузера.

4. Як можна впорядковувати листи в електронній скриньці?

Електронна скринька на будь-якому поштовому сервері містить набір стандартних папок, наприклад, *Вхідні*, *Надіслані*, *Видалені* та ін. Для ефективної роботи з листами можна створювати додаткові вкладені папки аналогічно до структури папок на дисках. Для внесення змін у структуру папок електронної скриньки слід обрати інструмент *Керування папками*. При цьому можна створювати нові папки в папці *Вхідні*, перейменовувати раніше створені папки або видаляти їх за допомогою відповідних інструментів.

Перелік створених вкладених папок відображається, якщо обрати посилання *Інші папки*.



Щоб перемістити листи до потрібної папки, слід позначити ці листи й обрати вказівку *Перемістити*. У списку наявних папок необхідно обрати ім'я потрібної папки (мал. 2.5).

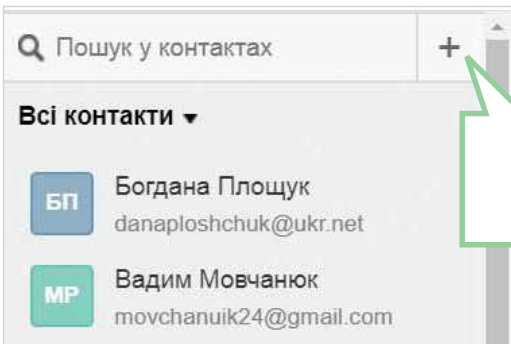


Мал. 2.5

5. Як користуватись адресною книгою та створювати списки розсилання?

Роботу з великою кількістю електронних листів від різних адресатів можна організувати ефективно. Доцільно скласти список людей, з якими ви постійно листуєтесь, та внести їхні електронні адреси до адресної книги. Адресну книгу призначено для зберігання на поштово-сервері різних відомостей про адресатів: адреси електронної пошти, домашньої та службової адреси, номерів телефонів тощо. У разі потреби можна внести на сторінку нотаток будь-які додаткові відомості. Для роботи з адресною книгою слід навести вказівник миші на об-

ласть у правій частині вікна . При цьому буде відображено *Панель контактів* (мал. 2.6). Записи, що додані до адресної книги, можна переглядати, редагувати (мал. 2.7) та видаляти. Для створення нових записів в адресній книзі використовують інструмент *Додати контакт* . Щоб знайти контакт у адресній книзі, використовують поле пошуку. Пошук здійснюється навіть за фрагментом опису контакту.



Інструмент
Додати контакт

Мал. 2.6

Редагування контакту [X]

Ім'я: Анастасія

Прізвище: Назар, 7

Email: nastyasupercrazy@gmail.com

Група: Учні ▼

Телефон: []

Компанія: []

Дата народження: дд.мм.рррр

Відмітки: []

[Зберегти] [Відміна]

Мал. 2.7

Коли адресна книга заповнена, то можна замість введення електронної адреси в полі листа *Кому* або *Копія* скористатися списком адрес, що розкривається. Адреса, вибрана з адресної книги, автоматично вставиться до відповідного поля. Часто при листуванні потрібно відправляти листи одній тій самій групі адресатів. У цьому разі слід об'єднати адреси в групу під одним іменем. Створювати нові групи можна при додаванні та редагуванні контактів в адресній книзі (мал. 2.7).





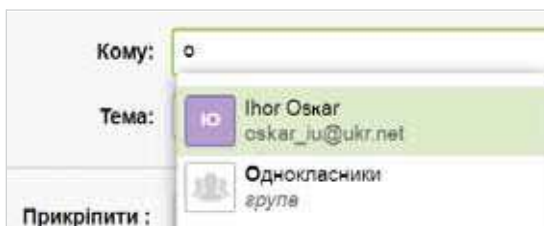
ДІЄМО

Вправа 3. Робота з адресною книгою.

Завдання. Створіть в адресній книзі групу *Однокласники*, внесіть до адресної книги електронні адреси однокласників/однокласниць, що працюють ліворуч і праворуч. Відправте листа з темою *Адресна книга* всім адресатам групи *Однокласники*.

1. Відкрийте вікно браузера, перейдіть на головну сторінку поштового сервера *mail.ukr.net*, введіть свій логін і пароль доступу до поштової скриньки й натисніть кнопку *Увійти*.

- У вікні своєї поштової скриньки наведіть вказівник миші на область у правій частині вікна , щоб відкрити *Панель контактів*.
- Оберіть інструмент *Додати контакт* . Заповніть поля у вікні *Новий контакт*, щоб додати відомості про однокласника чи однокласницю, що працює праворуч. У розділі *Група* оберіть посилання *Нова група*, введіть назву *Однокласники* та натисніть кнопку *Додати* (мал. 2.7).
- Оберіть створену групу *Однокласники* (мал. 2.7). Збережіть зміни. Аналогічно створіть новий контакт і додайте відомості про людину, з якою працюєте разом.
- Оберіть вказівку *Написати листа*. У поле *Кому* введіть першу літеру назви групи — «О». Зі списку, що відкриється, оберіть *Однокласники* (група) (мал. 2.8).



Мал. 2.8

- Підготуйте лист із темою *Адресна книга*, призначений усім адресатам із групи *Однокласники*. У тексті листа опишіть переваги використання адресної книги при створенні листів.
- Перегляньте отримані листи та дайте на них відповіді.

Вправа 4. Створення власної папки та переміщення листів.

Завдання. Створіть у папці *Вхідні* папку *Навчання*, перемістіть до неї листи, що надійшли від однокласників.

- У вікні своєї поштової скриньки оберіть інструмент *Керування папками*.
- Натисніть кнопку *Створити папку* (мал. 2.9). У полі *Нова папка* введіть назву папки *Навчання* та натисніть кнопку *Зберегти*.



Мал. 2.9

- Оберіть вказівку *Вхідні*, щоб повернутися до списку листів.
- Позначте листи, які надійшли від однокласників/однокласниць, установивши позначки ліворуч від заголовків потрібних листів.

5. Оберіть вказівку *Перемістити* (мал. 2.5), далі в списку папок оберіть *Навчання*.
6. Відкрийте папку *Навчання*. Переконайтесь, що всі необхідні листи переміщені до цієї папки.



Досліджуємо

Вправа 5. Видалення вкладених файлів.

Завдання. Дослідіть, як можна видалити помилково вкладені файли, що прикріплені до листа.



Обговорюємо

1. Які стандартні папки існують у поштової скриньці на поштовому сервері? Поясніть призначення кожної.
2. Які розділи в службовій частині електронного листа є обов'язковими для заповнення? Чому?
3. Які дії можна виконувати з електронними повідомленнями?
4. Чи можна записати в полі *Кому* при заповненні бланка електронного листа кілька адрес?
5. Чим відрізняються дії *Відповісти* й *Переслати* під час електронного листування?
6. За якими ознаками можна визначити, що до електронного листа було прикріплено та надіслано файл? Як переглянути вміст вкладеного файла?
7. Для чого використовують адресну книгу? Які переваги її використання? Де в житті вам доводилося користуватися адресною книгою; чи доводилося бачити, як використовується адресна книга?
8. Порівняйте роботу з адресною книгою на електронній пошті та в мобільному телефоні.
9. Як створити групу контактів? У яких випадках зручно користуватися групами контактів?
10. Для чого створюють нові папки в електронній скриньці? Як перемістити листи до створеної папки?



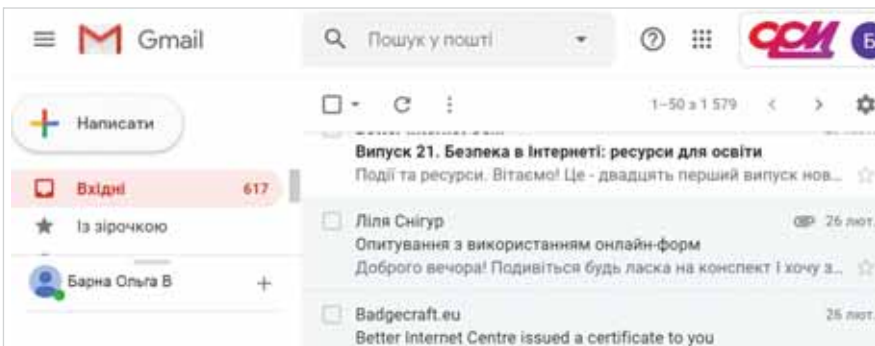
Працюємо в парах

11. Пограйте в гру «Тема листа». По черзі називайте зміст листа й добирайте тему для такого листа. Обговоріть для кожного запропонованого змісту, чи однаково ви сформулювали тему. Яка тема є більш вдалою?
12. Виділіть лист із темою *Запрошення* та оберіть вказівку *Видалити*. Відкрийте папку *Видалені*. Визначте й обговоріть, за допо-

могою яких вказівок можна відновити видалений лист, видалити його назавжди, остаточно видалити всі листи з цієї папки. Сформулюйте поради, у яких випадках доцільно використовувати вказівки *Очистити папку* та *Видалити назавжди*. Власні поради надішліть для обговорення на електронну пошту однокласника чи однокласниці з іншої пари. Надішліть відповідь на листа від іншої пари, де прокоментуйте подані поради.

★ **13.** Пограйте в гру «Дія — алгоритм». По черзі хтось із учасників називає дію, яку можна виконувати з електронними листами: створювати, отримувати, переглядати, відповідати, пересилати, видаляти, а хтось — алгоритм для виконання такої дії.

★ ★ **14.** Визначте, чи однакові дії можна виконувати з електронними листами, якщо скриньки розміщені на різних поштових серверах. Розподіліть ролі: хтось працює з електронною скринькою, розміщеною на поштовому сервері *mail.ukr.net* (мал. 2.1), а хтось — на *gmail.com* (мал. 2.10). По черзі давайте одне одному завдання щодо роботи з електронними листами та пояснюйте, як виконати ці дії в кожному з випадків. За результатами порівняння побудуйте діаграму Венна.



📌 **Мал. 2.10**

Працюємо самостійно

★ **15.** Знайдіть в інтернеті зображення, що ілюструють ваше хобі. Відправте своїм товаришам відповідні листи про своє хобі з вкладеними файлами, у яких зберігається одне зі знайдених зображень. Збережіть зображення, що були вкладені до отриманих листів, у своїй структурі папок.

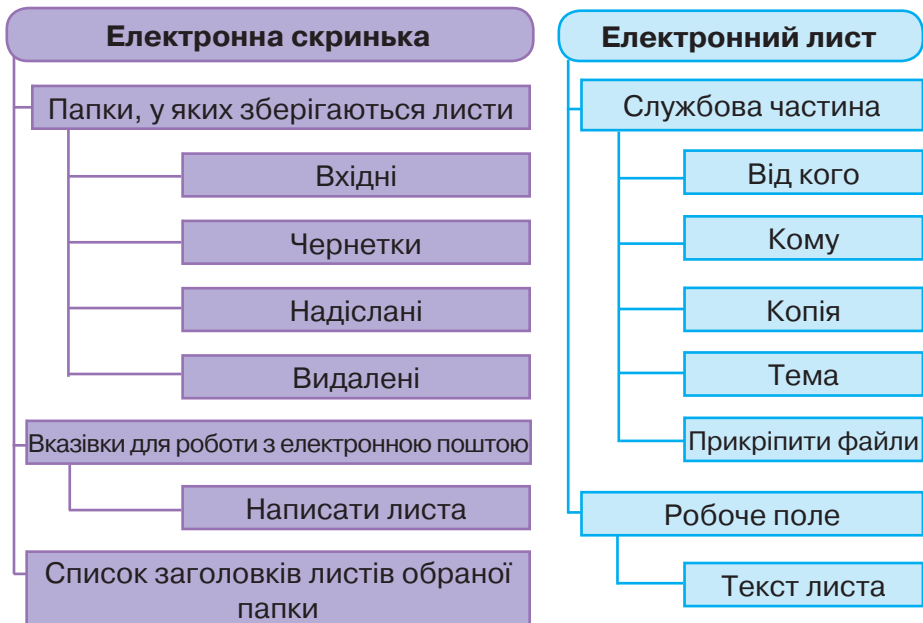
★ **16.** Створіть текстовий документ, що містить опис одного з природних чудес України. Тему та зміст листа придумайте самостійно. Відправте своєму сусіду чи сусідці листа, до якого приєднайте створений файл. Копію листа відправте на навчальну

електронну скриньку. Перевірте пошту. Перегляньте отриманого листа та збережіть у своїй папці файл, прикріплений до нього.

- ★ 17. Видаліть непотрібні листи. Відкрийте папку *Видалені*. Позначте один з листів і виконайте вказівку *Відновити*, оберіть папку *Вхідні*. Виконайте вказівку *Очистити папку*.
- ★★ 18. Створіть схеми, що відображають алгоритми видалення листів з папки *Вхідні* та відновлення випадково видалених листів з папки *Видалені*. Оберіть середовище для створення схем самостійно.
19. Створіть у папці *Вхідні* власну структуру папок для впорядкування листів. Перемістіть листи з папки *Вхідні* у створені папки. Поміркуйте, чи доречно створювати папку *Навчання* в такій структурі.
- ★ 20. Створіть групи контактів *Сім'я* та *Друзі*. Які ще групи контактів були б корисними для вас? Створіть їх.
- ★★ 21. Створіть схему для відображення дій, які можна виконувати за допомогою адресної книги. Схему можна підготувати, наприклад, за допомогою графічного або текстового редакторів. Відправте учителю або вчительці листа, до якого долучить схему як вкладення. Тему листа й потрібний текст придумайте самостійно.

Повторюємо

Розгляньте схеми та складіть розповідь, як пов'язані їхні складові.





3

Безпека в інтернеті

ВИ ДІЗНАЄТЕСЬ:

- ▶ що передбачає етикет електронного листування;
- ▶ яких правил слід дотримуватися для безпечного електронного листування;
- ▶ що таке спам і фішинг.



Вивчаємо

1. Що передбачає етикет електронного листування?

В інтернеті існує неформальний кодекс поведінки, за яким певним чином регулюється спілкування користувачів між собою — етикет електронного листування. Згідно з ним необхідно дотримуватися таких правил:

- чітко формулювати тему повідомлення та не залишати поле *Тема* порожнім;
- починати лист зі звернення до адресата/адресатки та привітання;
- свою думку в листі висловлювати коректно, співрозмовника не критикувати й не повчати, не писати погано про третю особу;
- не друкувати весь текст листа великими літерами — його важко читати, це сприймається як крик;
- завжди перевіряти правопис електронного повідомлення;
- поводитися ввічливо — не забувати вживати слова «будь ласка», якщо звертаєтесь до кого-небудь із проханням, подякувати, якщо хтось допомагає вам;
- повідомляти адресатові, які файли додаються до листа, оскільки часто через вкладені файли розповсюджуються комп'ютерні віруси, не рекомендується відправляти великі за обсягом вкладення до листа;
- вчасно відповідати на кожне особисте повідомлення. Якщо ви не маєте можливості зразу дати розгорнуту відповідь, доцільно підтвердити одержання листа та пообіцяти відповісти найближчим часом;
- у відповіді на повідомлення залишати його фрагменти — тобто цитувати найсуттєвіші частини листа.

Листуючись в інтернеті, ви зазвичай не бачите співрозмовника чи співрозмовницю, тому не можете знати, які емоції вкладені в те або

інше повідомлення. Для передавання своїх емоцій у листах часто використовують смайлики (від англ. *smile* — посмішка), або емотикони (від англ. слів *emotion* — емоція та *icon* — образ). Наприклад:

- :) — посмішка; не сприймайте мене занадто серйозно;
- :(— сумна або сердита особа;
- ;) — щаслива особа, що підморгує;
- :D — занадто широко відкритий рот (від сміху);
- :-e — розчарування.



ДІЄМО

Вправа 1. Порадник.

Завдання. У файлі *Фрагменти листів*, який зберігається в папці *Сервіси інтернету*, запишіть свої поради, які можуть допомогти їхнім авторам/авторкам виправити листи так, щоб вони відповідали правилам етикету електронного листування. Створіть електронне повідомлення, вкладіть до нього змінений файл і надішліть його на навчальну електронну скриньку.

1. Відкрийте файл *Фрагменти листів* у папці *Сервіси інтернету*.
2. Розгляньте приклад виконання завдання (мал. 3.1).

Порада: Варто заповнити тему листа. Наприклад, *Про екскурсію*.

Мал. 3.1

3. Продовжте заповнювати таблицю порад. Зверніть увагу на приклад: указано не тільки, що слід змінити, а й запропоновано як саме.
4. Збережіть файл *Фрагменти листів* у свій папці, додайте до імені файла своє прізвище.
5. Напишіть зі своєї електронної скриньки листа на адресу навчальної скриньки. Сформулюйте тему листа: *Порадник_Прізвище*. Текст листа заповніть самостійно відповідно до етикету електронного листування. Прикріпіть до листа файл із виконаним завданням.
6. Надішліть листа й завершіть роботу з поштовим сервером.

2. Яких правил слід дотримуватися для безпечного електронного листування?

Для безпечного листування завжди дотримуйтеся таких правил:

- не наводьте в листах приватні дані про себе та свою родину, якщо особисто не знайомі з адресатом;
- не погоджуйтесь на особисту зустріч із особами, які знайомі вам тільки з електронного листування, — вони можуть виявитися не тими, за кого себе видають;
- не повідомляйте пароль від електронної поштової скриньки стороннім особам, щоб листування не потрапило до чужих рук і щоб ніхто не міг надсилати листи від вашого імені;
- ставтесь критично до вмісту отриманих електронних листів — не все, що в них наведено, є правдою;
- не відповідайте на листи, що надходять від невідомих осіб;
- не відкривайте прикріплені файли, отримані в листах від невідомих осіб: у них можуть міститися шкідливі програми; такі листи рекомендується видаляти.

Коли людина публікує фотографії, аудіозаписи, відео, текстові повідомлення чи пише на сторінках друзів, то залишає в мережі так званий слід. Навіть якщо електронне повідомлення видалено з вашої скриньки, то адресат, який його отримав, може цим повідомленням скористатись.

Цифровий слід (від англ. *footprinting*) — це всі дії людини в інтернеті, з яких складається її електронний портрет.

Особливістю мережі інтернет є те, що з неї ніщо не зникає безслідно. Тому:

- цифровий слід користувача інтернету (як і будь-який інший вміст у мережі) може побачити хто завгодно;
- вміст, опублікований у мережі, важко або навіть неможливо видалити.

Ось чому так важливо піклуватися про свою конфіденційність, думати, перед тим як повідомляти будь-яку інформацію про себе чи свою родину. Варто публікувати лише вміст, який ви вважаєте цілком безпечним для себе й своєї теперішньої та майбутньої репутації.



До цифрового сліду також відносять активність користувача чи користувачки інтернету. За кількістю відвідувань деякого сайту чи сторінки в соціальній мережі, наявних контактів можна відслідкувати інтереси, переваги, коло спілкувань тощо.



Вправа 2. Репутація в інтернеті.

Завдання. Перегляньте відео за посиланням <https://cutt.ly/frlMhQ6>.

У файлі *Репутація в інтернеті*, який збережено в папці *Сервіси інтернету*, заповніть схему правил задля підтримки доброї репутації в інтернеті.

1. Відкрийте файл *Репутація в інтернеті*, який збережено в папці *Сервіси інтернету*. Пригадайте, як у текстовому процесорі працювати зі схемами.
2. Перегляньте відео за вказаним посиланням. Прослухайте рекомендації від *Google for Education*. Заповніть схему після перегляду або під час її. Для цього зупиняйте трансляцію відео, як тільки захочете додати щось до схеми. Для цього клацніть під відео . Для продовження відтворення клацніть .
3. Збережіть файл *Репутація в інтернеті* у своїй папці, додавши до імені файла своє прізвище.
4. Надішліть файл виконання завдання на навчальну скриньку.

3. Що таке спам і фішинг?

Електронна пошта часто використовується для інформаційних атак рекламодавцями. Листи, що містять такі повідомлення, належать до *спаму*, а людей, які займаються їх розсиланням, називають спамерами.

Спам — небажані рекламні електронні листи, повідомлення на форуми, телефонні дзвінки чи текстові повідомлення, що надходять без згоди користувача.

На даний момент поняття спаму включає в себе всі різновиди масової реклами.

Щоб убезпечити себе від спаму, дотримуйтесь таких правил:

- не зазначайте свою електронну адресу на сумнівних сайтах;
- якщо спам все-таки потрапив до вашої поштової скриньки, не відповідайте на ці повідомлення і не користуйтеся вказаними в них посиланнями.

Для того щоб повідомлення із цієї адреси не приходили в папку *Вхідні*, потрібно сумнівний лист перемістити в папку *Спам*. Для цього необхідно:

- 1) позначити  сумнівний лист, не відкриваючи його;

2) обрати вказівку **Спам!** .

Файли з цією адресою будуть переміщуватися в папку *Спам*. Якщо лист виявиться не зловмисним, тоді позначку спаму можна скасувати за допомогою вказівки **Не спам** .

Фішинг (від англ. *fishing* — ловити рибу) — це спроба викрасти ім'я користувача, облікові дані, наприклад, деякої людини, переконавши її, що вона спілкується з кимось, кому довіряє. Зазвичай для фішингу застосовуються текстові повідомлення, електронні листи й інші види онлайн-комунікацій. Фішингові листи та вкладені в них файли (а також сайти й завантажувані файли, посилання на які містять такі листи) можуть заражати комп'ютери користувачів вірусами, що отримують доступ до списку контактів і розсилають людям у ньому нові фішингові повідомлення. Інші шахрайські повідомлення можуть містити сповіщення про те, що пристрій, наприклад, заражений вірусами й користувач повинен негайно завантажити «антивірусне» програмне забезпечення (насправді воно, звісно ж, зловмисне). Тож пам'ятайте: жоден сайт або оголошення не здатні визначити, чи правильно працює ваш комп'ютер!



Отримавши певне повідомлення чи відкривши сайт, завжди перевіряйте, чи вони справжні: чи можна довіряти даним, які на ньому містяться.

Зокрема, завжди спробуйте дати відповіді на запитання:

- ✓ Чи є якісь ознаки сайта, які свідчать про його надійність (значки тощо)?
- ✓ Чи відповідає URL-адреса сайта його назві та заголовку?
- ✓ Чи є на сайті спливаючі вікна? (Такі вікна часто використовуються на шахрайських ресурсах.)
- ✓ Чи починається URL-адреса сторінки з префікса *https://*, перед яким відображається значок зеленого замка? (Це вказує на те, що з'єднання є безпечним і зашифрованим.)
- ✓ Що написано дрібним шрифтом? (Усякі хитрі нюанси та невігідні умови часто пишуть саме так).

Але що робити, якщо ви потрапили на гачок фішингу? Перш за все — не панікуйте!

- Негайно повідомте батьків, учителя чи дорослих, яким довіряєте. Що довше ви зволікаєте, то гіршими будуть наслідки.
- Змініть паролі своїх облікових записів.
- Повідомте всіх друзів, яких ви могли підставити під удар.
- Якщо можливо, то позначте фішингове повідомлення як спам.

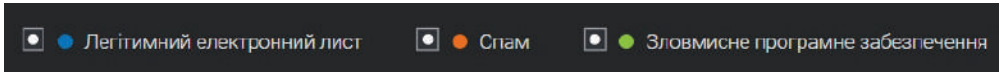


ДІЄМО

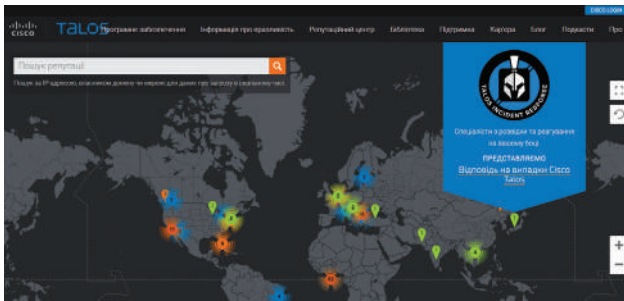
Вправа 3. Важливість правил безпеки.

Завдання. Визначте місця, з яких відбувається розсилання найбільшої кількості спаму та зловмисного програмного забезпечення на поточний час. Скористайтеся сайтом виявлення загроз у режимі реального часу <https://talosintelligence.com/>.

1. Перейдіть на сайт *Talos* за вказаним посиланням.
2. У вікні браузера розгляньте карту світу та позначки:



3. Визначте небезпеки, які відстежено за допомогою цього сервісу на різних континентах світу. Дізнайтеся, звідки найбільше розсилається спаму (мал. 3.2).



Мал. 3.2

4. У меню *Репутаційний центр* оберіть першу вказівку. На сторінці, що відобразиться, перейдіть до показу *Дані електронної пошти та спаму*.
5. Визначте обсяги електронної пошти, спаму на поточну дату. Дослідіть, як вони змінилися у порівнянні з попереднім днем. Зробіть висновки про важливість дотримання правил безпеки електронного листування.



Обговорюємо

1. Що таке етикет електронного листування?
2. Як бути ввічливими при користуванні електронною поштою? Чи завжди потрібно починати лист зі звернення й завершувати подякою?
3. Для чого в електронних листах використовують смайлики?
4. Які небажані листи можна отримати електронною поштою?
5. Що таке цифровий слід? Що потрібно знати при розміщенні будь-якого контенту в інтернеті?
6. Яких правил слід дотримуватися, щоб робота з електронною поштою була безпечною?

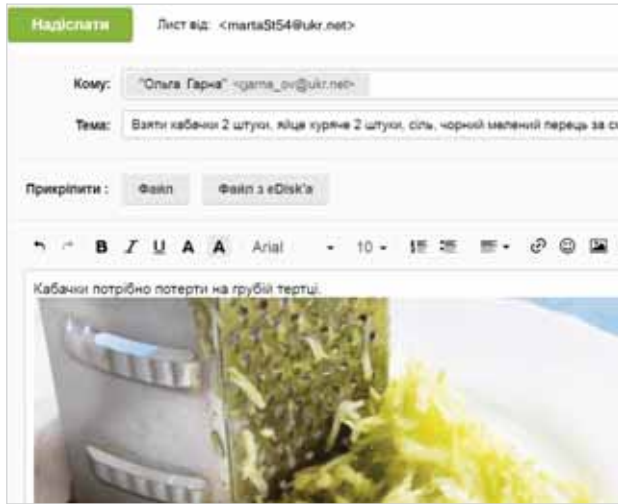
7. Як запобігти потраплянню спаму до комп'ютера?
8. Що таке фішинг? Які загрози розуміють під цим терміном?
9. Як розпізнати й уникнути загроз від фішингу?

Працюємо в парах

10. По черзі ставте одне одному запитання щодо правил етикету електронного листування. Обговоріть, чому необхідно дотримуватися таких правил. Наведіть одне одному приклади, з якими стикалися ваші друзі чи батьки, коли при листуванні не було дотримано такого етикету.
11. Знайдіть в інтернеті, які смайлики використовують під час електронного листування. Які графічні зображення відповідають цим смайликам? Складіть лист, у якому запропонуйте три зображення смайликів, які б ви порадили використовувати під час листування з однокласниками. Відправте листа, використовуючи контакти з групи *Однокласники*, для одержання відгуків на вашу пропозицію.
- ★ 12. Надішліть одне одному листи з темою *Етикет листування*. У тексті листа опишіть правила, яких слід дотримуватися при листуванні. Надішліть одне одному лист у відповідь. У відповіді потрібно внести зміни до вмісту отриманого листа: обрати й залишити лише одне з правил на свій вибір, решту тексту з отриманого листа видалити. На початку листа слід навести власні аргументи, чому важливо дотримуватися цього правила під час листування.
- ★★ 13. Обговоріть, яку інформацію можуть запитувати фішингові сайти:
 - ◆ імена користувача та паролі;
 - ◆ номери соціального страхування;
 - ◆ номери банківських рахунків;
 - ◆ PIN-коди (особисті ідентифікаційні номери);
 - ◆ номери кредитних карток;
 - ◆ дівоче прізвище матері;
 - ◆ ваш день народження;
 - ◆ адресу проживання.
 Поясніть одне одному, з якою метою можуть збирати з фішингових сайтів такі дані та які небезпеки при цьому можуть виникнути.

Працюємо самостійно

14. У вправі за посиланням <https://learningapps.org/view1704874> розмістіть твердження у дві групи.
15. Бабуся хоче надіслати електронне повідомлення своїй подрузі з рецептом приготування нової страви з кабачків. Визначте, яких помилок припустилася бабуся (мал. 3.3).



Мал. 3.3

- ★ 16. Розгляньте відео за посиланнями:

<https://cutt.ly/PrIMxqs>,
<https://cutt.ly/wrIMvBu>,
<https://cutt.ly/grIMne9>.

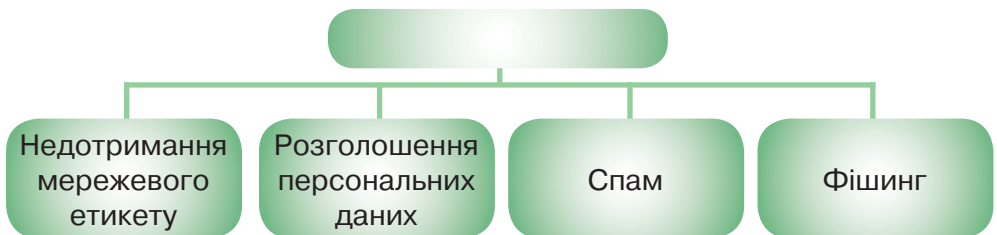
Складіть у текстовому документі контрольний список дій, які б ви порекомендували героям відео. Надішліть цей список усім, чий контакт збережено в папці *Однокласники* в адресній книзі вашої електронної пошти, та як копію — на навчальну електронну скриньку.



- ★ 17. Проаналізуйте фрагмент *Посібника із цифрового громадянства й безпеки*, який збережено у файлі *Фішинг* у папці *Сервіси інтернету*. Визначте, який із сайтів (1–5) є справжнім, а який — шахрайським. Надішліть свою відповідь на навчальну електронну скриньку.

Повторюємо

Розгляньте схему та складіть розповідь, як пов'язані їхні складові. Що можна записати в порожній блок?





Інтернет-ресурси для спільної роботи

4

ВИ ДІЗНАЄТЕСЬ:

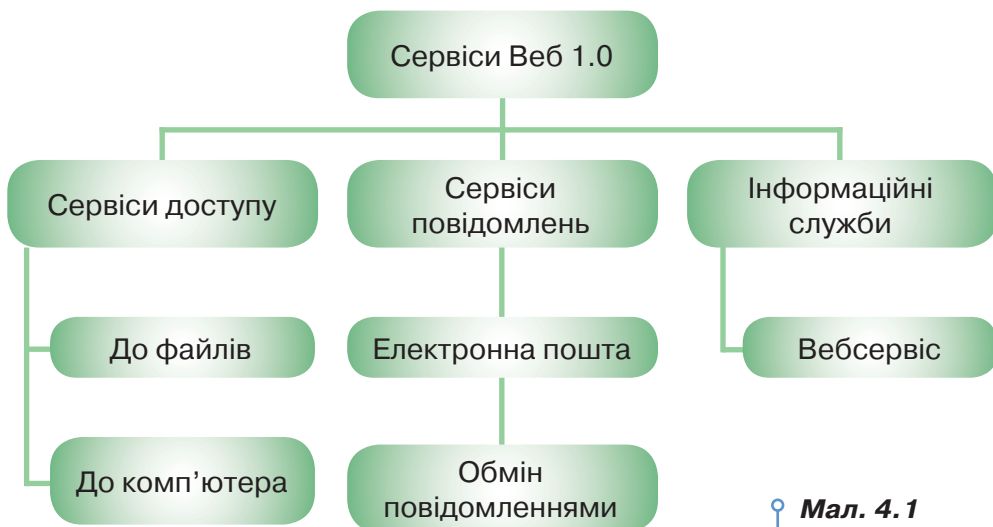
- ▶ якими є сервіси інтернету;
- ▶ що таке персональне освітнє середовище;
- ▶ як у персональному освітньому середовищі застосовують хмарні технології;
- ▶ як організувати доступ до інформаційних матеріалів у персональному освітньому середовищі.



Вивчаємо

1. Якими є сервіси інтернету?

Електронна пошта — це один із сервісів, які доступні користувачам інтернету. Цей сервіс відносять до групи першого покоління, яке називають сервісами Веб 1.0 (мал. 4.1). Наповнювати вмістом сайти та розробляти самі сервіси цієї групи в мережі могли тільки спеціалісти з вебпрограмування, тому основною їхньою властивістю, доступною простому користувачеві/користувачці, є можливість «тільки читати» їхній вміст.



Мал. 4.1

Розрізняють й інші покоління мережевих сервісів інтернету. Розглянемо друге покоління мережевих сервісів інтернету, які об'єднані назвою Веб 2.0.

Користування сервісами Веб 2.0 не потребує спеціальних навичок з вебпрограмування. Кожен може «писати» до вебпростору та співпрацювати з іншими користувачами — обмінюватися даними, зберігати посилання та мультимедійні документи на серверах для хостингу, спільно створювати й редагувати публікації, управляти зв'язками між своїми матеріалами та контентом, створеним іншими авторами, співпрацювати з користувачами інтернету незалежно від індивідуального графіка включення учасників в процес комунікації (мал. 4.2).



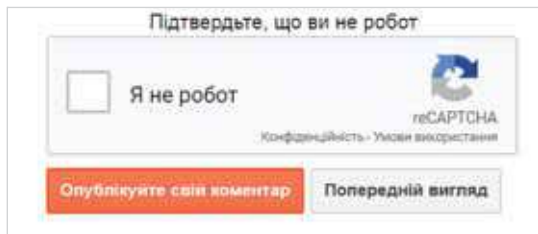
Мал. 4.2

Визначальною особливістю застосування сервісів Веб 2.0 здебільшого є необхідність попередньої реєстрації, коли користувач/користувачка в реєстраційній формі має вказати свій логін і пароль, заповнити інші поля. Як правило, при реєстрації потрібно вводити адресу своєї електронної поштової скриньки. Після успішної реєстрації на вказану електронну адресу надходить повідомлення, у якому підтверджується логін і пароль облікового запису в сервісі та вказується посилання, за яким слід перейти для підтвердження реєстрації або її скасування.

Маючи обліковий запис в окремому сервісі, як, наприклад, акаунт *Google*, користувач/користувачка отримує доступ до інших сервісів.

Деякі сервіси доступні користувачам, яких зареєстрували старші, наприклад, адміністратор мережі у школі. Тоді пароль і логін доступу до сервісу школярі отримують в адміністратора/адміністраторки.

Публікування коментарів або дописів на окремих сервісах передбачає автоматизоване розпізнавання людини чи робота. Зазвичай воно потребує підтвердження від користувача того, що це жива людина, а не робот. Перевірка відбувається або шляхом введення якогось спотвореного тексту, або визначенням серії малюнків (мал. 4.3, а, б, в).



Мал. 4.3, а



Мал. 4.3, б

Мал. 4.3, в

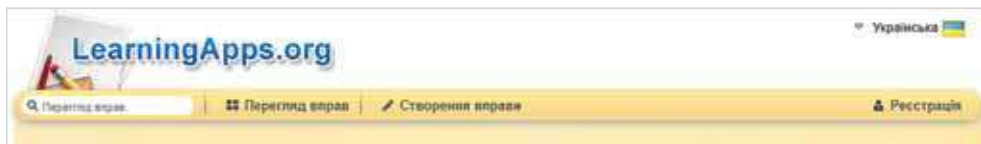


ДІЄМО

Вправа 1. Доступ до навчальних завдань.

Завдання. Отримайте від вчителя або вчительки логін і пароль до навчальних завдань класу в сервісі <https://learningapps.org/>. Змініть налаштування профілю. Виконайте завдання для класу.

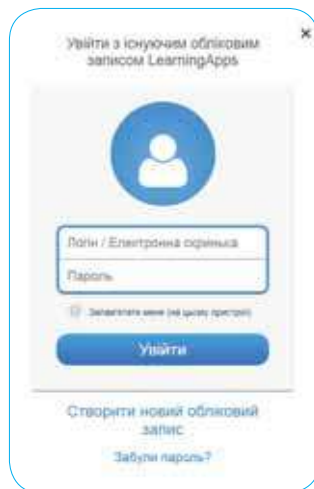
1. Перейдіть на сайт за вказаним посиланням.
2. За потреби змініть мову налаштування. Оберіть вказівку *Реєстрація* (мал. 4.4).



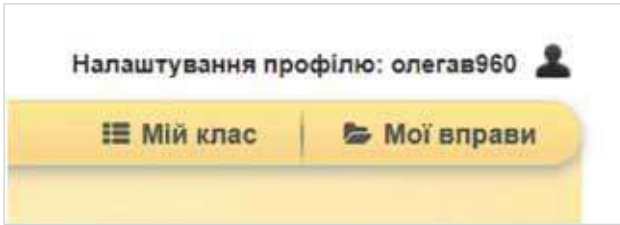
Мал. 4.4

3. У вікні, що відкриється, у поле *Логін/Електронна скринька* введіть наданий вам логін, у поле *Пароль* — відповідно пароль (мал. 4.5). Чому не варто дозволяти запам'ятовувати ваш обліковий запис на комп'ютері у школі? Натисніть *Увійти*.

Мал. 4.5



4. Перевірте, чи отримано подібні зміни у правому верхньому куті сервісу (мал. 4.6).



Мал. 4.6

5. Змініть налаштування профілю, пароль та адресу електронної скриньки (мал. 4.7).



Мал. 4.7

6. Виконайте завдання у вправі, яка розміщена в папці *Мій клас*.
7. Завершіть роботу із сервісом, вийдіть зі свого облікового запису.

2. Що таке персональне освітнє середовище?

Кожен із нас перебуває в деякому інформаційному просторі: навчається у школі, отримує відомості з газет, книжок, телебачення, очно спілкується з рідними та друзями й віртуально — у соціальних мережах. Але інформаційний простір при цьому не вичерпується територією проживання чи колом спілкування. Він розширюється в міру того, наскільки людина залучає до нього нові джерела знань, зокрема спільноти: групи, які створюються в мережі та реальному житті за інтересами. Учасники таких спільнот часто обмінюються своїми знаннями, дають рекомендації щодо використання нових ресурсів, допомагають оволодіти ними. Користувач інтернету, включений у роботу таких віртуальних груп, поступово накопичує дані й перелік засобів, які використовує у своїй діяльності зараз, або планує переглянути чи використати в майбутньому. Таким чином формується його електронне інформаційне середовище.

Інформаційне середовище, яке створює навколо себе людина з метою задоволення власних навчальних потреб (досягнення відповідних навчальних цілей), має назву **персонального освітнього середовища**.

Персональне освітнє середовище складається з інформаційних матеріалів, які створені іншими користувачами та персонально, а також людей: вчителів, батьків, однолітків та професіоналів.

3. Як у персональному освітньому середовищі застосовують хмарні технології?

Для організації персонального освітнього середовища зазвичай використовують хмарні технології.

Хмарні технології — це інформаційні технології, що передбачають віддалене опрацювання та зберігання даних.

Хмара — це деякий сервер або мережа розподілених центрів опрацювання даних, де зберігаються дані та програми, які використовують користувачі за допомогою глобальної мережі. Хмарні технології дають змогу споживачам використовувати програми без їх встановлення на власний комп'ютер і доступу до особистих файлів з будь-якого комп'ютера, що має доступ до інтернету.





Користування хмарною системою зберігання даних надається за допомогою інтернет-сервісу. Зростання попиту користувачів на такі сервіси зумовлено зручністю користування інформацією, зокрема:

- не потрібні власні потужні комп'ютери;
- зменшуються витрати на придбання програмного забезпечення та його систематичне оновлення;
- необмежений обсяг збереження даних;
- доступність із різних пристроїв і відсутність прив'язки до робочого місця;
- забезпечення захисту даних від втрат і виконання багатьох видів навчальної діяльності, контролю й оцінювання, тестування, відкритості освітнього середовища;
- економія коштів на утримання технічних фахівців.

Але користувачу доведеться пам'ятати пароль доступу, без якого дані назавжди залишаться у хмарному сховищі під замком. Найпростішим хмарним сховищем можна вважати скриньку електронної пошти, на яку можна відправляти дані самому собі.

Найпопулярнішими серед спеціалізованих хмарних сервісів на сьогодні є *Dropbox* і *Google Drive*. Безкоштовний обсяг даних, що можуть бути збережені зараз у хмарних сховищах, подано в таблиці 4.1.

Таблиця 4.1

Назва				
Безкоштовний обсяг	2 GB + 36 GB — за виконання завдань	15 GB	50 GB	7 GB

Хмарне сховище можна використовувати для розміщення даних, які збережено у файлах на локальному диску, створення документів і надання доступу до їхнього перегляду, редагування та коментування іншими користувачами мережі, що належать до вашого кола спілкування. Наприклад, у сховищі *Google Диск* можна використовувати стандартні офісні вебпрограми (табл. 4.2) та доповнювати їх список, підключивши інші додатки.

Таблиця 4.2

Офісна вебпрограма	Призначення	Спрощений відповідник програми з пакетів <i>Office Offline</i>
 <i>Google Документи</i>	Створення і редагування текстових документів	<i>Microsoft Word, LibreOffice Writer</i>
 <i>Google Таблиці</i>	Створення і редагування електронних таблиць	<i>Microsoft Excel, LibreOffice Calc</i>
 <i>Google Презентації</i>	Створення і редагування презентацій	<i>Microsoft PowerPoint, LibreOffice Impress</i>
 <i>Google Малюнки</i>	Створення і редагування малюнків	Убудований редактор графіки в офісних програмах



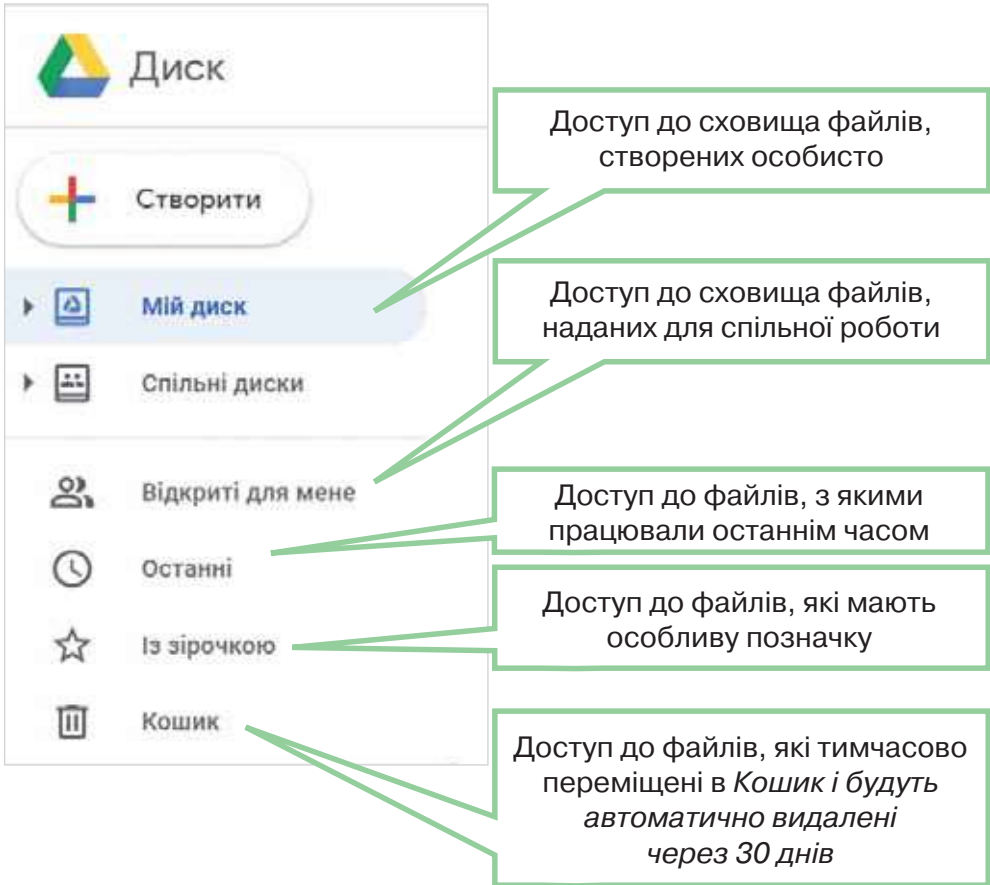
ДІЄМО

Вправа 2. *Google Диск*.

Завдання. Створіть сховище для зберігання файлів у електронному персональному освітньому середовищі засобами *Google Диск*.

1. У браузері відкрийте сторінку пошукової системи *Google*.
2. Увійдіть до сервісу під своїм акаунтом, наданим вам адміністратором/адміністраторкою шкільної мережі.
3. Перейдіть до списку сервісів, обравши інструмент *Додатки Google* у правому верхньому куті вікна браузера.
4. У списку сервісів, що відкрився, оберіть сервіс *Диск*.
5. На запит служби підтримки оберіть персональний акаунт доступу до диска, якщо на комп'ютері працює ще декілька користувачів. Натисніть кнопку *Продовжити (Continue)*.

6. Розгляньте структуру сховища (мал. 4.8).

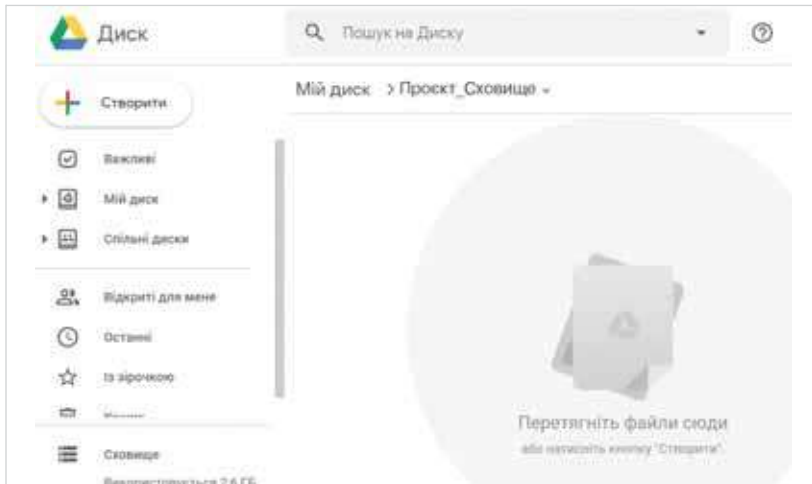


 **Мал. 4.8**

Вправа 3. Публікація документів на Google Диску.

Завдання. Розмістіть в електронному персональному освітньому сховищі документи, збережені на локальному диску.

1. Увійдіть у середовище власного *Google Диска* й натисніть кнопку *Створити*.
2. У списку об'єктів для створення оберіть вказівку *Папка*.
3. У вікні *Нова папка* введіть назву папки — *Проект_сховище* та натисніть кнопку *Створити*.
4. Відкрийте папку *Проект_сховище* у сховищі *Мій диск*. Перетягніть у поле папки файл із папки *Проект* (мал. 4.9).
5. Перевірте, чи відображено у сховищі файли з папки *Проект*. Ознайомтесь із вмістом файлів. Визначте, які питання будуть розв'язуватись у ході проекту.



Мал. 4.9

Вправа 4. Коментування спільного документа.

Завдання. Відкрийте в персональному освітньому середовищі документ *Презентація_проекту*, доступ до якого вам надано. У коментарях додайте коментар про те, у якій групі ви б хотіли працювати над проектом. Урахуйте кількість учасників у кожній групі, яку вкаже вчитель чи вчителька.

1. У персональному освітньому середовищі оберіть сховище документів, доступ до яких вам надано



Відкриті для мене

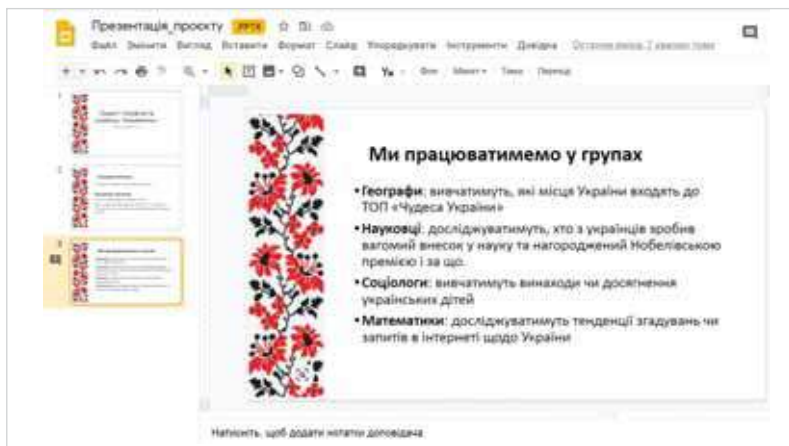
2. Оберіть документ *Презентація_проекту*. Відкрийте його в сервісі

Google Презентації



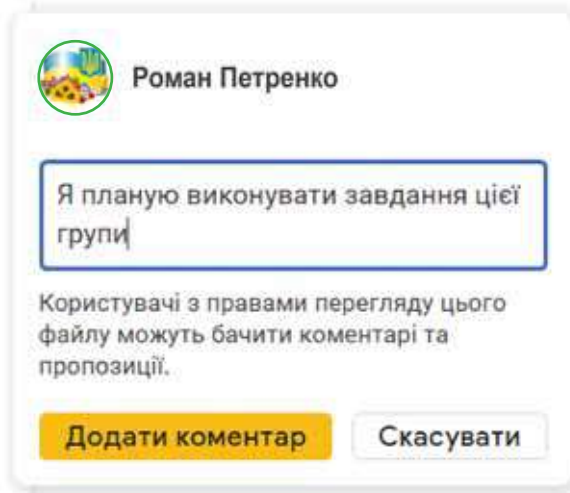
Відкрити за допомогою Googl...

3. Оберіть третій слайд презентації (мал. 4.10).



Мал. 4.10

4. Виділіть назву групи, у якій ви плануєте працювати над проектом. Додайте до виділеного слова коментар. Для цього скористайтесь вказівкою *Вставити/Коментар*. Наприклад, як на малюнку 4.11.



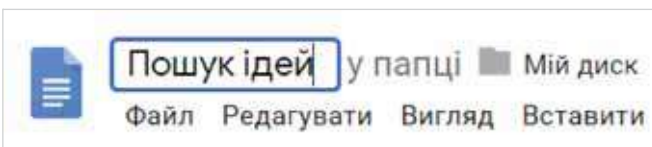
 **Мал. 4.11**

5. Додайте коментар. Слідкуйте за коментарями інших учнів.

Вправа 5. Створення документа на Google Диску.

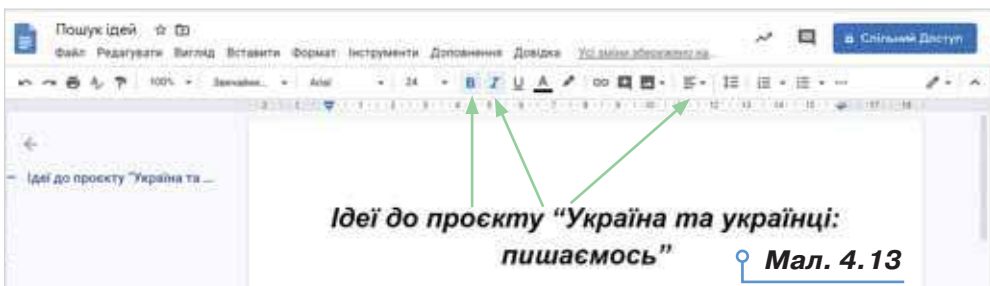
Завдання. Створіть у персональному освітньому середовищі текстовий документ для формування списку учасників проекту «Україна та українці: пишаємось». Надайте до нього доступ іншим учням класу за їхньою електронною адресою з правами доступу: *Може редагувати*.

1. У списку об'єктів, який відкривається після натиснення кнопки *Створити* вашого Google Диска, оберіть *Google Документи*.
2. Переіменуйте *Документ без назви* на *Пошук ідей* (мал. 4.12).



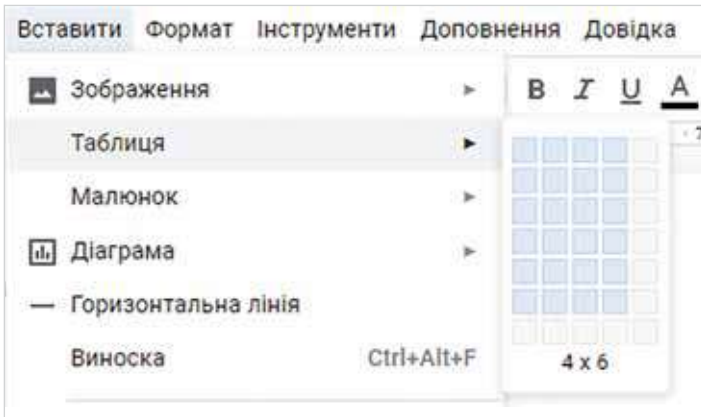
 **Мал. 4.12**

3. Запишіть заголовок документа. Використайте позначені на малюнку 4.13 інструменти онлайн-редактора для форматування тексту.



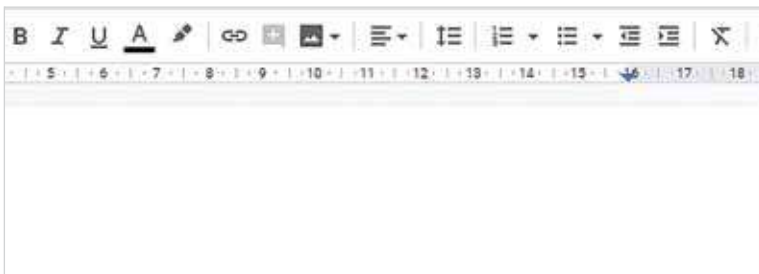
 **Мал. 4.13**

4. Додайте після заголовка таблицю подібно до того, як ви додавали таблицю до текстового документа в офісних текстових процесорах (мал. 4.14).



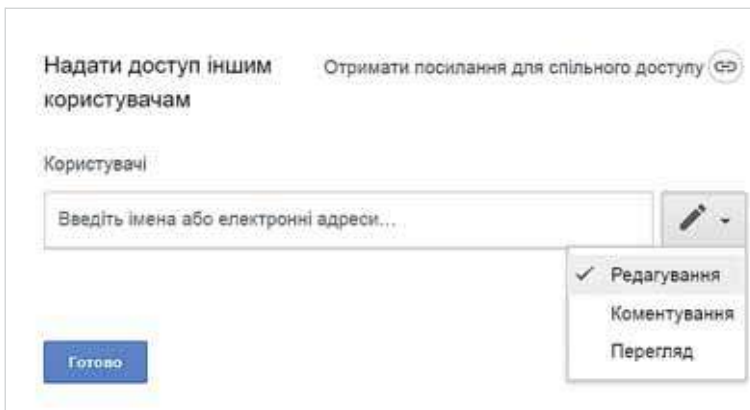
Мал. 4.14

Сформуйте заголовки таблиці: *№*, *Автор ідеї*, *Ідея*, *Посилання на матеріали*. Оформіть їх на власний розсуд, використавши повне меню інструментів (вказівка *Більше*) (мал. 4.15).



Мал. 4.15

5. Надайте спільний доступ до редагування іншим учням класу, які братимуть участь у вашій групі. Для цього натисніть *Спільний доступ* й у вікні, що відкриється, введіть електронні адреси учнів класу. Налаштуйте рівень доступу: *Редагування* (мал. 4.16). Завершіть надання доступу натисненням кнопки *Готово*.



Мал. 4.16

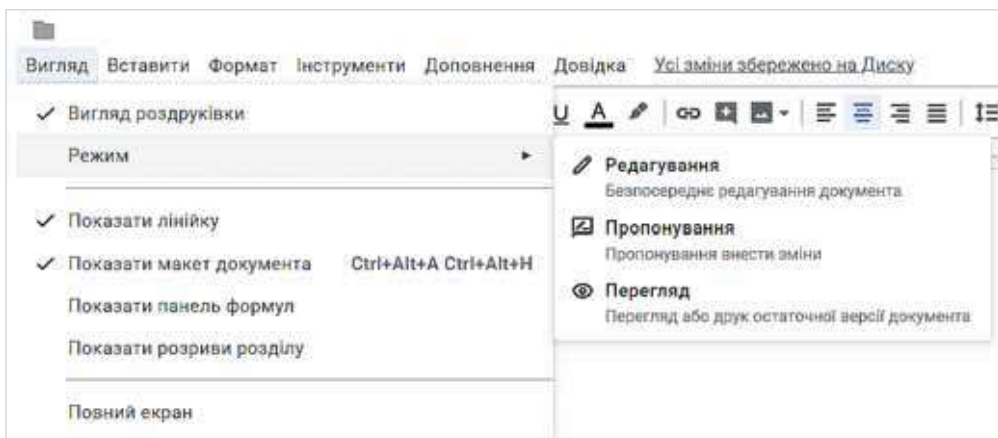
6. Зверніть увагу на те, що всі, хто відкривають одночасно з вами документ, отримують автоматично колір маркування тексту та можуть одночасно працювати з документом.
7. Дослідіть, чи можна надати доступ певним користувачам до редагування документа, не вказуючи їхньої електронної пошти.
8. Завершіть редагування документа.

Досліджуємо

Вправа 6. Додавання завдань.

Завдання. Дослідіть розширені можливості спільної роботи над документом у *Google Диску*.

1. У файлі *Пошук ідей*, створеному в персональному освітньому середовищі, додайте коментар до ідеї, яка вас найбільше зацікавила.
2. У тексті коментаря додайте значок «+».
3. У списку електронних адрес, що відкриється, оберіть адресу когось із учасників групи. Оберіть вказівку *Призначити*.
4. Дослідіть разом з адресатом, яке повідомлення прийшло йому чи їй на пошту, як це відображається у вікні середовища. Зробіть висновок.
5. У меню *Вигляд* оберіть режим *Пропонування* (мал. 4.17).



Мал. 4.17

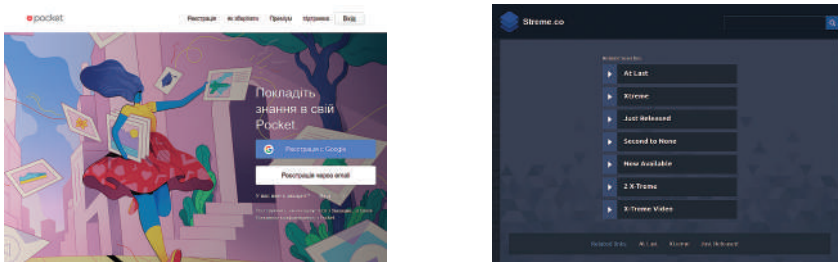
6. Після заголовка документа додайте підзаголовок з назвою групи. Перевірте, як будуть відобразитися зміни. Дослідіть, як прийняти або відхилити пропонувані зміни. Зробіть висновок.
7. Вийдіть зі свого акаунта. Завершіть роботу з *Google Диском*.

4. Як організувати доступ до інформаційних матеріалів у персональному освітньому середовищі?

Доступ до інформаційних матеріалів персонального освітнього середовища можна організувати декількома способами:

- використати онлайн-сервіс електронних закладок для швидкого доступу до сайтів з будь-якого пристрою, які використовує користувач;
- розмістити документи, які збережено на локальному диску, у хмару;
- синхронізувати *Google Диск* із локальним диском комп'ютера.


Існує декілька різноманітних онлайн-сервісів (мал. 4.18) для створення електронних закладок, що є аналогами звичайних закладок, які створюють засобами браузера. Прикладом є сервіс *Pocket* (<https://getpocket.com/>), у якому можна зберігати посилання на сайти й отримувати до них доступ через телефон, планшет і комп'ютер, навіть без підключення до інтернету.



Мал. 4.18

Іншим сервісом, який використовують для таких цілей, є *Streme* (<http://streme.co/>). За допомогою сервісу *Streme* можна створювати групи закладок, якими легко обмінюватися. Групи закладок можуть бути доступними тільки для читання, або їх можуть редагувати інші користувачі. Таким чином створюється спільнота навколо обраної теми. *Streme* автоматично визначає тип даних, на які створено закладку — відео, аудіо та інші види мультимедіа, щоб подати їх у найзручніший спосіб. А це означає, що деякі інформаційні матеріали можна переглянути без переходу на сайт-джерело.

Для того щоб отримувати повідомлення про зміни, які відбулися на сайті, чи появу нових дописів, наприклад, у вебжурналах або блогах, використовують так звані канали новин, що організують за допомогою технології RSS, яку ще називають фідом


(від англ. *feed* — живлення, постачання, канал)  . RSS (від англ. *Really Simple Syndication* (RSS 2.0), *Rich Site Summary* (RSS 0.91, RSS 1.0)) — це набір форматів, що використовується для публікації та надсилання відомостей, які часто змінюються.

Стрічки новин RSS дають змогу швидко ознайомитися зі змінами змісту сайту, безпосередньо не заходячи на нього за допомогою

браузера. Для того щоб мати доступ до власних документів з будь-яких пристроїв, можна синхронізувати дані, які розміщено на різних пристроях. Для цього встановлюють додаток *Google Диск* на комп'ютері. За замовчуванням усі дані в розділі *Мій диск* будуть синхронізовані з *Google Диском* на комп'ютері.

Синхронізація даних — це процес, який забезпечує встановлення відповідності даних на всіх пристроях, незалежно від того, на якому з них відбувалися зміни. Наприклад, файл, який змінено у хмарному офісному пакеті, матиме такий само зміст, як його відповідник на локальному *Google Диску*.

Щоб налаштувати синхронізацію, виконують такі дії:

1. Відкрити *Google Диск*. Обрати *Пуск /Всі програми/ Google Диск*.
2. Натиснути значок *Google Диск*  на панелі завдань у правому нижньому куті екрана.
3. Натиснути в правому верхньому куті вікна та обрати *Налаштування/Параметри синхронізації*.
4. Обрати дані для синхронізації: *Синхронізувати всі об'єкти* в розділі *Мій диск* чи *Синхронізувати тільки ці папки*. Якщо обрано другий варіант, то позначити папки для синхронізації.
5. Натиснути *Застосувати*.



Обговорюємо

1. Що означає персональне освітнє середовище для вас особисто?
2. Що таке хмарні технології та які переваги їхнього застосування?
3. Які хмарні технології ви використовуєте у своєму електронному освітньому середовищі? Наведіть приклади.
4. Які типи документів можна завантажувати до *Google Диска*, а які — створювати?
5. Як надавати спільний доступ до документів у своєму хмарному сховищі? Які рівні доступу можна надати?
6. Для чого використовують електронні закладки на ресурси інтернету? Чим вони відрізняються від закладок браузера?
7. Як дізнаватися про зміни на сайті, не заходячи на нього у браузері?
8. Що таке синхронізація даних і для яких цілей її застосовують?



Працюємо в парах

9. Наведіть три причини, чому сучасній молоді важливо створювати персональне освітнє середовище. Обговоріть у парі, які з наведених причин збігаються та як проранжувати спільні причини за ступенем важливості для планування свого майбутнього.

- ★ **10.** Обговоріть у парі, у чому подібність і чим відрізняється вікно текстового процесора та сервісу для створення текстових документів у *Google Диску*. Відповідь подайте в таблиці порівняння. Надішліть її на навчальну скриньку або дайте доступ до неї вчителю або вчительці.
- ★ **11.** Обговоріть у парі, у чому подібність і чим відрізняється вікно редактора презентацій та вікно сервісу для створення презентацій у *Google Диску*. Поміркуйте, як краще подати свою відповідь. Створіть відповідь у вибраному середовищі та надішліть її на навчальну скриньку чи дайте доступ до неї вчителю або вчительці.
- ★★ **12.** Обговоріть, у чому подібність і відмінність кнопки RSS та кнопка *Оповіщення й Підписатись* на каналі *Цікава наука* (<https://cutt.ly/erdvhud>). Як змінюється адреса каналу після натиснення кнопок *Оповіщення* та *Підписатись*?
- ★★ **13.** Розгляньте список сервісів закладок, які описані у файлі *Закладки папки Сервіси інтернету*. Зареєструйтесь в одному з пропонувананих ресурсів та дослідіть, чи всі можливості для створення електронних закладок описано у статті. Обговоріть, як би ви доповнили подані у файлі відомості.

підписок: 1 тис.



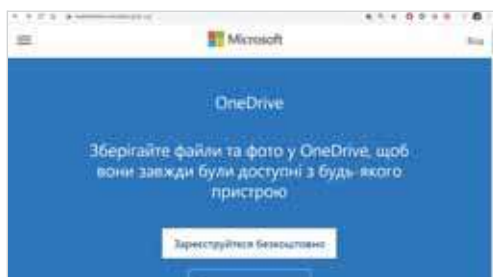
Працюємо самостійно

- ★ **14.** Створіть модель власного персонального освітнього середовища у вигляді схеми. Додайте в блоки схеми назви папок цього середовища. Збережіть схему в файлі *Я навчаюсь* у папці *Проекти* своєї структури папок. Поділіться файлом із вчителем або вчителькою та кимось із класу.
- ★ **15.** Створіть у *Google Диску* презентацію роботи у проекті «Україна та українці: пишаємось». Надайте доступ для коментування документа двом учням чи ученицям класу, щоб отримати від них дві похвали щодо змісту пропонованої презентації та однієї пропозиції щодо її удосконалення. Візьміть участь у такому коментуванні документів, які створено іншими учнями чи ученицями.
- ★ **16.** Створіть на *Google Диску* документ для спільного створення оголошення про роботу в проекті «Україна та українці: пишаємось». Долучіть до спільного редагування ще одного учня чи ученицю класу. Надайте доступ для коментування результатів своєї роботи вчителю або вчительці та учневі-експерту, якого визначить вчитель або вчителька.

17. Установіть додаток *Google Диск* на персональний комп'ютер. Синхронізуйте папки, які ви використовуєте в навчанні, з хмарним диском *Мій диск*. Оберіть для цього потрібні параметри налаштувань синхронізації.

18. Розгляньте пропозиції щодо організації персонального освітнього середовища на базі сервісу <https://onedrive.live.com/about/uk-ua/> (мал. 4.19). Визначте спільне та відмінне з розглядуваними сервісами. Результати своєї роботи подайте у презентації. Надішліть її на навчальну скриньку або надайте доступ вчителю або вчительці до файла, якщо ви його створювали в персональному освітньому середовищі.

Мал. 4.19

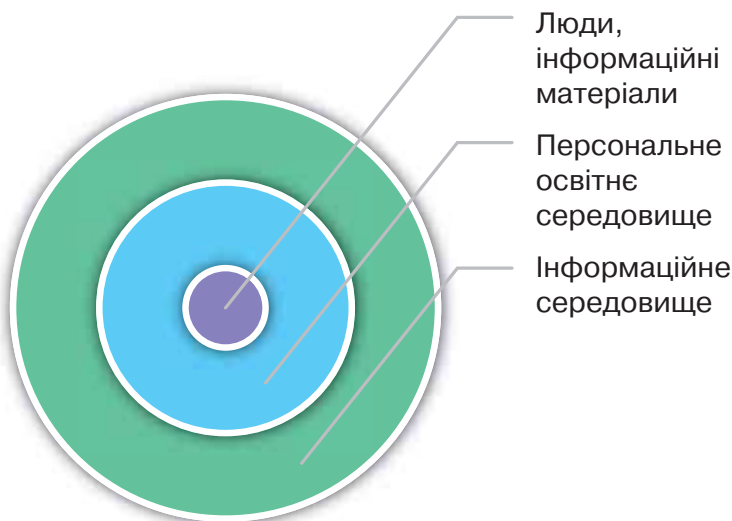


19. Знайдіть два підтвердження того, що сервіс <https://www.symbaloo.com/home> можна використати для організації персонального освітнього середовища.

Повторюємо



Розгляньте схему та складіть розповідь, як пов'язані її складові. Яке місце у схемі мають займати сервіси Веб 1.0 та Веб 2.0?





Навчання в інтернеті

ВИ ДІЗНАЄТЕСЬ:

- ▶ які ресурси інтернету можна використати для навчання;
- ▶ як інтернет допомагає вивчати іноземну мову.



Вивчаємо

1. Які ресурси інтернету можна використати для навчання?

Інтернет можна використовувати для навчання: здійснювати пошук навчальних матеріалів, зображень, фотографій, відео, навчальних програм тощо. Крім того, в інтернеті є безліч сайтів, які створено спеціально для навчання. Серед них — електронні словники, перекладачі, енциклопедії.

Перевірити правопис, знайти тлумачення або переклад слів можна за допомогою електронних словників. Збірки електронних словників можна знайти на сайтах *Мова Інфо* (www.mova.info), *Словopedia* (slovopedia.org.ua) та інших.

Ті, хто цікавиться українською мовою, можуть скористатись сайтом «Мова — ДНК нації» (<https://ukr-mova.in.ua/>), на якому є можливість виконати вправи з української мови, написати та перевірити диктант, перевірити текст на наявність орфографічних чи граматичних помилок.

Щоб опанувати основи наук, можна переглядати невеликі навчальні відео на каналі *Цікава наука* (<https://www.youtube.com/channel/UCMIVE71tHEUDkuw8tPxtzSQ>). Бути в курсі новинок технологій допоможе сервіс *МанЛаб* (<http://manlab.inhost.com.ua/>).

Крім матеріалів для поглиблення знань зі шкільних предметів, за допомогою інтернету можна дізнатися про новинки книговидавництва, техніки, культурні події у світі, в Україні та вашій місцевості; брати участь у різних електронних конкурсах, вікторинах й олімпіадах; користуватися електронними бібліотеками; отримати допомогу у вирішенні як навчальних, так і повсякденних проблем, наприклад, як купувати потрібні речі в інтернет-магазинах, як здійснювати електронні платежі, як налаштувати підключення тих або інших пристроїв, як обрати місце навчання чи відпочинку, та багато інших.



ДІЄМО

Вправа 1. Говоримо та читаємо українською.

Завдання. Семикласники створили список онлайн-ресурсів, які вони використовують для вивчення української мови та читання творів українською. Доповніть список коротким описом пропонованого ресурсу, використовуючи навички спільної роботи над документом.

1. Відкрийте документ *Інтернет для української мови*, доступ до якого вам надано в *Google Діску* вчителем або вчителькою.
2. Розподіліть обов'язки у класі так, щоб переглянути всі запропоновані ресурси, використати матеріали з довідки або самостійно додати короткий опис вмісту чи призначення навчального ресурсу.
3. Доповніть список за зразком, поданим у таблиці 5.1.

Таблиця 5.1

№	Назва ресурсу	Адреса ресурсу	Призначення	Оцінка (від 1 до 5)
1	Мова — ДНК нації	https://ukr-mova.in.ua/	Сайт містить бібліотеку навчальних ілюстрацій з мови, вправи, диктанти, сервіс для перевірки тексту. Не потребує реєстрації	

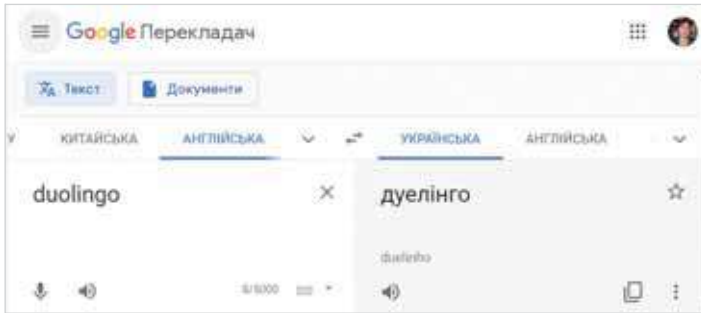
4. Поставте оцінку ресурсу за критеріями: легкість у користуванні, відсутність реклами, обсяг інформації, різноманітність ресурсів, захопливий дизайн.

2. Як інтернет допомагає вивчати іноземну мову?

У процесі роботи, навчання чи отримання потрібних відомостей з інтернету дедалі частіше виникає потреба опрацювання текстів іноземною мовою. Сучасний ринок програмного забезпечення пропонує для автоматизації перекладу різні програми-словники та перекладачі. Якщо потреба в перекладі виникає доволі рідко, то купувати й встановлювати на свій комп'ютер спеціальне ліцензійне програмне забезпечення недоцільно. У таких випадках можна використовувати безкоштовні онлайн-словники та служби перекладу текстів.

Онлайн-перекладач — це програма для автоматичного перекладу тексту з однієї мови на іншу.

Онлайнними перекладачами є <http://translate.meta.ua/>, <https://pereklad.online.ua/ukr/>, <http://ua.freeonlinetranslators.net/>. За кількістю підтримуваних мов (51) і напрямів — найбільш універсальним перекладачем є *Google Перекладач* (<https://translate.google.com/?hl=uk>) (мал. 5.1).



Мал. 5.1

При перекладі окремих слів програму можна використовувати як онлайнний словник: *Google Translate* виводить опис усіх значень слова, яке користувач хоче перекласти. Крім того, є опція «показати транслітерацію», що особливо важливо в ситуаціях, коли є потреба в перекладі важких для вимови слів або необхідно правильно написати адресу на конверті при відправці листа за кордон.


Опанувати іноземну мову, наприклад англійську, можна за курсами, які пропонує сайт *Британської ради* (<http://www.britishcouncil.org.ua/english/mooc>). Для тих, хто хоче в ігровій формі вивчати іноземну мову та відслідковувати свій прогрес, цікавим буде сайт *Дуелінго* (<https://uk.duolingo.com/>).

Окрім перекладачів і спеціалізованих навчальних сайтів можна поповнювати свій словниковий запас іноземною за допомогою онлайн-нових словників. Такі словники, як правило, містять спеціальні засоби пошуку потрібного слова. Наприклад, словник *Visual Dictionary Online* за посиланням <https://cutt.ly/atcK3qH> допомагає за ілюстрацією знайти потрібне слово з обраної категорії.

Багато музеїв світу пропонують віртуальні екскурсії, які можна здійснювати, використовуючи інтернет. Наприклад, на порталі Музею історії природи можна здійснити віртуальний тур (https://naturalhistory.si.edu/VT3/NMNH/z_NMNH-016.html). А на сайті Музею Галілео (<https://www.museogalileo.it/>) — отримати доступ до віртуального музею та електронної бібліотеки. Щоб швидко зрозуміти вміст сайту іноземною, можна скористатись автоматичним перекладом вебсторінки у браузері.

Вправа 2. Налаштування параметрів перекладу вебсторінок у браузері.

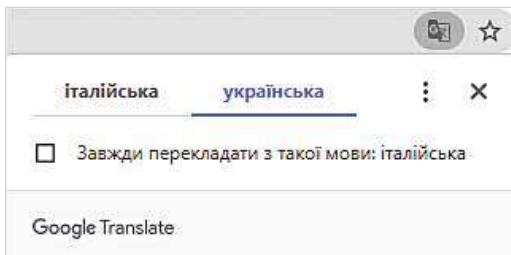
Завдання. Налаштуйте автоматичний переклад усіх вебсторінок увімкненням панелі автоматичного перекладу в налаштуваннях браузера.

1. Оберіть *меню Chrome*  у вікні браузера *Google Chrome*.
2. Виконайте вказівку *Налаштування*.
3. Оберіть *Показати розширені налаштування*.
4. Увімкніть режим *Пропонувати перекладати сторінки, мова яких відмінна від тієї, якою я читаю* в розділі *Мови*.

Вправа 3. Переклад вебсторінки.

Завдання. Перекладіть вебсторінку на іншу мову. Налаштуйте мовну пару для перекладу сторінок рідною мовою.

1. Введіть в адресний рядок назву сайту Музею Галілео <https://www.museogalileo.it/it/>.
2. У повідомленні під адресним рядком оберіть *Перекласти* (мал. 5.2).



 **Мал. 5.2**

3. Переконайтеся, що сторінку перекладено на вашу рідну мову. На вкладці *Параметри* оберіть вказівку *Змінити мови*. Перевірте, чи вказано переклад з італійської мови на українську.

Вправа 4. Історія.

Завдання. Установіть відповідність тексту англomовної Вікіпедії про трипільську культуру (http://en.wikipedia.org/wiki/Cucuteni-Trypillian_culture) і відомостей з української Вікіпедії.



1. В адресному рядку браузера введіть адресу сторінки <https://cutt.ly/ftcK6OJ>.
2. Знайдіть загальні відомості про трипільську культуру. Виділіть перші два абзаци тексту та скопіюйте його в буфер обміну.
3. У браузері відкрийте нову вкладку. Відкрийте *Перекладач Google*. Перекладіть вміст буфера обміну на українську мову.
4. У браузері відкрийте нову вкладку. Оберіть зручний засіб простого пошуку у браузері та введіть ключові слова пошуку «трипільська культура». У списку отриманих посилань знайдіть посилання на статтю з української Вікіпедії.
5. Прочитайте початок статті про трипільську культуру українською мовою. Зробіть висновок про відповідність відомостей у різних мовних версіях Вікіпедії.



Досліджуємо

Вправа 5. Текст.

Завдання. Складіть алгоритм перекладу за допомогою *Перекладача Google* текстового документа. Використайте файл *Дар_волхвів* з папки *Сервіси інтернету*.



Обговорюємо

1. Як школярі можуть використати інтернет для навчання? Як це може вплинути на їхній успіх у майбутньому? Назвіть три прогнози.
2. Чи використовують інтернет в освітніх цілях дорослі? Назвіть дві причини.
3. Яким чином можна використовувати для навчання сайти, що створені мовами, якими ви не володієте? Як розробники деяких сайтів цьому сприяють?
4. Чим електронні словники відрізняються від онлайн-перекладачів? Наведіть приклади.



Працюємо в парах

5. Обговоріть, який спосіб онлайн-перекладу варто застосувати в кожному випадку:
 - 1) у буфері обміну міститься фрагмент вебсторінки іноземною мовою;
 - 2) за введеною в рядку адреси браузера адресою вебсторінки завантажено головну сторінку вебсайта, відомості на якій подано іноземною мовою;
 - 3) при прослуховуванні звукозапису для підготовки до уроку англійської мови вами було почуто незнайоме слово.
6. Обговоріть, які курси з платформи *Курсера* (<https://www.coursera.org/browse/computer-science>) можна використати для поглиблення власного навчання інформатики. Визначте, які з обраних курсів можуть бути вам доступними та чому.
- ★ 7. Обговоріть, чи корисним для вас був би курс *Фінансова грамотність громадянина* на порталі *Вільного університету Майдану* (<https://vumonline.ua/course/abc-of-finance/>). Визначте спільні аргументи.
- ★ 8. Запропонуйте власну ідею інтернет-енциклопедії для підтримки навчання одного з предметів сьомого класу. Передбачте, які будуть розділи (вебсторінки), як буде здійснюватись навігація між

ними, хто наповнюватиме енциклопедію матеріалами та якими. Обговоріть свою ідею з іншою парою учнів.

Працюємо самостійно

9. Поставте у відповідність опис ресурсу для навчання математики та його адресу:

Онлайнова школа з математики

<https://learning.ua/matematyka/somyi-klas/vyznachaiemo-chyslo>

Онлайнний підручник з математики

<https://cubens.com/uk>

Тренажер з основних тем математики

<http://ua.onlinemschool.com>

Навчальний сайт із математики

<https://formula.co.ua/uk>

10. Складіть рекламу для учнів щодо використання матеріалів сайта:
- 1) *Добра фізика* (<http://gutpfusik.blogspot.com/>) для вивчення шкільної фізики;
 - 2) *Целестія* (<https://celestia.space/>) для дослідження Всесвіту;
 - 3) *Уся біологія* (<http://www.sbio.info/>) для вивчення біології;
 - 4) відеотаблиця Менделєєва (<http://ed.ted.com/periodic-videos>) для вивчення хімії;
 - 5) канал *Історія Українських земель* для вивчення історії;
 - 6) список ресурсів від Британської ради (<https://cutt.ly/atvjjiZ>) для вивчення англійської мови.

Повторюємо

Розгляньте схему та складіть розповідь про те, як пов'язані її складові.





Інтернет речей

ВИ ДІЗНАЄТЕСЬ:

- ▶ що розуміють під терміном «інтернет речей»;
- ▶ як речі взаємодіють між собою;
- ▶ де застосовують концепцію інтернету речей.



Вивчаємо

1. Що розуміють під терміном «інтернет речей»?

Чи маєте ви пристрій, що відправляє дані на ваш смартфон? Як, перебуваючи в дорозі, за допомогою лише комп'ютерного пристрою і мобільного інтернету, ваші рідні можуть зберігати свою оселю в безпеці й контролювати витрати електроенергії? Це може забезпечувати інтернет речей — технологія, що набирає популярності у світі. Квартири, будинки та цілі міста дедалі активніше використовують цю концепцію у своєму повсякденному житті. Їх часто називають смарт (від англ. *smart* — «розумні»).



Мал. 6.1

Термін «інтернет речей» (від англ. *internet of things*) уперше з'явився ще в 1999 році. Він являє собою концепцію комунікаційної мережі об'єктів («речей»), у які вбудовано технологію для взаємодії між собою та з навколишнім середовищем (мал. 6.1). Крім того, ця

технологія передбачає можливість виконувати певні дії без втручання людини. Тобто пристрої, що розміщені в будинках, убудовані в автомобілі, виконують опрацювання інформації, її аналіз та обмін між собою та, залежно від результатів, приймають рішення і виконують певні передбачені дії. Середовище роботи таких пристроїв не обмежується лише «розумним» будинком, воно може охоплювати все місто.

Автором цієї концепції є британський учений Кевін Ештон. За словами Ештона, до цього часу інтернет слугував переважно для пошуку інформації і був повністю залежним від людей: тих, які подають інформацію, натискають на кнопки, завантажують відео та фото на сервер, сканують коди тощо. У майбутньому, за уявленням Ештона, речі самі вноситимуть дані про себе.

Інтернет речей (Internet of things, IoT) — концепція, яка дає змогу об'єктам (фізичним речам), підключеним до інтернету, взаємодіяти одне з одним, або із зовнішнім світом, здійснювати опрацювання інформації та її аналіз, обмінюватись інформацією між собою та, залежно від результатів, приймати рішення частково або повністю без людини й виконувати певні дії.

У найзагальнішому вигляді з інформаційно-комунікаційної точки зору інтернет речей можна записати у вигляді такої символічної формули:

IoT = Сенсори (датчики) + Дані + Мережі + Послуги.

Найважливішими відмінностями інтернету речей від існуючого інтернету людей є:

- фокус на речах, а не на людині;
- істотно більша кількість підключених об'єктів;
- істотно менші розміри об'єктів і невисокі швидкості передавання даних;
- фокус на зчитуванні інформації, а не на комунікаціях;
- необхідність створення нової інфраструктури й альтернативних стандартів.

2. Як речі взаємодіють між собою?

Для реалізації інтернету речей потрібна екосистема: гаджети з датчиками; мережа доступу та передачі інформації; платформи для управління мережею, пристроями й додатками. Вона передбачає чотири рівні:

1. Сенсори (датчики), які вбудовані в «розумні об'єкти». Існують різні типи датчиків для відповідних цілей, наприклад, для вимірювання температури, тиску, швидкості руху, місця розташування та ін. Датчики можуть мати невелику пам'ять, даючи можливість записувати кілька результатів вимірювань. Датчик може вимірювати

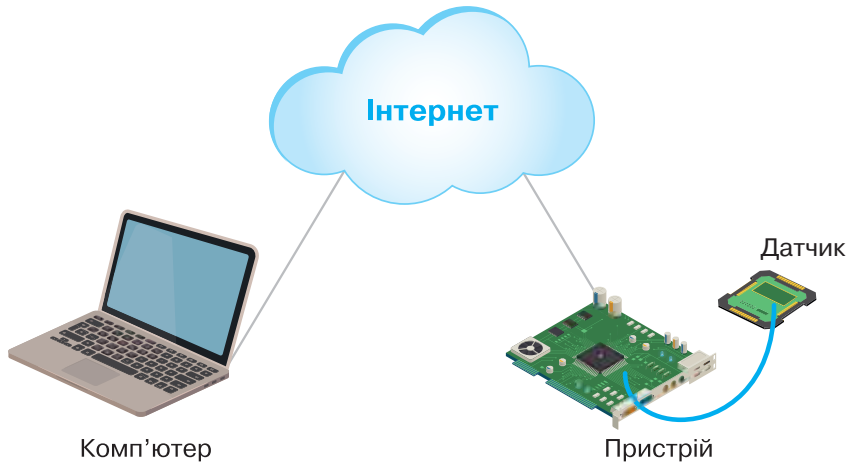
фізичні параметри контрольованого об'єкта або явища й перетворювати їх у сигнал, який може бути прийнятий відповідним **проміжним мережевим пристроєм (комп'ютером)**.

2. Мережа доступу й передавання інформаційних даних. Може бути реалізована на базі високопродуктивної дротової або бездротової мережевої інфраструктури як транспортне середовище (локальної мережі, Wi-Fi або персональної мережі).

3. Платформа, яка здійснює отримання, зберігання, аналіз, опрацювання великої кількості даних, що надходять у режимі реального часу, та забезпечує управління мережею, пристроями та додатками.

4. Кінцевий пристрій, який діє за допомогою спеціальних додатків чи програм відповідно до алгоритму, що передається.

Наприклад, для вирішення певної задачі комп'ютер зв'язується через публічний інтернет з невеликим пристроєм, до якого підключено відповідний датчик (наприклад, температури), як це показано на малюнку 6.2.



Мал. 6.2

Основна відмінність мереж наступного покоління від традиційних мереж — у тому, що всю інформацію, яка циркулює в мережі, розбито на дві складові. Це сигнальна інформація, що забезпечує комутацію абонентів та надання послуг, і безпосередньо дані користувача, які містять корисну інформацію, призначену абоненту (голос, відео, дані).

Розглянемо принцип функціонування інтернету речей на прикладі «розумної теплиці» (мал. 6.3).

У речі можна вмонтовувати сенсори. Вони здатні навіть без власного джерела струму комунікувати з приладами зчитування. Кожна річ має індивідуальний номер. У примітивному варіанті таку методику застосовують, скажімо, у сфері логістики: посилки

самостійно розпізнаються зчитувальними приладами й пересилаються у правильному напрямку.



Мал. 6.3

Наприклад, за допомогою вмонтованих сенсорів «розумний» будинок має інформацію не лише про рівень тепла в різних приміщеннях. Він «знає» також, у яких кімнатах перебувають люди, як вони пересядають, що роблять. Якщо пристрої розумного будинку з'єднані зі смартфоном власника, то вони можуть також «вираховувати» місцезнаходження своїх мешканців навіть за його межами. Наприклад, якщо людина налаштовує будильник на смартфоні на певну годину, «розумний» будинок уже за пів години до цього увімкне опалення, причому отримає дані від метеостанції та підлаштує мікроклімат відповідно до змін температури на вулиці й переваг щодо мікроклімату в будинку господарів.

3. Де застосовують концепцію інтернету речей?

Експерти стверджують, що інтернет речей є однією з найперспективніших технологій, яка вже сьогодні фактично створює сотні нових продуктів і використовується в побуті, облаштуванні дому, підтримці

систем безпеки, у медицині, транспорті, торгівлі, промисловості, сільському господарстві тощо.

На основі інтернету речей можуть бути реалізовані всілякі «розумні» додатки в різних сферах діяльності й життя людини: «розумна планета», «розумне місто», «розумний будинок», «розумна енергетика», «розумний транспорт».

Проект «Розумне місто» передбачає, що міська інфраструктура й супутні муніципальні послуги, такі як освіта, охорона здоров'я, громадська безпека, житлово-комунальне господарство, стануть більше пов'язаними й ефективними. Датчики на громадському транспорті, контроль за рівнем води у водоймах, датчики шуму та забруднення, сміттєві баки з датчиками заповнюваності. Це зробить міста зручнішими та безпечнішими. А збір великих даних на основі датчиків допоможе місцевій владі краще розуміти потреби міста й жителів. З появою розумних міст автомобілі теж під'єднаються до інтернету. Вони самостійно оцінюватимуть ситуацію на дорозі. Або писатимуть повідомлення, якщо палива не вистачає на дорогу до роботи, розряджається акумулятор чи підскочить тиск у шинах тощо.

Проект «Розумний будинок» передбачає, що система розпізнаватиме конкретні ситуації, які відбуваються в будинку, і реагуватиме на них відповідним чином, що забезпечить мешканцям безпеку, комфорт і ресурсозбереження. За проектом «Розумна медицина» лікарі й пацієнти зможуть отримати віддалений доступ до дорогого медичного обладнання або до електронної історії хвороби в будь-якому місці, буде реалізовано систему віддаленого моніторингу здоров'я, автоматизовано видачу лікарських препаратів хворим і багато іншого.

Магазини без касирів, віртуальна й доповнена реальність, камери відеоспостереження з розпізнаванням облич у метро — це теж інтернет речей. Інтернет речей змінює процес покупок. Продавець відстежує маршрут покупця та збирає дані про моделі поведінки. Це допомагає визначити, чи потрібно надавати знижку, і якщо так, то на скільки відсотків. Усі дані від тисяч покупців допоможуть продавцеві вдосконалити свій магазин. Наприклад, змінити розташування товарів на полицях, що приведе до зростання продажів.

Сьогодні вже багато хто з дорослих і дітей користуються пристроями, до яких вбудована технологія інтернету речей. Найбільш поширеними є фітнес-браслети, смарт-годинники, «розумні окуляри» тощо (мал. 6.4). Для багатьох ці пристрої стали незамінними в ранкових пробіжках або заняттях на тренажерах. Вони можуть слідкувати за фазами сну користувача та активністю протягом дня, відстежувати ритм серцебиття, раціон. Усі дані синхронізуються із власним додатком на смартфоні, який дає рекомендації щодо здоров'я, а також може надсилати повідомлення в персональну базу здоров'я та регулювати потребу в медикаментах, обмежувати рухову активність тощо.



Мал. 6.4

Для тих, хто має домашніх улюбленців, інтернет речей також стане у пригоді. Пристрої автоматичної годівниці з IoT забезпечать домашньому улюбленцю безперебійний доступ до їжі, навіть якщо господарі вирушили у тривалу відпустку. На відміну від звичайних автоматичних годівниць, цей гаджет обладнано системою розпізнавання «облич» тварин, що знадобиться за наявності в родині декількох пухнастих і дасть змогу господарям контролювати режим харчування кожного улюбленця.



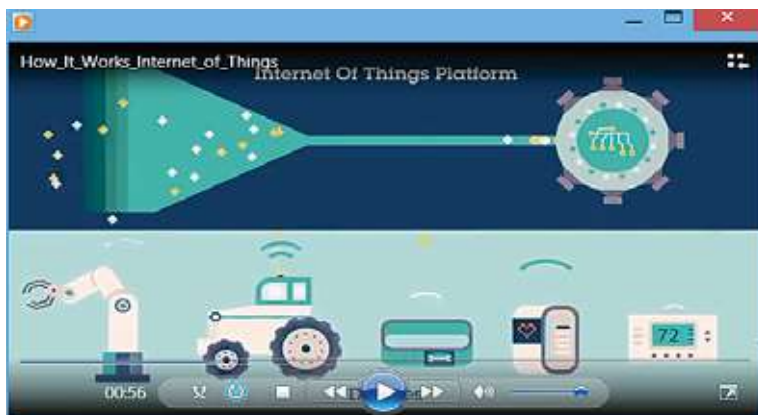
Діємо



Вправа 1. Як працює інтернет речей.

Завдання. Розгляньте відео «Як працює інтернет речей» за посиланням <https://cutt.ly/drlMYnN>. Визначте, який приклад використано у відео для пояснення концепції інтернету речей.

1. У браузері відкрийте посилання за вказаною адресою.
2. Завантажте відео на комп'ютер. Керуйте відтворенням відео у програмі перегляду відео (мал. 6.5).



Мал. 6.5

3. Визначте, який приклад використано у відео для пояснення концепції інтернету речей. З якого часу перегляду відео розпочинається демонстрація прикладу?
4. Поясніть складові інтернету речей, використані у відеопоясненні.

Вправа 2. Як працює інтернет речей у виробництві.



Завдання. Розгляньте відео «Як це працює: інтернет речей у виробництві» за посиланням <https://cutt.ly/BrIMU1P>. Визначте, який вплив має інтернет речей на виробництво.

1. У браузері відкрийте посилання за вказаною адресою.
2. Скористайтесь онлайнним перекладачем для отримання інформації, про що йтиметься у відео (мал. 6.6).



Мал. 6.6

3. Увімкніть перегляд відео.
4. Визначте, який вплив має інтернет речей на виробництво.

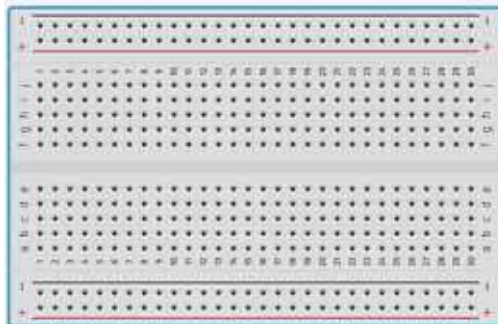


Досліджуємо

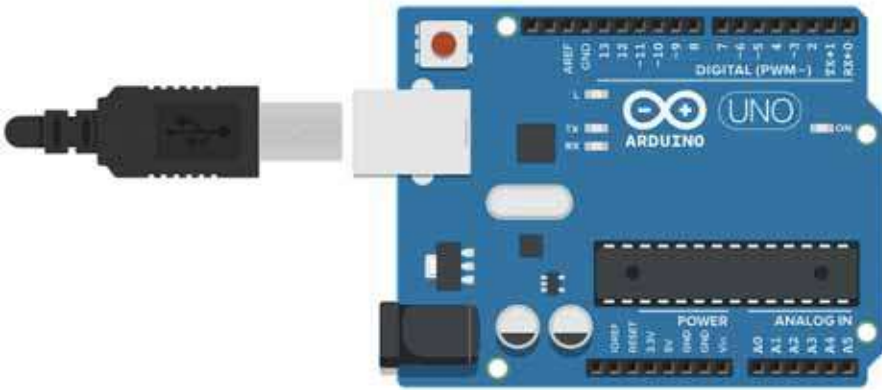
Вправа 3. Власний проєкт.

Завдання. Створіть у середовищі <https://www.tinkercad.com/> проєкт, у якому світлодіод блиматиме з деякою частотою. Дослідіть, чи можна вважати цей проєкт прототипом інтернету речей.


1. Увійдіть у клас на сервісі <https://www.tinkercad.com/>, код якого та доступ надасть вчитель або вчителька. За наданим псевдонімом увійдіть у середовище.
2. У меню сервісів оберіть **Circuits**.
3. Створіть новий проєкт. У поле для створення проєкту з бібліотеки складових перетягніть складові проєкту:
 - а) макетна плата, на яку кріпляться елементи схеми:



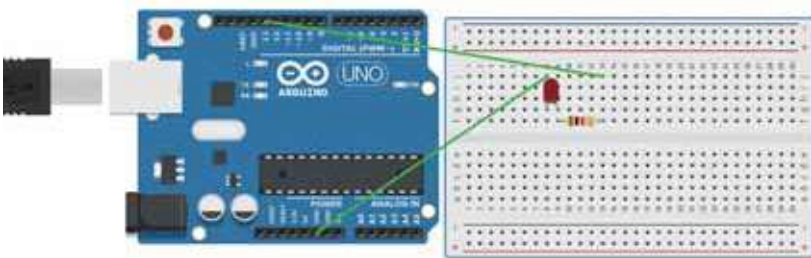
б) мікроконтролер *Ардуїно*, за допомогою якого відбувається виконання програми у проєкті:



в)  — світлодіод, що є виконавцем програмного коду;


г)  — резистор, який забезпечує можливість безпечного приєднання світлодіода до джерела живлення.

4. Складіть схему, як показано на малюнку 6.7. Урахуйте, що лінії з'єднання утворюються одразу після натиснення місця з'єднання на платі й точки входу/виходу на мікроконтролері. Коротка ніжка світлодіода приєднується до GND (земля), а довга через резистор — до цифрового виходу 13, на який надсилається сигнал з мікропроцесора.




Мал. 6.7

5. Відкрийте код, який автоматично згенерувався системою. Для цього

натисніть  Код . Навіщо подібний редактор коду?

6. Запустіть програму на виконання. Для цього натисніть *Почати*

моделювання . Перевірте, чи буде вмикатись і вимикатись ваш маячок.

7. Яких елементів бракує у проєкті, щоб його можна було віднести до інтернету речей? Розгляньте базовий набір елементів. Чи є серед них датчики? Які? Які проєкти можна було б з ними реалізувати? Що потрібно, щоб об'єкти на макетній платі обмінювались даними?
8. Обговоріть і зробіть висновок за умовою задачі.



Обговорюємо

1. Чи звичайний інтернет відрізняється від інтернету речей?
2. Які пристрої використовують в IoT?
3. Якими є складові інтернету речей?
4. Як можуть взаємодіяти речі між собою в мережі?
5. Чи можуть втручатись люди у взаємодію речей через мережу?
6. Як концепція інтернету речей використовується в житті людей?
7. Якими є сфери застосування інтернету речей?
8. Що з інтернету речей застосовують ваші рідні вже зараз?



Працюємо в парах

- ★ 9. Розгляньте відео про життя героя в розумному будинку за посиланням: <https://cutt.ly/nrIMDJ7>. Обговоріть і придумайте, як озвучити історію, що продемонстрована у відео.
- ★ 10. Розгляньте відео про роботу героя в розумному офісі за посиланням: <https://cutt.ly/lrIMHsn>. Обговоріть і придумайте, як озвучити історію, яку продемонстровано у відео.
- ★ 11. Розгляньте відео про особливості покупок у «розумному магазині» за посиланням: <https://cutt.ly/zrIMLyO>. Обговоріть і придумайте, як озвучити історію, яку продемонстровано у відео.
- ★ 12. Розгляньте відео про особливості використання інтернету речей у сільському господарстві за посиланням: <https://cutt.ly/TrIMXo0>. Обговоріть і придумайте, як озвучити історію, яку продемонстровано у відео.
- ★ 13. Розгляньте відео про роботу з використанням інтернету речей за посиланням: <https://cutt.ly/ErIMVPV>. Обговоріть і придумайте, як озвучити історію, яку продемонстровано у відео.

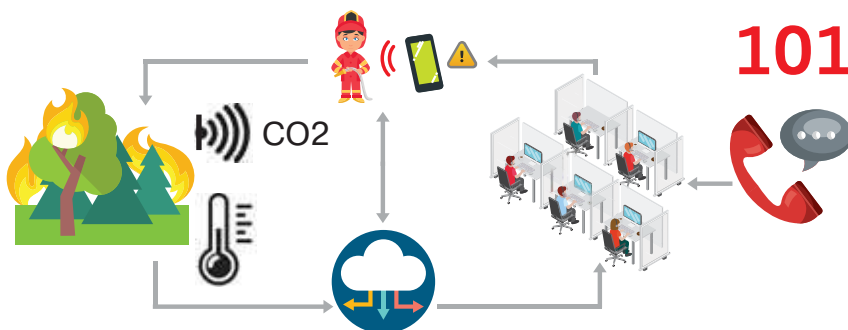


- ★ 14. Розгляньте відео про організацію виробництва та концепції інтернету речей за посиланням: <https://cutt.ly/2r1MNK3>. Обговоріть і придумайте, як озвучити історію, яку продемонстровано у відео.
- ★ 15. Поставте одне одному запитання за матеріалами відео за посиланням <https://cutt.ly/kr1MOKG>.
- ★ 16. Обговоріть і сплануйте зі старшими проект, який можна створити на базі мікроконтролера Ардуїно за концепцією інтернету речей. Залучіть до своєї роботи старшокласників чи батьків. Протестуйте його в середовищі <https://www.tinkercad.com>. Використайте реальні складові для створення свого проекту.



Працюємо самостійно

17. Складіть розповідь, як інтернет речей допомагає лісівникам і рятувальникам у боротьбі з лісовими пожежами (мал. 6.8)



Мал. 6.8

- ★ 18. Складіть субтитри до відео про використання інтернету речей в агробізнесі за посиланнями: <https://cutt.ly/8r1M6rH>, <https://cutt.ly/url1uFW>.
- ★ 19. Складіть розповідь про шкільне життя з інтернетом речей за відео: <https://cutt.ly/Brl1guJ>.
- ★ 20. Складіть презентацію про прогнози використання інтернету речей. Скористайтесь матеріалами сайту за посиланням: <https://cutt.ly/Nu4KA95>.



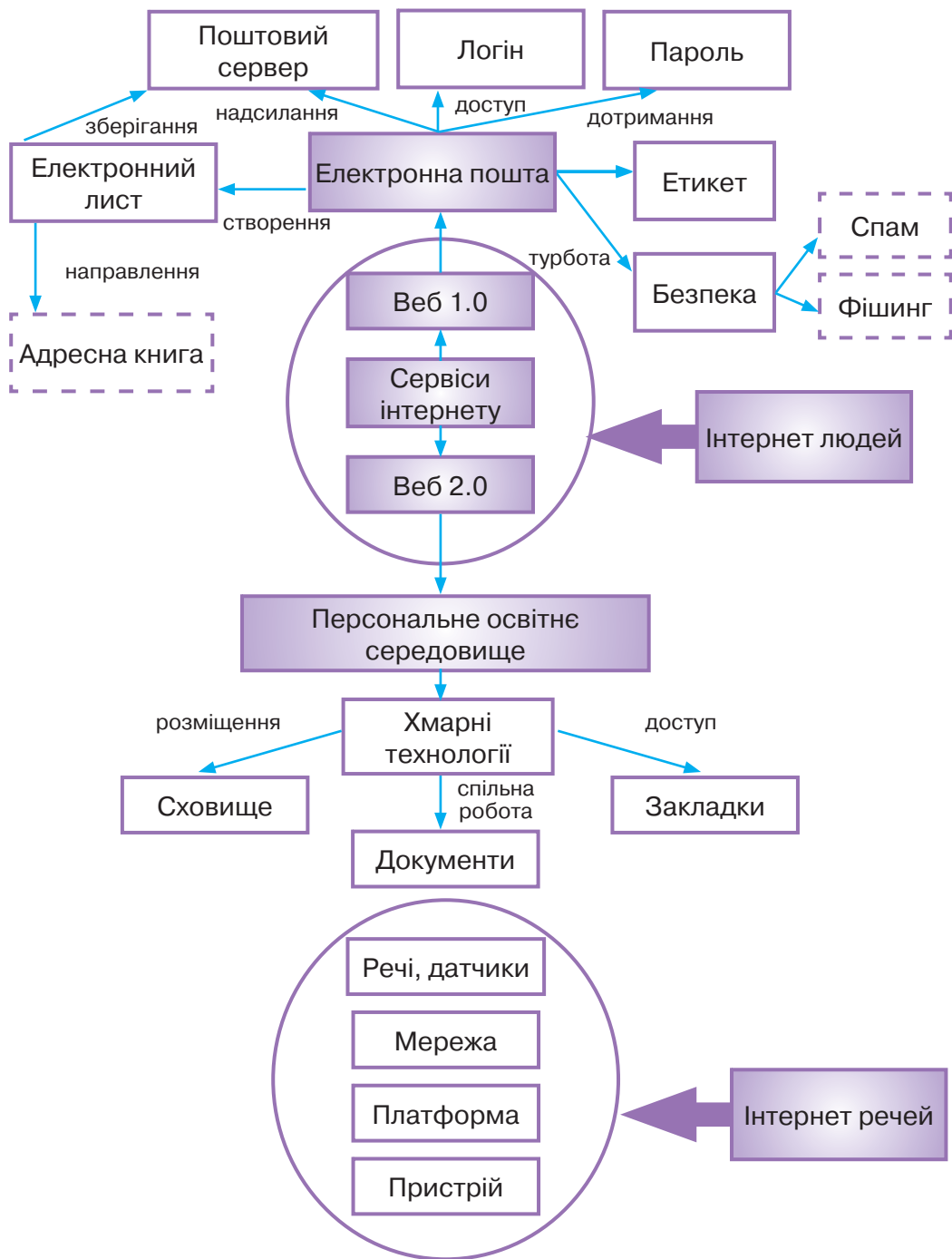
Повторюємо



Побудуйте схему, яка пояснює застосування інтернету речей, і складіть за нею розповідь.



Узагальнюємо



Розділ 2

Алгоритми та програми

7



Мова програмування.
Середовище програмування

ВИ ДІЗНАЄТЕСЬ:

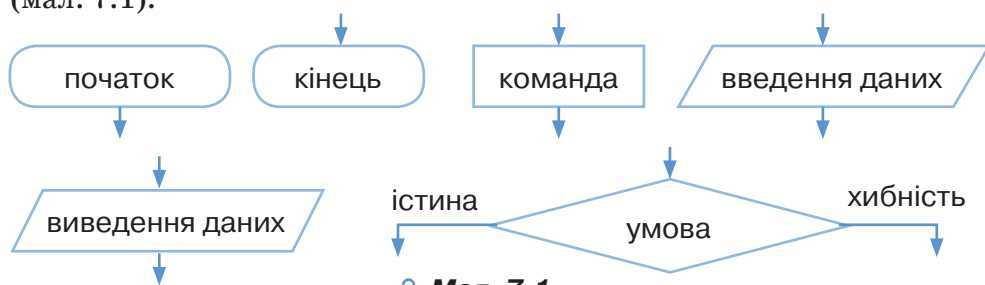
- ▶ що таке мова програмування;
- ▶ яке середовище використовують для створення програм;
- ▶ чи можна використати середовище програмування як звичайний калькулятор;
- ▶ як у програмі мовою *Python* використовують команду виведення.



Вивчаємо

1. Що таке мова програмування?

Ви вже знаєте, що алгоритми для виконавців описують різними способами та використовують різні форми їх подання. Якщо алгоритм створюють для виконавця людини, то, як правило, його подають словесно або графічно. Графічний спосіб подання алгоритмів передбачає також використання спеціальних графічних позначок — **блок-схем** (мал. 7.1).



Мал. 7.1


Ви вже використовували середовища для складання та виконання алгоритмів, наприклад *Скретч*, у яких алгоритм будували за допомогою інших блоків-команд. Для опису алгоритмів, що виконуються за допомогою комп'ютера, використовують спеціальну мову — **мову програмування**.

Алгоритм, записаний мовою програмування, називають **програмою**.

Мова програмування — це система позначень для точного опису алгоритму, який потрібно виконати за допомогою комп'ютера.

Мова програмування, як і будь-яка інша мова, має такі складові:

- набір символів, з яких утворюють слова та речення цієї мови, — **алфавіт**;
- сукупність спеціальних слів, які мають однозначне пояснення та застосування, — **словник**;
- систему правил складання базових конструкцій мови — **синтаксис**;
- правила **семантики**, що пояснюють, яке смислове значення має опис кожної з вказівок програми та які дії повинен виконати комп'ютер під час виконання кожної з таких вказівок.

Наприклад, щоб отримати повідомлення , яке ви створювали в навчальному середовищі *Скретч* за допомогою команди **говорити** **Привіт!**, мовою програмування *Python* потрібно ввести команду:

```
print ('Привіт!')
```

До алфавіту цієї мови належать латинські літери та спеціальні символи для запису команд, цифри — для числових даних, українські літери — для коментарів і текстових даних. Слово `print` належить до словника мови й задає команду *Друкувати*. Щоб надрукувати потрібний текст, його записують у дужках між символами «'» — такі правила синтаксису.

Щоб команди програми могли виконуватися за допомогою комп'ютера, слід дотримуватись усіх правил обраної мови програмування. Якщо при складанні програми використати символи, які не входять до алфавіту обраної мови програмування, неправильно написати спеціальні слова чи скласти із цих слів структуру алгоритму, не дотримуючись прийнятих правил, це буде сприйнято як помилку, і програма не виконається.



Діємо

Вправа 1. Ознайомлення з мовами програмування.

Завдання. Реалізуйте в навчальному середовищі створення програм *Блоклі* (<https://blockly-demo.appspot.com/static/demos/index.html>) алгоритм, за яким у вікні повідомлення з'явиться текст *Я вивчаю мови програмування*.



1. Відкрийте вікно браузера. Введіть у поле адреси `https://blockly-demo.appspot.com/static/demos/index.html`.
2. На головній сторінці *Блоклі* оберіть середовище *Code Editor* (мал. 7.2).

```

; 1 <= 100
3 == 0) (
.alert ('F1
f (1 % 5 =

```

[Code Editor](#)


[Мал. 7.2](#)

3. У списку обрання мови середовища , розташованому у верхньому правому куті вебсторінки, оберіть українську мову.
4. Складіть програму з блоків групи *Текст* (мал. 7.3), так само, як ви робили це в середовищі *Скретч*.

друк

“ Я вивчаю мови програмування ”

[Мал. 7.3](#)

5. Запустіть програму на виконання, натиснувши в правому верхньому куті середовища кнопку *Запуск*  . Перевірте, чи відображено у вікні повідомлення текст *Я вивчаю мови програмування*.
6. Перейдіть на сторінку кожної із запропонованих мов програмування (мал. 7.4).

JavaScript

Python

PHP

Dart

XML

[Мал. 7.4](#)

7. Ознайомтесь із текстом програми на кожній з мов програмування. Визначте, яка команда відповідає використаним блокам (мал. 7.3).
8. Закрийте вікно браузера.

2. Яке середовище використовують для створення програм?

Для створення, редагування та налагодження програм, описаних мовою програмування, використовують **середовище програмування**.

Для мови *Python*, як і для інших мов програмування, розроблено багато середовищ програмування. Розглянемо одне з них.

Середовище *Thonny* можна завантажити для роботи на персональному комп'ютері з офіційного сайту (<https://thonny.org/>), обравши інсталяційний пакет для відповідної операційної системи (мал. 7.5).

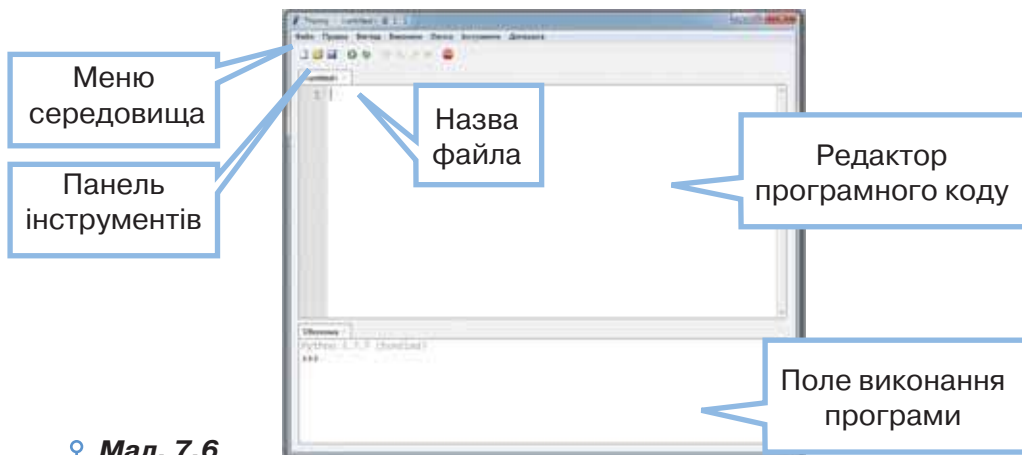


Download version **3.2.7** for
[Windows](#) • [Mac](#) • [Linux](#)

[Мал. 7.5](#)

Після відкриття файла інсталяції потрібно виконати прості кроки для встановлення програми.

Вікно середовища має вигляд (мал. 7.6).



Мал. 7.6



ДІЄМО

Вправа 2. Створення першої програми.

Завдання. Створіть програму мовою програмування *Python* для виведення в полі виконання програми повідомлення *Я вивчаю мову програмування*.

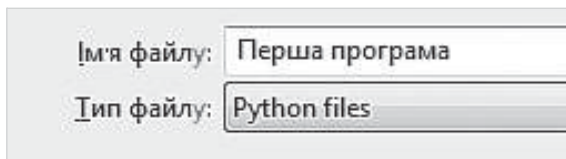
1. Завантажте середовище *Thonny* одним з відомих способів запуску програми.
2. У редакторі програмного коду введіть текст програми (мал. 7.7).

```
<untitled> * * <untitled>
1 #Запишіть текст програми
2 print('Я вивчаю мову програмування')
```

Рядок, що починається символом #, — це **коментар** у тексті програми. Коментарі використовують, щоб допомогти тим, хто ознайомлюється з текстом програми. Вони не впливають на хід виконання програми. Багаторядкові коментарі відділяються від тексту програми потрійними лапками.

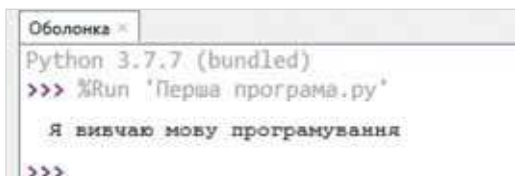
Мал. 7.7

3. У меню *Файл* оберіть команду *Зберегти як* і збережіть у власній папці з іменем *Перша програма* (мал. 7.8). Зверніть увагу на тип файла при його збереженні.



Мал. 7.8

4. Запустіть програму на виконання. Для цього в меню *Виконати* оберіть вказівку *Запустити поточний скрипт*.
5. Перевірте, чи отримали ви подібне повідомлення (мал. 7.9).



Мал. 7.9


6. Очистіть поле виконання програми. Для цього скористайтесь вказівкою з меню *Правка* — *Очистити оболонку* або комбінацію клавіш *Ctrl+I*.
7. Закрийте вікно середовища.



Досліджуємо

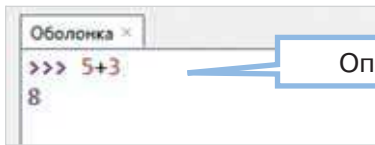
Вправа 3. Використання команди *Налагодити поточний скрипт*.

Завдання. Відкрийте програму *Перша програма*. Використайте комбінацію клавіш *Ctrl+F5*. Зробіть висновок про використання відповідної команди.

1. У своїй папці відкрийте програму *Перша програма.py*. Перевірте, чи відкривається одразу середовище *Thonny*.
2. Використайте комбінацію клавіш *Ctrl+F5*, яка відповідає команді *Налагодити поточний скрипт* із меню *Виконати*.
3. Перевірте, чи отримали ви результат, подібний до малюнка 7.9. На якому рядку програми припинилось її виконання? Як це відображено в редакторі програмного коду?
4. На панелі інструментів натисніть  або клавішу *F8*. Перевірте, чи буде продовжено виконання програми.
5. Порівняйте результат у полі виконання програми з результатом малюнка 7.9. Зробіть висновок про призначення команд *Run* та *Debug*.
6. Закрийте вікно середовища.

3. Чи можна використати середовище програмування як звичайний калькулятор?

У середовищі *Thonny* можна знаходити значення виразів безпосередньо в полі виконання програми. Наприклад, як на малюнку 7.10.



Операція додавання позначається +

Мал. 7.10

Операція віднімання позначається $-$, множення — $*$, ділення — $/$, піднесення до степеня — $**$.

Для того щоб добути квадратний корінь із числа x , використовують оператор $x**0.5$, а корінь n -го степеня — $x**(1/n)$.

На відміну від звичайного калькулятора, у середовищі *Thonny* можна використати спеціальний оператор ділення за модулем ($\%$), який обчислює остачу від ділення, та операцію цілочисельного ділення $//$:

```
>>> 15%6
3
```

```
>>>15//6
2
```

Справді, $15 : 6 = 2$ (ост. 3).

У складених математичних виразах використовують порядок виконання дій за вказаними пріоритетами:

- 1) дужки $()$;
- 2) піднесення до степеня $**$;
- 3) множення $*$, ділення $/$, остача від ділення $\%$, цілочисельне ділення $//$;
- 4) додавання $+$, віднімання $-$.

4. Як у програмі мовою Python використовують команду виведення?

Ви вже знаєте, що для виведення результату виконання програми застосовують команду `print()`. Приклади її застосування та отримані результати описано в таблиці 7.1.

Таблиця 7.1

Команда	Результат
<code>print ("Комп'ютер - мій помічник")</code>	Комп'ютер - мій помічник
<code>print ('Комп\`ютер - мій помічник')</code>	Комп'ютер - мій помічник
<code>print (15*3)</code>	45
<code>print ('15'*3)</code>	151515
<code>print ('15'+ '3')</code>	153
<code>print ('15+3=', 15+3)</code>	15+3=18
<code>print ('15+3=', 15+3, sep='')</code>	15+3= 18
<code>print ('15', '3', sep=' + ', end='')</code> <code>print ('=', 15+3, sep='')</code>	15 + 3 = 18

У команді `print()` іменованій параметр `sep` (від англ. *separator* — роздільник) у передостанньому рядку таблиці використовує порожній рядок (дві одинарні лапки `' '` підряд) як роздільник. Як параметр `sep` можна використовувати будь-який рядок, у тому числі такий, що складається з декількох символів. Якщо нам потрібно зробити кілька різних роздільників для різних частин рядків, то не залишається іншого вибору, окрім як використовувати кілька команд `print()` поспіль, як, наприклад, в останньому рядку таблиці. Зверніть увагу, що в першій команді використано ще один іменованій параметр `end`. Оскільки після кожної команди `print()` наступне виведення здійснюється з нового рядка, то параметр `end=' '` дає змогу «склеїти» два виведення.

Обговорюємо

1. Які є способи подання алгоритмів?
2. Що таке мова програмування? Чим вона відрізняється від мови людського спілкування?
3. У чому особливість установлення середовища програмування *Thonny* на персональній комп'ютер?
4. Як вивести повідомлення в поле виконання програми?
5. Які правила запису команди `print()` мовою програмування *Python*?
6. Які є режими запуску програми на виконання в середовищі програмування *Thonny*? Коли варто використовувати кожен з них?
7. Як виконують арифметичні операції в середовищі *Thonny*? Чим вони різняться від звичайних математичних обчислень?

Працюємо в парах

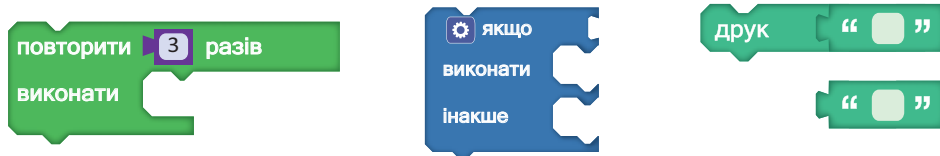
8. Обговоріть, що спільного та які відмінності в навчальних середовищах програмування *Скретч* і *Блоклі*. Результат обговорення подайте діаграмою *Венна*.
9. Обговоріть, що спільного та які відмінності в середовищі *Thonny* та онлайн-овому середовищі *Pynative* (<https://cutt.ly/mu5dKzP>). Поясніть одне одному, як працювати в середовищі *Pynative*. Висловіть припущення, коли вам може знадобитися це середовище.
10. Обговоріть, чому варто вивчати мову програмування *Python*. Скористайтесь інфографікою <https://cutt.ly/TrlNsx1>.





Працюємо самостійно

11. У середовищі *Блоклі* створіть програму, яка у вікні повідомлення тричі виводитиме на екран деякий текст. Обговоріть, які із запропонованих блоків (мал. 7.11) можна для цього використати.



Мал. 7.11

Яким командам у середовищі *Скретч* відповідають обрані блоки? Визначте, як команди повторення та друку реалізуються в різних мовах програмування.

- ★ 12. Складіть програму мовою *Python*, за допомогою якої в полі виконання програми отримують результат (мал. 7.12).

```
>>> %Run 'Завдання 3. py'
Мова програмування Python
не складна для вивчення
>>>
```

Мал. 7.12

- ★ 13. У середовищі, яке підтримує мову *Python*, знайдіть значення виразів:

1) $125 : 25 + 10$; 2) $2 * 3 * 4 * 5 : 6$; 3) $765 : 9 + 48 * 4 - 121$.

Отримані результати запишіть у полі виконання програми поряд з виразом, що обчислюється.

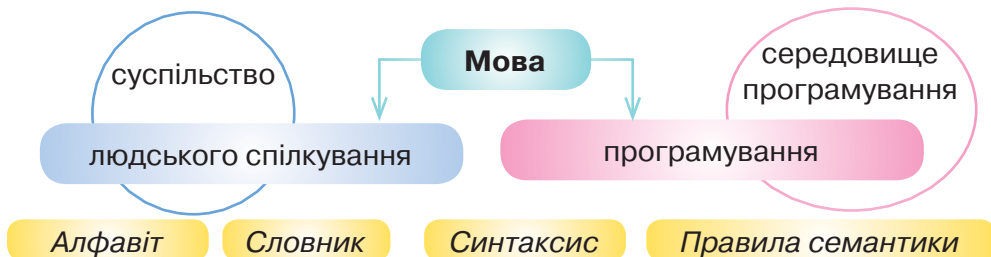
- ★★ 14. Складіть програму мовою *Python*, за допомогою якої в полі виконання програми отримують:

1) у трьох рядках — повідомлення про число, місяць і рік її виконання;
2) тричі слово *Успіх!*

Повторюємо



Розгляньте схему та складіть розповідь, як пов'язані її складові.





8

Величини. Змінні. Вказівка присвоєння

ВИ ДІЗНАЄТЕСЬ:

- ▶ що таке величина та які властивості вона має;
- ▶ якими бувають величини;
- ▶ які розрізняють величини за типами;
- ▶ як надати значення змінній у програмі;
- ▶ які дії виконують над величинами числового типу в *Python*-програмах.



Вивчаємо

1. Що таке величина та які властивості вона має?

Для опису об'єктів і процесів у матеріальному світі ми використовуємо величини. З прикладами величин ви стикаєтеся щодня: відстань між будинком і школою, температура повітря тощо. За допомогою величин можна позначити довжину відрізка, площу земельної ділянки, висоту будинку, швидкість пішохода або автомобіля, час обертання планети навколо Сонця. Кожна величина характеризується певним значенням та одиницями, у яких вимірюється це значення, наприклад, швидкість може дорівнювати 80 км/год, відстань — 700 м, а температура — 15°. Поняття величини відіграє важливу роль у науці взагалі й відображає можливість фіксувати різні стани деяких об'єктів, зокрема кількісну сторону проявів навколишньої дійсності.

Величина має ім'я та може набувати різних значень з деякої множини допустимих значень. Тип цих значень визначає **тип самої величини**.



Це цікаво

Поняття величини прийшло до нас із математики. Уперше властивості величини чітко були сформульовані Евклідом у його «Початках» (III ст. до н. е.).

Визначення значень деяких величин можна здійснювати їх безпосереднім вимірюванням і за певним алгоритмом, якщо значення можуть змінюватися. Так, алгоритм розв'язування рівняння використовують для визначення значень невідомих величин — коренів рівняння. Комп'ютерна модель автомобіля, що подається за допо-

могою математичного рівняння, дає змогу визначати значення величини, що відповідає витратам палива, залежно від його швидкості. З визначенням значень величин пов'язані також алгоритми одержання або створення текстів, різних списків тощо. У цих випадках величини набувають значень, що відповідають фрагментам тексту, елементам списків, значенням *Істина* або *Хибність* тощо.

Для посилань на величини у виразах під час створення програми використовують **імена** величин. Позначення імен називають також **ідентифікаторами**. Ідентифікатори добирають у вигляді деякого скінченного впорядкованого набору літер і цифр, який починається з літери або символу підкреслення `_`. Прикладами ідентифікаторів величин можуть бути такі послідовності символів: `a`, `B2C`, `_I5`, `x`, `y`, `SI`, `DAT_33`, `My_program` тощо. Хорошим стилем у програмуванні вважається, коли величинам надають імена, які б до деякої міри характеризували тип величини та її роль у програмі.



Це цікаво

При написанні програм мовою *Python* імена `a1` і `A1` відрізняються і можуть бути використані для позначення різних величин.

Під час виконання програми в кожний конкретний момент величина має деяке значення або не визначена.

2. Якими бувають величини?

Величини поділяються на змінні та постійні (константи).

Величина, яка має одне і те ж значення в будь-які моменти часу, називається **постійною**, або **константою**.

Константам присвоюються значення в описовій частині програми й у процесі виконання програми їх змінювати заборонено.

Існують константи, до значень яких можна звертатися в програмі без попереднього опису (табл. 8.1).

Таблиця 8.1

Константа	Значення	Опис
True	<i>True</i>	«Істина»
False	<i>False</i>	«Хибність»

Величина, яка в різні моменти часу може набувати різних значень, називається **змінною**.

Під час виконання програми в кожний момент часу величина, як правило, має деяке значення, яке називають **поточним (біжучим)** значенням. При цьому змінна величина може мати лише одне значення

або не мати жодного. У процесі виконання програми величині може бути не надано ніякого конкретного значення. Тоді величина залишається **невизначеною**.

Мова програмування *Python* є досить гнучкою й дає змогу створювати змінні будь-де в коді програми. Але гарний стиль оформлення програм передбачає, що змінні, за можливості, оголошуються в одному місці — їм надаються початкові значення. Це зручніше для програмістів, які одразу бачать, які змінні будуть використовуватися і яких вони типів. Якщо конкретні значення не відомі або не потрібні на початку роботи, можна присвоїти спеціальне значення *None* — «нічого». Тобто це не нуль, не одиниця, не порожній рядок, а взагалі відсутність значення.

3. Які розрізняють величини за типами?

Кожна змінна й постійна величини належать до визначеного типу.

Числові величини — це величини, які можуть набувати значень з деяких числових множин. Наприклад, ціла числова величина *a* може набувати довільних значень із множини цілих чисел (... , -3, -2, -1, 0, 1, 2, 3, ...).

Величини з текстовими значеннями можуть мати символні або рядкові типи. **Символьні величини** можуть набувати значень з деякої множини символів, і кожне значення може містити лише один символ. **Рядкові величини** — це величини, що можуть набувати значень з деякої множини послідовностей символів, зокрема слів або наборів слів. Наприклад, («понеділок», «вівторок», ... , «неділя») — множина значень рядкової величини з іменем *День_тижня*.

Логічні величини можуть набувати тільки одне із двох значень: *true* (*істина*) або *false* (*хибність*).

Від типу значень, яких може набувати величина, залежить множина допустимих операцій. Наприклад, не можна виконувати арифметичні операції з текстовими величинами, операції ділення та віднімання над величинами логічного типу.

Тип величини — це сукупність множини допустимих значень й операцій, які дозволяється виконувати над цими значеннями.

Тип величини визначає обсяг пам'яті, необхідний для зберігання її значень, а також структуру даних. Під час виконання програм, написаних мовою *Python*, система сама визначає обсяг, який числові величини можуть займати в пам'яті комп'ютера, залежно від введеного їх значення.

Тип величини можна визначити за допомогою функції `type()`.
Наприклад:

<code>>>> type(2)</code> <code><class 'int'></code>	<code>>>> type(2.0)</code> <code><class 'float'></code>	<code>>>> type('2')</code> <code><class 'str'></code>	<code>>>> type(4<5)</code> <code><class 'bool'></code>
-----------------------------------------------------------------------	---------------------------------------------------------------------------	-------------------------------------------------------------------------	-----------------------------------------------------------------------------

4. Як надати значення змінній у програмі?

Для того щоб надати значення деякій змінній, у програмі використовують **оператор присвоювання**. У різних мовах програмування цей оператор може мати різний вигляд. У мові *Python* він позначається знаком `=`.

Наприклад, `a=15` означає, що змінній `a` надано значення 15. Результатом виконання програми на малюнку 8.1 буде 45.

```
1 a=15
2 b=3
3 print (a*b)
```

 Мал. 8.1

Надавати значення можна декільком величинам одночасно.

Наприклад, в обох програмах (мал. 8.2) отримаємо однаковий результат: 15 30 45.

```
1 a=b=c=15
2 print (a,b*2,c*2)
```

```
1 a=b=c=15,30,45
2 print (a,b,c)
```

 Мал. 8.2

У мові програмування *Python* можна використовувати комбіновані операції присвоювання (`+=`, `-=`, `*=`, `/=`, `//=`, `%=`, `**=` та інші). Наприклад:

`x+=3` — додавання до значення змінної `x` числа 3;

`x-=2` — віднімання від значення змінної `x` числа 2.

Для введення значення змінної під час виконання програми використовують команду `input()`. Ця функція зчитує рядок з поля виконання програми, а щоб закінчити введення рядка, потрібно натиснути *Enter*. Так, після виконання програми

```
name=input()
input('Я люблю',name)
```

у полі виконання слід ввести слово, наприклад *Python*. Отримаємо результат:

```
Python
Я люблю Python
>>>
```

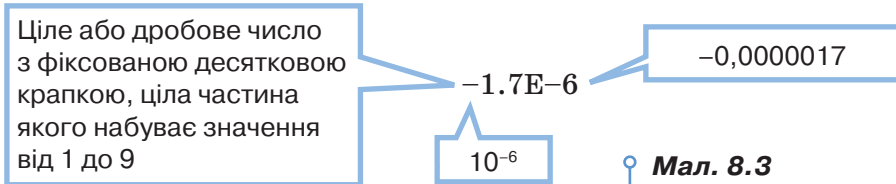
5. Які дії виконують над величинами числового типу в Python-програмах?

Для запису арифметичних виразів, аргументів математичних та деяких інших функцій можуть використовуватися числа або величини числового типу. Значення цих величин можуть бути цілими або дійсними числами. Для опису числових величин мовою програмування *Python* використовують службові слова (табл. 8.2).

Таблиця 8.2

Тип	Опис числових величин мовою програмування	Можливе значення
Цілий	int	Довільне ціле число
Дійсний	float	Дійсне число

Мова програмування *Python* допускає представлення дійсних чисел як з фіксованою, так і з плаваючою крапкою. Дійсні *десяткові числа з фіксованою крапкою* записуються за звичайними правилами арифметики. Єдине, що відрізняє цей формат від математичного — це те, що ціла частина від дробової відокремлюється десятковою крапкою, а не комою. Якщо десяткова крапка відсутня, число вважається цілим. Перед числом може бути записано знак «+» або «-». Якщо знак відсутній, за замовчуванням число вважається додатним. Наприклад, 9.23, 0.05. Дійсні *десяткові числа з плаваючою крапкою* подаються в експоненційному вигляді (мал. 8.3):



Мал. 8.3

Наприклад, відстань від Землі до Сонця становить $1,496 \cdot 10^{11}$, або $1.496e11$ в *Python*. Маса однієї молекули води $2,99 \cdot 10^{-23}$, або $2.99e-23$ в *Python*.

Над величинами цілого та дійсного типу можна виконувати операції додавання (+), віднімання (-), ділення (/), піднесення до степеня (**), цілочисельного ділення (//) та ділення з остачею (%) (табл. 8.3).

Таблиця 8.3

Значення величини a	Значення величини b	Операція	Тип результату
Ціле	Ціле	+, -, *, **, //, %	Ціле
Ціле	Ціле	/	Дійсне
Дійсне	Дійсне	+, -, *, /, **	Дійсне

Величини дійсного типу можна округлювати. Для цього використовують функцію `round()`. Наприклад,

```
print(round(1.3)) # отримаємо 1
print(round(1.7)) # отримаємо 2
print(round(-1.3)) # отримаємо -1
print(round(-1.7)) # отримаємо -2
```

Функція `round()` може мати один параметр — дійсне число, що округлюється, і може містити ще один параметр — ціле число,

що вказує на кількість знаків після цілої частини. Якщо це число 2, то округлення відбувається до сотих, якщо 0, то до цілого, якщо -1, то округлюється до десятків. За відсутності другого параметра за замовчуванням число округлюється до цілого.

```
a=12.473
print(round(a)) # отримаємо 12
print(round(a,2)) # отримаємо 12.47
print(round(a,-2)) # отримаємо 10
```

У багатьох задачах нам потрібно працювати з введеними числами, а результатом команди `input()` є текст. Щоб перетворити текст, який складається із цифр (і, можливо, знака «-» перед ними), у число, можна скористатися функціями:

`int()` — перетворення тексту в ціле число;
`float()` — перетворення тексту в дійсне число.



ДІЄМО

Вправа 1. Равлик.

Завдання. Равлик рухається по вертикальній жердині, піднімаючись за день на a м, а за ніч спускаючись на b м. Скільки діб має пройти, щоб равлик опинився на жердині на висоті h м? Складіть програму в середовищі *Thonny*.

1. Запустіть середовище програмування *Thonny*.
2. Введіть значення змінних за зразком:

```
h=int(input('Введіть висоту h='))
```

3. Для обчислення кількості днів використайте підказку (мал. 8.4):

Мал. 8.4 $(h - 2 * b + a - 1) // (a - b)$

Обчислюємо відстань, яку потрібно пройти

Обчислюємо швидкість піднімання равлика

4. Виведіть отриманий результат.
5. Збережіть файл програми з іменем *Равлик* у папці *Програми* своєї структури папок.

Вправа 2. Щасливе число.

Завдання. Семикласники придумали гру. Якщо у квитку на сеанс фільму, який містить тризначний номер, сума цифр цього номера виявиться такою, що дорівнює 20, то квиток щасливий. Складіть програму, яка за введеним значенням номера обчислює його суму цифр.

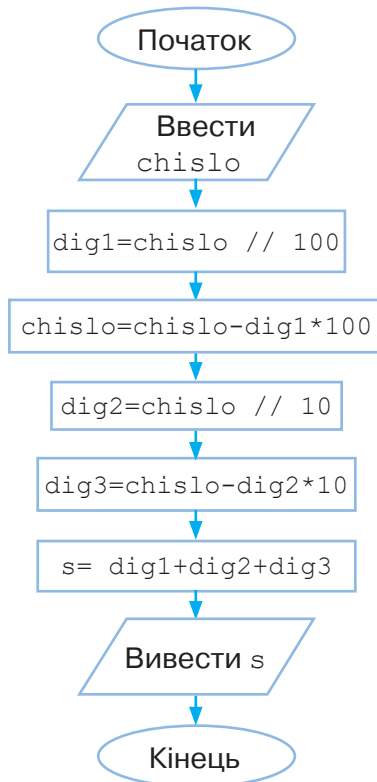
1. Запустіть середовище програмування *Thonny*.

2. Сплануйте, які змінні будуть використані у програмі. Скористайтеся для цього таблицею 8.4.

Таблиця 8.4

Ідентифікатор	Тип	Призначення
chislo	<i>integer</i>	Введене трицифрове число
dig1	<i>integer</i>	Перша цифра числа
dig2	<i>integer</i>	Друга цифра числа
dig3	<i>integer</i>	Третя цифра числа
s	<i>integer</i>	Сума цифр

3. Для складання алгоритму розв'язування задачі врахуйте, що трицифрове число можна подати у вигляді суми розрядних одиниць: $abc = a \cdot 100 + b \cdot 10 + c \cdot 1$, де a — перша, b — друга, c — третя цифри числа.
4. Запишіть програму відповідно до блок-схеми (мал. 8.5).
5. Збережіть програму з іменем *Число* в папці *Програми* своєї структури папок.
6. Запустіть програму на виконання з такими числами: 925, 947. Чи є серед них «щасливе»?



Мал. 8.5

Обговорюємо

1. Які величини ми використовуємо в повсякденному житті та чим вони різняться?
2. Як мовою програмування *Python* описують константи та змінні величини?
3. Опишіть правила вибору ідентифікаторів величин. Наведіть приклади.
4. Яку функцію використовують для введення даних користувачем? Як значення цієї функції перетворити на числове?
5. Чим різняться операції над числовими даними цілого та дійсного типів мовою програмування *Python*?
6. Коли доцільно використовувати функцію округлення та операції цілочисельного ділення або знаходження остачі від ділення?

Працюємо в парах

7. Пограйте у гру «Значення величини — тип величини». Хтось називає множину значень деякої величини, а хтось — відповідний тип величини. Гра продовжується до першої помилки, після цього учасники міняються ролями.
8. Запропонуйте одне одному вирази для обчислення значення змінної-результату, у яких вхідними даними є цілі числа, а результатом — дійсне число. Запишіть програмний код оголошення змінних та обчислення значення змінної-результату мовою *Python*. Обміняйтеся записами й перевірте одне одного.
9. Семикласники розв'язували задачу, за якою при введенні числа 1 воно змінюється на 0 і навпаки — з 0 на 1. Обговоріть, яку із запропонованих програм ви б обрали для розв'язування цієї задачі, якщо n — введене число.

```
n=int(input())
print((1-n)//1)
```

```
n=int(input())
print((1-n)%1)
```

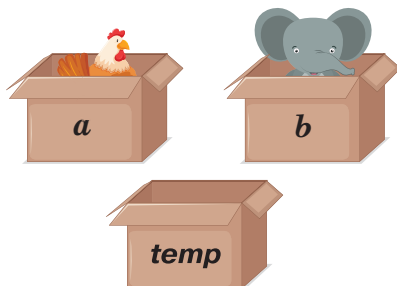
```
n=int(input())
print(1-n)
```

10. Дано a ($a < 1440$) — кількість секунд, які минули від початку доби. Що буде відображено на табло електронного годинника, якщо покази подаються у форматі години:хвилини:секунди. Обговоріть призначення змінних і команд у програмі:

```
a=int(input())
h=a//3600%24
m=a%3600//60
s=a%60
print(h, ':', m//10, m%10, ':', s//10, s%10, sep= '')
```

Перевірте програму в середовищі програмування.

- 11.** Сплануйте розв'язок задачі: скільки кроків потрібно, щоб перенести курку в ящик b , а слона — у коробку a , знаючи, що в коробці не може бути більше однієї тварини? Користайтеся запропонованим малюнком.



Працюємо самостійно

- 12.** У школі облаштовують три нові класи для семикласників. Кількість учнів у кожному з класів потрібно ввести під час виконання програми. За однією партою може сидіти не більше двох учнів. Яку найменшу кількість парт потрібно замовити? Побудуйте блок-схему для розв'язування задачі та реалізуйте відповідну програму мовою *Python* у середовищі програмування для таких значень кількості учнів: 25, 22, 23.
- 13.** Плановий відділ швейної компанії розраховує площу тканини, яка залишиться після того, як із заготовки розміром $a \times a$ вирізали круг радіусом r . Побудуйте блок-схему для розв'язування задачі та реалізуйте відповідну програму мовою *Python* у середовищі програмування для $a = 4$ м, $r = 2$ м.
- 14.** Для виготовлення жерстяної коробки використовують квадратний лист жерсті розміром $a \times a$ см². Висота коробки становить b см. Маса 1 см² жерсті становить c г. Визначте масу коробки. Побудуйте блок-схему для розв'язування задачі та реалізуйте відповідну програму мовою *Python* у середовищі програмування для $a = 120$ см, $b = 20$ см, $c = 0,4$ г.
- 15.** В електронній касі у двох комірках містяться кошти в розмірі a грн та b грн. Складіть алгоритм, за допомогою якого кошти в цих комірках будуть обміняні значеннями за умови, що додаткової касової комірки немає. Примітка: можна використовувати операції додавання й віднімання. Побудуйте блок-схему для розв'язування задачі та реалізуйте відповідну програму мовою *Python* у середовищі програмування для $a = 20$, $b = 30$.

Повторюємо

Пригадайте поняття уроку і складіть схему, яка їх пов'язує.



Додаткові модулі

ВИ ДІЗНАЄТЕСЬ:

- ▶ якими командами можна доповнити стандартні команди мови *Python*;
- ▶ як у середовищі програмування створювати графічні примітиви.



Вивчаємо

1. Якими командами можна доповнити стандартні команди мови *Python*?

Стандартні команди мови *Python* можна доповнити додатковими командами, які об'єднано в **модулі**. Модуль у *Python* — це спосіб надання деякого коду для використання у програмі.

Щоб підключити додатковий модуль до програми, застосовують таку конструкцію:

```
import назва_модуля.
```

Модулі оголошуються на початку програми.

Доступ до команди додаткового модуля здійснюють за шаблоном:
назва_модуля.команда.

Наприклад, модуль `math` містить додаткові команди для роботи з числами, які не входять до стандартного набору. У таблиці 9.1 наведено деякі з функцій модуля `math`.

Таблиця 9.1

Команда	Результат	Приклад
<code>math.fabs(x)</code>	Модуль x	<code>math.fabs(-10)</code> результат: 10
<code>math.ceil(x)</code>	Округлення до найближчого більшого числа	<code>math.ceil(9.1)</code> результат: 10.0
<code>math.factorial(x)</code>	Факторіал числа x — добуток послідовних натуральних чисел від 1 до x	<code>math.factorial(5)</code> результат: 120
<code>math.sqrt(x)</code>	Квадратний корінь із числа x	<code>math.sqrt(100)</code> результат: 10

Модуль `random` містить функції для генерування випадкових чисел, букв, випадкового набору послідовності.

`random.randint(a, b)` — випадкове ціле число n , $a \leq n \leq b$.

`random.random()` — випадкове число від 0 до 1.

`random.choice(['a', 'o', 'y', 'e', 'и', 'і', 'ї', 'ю'])` — вибір випадкового значення зі списку, наприклад, зі списку голосних літер українського алфавіту.



ДІМО

Вправа 1. Відгадайте число.

Завдання. Складіть програму, яка обчислює факторіал випадкового числа з проміжку від 1 до 6.

1. Запустіть середовище програмування *Thonny*.
2. У програмі підключіть модулі: `import math, random`.
3. Надайте змінній x випадкового значення з проміжку від 1 до 6:
`x=random.randint(1, 6)`.
4. Знайдіть значення факторіалу «задуманого» числа:
`y=math.factorial(x)`.
5. Виведіть отриманий результат. Відгадайте, яке число «задумав» комп'ютер.

2. Як у середовищі програмування створювати графічні примітиви?

У мові *Python* можна підключити модуль створення векторної графіки, зокрема створення графічних примітивів з ліній, точок і кривих. Це — модуль `turtle`. Команди, які входять до модуля `turtle`, дозволяють у програмі управляти деяким об'єктом — назовемо його черепахою. Деякі команди із цього модуля подано в таблиці 9.2.

Таблиця 9.2

Команда	Значення
<code>turtle.forward(x)</code>	Пройти вперед x пікселів
<code>turtle.backward(x)</code>	Пройти назад x пікселів
<code>turtle.left(x)</code>	Повернутись ліворуч на x градусів
<code>turtle.right(x)</code>	Повернутись праворуч на x градусів
<code>turtle.penup()</code>	Не залишати слід під час руху
<code>turtle.pendown()</code>	Залишати слід під час руху

Команда	Значення
<code>turtle.shape(x)</code>	Змінити значок виконавця ('arrow', 'turtle', 'circle', 'square', 'triangle', 'classic')
<code>turtle.stamp()</code>	Намалювати копію виконавця в поточному місці
<code>turtle.color()</code>	Установити колір
<code>turtle.begin_fill()</code>	Необхідно викликати перед малюванням фігури, яку слід зафарбувати
<code>turtle.end_fill()</code>	Викликати після завершення малювання фігури
<code>turtle.width()</code>	Установити товщину лінії
<code>turtle.goto(x, y)</code>	Перемістити виконавця в точку (x, y)

Наприклад, після запуску програми (мал. 9.1) у вікні малювання буде намальована літера S:

```
import turtle
turtle.shape('turtle')
turtle.forward(50)
turtle.left(90)
turtle.forward(50)
turtle.left(90)
turtle.forward(50)
turtle.right(90)
turtle.forward(50)
turtle.right(90)
turtle.forward(50)
```

 Мал. 9.1



ДІМО

Вправа 2. Квадрат.

Завдання. Складіть програму, яка малює на екрані квадрат з кольоровими сторонами, довжину та товщину яких вказує виконавець.

1. Запустіть середовище програмування *Thonny*.
2. Оголосіть приєднання модуля побудови графічних примітивів `import turtle`.
3. Задайте в програмі команду введення значення змінної d — довжини сторони квадрата від 50 до 300.

4. Задайте в програмі команду введення значення змінної h — товщини пензлика від 1 до 10.
5. Доповніть програму (мал. 9.2) командами:

```
turtle.shape('arrow')
```

Вибір форми виконавця

```
turtle.width(h)
```

Установлення товщини пензлика малювання виконавця

```
turtle.color('blue')
```

Установлення кольору сторони квадрата

```
turtle.forward(d)
```

```
turtle.left(90)
```

```
turtle.color('yellow')
```

Малювання сторони квадрата

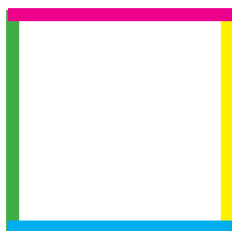
```
turtle.forward(d)
```

```
turtle.left(90)
```

Поворот ліворуч на 90°

Мал. 9.2

6. Завершіть код побудови сторін з кольорами «red», «green».
7. Збережіть програму з іменем *Квадрат_Прізвище* у своїй структурі папок.
8. Запустіть програму на виконання. Перевірте, чи отримали ви подібний малюнок (мал. 9.3):



Мал. 9.3

Досліджуємо

Вправа 3. Квадрат.

Завдання. Дослідіть, як зміниться результат програми побудови квадрата, якщо довжину й товщину сторони задати випадково. Внесіть потрібні зміни у програму *Квадрат_Прізвище* та запустіть її на виконання.

Вправа 4. Гра.

Завдання. Визначте у грі *Кодекомбат*, які команди мови програмування *Python* уміє виконувати виконавець.

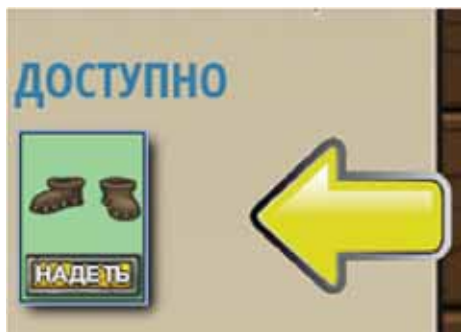
1. Перейдіть до гри за посиланням <https://codecombat.com/play/dungeon>. Оберіть Підземелля Кітгарда.

2. Розпочніть гру



У грі візьміть самоцвіт у підземеллі, нічого не чіпаючи. На цьому рівні ви засвоїте основи переміщення героя гри.

3. Оберіть героя та мову програмування.
4. Ознайомтесь із командами, що доступні герою, який одягне чобітки руху (мал. 9.4).



moveDown — рух униз;
moveLeft — рух ліворуч;
moveRight — рух праворуч;
moveUp — рух вгору.

Мал. 9.4

5. Натисніть — *Надягнути* та *Грати*.
6. Ознайомтесь із завданням для героя: рухатись лабіринтом, уникати шипів і взяти самоцвіт (мал. 9.5)



Мал. 9.5

7. Продовжте написання коду. Використовуйте підказки з вікна (мал. 9.6).

```

5 hero.moveRight()
6 he|
7 hero.moveUp      натисність enter
  hero.moveDown
  hero.moveLeft
  hero.moveRight
  
```

Мал. 9.6

8. Запустіть складену програму (*Запустити*). Перейдіть до наступного рівня (*Готово*).
9. Зробіть висновок про назву додаткового модуля мови *Python*, який використано у грі. Продовжуйте грати. Визначте, які команди може виконувати виконавець.
10. Завершіть роботу в грі за пропозицією вчителя або вчительки.



Обговорюємо

1. Що таке модуль у мові програмування *Python*?
2. Як записують команду деякого модуля в кодї програми в мові програмування *Python*?
3. Які команди містить модуль `math`?
4. З якою метою використовують модуль `random`?
5. За допомогою яких команд можна малювати в мові програмування *Python*?
6. Чому команди малювання модуля `turtle` відносять до команд векторної графіки?



Працюємо в парах

7. Обговоріть переваги та недоліки використання модулів у мові програмування *Python*.
- ★ 8. Пограйте у гру «Запитання — відповідь». Запропонуйте одне одному завдання на використання команд модуля `math`, `random` у мові програмування *Python*. Завдання задаються до першої помилки іншого. Потім гравці міняються ролями.
- ★ 9. Обговоріть, чим подібні та відрізняються модуль малювання в середовищі програмування мовою програмування *Python* і команди групи *Олівець* із середовища *Скретч*.
- ★★ 10. Пограйте у гру «Команда — виконавець». Хтось називає команду, а хтось — виконує за цією командою відповідну побудову. Завдання задаються до першої помилки іншого. Потім гравці міняються ролями.
- ★★ 11. На сайті *Realpython* автори пропонують серію електронних підручників з питань вивчення мови програмування *Python*. Один із них навчає початківців використовувати модуль `turtle`. Перейдіть за посиланням:
<https://realpython.com/beginners-guide-python-turtle/#moving-the-turtle>. Обговоріть у парі, які розділи пропонованого електронного підручника



ви можете вже використовувати, а які — ні, і чому. Розгляньте підсумкове завдання. Якими порадами щодо його розв'язування варто скористатись для складання власних проєктів, ігор та програм мовою *Python*.



Працюємо самостійно

У завданнях 12–15 побудуйте блок-схему, складіть програму мовою *Python* і реалізуйте її в середовищі програмування.

- ★ **12.** Програма «Чарівне число», у якій комп'ютер має «загадати» деяке ціле число від 1 до 10. Виконати наступні дії: до задуманого числа додати 3; від того, що вийшло, відняти 1; до того, що вийшло, додати 2; до того, що вийшло, додати 1; до того, що вийшло, додати 2; від того, що вийшло, відняти задумане число. Вивести результат. Перевірте, який результат отримано. Чи можна після повторного запуску програми отримати те саме число? Чи означає це, що комп'ютер задумав одне й те саме число?
- ★ **13.** Програма «Решта», у якій знаходять різницю між округленням деякого дійсного числа із проміжку від 1 до 10 та його округленням до найближчого більшого числа.
- ★★ **14.** Програма «Таблиця квадратів та коренів», у якій для введеного числа знаходять його квадрат і квадратний корінь. У результаті виводиться три числа.
- ★★ **15.** Програма «Ім'я», у якій виконавець малює літери вашого імені.

Повторюємо



Розгляньте схему та складіть розповідь, як пов'язані її складові.



`назва_модуля.команда`

`import назва_модуля`



Текстові величини та операції над ними

ВИ ДІЗНАЄТЕСЬ:

- ▶ як вводять й описують текстові величини мовами програмування;
- ▶ які операції виконують над текстовими величинами;
- ▶ які виникають помилки при створенні та виконанні програм.



Вивчаємо

1. Як вводять й описують текстові величини мовами програмування?

Для опрацювання текстових даних мовою програмування використовують величини символного та рядкового типів.

Значенням величини **символьного типу** може бути один символ — літера, цифра чи знак. Набір символів утворює рядок.

Надавати значення текстовим величинам, як і числовим, можна при написанні програмного коду за допомогою оператора присвоювання або під час виконання програми, що містить команду введення.

Ви вже знаєте, що у програмному кодї мовою програмування *Python* можна використовувати одинарні та подвійні лапки.

Рядки можуть містити набори символів, що починаються з похилої риски `\`; їх називають *escape*-послідовностями. При виведенні рядка такі символи опрацьовуються спеціальним чином і можуть повністю або частково не відобразитися на екрані. Найчастіше використовують такі:

- `\n` — перехід на новий рядок;
- `\t` — вставлення табуляції;
- `\"` — подвійні лапки (корисно, якщо необхідно вставити подвійні лапки в рядок, оточений подвійними лапками);
- `\'` — одинарні лапки (корисно, якщо необхідно вставити, наприклад, апостроф у рядок, оточений одинарними лапками).



Це цікаво

Поняття *escape-послідовності*, що походить від англomовного аналогу *escape sequences* (від англ. *to escape* — уникати), викори-

стовують для запису в текстових значеннях спеціальних символів, у яких уникається виведення на екран самої послідовності символів.

Усі символи в рядку мають свій порядковий номер. За цим номером можна отримати значення символу, вказавши номер у квадратних дужках після імені змінної. У мові *Python* нумерація символів розпочинається з нуля:

```
z='школа' z[4] - а.
```

2. Які операції виконують над текстовими величинами?

Над текстовими величинами виконують операцію склеювання — поєднання кількох рядкових величин, яка позначається символом «+» (мал. 10.1).

Приклад набору команд

```
f='алго'
l='ритм'
r=f+l;
```

Результат — значення змінної r

```
'алгоритм'
```

 **Мал. 10.1**

Текстові величини є незмінюваними, це означає, що не можна змінити частину рядка, не створивши нового.

У програмі мовою програмування *Python* при роботі з текстовими величинами можна використовувати операцію зрізу, за допомогою якої копіюється послідовність або її частина. Наприклад, для змінної `s = 'Hello world'` операції зрізу подано в таблиці 10.1.

Таблиця 10.1

Опис	Приклад команди	Результат — значення змінної <code>s1</code>
Зріз від початку рядка до символу з номером n	<code>s1 = s[:6] + 'Python!'</code>	'Hello Python!'
Зріз від символу з номером n рядка включно до символу з номером m	<code>s2=s[3:5]</code>	'lo'
Зріз від символу з номером n рядка включно до кінця рядка	<code>s3=s[6:]</code>	'world'

Щоб записати символи в рядку у зворотному порядку, використовують операцію `s4=s[::-1]`.

Тоді змінна `s4` буде мати значення 'dlrow olleH'.

Текстову величину можна повторити декілька разів. Наприклад:

```
>>> print('spam' * 3)
spamspamspam
```

Ви вже використовували функції для перетворення текстової величини у величину числового типу. Щоб перетворити число у величину текстового типу, у мові програмування *Python* використовують функцію `len()`.



ДІЄМО

Вправа 1. Слова.

Завдання. Складіть програму мовою *Python*, за допомогою якої з введеного слова *інформатика* будуть утворюватися слова *форма*, *романтика*.

1. Запустіть середовище програмування.
2. Створіть новий файл програми мовою *Python* з іменем *Слова* в папці *Програми* своєї структури папок.
3. Нехай у програмі будуть використані змінні: *s* — значенням якої буде введене з клавіатури слово *інформатика*, *s1* — отримане значення якої має бути *форма*, *s2* — отримане значення якої має бути *романтика*.
4. Визначте номер позиції кожної літери у слові:

0	1	2	3	4	5	6	7	8	9	10
і	н	ф	о	р	м	а	т	и	к	а

5. У вікні редактора коду запишіть команди введення змінної *s* та знаходження значень змінних *s1*, *s2*:


```
s=input('Введіть слово')
s1=s[2:7]
s2=s[4]+s[3]+s[5:7]+s[1]+s[7:]
```
6. Запишіть команди виведення отриманих значень.
7. Запустіть проєкт на виконання. Введіть у вікні виконання програми значення змінної *s*: *інформатика*. Перевірте отримані результати. За потреби виправте помилки.
8. Завершіть роботу із середовищем.

3. Які виникають помилки при створенні та виконанні програм?

Створена в середовищі програмування програма може не виконуватися взагалі, або результат її виконання не відповідатиме очікуваному результату. Це відбувається тоді, коли при складанні алгоритму розв'язування завдання чи написанні коду програми були допущені помилки. Розрізняють три групи помилок:

- синтаксичні;
- помилки під час виконання;
- логічні.

Синтаксичні помилки можна виявити як у процесі написання програмного коду, так і після запуску програми на виконання. Якщо деяка команда у програмному коді написана програмістом не за правилами, прийнятими в мові програмування, то вона може набувати іншого кольору, який відмінний від кольору правильно написаних команд.

Після запуску програми на виконання у вікні виконання програми в середовищі *Thonny* буде відображено не тільки місце розташування помилки у програмному коді, а й сама команда, у якій допущено помилку. Наприклад,

```
Traceback (most recent call last):
  File «D:\Учні\7 клас\Зразок\Помилка1.py», line 1, in
<module>
    a = int(inrut())
NameError: name 'inrut' is not
defined
>>>
```

При цьому в середовищі з'явиться ще одне поле *Асистент*, у якому може бути не тільки пояснення помилки, а й спосіб її усунення (мал. 10.2).

Типовими є помилки, що позначають невідповідність типів описаних величин і значень, які їм присвоюються у процесі виконання програми. Наприклад, після запуску програми на виконання отримуємо повідомлення про неможливість застосування операції у виразі даних визначеного типу (мал. 10.3).

```
a = input()
b = input()
k = (a % b) // a * b
Traceback (most recent call last):
  File «D:\Учні\7 клас\Зразок\Помилка2.py», line 3, in
<module>
    k = (a % b) // a * b
TypeError: not all arguments converted during string
formatting
```



Мал. 10.2

Мал. 10.3

До помилок під час виконання відносять помилки, пов'язані з неправильними числовими обчисленнями, помилки при знаходженні значень величин за формулами. Їх розпізнають тільки під час виконання програми. Наприклад, якщо у програмі мовою *Python* використати програмний код

```
f=5
c=0
r=f/c
print(r)
```

то після запуску програми на виконання у вікні виконання отримаємо повідомлення

```
Traceback (most recent call last):
  File «D:\Учні\7 клас\Зразок\Помилка3.py», line 3, in
<module>
    r=f/c
ZeroDivisionError: division by zero
```

Логічні помилки — це помилки алгоритму, який лежить в основі програми. Результат, отриманий у ході виконання програми, не збігається з очікуваним результатом. Такі помилки не можна виявити засобами програмного середовища.

Виправити синтаксичні помилки та помилки часу виконання програми можна після їх виявлення. А тоді знову запустити програму на виконання. Для того щоб перевірити наявність логічних помилок, потрібно запустити програму з тестовим (вхідним) набором даних і перевірити, чи збігається результат «ручного» підрахунку та програмного.

Процес пошуку логічних помилок у тексті програми з використанням тестових наборів вхідних даних називають **тестуванням програми**.



Обговорюємо

1. Наведіть приклади значень текстових величин.
2. Чим відрізняються значення символьного та рядкового типів?
3. Як описати величини символьного та рядкового типів мовою програмування *Python*?
4. Які операції можна виконувати з текстовими величинами?
5. Які функції використовують у мові програмування *Python* для опрацювання текстових величин?
6. Чому програмний код у проекті може не виконатися взагалі або в програмі можуть бути отримані неправильні результати?

7. Яких помилок допускаються під час складання програм у середовищах програмування?
8. У чому полягає процес налагодження програми?
9. Як у середовищах програмування здійснюють контроль над значеннями змінних у процесі виконання програми?



Працюємо в парах

10. У програмі, написаній мовою програмування *Python*, використавши значення текстової змінної *s1*, отримали *s2* (табл. 10.2). Складіть відповідний програмний код. Перевірте, чи однакові засоби отримання значення змінної *s2* зі змінної *s1* ви використали. Зробіть висновки.

Таблиця 10.2

Значення змінної <i>s1</i>	Значення змінної <i>s2</i>
Слово — не спіймаєш, вилетить — не горобець	Слово — не горобець, вилетить — не спіймаєш
Світнебездобрихлюдей	Світ не без добрих людей
Вік учись — вік живи	Вік живи — вік учись
ьтюанзіп ітуркс в агурД	Друга в скрутї пізнають

11. Обговоріть, як мовою програмування *Python* можна вивести текст заяви про зарахування до гуртка технічної творчості у прийнятому для такого документа вигляді, у якій дані про заявника вносяться у процесі виконання програми.
12. Обговоріть, у чому схожість процесу налагодження програми в середовищі програмування та редагування тексту в середовищі текстового процесора. Чи можна вважати редагування і налагодження синонімами?



Працюємо самостійно

У задачах 13–15 побудуйте блок-схему та реалізуйте відповідну програму в середовищі програмування. Збережіть програму в папці *Програми* власної структури папок.

- 13.** Складіть програму, за допомогою якої зі слова «університет» можна отримати слово «турист».
- ★ **14.** У програмі *Шифрувальник* вводять слово з п'яти літер, а після запуску програми на виконання отримують результат за правилами, описаними в таблиці 10.3.

Таблиця 10.3

№	Результат	Введене значення	Отримане значення
1	Усі літери написано двічі	школа	шшккооллаа
2	Змінено порядок літер у парах	книга	нкгиа
3	Після кожної літери додано останню літеру слова	голуб	гбоблбубб
4	Кожну літеру замінено на відповідний код у кодовій таблиці	зошит	10791086109610801090
5	Кожна літера замінена на наступну в кодовій таблиці	аркуш	бслфщ

- ★ **15.** У програмі *Розрядні одиниці* вводять чотиризначне ціле число й отримують запис числа у вигляді суми розрядних одиниць. Наприклад, для введеного числа *5843* отримують $5*1000+8*100+4*10+3*1$.

Повторюємо



Склади схему з термінів, які вивчались у даному уроці.



Опрацювання величин логічного типу. Команда розгалуження

ВИ ДІЗНАЄТЕСЬ:

- ▶ як порівнюють значення величин у програмах;
- ▶ які операції виконують над логічними величинами;
- ▶ як описати алгоритмічну структуру неповного розгалуження мовою програмування *Python*;
- ▶ як описати алгоритмічну структуру повного розгалуження мовою програмування *Python*.



Вивчаємо

1. Як порівнюють значення величин у програмах?

Ми вже розглядали програми, які реалізують **лінійні** алгоритми. Для опису умови в алгоритмах із **розгалуженням** використовують логічні вирази з використанням операцій порівняння (табл. 11.1). Результатом виконання операції порівняння значень двох величин є величина **логічного типу**, яка може набувати одного із двох значень: *True* або *False*. Для опису логічних величин у мові програмування *Python* використовують службове слово `bool`.

Таблиця 11.1

Операція	Операція мовою <i>Python</i>	Приклад $A=10$	Результат
Більше	<code>></code>	$A > 100$	<i>False</i>
Менше	<code><</code>	$A < 100$	<i>True</i>
Не більше	<code><=</code>	$A <= 10$	<i>True</i>
Не менше	<code>>=</code>	$A >= 10$	<i>True</i>
Дорівнює	<code>==</code>	$A == 100$	<i>False</i>
Не дорівнює	<code>!=</code>	$A != 100$	<i>True</i>



Це цікаво

Логічний тип отримав свою назву на честь англійського математика та логіка середини XIX століття, одного із засновників математичної логіки Джорджа Буля.

Порівняння значень величин різних типів має особливості. Числові величини порівнюються за правилами математики. Порівняння текстових величин мовою *Python*, у результаті якого отримано логічне значення *True*, продемонстровано на малюнку 11.1.

```
'алгоритм' > 'алго' — довжина першої величини більша за
                    довжину другої
'алгоритм' != 'alhoritm'
'алл' == 'алл'
```

 Мал. 11.1

Для логічних величин $a = True$, $b = False$ результатом виконання операції $a > b$ буде значення *True*, оскільки істинне значення *True* інтерпретується, як 1, а хибне *False* — 0. Зрозуміло, що $1 > 0$.

Для того щоб перевірити, чи входить деякий фрагмент текстової величини в іншу текстову величину, використовують оператор входження *in*.

Наприклад, результатом перевірки умови `'y' in 'Python'` є *True*.

2. Які операції виконують над логічними величинами?

Окрім операцій порівняння над логічними величинами, в усіх мовах програмування для запису складених висловлювань використовують логічні операції:

- *not* (не) — заперечення;
- *and* (і) — кон'юнкція;
- *or* (або) — диз'юнкція (нестрога);
- *xor* (виключне або) — диз'юнкція (строга).



Це цікаво

Операцію *or* ще називають логічним додаванням, а *and* — логічним множенням. Назви кон'юнкція та диз'юнкція походять від англ. *conjunction* — об'єднання та *disjunction* — роз'єднання. У математиці та логіці для запису логічних операцій прийняті спеціальні позначення. Операцію *and* позначають \wedge , операцію *or* — \vee .

Результати виконання цих операцій над змінними A і B логічного типу, які набувають значень *True* (1) і *False* (0), наведено в таблиці істинності (табл. 11.2).

Таблиця 11.2

A	B	$\text{not } A$	$A \text{ and } B$	$A \text{ or } B$	$A \text{ xor } B$
1	1	0	1	1	0
1	0	0	0	1	1
0	1	1	0	1	1
0	0	1	0	0	0

З таблиці істинності видно, що за допомогою логічної операції `not` змінюють значення логічної величини на протилежне. Результат операції `and` збігається з результатом множення значень `1` і `0` — є істинним лише за умови, що обидві величини A і B набувають значення *True* — істина. Результат операції `or` буде хибним лише за умови, що величини A і B набувають значення *False*, та істинним в усіх інших випадках. Наприклад, якщо $A = 5$, а $B = 7$, то значенням логічного виразу $(A < B)$ `and` $(B = 7)$ є *True*, оскільки логічні вирази $5 < 7$ та $B = 7$ набувають значення *True*, а тому, за таблицею істинності, результатом операції `and` є значення *True*.

Щоб визначити значення складеного логічного виразу, який містить кілька логічних операцій, використовують таблицю істинності та враховують пріоритет виконання операцій: у першу чергу виконується операція `not`, далі — `and`, в останню чергу — `or`. Як і для числових виразів, для зміни порядку виконання логічних операцій використовують дужки.

У мові програмування *Python* передбачено, що логічні значення *True* та *False* можна перетворювати на значення інших типів. Наприклад, при перетворенні їх у рядкові величини отримаємо слова *True* та *False* відповідно. Для перетворення логічних значень у значення числових типів використовують функції перетворення `int`, `float`. У цьому разі *True* відповідає одиниці, а *False* — нулю як для цілих, так і для дійсних чисел. Наприклад:

```
int(True) == 1           float(False) == 0.0
int(False) == 0         str(True) == 'True'
float(True) == 1.0     str(False) == 'False'
```

У зворотному порядку перетворення працює трохи інакше, але запам'ятати дуже просто: будь-які «непорожні» значення конвертуються в *True*, будь-які «нульові» — у *False*. Для примусового приведення значення до логічного типу використовується вбудована функція `bool`. Наприклад:

```
bool(None) == False
bool(1) == True
bool(-1.1) == True
```



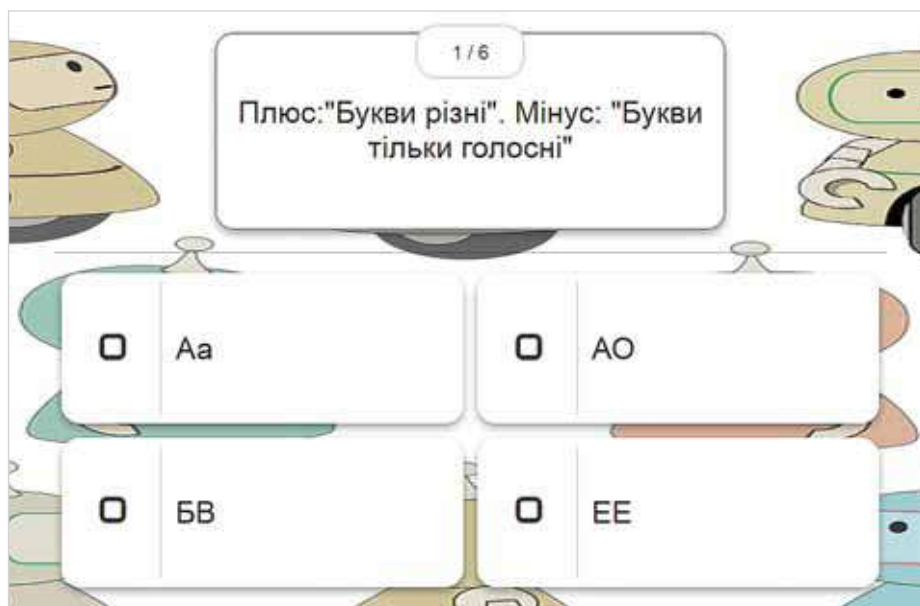
ДІЄМО

Вправа 1. Міркуємо логічно.

Завдання. У вправі за посиланням <https://learningapps.org/view13123820> два роботи допомагають вам виконати завдання. Робот Плюс завжди говорить правду. Робот Мінус — завжди говорить неправду. Виконайте завдання за підказками роботів.



1. Відкрийте браузер та перейдіть за посиланням <https://learningapps.org/view13123820> або QR-кодом.
2. Розгляньте перше завдання (мал. 11.2).



Мал. 11.2

3. Міркуйте так. Нехай твердження робота *Плюс* ми позначимо літерою *A*, а твердження робота *Мінус* — літерою *B*. Складемо таблицю істинності для двох тверджень. Позначимо правдиву відповідь 1, а неправдиву — 0: (табл. 11.3)

Таблиця 11.3

Твердження	<i>A</i>	<i>B</i>	not <i>B</i>	<i>A</i> and (not <i>B</i>)
<i>Aa</i>	0	1	0	0
<i>БВ</i>	1	0	1	1
<i>AO</i>	1	1	0	0
<i>ЕЕ</i>	0	1	0	0

4. Поставте позначку біля відповіді, що відповідає 1 в останньому стовпчику.
5. Виконайте наступне завдання. За запропонованим зразком створіть для нього таблицю істинності. Прослідкуйте, скільки часу витратите на його повне виконання.
6. Виконайте завдання 3 без таблиці істинності. Скільки часу витрачено? Зробіть висновок, чи допомагає таблиця істинності у передбаченні результату.
7. Зручним для вас способом завершіть виконання завдань.

3. Як описати алгоритмічну структуру неповного розгалуження мовою програмування Python?

Прості та складені логічні вирази, значеннями яких є *True* або *False*, використовують в описі алгоритмічної структури розгалуження, за допомогою якої виконавець алгоритму може вибрати один зі сценаріїв подальших дій залежно від виконання певної умови.

Для опису алгоритмічної структури розгалуження мовою програмування, як і в середовищі *Скретч*, використовують *оператор неповного розгалуження* (мал. 11.3). Команда або блок команд в операторі неповного розгалуження буде виконуватись тільки в тому разі, коли логічний вираз набуває значення *True*.

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

Зверніть увагу на знак :
Пропуск 4 символи

Мал. 11.3

Мовою програмування *Python* в операторі неповного розгалуження після логічного виразу записують символ «:», і наступний рядок розпочинають з відступу в 4 символи. Усі команди, що мають такий відступ у програмному коді, входять до блоку команд, які будуть виконуватися в тому разі, якщо умова істинна.

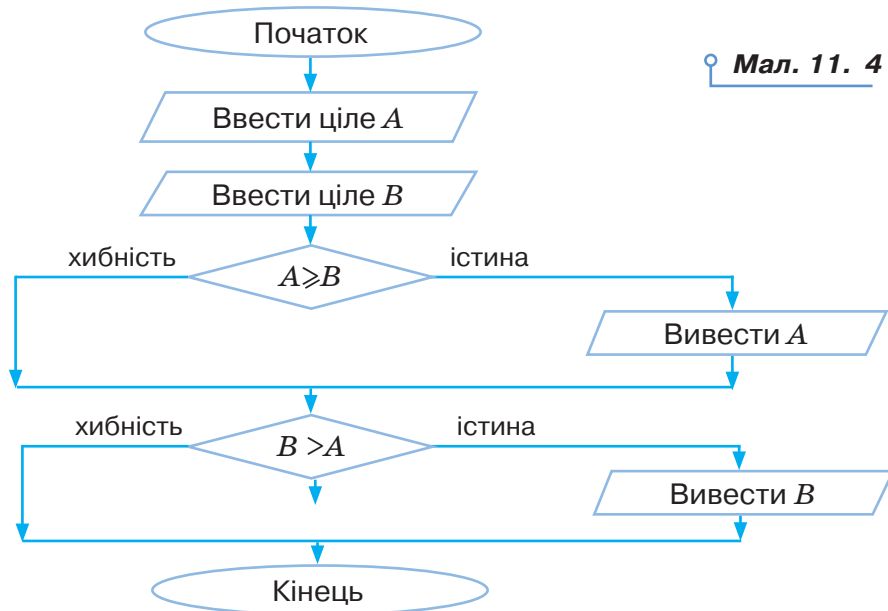


ДІЄМО

Вправа 2. Максимум із двох.

Завдання. Складіть програму мовою *Python*, у якій із двох введених цілих чисел A і B (<1000) виводиться найбільше з них.

1. Розгляньте блок-схему розв'язування завдання (мал. 11.4).



Мал. 11.4

2. Передбачте, які команди мови *Python* відповідають блокам на малюнку 11.4.
3. Відкрийте середовище програмування *Thonny*.
4. Створіть новий файл програми мовою *Python* з іменем *Більше з двох* у папці *Програми* своєї структури папок.

Якщо значення логічного виразу *True*, то виконується блок команд 1. Якщо значення логічного виразу *False*, то виконується блок команд 2.

5. Запишіть програму, використавши сплановані команди. Перевірте, чи є серед них команда:

```
if A>B:
    print (A)
```

6. Запустіть програму на виконання. Доберіть самостійно значення змінних для тестування програми.

4. Як описати алгоритмічну структуру повного розгалуження мовою програмування *Python*?

У задачах, які передбачають виконання альтернативних дій, використовують **оператор повного розгалуження**. Для запису оператора повного розгалуження додатково використовують службове слово *else*:

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

Якщо є потреба врахувати значення декількох логічних виразів, то використовують **вкладені розгалуження**:

```
if <логічний вираз1>:
    блок команд 1
elif <логічний вираз2>:
    блок команд 2
...
elif <логічний вираз n>:
    блок команд n
else:
    <блок команд інакше>
```

В описаних вкладених розгалуженнях, якщо значення логічного виразу 1 *True*, то виконується блок команд 1. Якщо значення логічного виразу 1 *False* і значення логічного виразу 2 *True*, то виконується блок команд 2, і т.д. Інакше виконується команда *інакше* або блок команд *інакше*.

Якщо вкладених умовних операторів декілька, то до якого з них належить *else*, можна зрозуміти за відступом. Відступ у *else* має бути такий само, як в *if*, до якого він належить.

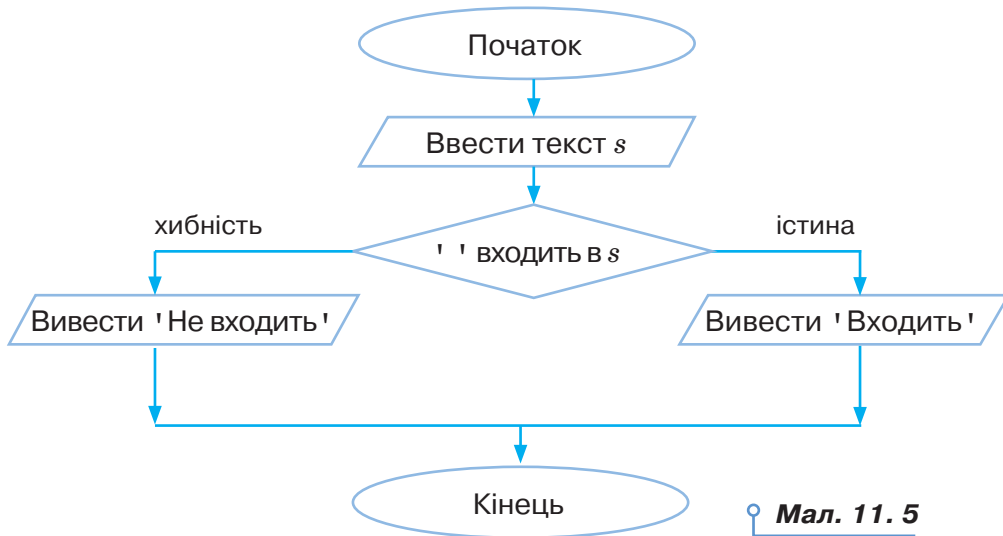


ДІЄМО

Вправа 3. Слово чи декілька.

Завдання. Складіть програму, за допомогою якої можна визначити, чи введений текст складається з одного слова, чи з декількох.

1. Відкрийте середовище програмування *Thonny*.
2. Створіть новий файл програми мовою *Python* з іменем *Слова* в папці *Програми* своєї структури папок.
3. Передбачте, що введене текстове значення буде надано змінній *s*.
4. У програмному коді запишіть команди за поданою блок-схемою (мал. 11.5).



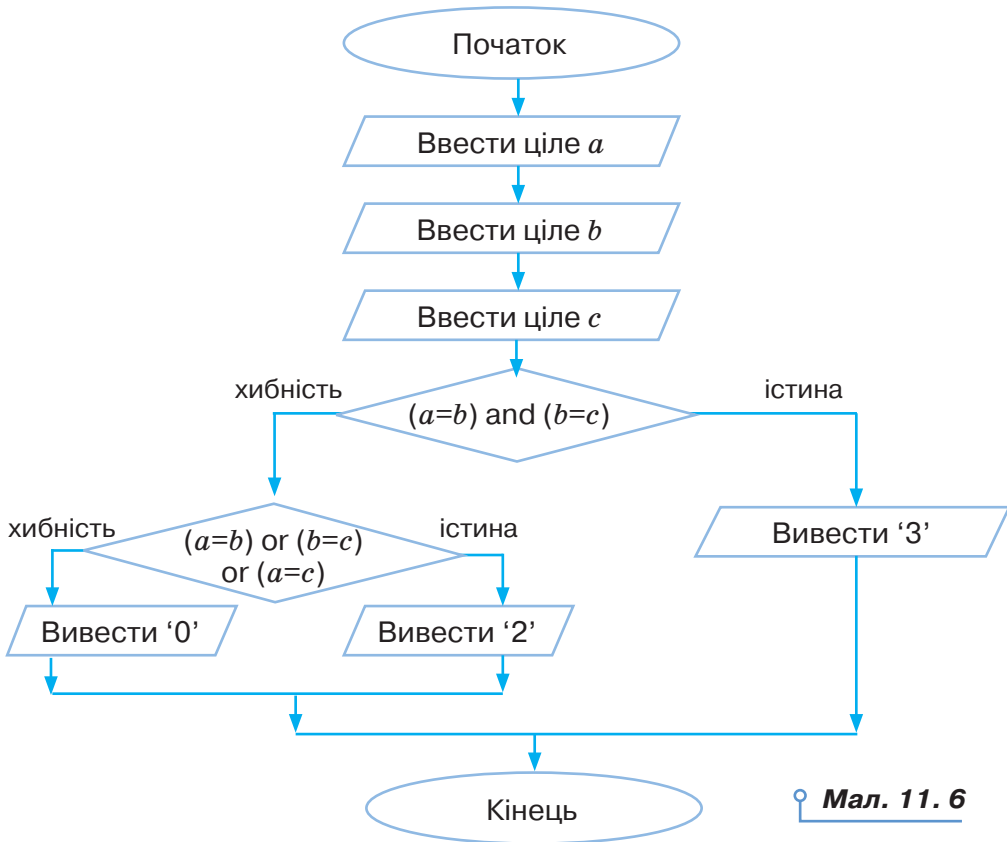
Мал. 11.5

5. Запустіть програму на виконання. Доберіть самостійно значення змінної *s* для тестування програми.

Вправа 4. Кількість однакових чисел.

Завдання. Дано три цілих числа. Визначте, скільки серед них збігаються. Складіть програму для виконання завдання, яка виводить одне із чисел: 3 (якщо всі збігаються), 2 (якщо два збігаються) або 0 (якщо всі числа різні).

1. Відкрийте середовище програмування *Thonny*.
2. Створіть новий файл програми мовою *Python* з іменем *Однакові числа* в папці *Програми* своєї структури папок.
3. Передбачте використання змінних *a*, *b*, *c* для заданих цілих чисел.
4. У програмному коді запишіть команди за поданою блок-схемою (мал. 11.6).



Мал. 11. 6

5. Запустіть програму на виконання. Доберіть самостійно значення змінних для тестування програми.



Обговорюємо

1. У чому особливість опису порівняння величин різних типів мовою програмування *Python*?
2. Як визначити значення логічного виразу, що містить логічні операції *and*, *or* і *not*? Як для пошуку відповіді на це запитання можна використати таблиці істинності?
3. Що спільного й відмінного мають логічні операції *and* і *or*?
4. Які засоби для перетворення логічних даних в інші типи передбачено в мові програмування *Python*?
5. У чому особливість опису повного розгалуження мовою програмування *Python*?
6. Наведіть приклади, коли у програмі варто застосовувати оператор неповного розгалуження.
7. Як мовою програмування *Python* реалізувати вкладені розгалуження?

Працюємо в парах

8. Складіть у парі 5 виразів на порівняння значень величин різних типів мовою програмування *Python*. Запропонуйте їх іншій парі. Перевірте правильність отриманих відповідей.
9. Обговоріть, який оператор розгалуження варто використати у програмі, за допомогою якої описують кількість n горобців, що сидять на гілці ($n < 100$). Наприклад, для $n = 1$ — отримують 1 горобець, $n = 2$ — отримують 2 горобці, $n = 8$ — отримують 8 горобців. Перевірте власні міркування за допомогою створення програми в середовищі програмування.
- ★ 10. Обговоріть, які зміни потрібно внести у програму *Максимум із двох*, щоб упорядкувати три введені числа за зростанням. Побудуйте блок-схему, створіть відповідну програму та протестуйте її в середовищі програмування.
- ★ 11. Обговоріть блок-схему розв'язування задачі для визначення агрегатного стану води, якщо відомо, що за температури $t < 0$ вода перетворюється в лід, $t > 100$ — у пару, інакше — залишається в рідкому стані. Скористайтесь файлом *Вода* з папки *Програмування*. За блок-схемою складіть програму. Обговоріть спільне та відмінне у програмах. Чи може програмний код відрізнятись?
- ★★ 12. Обговоріть блок-схему розв'язування задачі для перетворення введеного значення температури повітря в прийнятій одиниці (відомо, що ступінь Фаренгейта ($^{\circ}\text{F}$) і градус Цельсія ($^{\circ}\text{C}$) — дві основні одиниці вимірювання температури. Шкала Фаренгейта використовується переважно в США, тоді як інші країни, як правило, використовують шкалу Цельсія). Якщо користувач програми вводить значення температури у градусах $^{\circ}\text{C}$ і вказує шкалу F , то в результаті отримує введене число. Якщо вказує шкалу F , то отримує відповідне значення температури, і навпаки. Використовують при цьому такі формули переходу від однієї шкали до іншої:

$$^{\circ}\text{C} = (^{\circ}\text{F} - 32) \times 5/9$$

$$^{\circ}\text{F} = (^{\circ}\text{C} \times 9/5) + 32$$

Працюємо самостійно

13. Визначте, якого логічного значення набуває логічний вираз, записаний мовою програмування *Python*:

1) `False and False`

2) `1 == 1 and 2 == 1`

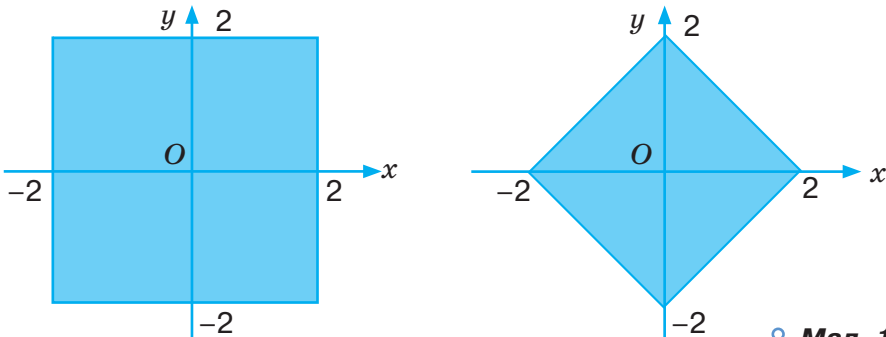
- 3) `'test' == 'testing'` ($1==1$) and ($2==1$)
- 4) `'test' != 'testing'`
- 5) `True and 0 != 0`

14. За допомогою умовного оператора мовою програмування *Python* запишіть фрагмент програмного коду для:

- 1) заміни значення змінної s дійсного типу його абсолютною величиною;
- 2) присвоювання змінній k значення 0, якщо її початкове значення належало інтервалу (0; 5);
- 3) надання змінній t значення *True*, якщо введені координати x і y належать першій або третій координатній чверті.

У завданнях 15–23 складіть блок-схему розв'язування завдання програму мовою *Python* та реалізуйте її в середовищі програмування.

- ★ **15.** Складіть програму *Мішень*, у якій за введеними дійсними числами x і y буде виведено повідомлення *Вітаємо! Влучний постріл!*, якщо точка з координатами x , y належить заштрихованій області, (мал. 11.7) та повідомлення *Шкода, не влучено!* — в іншому разі.



Мал. 11.7

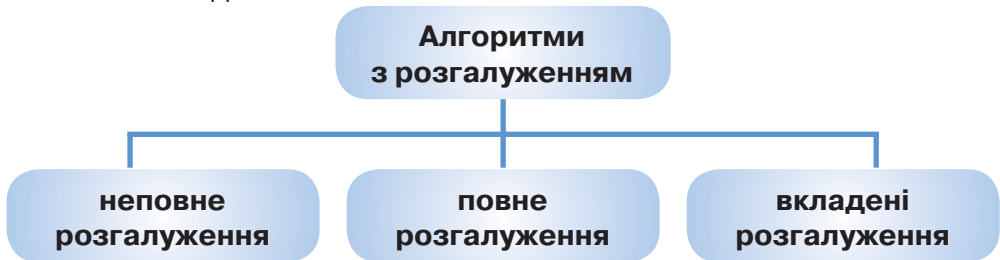
- ★ **16.** Відомо, що лев спить у середньому 18 год на день. Як тільки лев прокидається, він стає дуже голодним. У середовищі програмування розробіть програму *Лев*, щоб сказати власнику/власниці зоопарку, чи лев буде спати, голодний або щасливий. Лев буде:
- сонним, якщо час його сну не досяг 18 год;
 - голодним, якщо час його сну перевищує 18 год і його не годували;
 - щасливим, якщо час його сну перевищує 18 год і його нагодували.
- ★ **17.** Складіть програму *Рік*, у якій за введеним цілим числом року визначають, чи є рік високосним. Якщо рік є високосним, то слід вивести повідомлення YES, інакше — NO. Відомо, що відповідно

до григоріанського календаря рік є високосним, якщо його номер кратний 4, але не кратний 100, а також якщо він кратний 400.

- 18.** Складіть програму *Оцінки*, у якій у текстове поле виводиться опис досягнутого рівня на основі шкільної оцінки:
- 1) 1, 2, 3 — початковий рівень;
 - 2) 4, 5, 6 — середній рівень;
 - 3) 7,8,9 — достатній рівень;
 - 4) 10, 11, 12 — високий рівень.
- 19.** Складіть програму *Правила дорожнього руху*, у якій можна визначити, чи не порушив правила дорожнього руху водій, що проїхав відстань S за час t , якщо на шляху встановлено одне із запропонованих обмежень на швидкість: 40 км/год, 60 км/год, 90 км/год. Передбачте, що вводяться іменовані дані.
- 20.** Відомо таке співвідношення ідеальної ваги та зросту людини за віком: від значення зросту людини в сантиметрах беруться останні дві цифри; якщо вік людини до 25 років, то її вага має становити на 5 кг менше одержаного двоцифрового числа, для людей від 25 до 45 років — дорівнювати цьому двоцифровому числу, а для старших за 45 років — двоцифрове число потрібно збільшити на 5. У середовищі програмування розробіть програму *Ідеальна вага*, у якій користувач вводить свій зріст у сантиметрах і вагу, вказує вік у роках. У результаті виконання програми отримують висновок: вага ідеальна, надлишкова вага, недостатня вага.
- 21.** На сковорідку одночасно можна покласти k котлет. Кожну котлету потрібно з кожного боку обсмажувати m хв безперервно. За який найменший час вдасться підсмажити з обох сторін n котлет? Розробіть програму *Кухар*, яка повідомляє про час обсмаження котлет у хвилинах .

Повторюємо

Розгляньте схему та поясніть, як у мові програмування *Python* реалізують її складові. Наведіть приклади програм, які використовують описані команди.





Реалізація алгоритмів повторення мовою програмування

ВИ ДІЗНАЄТЕСЬ:

- ▶ як мовою програмування *Python* описують цикл із передумовою;
- ▶ як мовою програмування *Python* описують цикл із параметром;
- ▶ як керувати виконанням повторень у мові програмування *Python*.



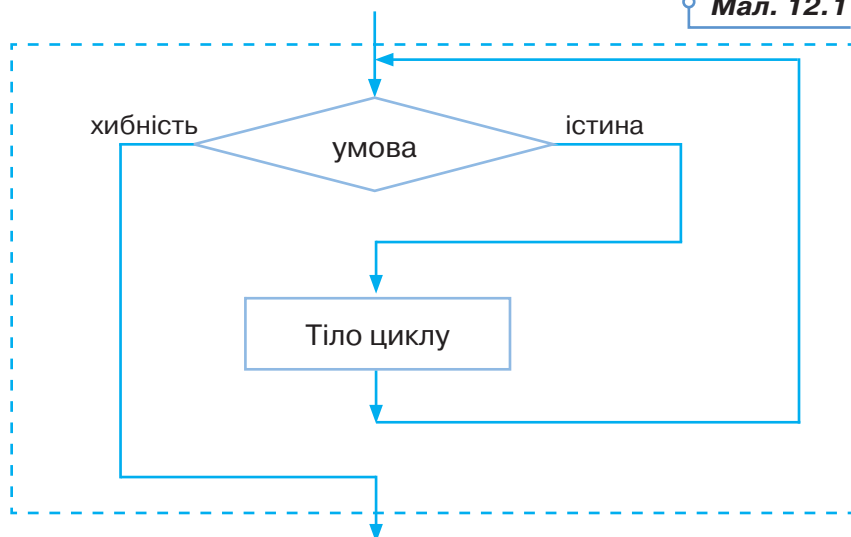
Вивчаємо

1. Як мовою програмування *Python* описують цикл із передумовою?

Ви знаєте, що алгоритм, у якому передбачається багаторазове виконання одного й того самого набору команд, називають **циклічним**. У циклічних алгоритмах використовують алгоритмічну структуру повторення. У мовах програмування для опису структури повторення використовують оператори циклу.

Будь-який оператор циклу складається з двох частин: заголовка та тіла. У заголовку циклу записуються умови, за яких виконання циклу триватиме або завершиться, а в тілі циклу містяться команди, виконання яких потрібно повторювати (мал. 12.1).

Мал. 12.1



У мові програмування *Python* цикл із передумовою описується оператором `while...` (табл. 12.1).

Таблиця 12.1

Опис		Приклад
while <логічний вираз>: <команда>	Скорочена форма	while a<5: a+=1
while <логічний вираз>: <команда1> <команда2> . . . <команда n> else: <команда інакше>	Повна форма	while a<5: s=s+a a+=1 else: print('Сума не змінюється')

Скорочена форма оператора `while` працює таким чином. Визначається значення логічного виразу. Якщо це значення `true`, то виконується команда, що слідує після рядка зі знаком «:» у рядках, які мають відступ 4 позиції. Якщо значення логічного виразу `false`, то цикл завершується й управління передається команді, яка записана у програмному коді після тіла циклу (з рядка, який перебуває на тому ж рівні, що і команда `while`). У тілі циклу обов'язково використовують команду, яка буде змінювати значення величини, що використовується в логічному виразі.

Повна форма оператора `while` мовою програмування *Python*, окрім команд, що входять до блоку `while`, використовує конструкцію `else`: після якої в новому рядку роблять відступ 4 позиції та записують команди, що мають виконуватись, якщо значення логічного виразу `false`.

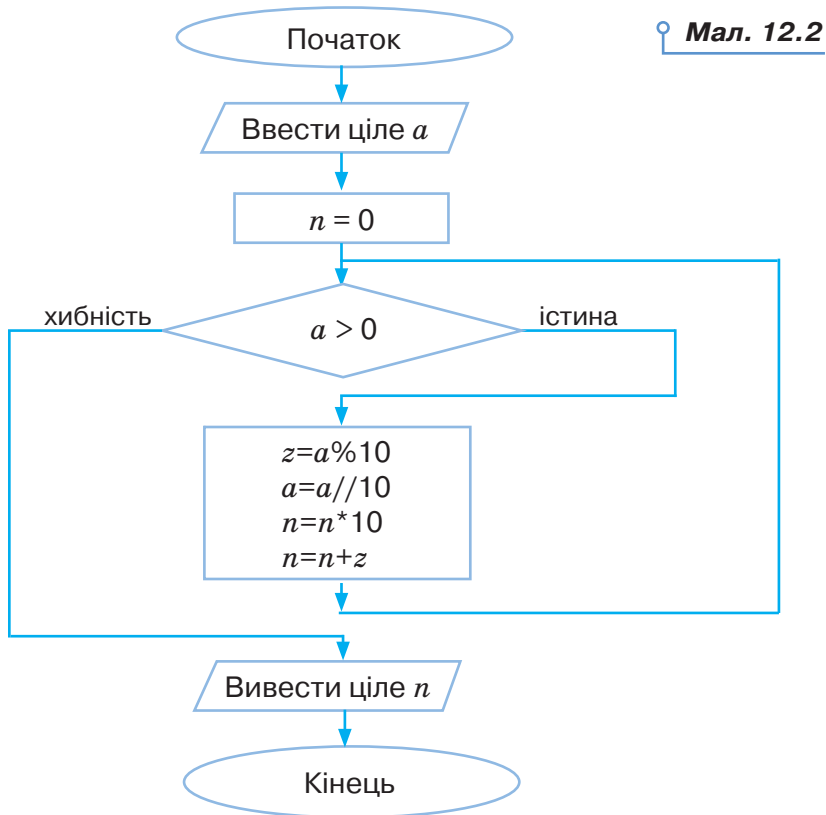


ДІЄМО

Вправа 1. Число навпаки.

Завдання. Складіть програму, яка задане ціле число перетворює на число, записане навпаки. Наприклад, при введенні числа 123 отримуємо 321.

1. Розгляньте анімацію «перетворення» числа за вказаною умовою у файлі *Перетворення*, який збережено в папці *Програмування*.
2. Проаналізуйте, які команди блок-схеми (мал. 12.2) реалізують розглянуту анімацію. Урахуйте, що у блок-схемі використано такі змінні:
 - a — задане число;
 - n — нове число;
 - z — виділена цифра.

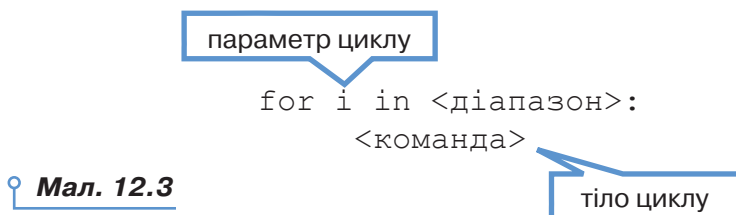


3. Передбачте, які команди мови *Python* відповідають блокам на малюнку 12.2.
4. Відкрийте середовище програмування *Thonny*.
5. Створіть новий файл програми мовою *Python* з іменем *Число навпаки* в папці *Програми* своєї структури папок.
6. Запишіть складену програму. Запустіть програму на виконання. Доберіть самостійно значення змінних для тестування програми.

2. Як мовою програмування *Python* описують цикл із параметром?

Для того щоб описати повторення визначену кількість разів, використовують оператор **циклу з параметром**.

У мові програмування *Python* цикл із параметром описується оператором `for...` (мал. 12.3).



Мал. 12.3

тіло циклу

Цикл `for...` виконується за такими алгоритмом:

1. Параметру циклу `i` присвоюється початкове значення.
2. Якщо значення параметра циклу більше, ніж його кінцеве значення, то цикл завершується. Інакше виконується п. 3.
3. Виконується команда.
4. Значення параметра циклу `i` змінюється на відповідне наступне значення, і здійснюється перехід до п. 2. і т.д.

Наприклад, результатом виконання наступного фрагмента програми:

```
for i in 'hello world':
    print(i * 2, end='')
```

є текст:

```
hheelllloo wwoorrlldd
```

Таким чином, на відміну від оператора циклу `while`, в операторі циклу з параметром зміна значення параметра здійснюється автоматично.

Для визначення діапазону значень параметра у програмах мовою програмування *Python* можна скористатись функцією `range` (табл. 12.3).

Таблиця 12.2

Опис	Значення	Приклад/Результат
<code>range(n)</code>	Послідовні значення цілих чисел від 0 до n (не включаючи)	<code>range(6)</code> 0, 1, 2, 3, 4, 5
<code>range(m, n)</code>	Послідовні значення цілих чисел від m до n (не включаючи)	<code>range(3, 8)</code> 3, 4, 5, 6, 7
<code>range(m, n, k)</code>	Послідовні значення цілих чисел від m до n (не включаючи) з кроком k	<code>range(7, 16, 2)</code> 7, 9, 11, 13, 15

Можна скористатись величиною **перелічуваного типу**. Наприклад, списком непарних чисел першого десятка: 1, 3, 5, 7, 9 — або списком голосних літер українського алфавіту: 'а', 'о', 'у', 'е', 'и', 'і', 'ї', 'ю'. Список значень у мові програмування *Python* обмежують квадратними дужками `[]`.

3. Як керувати виконанням повторень у мові програмування Python?

У мові програмування *Python* можна використовувати достроковий вихід із циклу. За допомогою функції `break` можна перервати виконання команд тіла циклу та перевести управління до команди, яка слідує після блоку команд, що належать до `while` або `for`.

Якщо в операторах циклу `for` або `while` використовують конструкцію `else`, то за таких умов здійснюється перевірка, чи був вихід із циклу здійснений за допомогою команди `break`, або ж «природнім» чином. Блок команд після `else` буде виконаний у тому разі, якщо вихід із циклу відбувся без використання команди `break`.

Оператор `continue` починає наступний прохід циклу, минаючи частину тіла циклу `for` або `while`, що залишилась.

Приклади використання команд керування повторень подано в таблиці 12.3.

Таблиця 12.3

Фрагмент програми	Результат
<pre>for i in 'hello world': if i == 'o': break print (i * 2, end='')</pre>	hheelllll
<pre>for i in 'hello world': if i == 'a': break else: print ('такої літери немає')</pre>	такої літери немає
<pre>for i in 'hello world': if i == 'o': continue print (i * 2, end='')</pre>	hheelllll wwrrlldd




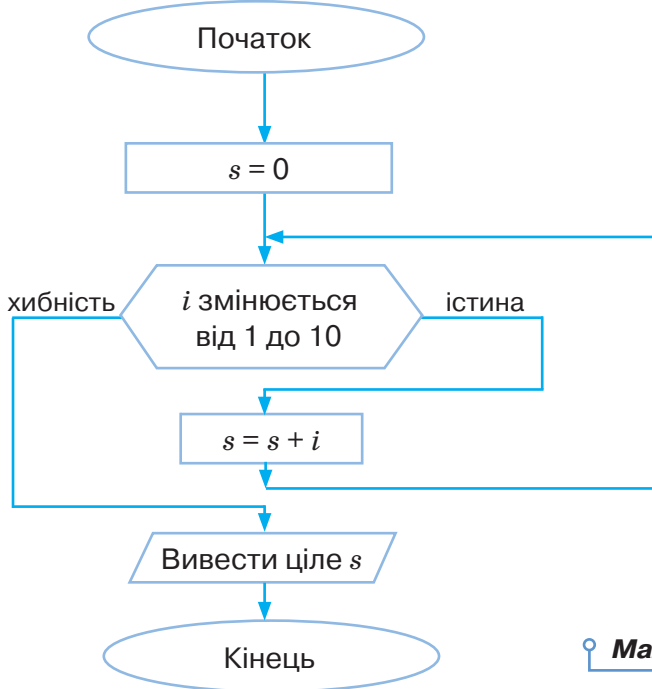
ДІЄМО

Вправа 2. Сума чисел першого десятка.

Завдання. Розробіть програму *Сума чисел*, за допомогою якої можна обчислити значення суми $1 + 2 + 3 + \dots + 10$.

- Проаналізуйте умову задачі та визначте змінні для її розв'язування. Нехай s — змінна цілого типу для обчислення потрібної суми, i — параметр циклу.

2. Розгляньте блок-схему розв'язування завдання (мал. 12.4). Зверніть увагу на блок . Він відповідає команді `for...`



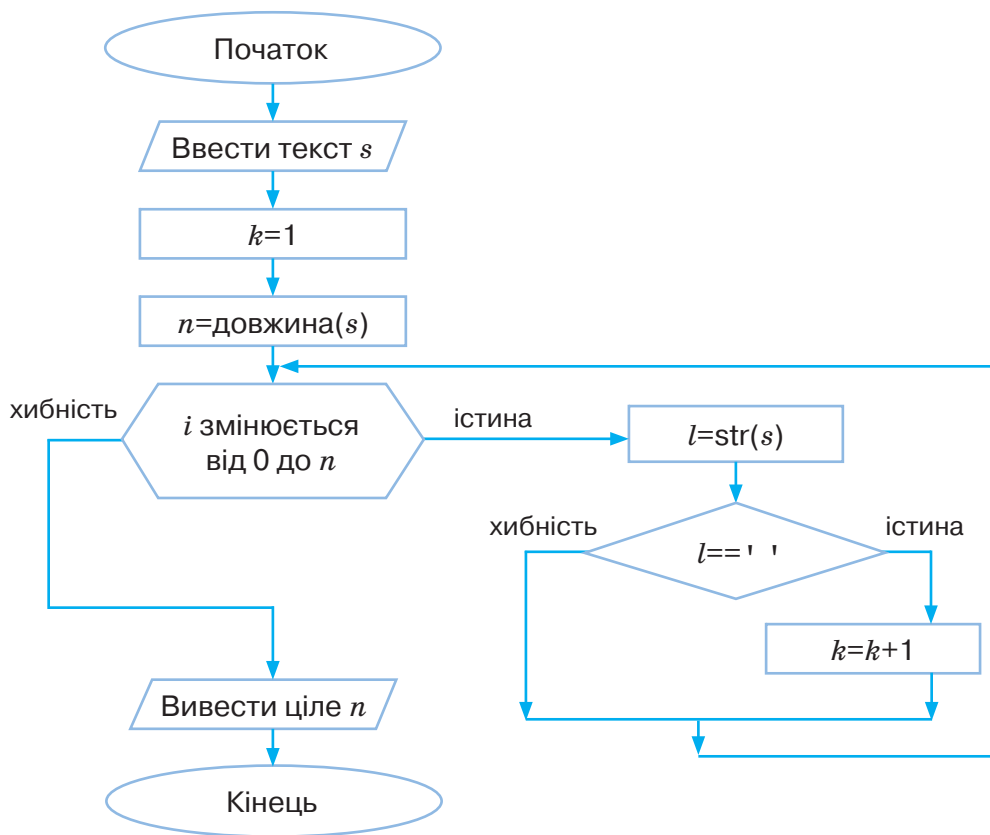
 Мал. 12.4

3. Передбачте, які команди мови *Python* відповідають блокам на малюнку 12.4. Урахуйте, що мовою *Python* команду `s=s+i` можна записати `s+=i`.
4. Відкрийте середовище програмування *Thonny*.
5. Створіть новий файл програми мовою *Python* з іменем *Сума чисел* у папці *Програми* своєї структури папок.
6. Запустіть програму на виконання. Порівняйте отриманий результат із числом 45.

Вправа 3. Кількість слів у реченні.

Завдання. Розробіть програму *Речення*, за допомогою якої можна визначити, скільки слів введено в текстове поле, якщо відомо, що між словами міститься лише один пропуск.

1. Проаналізуйте умову задачі та визначте змінні для її розв'язування. Нехай введене речення буде надано текстовій змінній s , кількість слів — змінній k цілого типу, n — довжина рядка s , l — виділена літера введеного слова.
2. Розгляньте блок-схему розв'язування завдання (мал. 12.5).



Мал. 12.5

3. Передбачте, які команди мови *Python* відповідають блокам на малюнку 12.5.
4. Відкрийте середовище програмування *Thonny*.
5. Створіть новий файл програми мовою *Python* з іменем *Речення* в папці *Програми* свої структури папок.
6. Запустіть програму на виконання. Доберіть самостійно значення змінних для тестування програми.



ДІЄМО

Вправа 4. Гра.

Завдання. Проаналізуйте програму мовою програмування *Python*, яка реалізує гру *Відгадай число* між комп'ютером і користувачем. Користувач не більше ніж за 6 спроб має відгадати «задумане» число з діапазону від 1 до 20. Якщо користувач відгадає, то на екран виводиться повідомлення про кількість спроб, якщо ні — «задумане число». Змініть програмний код так, щоб замість циклу `while` використати цикл `for`.

1. Відкрийте файл з іменем *Відгадай число* в папці *Програмування*.

2. Розгляньте програму. Проаналізуйте, яке призначення кожного рядка програмного коду.
3. Запустіть програму на виконання.
4. Збережіть файл з іменем *Гра* в папці *Програми* своєї структури папок. Внесіть зміни у програмний код так, щоб замість циклу `while` використати цикл `for`.
5. Запустіть програму на виконання. Перевірте, чи впливають внесені зміни на гру.
6. Завершіть роботу із середовищем.



Обговорюємо

1. Які оператори для опису алгоритмічної структури повторення використовують у програмах мовою *Python*?
2. Як в операторах циклу в мові програмування *Python* організують виконання декількох команд у тілі циклу?
3. Що спільного та чим різняться цикли `while` і `for` у мові програмування *Python*?
4. Поясніть, коли припиняється виконання команд у тілі циклу для циклів з передумовою та циклу з параметром. Чи можна вимушено перервати або продовжити виконання циклу?



Працюємо в парах

5. Обговоріть, що буде результатом виконання таких програм. Перевірте свої здогадки в середовищі *Thonny*.

```
1) i = 1
   for color in 'red', 'orange', 'yellow', 'green',
   'cyan', 'blue', 'violet':
       print('#', i, ' color of rainbow is ', color,
       sep = '')
       i += 1
```

```
2) for i in 1, 2, 3, 'one', 'two', 'three':
   print(i)
```

★ 3)

```
for i in range(4):
    print(i)
    print(i ** 2)
print('Кінець циклу')
```

★ 4)

```
sum = 0
n = 5
for i in range(1, n + 1):
    sum += i
print(sum)
```

★ 5)

```
i = 1
while i <= 10:
    print(i ** 2)
    i += 1
```

6. У програмах описаних завдань автори припустились деяких помилок. Відшукайте помилки. Обговоріть їх у парі. Запропонуйте одне одному шляхи вирішення.

★ 1) Дано два цілих числа a і b ($a \leq b$). Виведіть усі числа від a до b включно.

```
a=int(input('Введіть перше число'))
b=int(input('Введіть друге число'))
for i in range(a,b):
    print(i)
```

★ 2) Дано натуральне число n . Знайдіть $n!$ (добуток всіх натуральних чисел, що не перевищує n , тобто $1*2*3*...*n$).

```
n=int(input())
p=1
for i in range(1,n+1):
    p*=i
print(p)
```



3) У перший день спортсмен пробіг x кілометрів, а потім він кожного дня збільшував пробіг на 10 % від попереднього значення. За даним числом y визначте номер дня, за який пробіг спортсмена становитиме не менше y кілометрів.

```
x = int(input())
y = int(input())
z = 1
while x < y:
    x = x * 1,1
    z = z + 1
print(z)
```



Працюємо самостійно

7. Яких значень буде надано змінним, імена яких трапляються в командах, у результаті виконання таких фрагментів програм (для різних значень вхідних даних):

1) $i:=1; x:=0;$

```
while (I<n) and (y='ТАК'):
    x:=x+1;
    i:=i+1
```

при 1) $n=3; y='ТАК'$

2) $n=3; y='НЕМАЄ'$

3) $n=1; y='ТАК'$

4) $n=1; y='НЕМАЄ'$

5) $n=5; y='ТАК'$

2) $\text{for } i \text{ in range}(1, 5):$

```
    for j in range(1, 5):
```

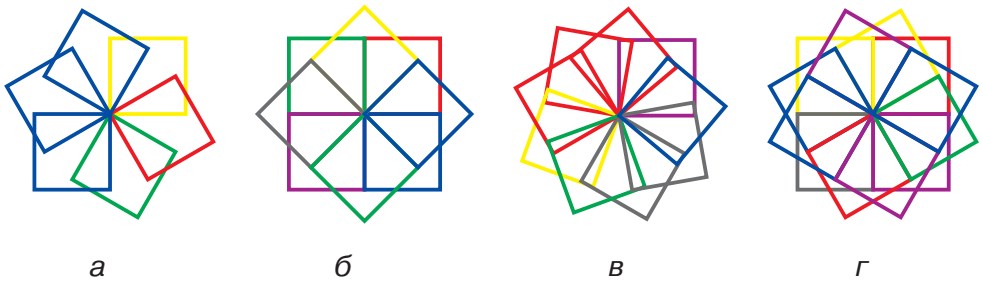
```
        r= i*j
```

```
        print(i+ ' ', що множиться на '+j+'
            'дорівнює'+r);
```

- ★ 8. Складіть блок-схему та відповідну програму мовою *Python* для виконання таких дій:
- 1) друкується таблиця множення на 9;
 - 2) друкуються квадрати чисел від 10 до 1;
 - 3) знаходиться сума цілих чисел від 10 до 100;
 - 4) обчислюється добуток парних чисел першого десятка;
 - 5) обчислюється сума дробів, знаменниками яких є непарні числа від 3 до 9;
 - 6) визначається кількість введених речень (речення закінчується крапкою, знаком оклику чи знаком запитання);
 - 7) визначається середнє значення всіх елементів послідовності, яка завершується числом 0.

У завданнях 9–12 складіть блок-схему виконання завдання та розробіть відповідну програму в середовищі програмування мовою *Python*.

- ★ 9. Програму *НСД*, у якій для двох цілих чисел визначають їх найменший спільний дільник.
- ★ 10. Програму *Дії*, у якій у введеному виразі, що містить арифметичні операції без дужок і математичних функцій, у повідомленні буде виводитись, які дії потрібно виконати.
- ★ 11. Програму *Фрагмент*, у якій з тексту, що вводиться з клавіатури, буде виводитись та його частина, що розташована до першої крапки.
- ★ 12. Програму *Малюнки*, у якій відбувається малювання таких фігур (колір задається випадково зі списку: 'blue', 'red', 'yellow', 'green', 'gray', 'brown', 'purple') (мал. 12.6).



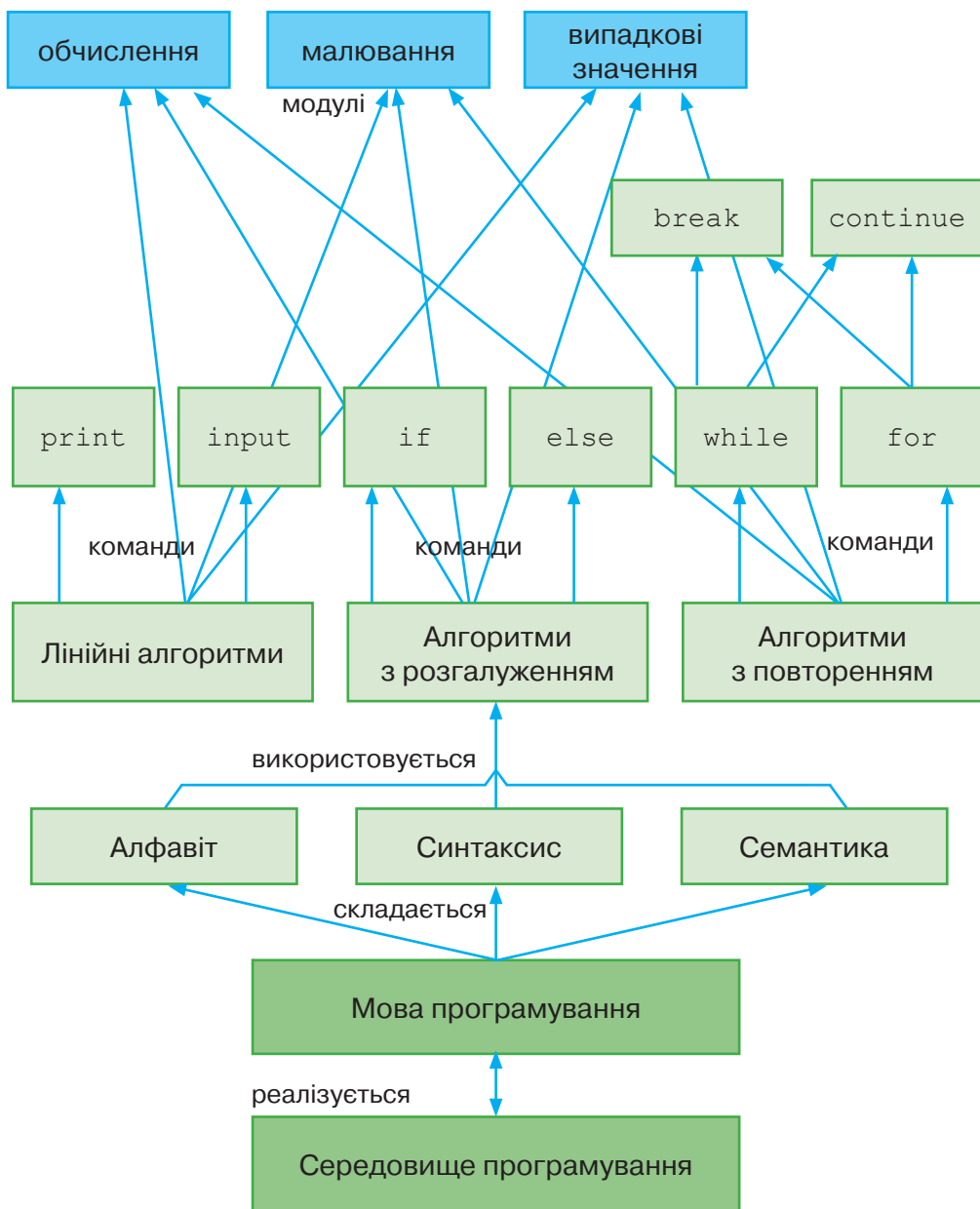
Мал. 12.6

Повторюємо

Поясніть, як у мові програмування *Python* реалізують команди повторення.



Узагальнюємо



Опрацювання табличних даних

13



Електронні таблиці та їхні об'єкти

ВИ ДІЗНАЄТЕСЬ:

- ▶ які програми використовують для опрацювання даних в електронних таблицях;
- ▶ з якими основними об'єктами можна працювати в середовищі табличного процесора;
- ▶ як переміщуватись по електронній таблиці;
- ▶ що відбувається при введенні нових даних до клітинки.



Вивчаємо

1. Які програми використовують для опрацювання даних в електронних таблицях?

Для опрацювання числових даних використовують табличні процесори. Ними користуються представники багатьох професій, наприклад, продавці в магазинах, бухгалтери, банкіри тощо.

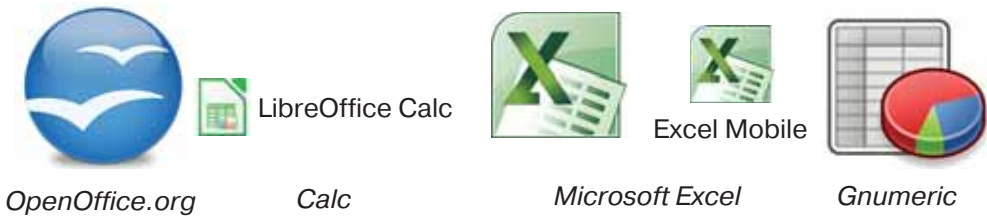
За допомогою табличних процесорів можна швидко опрацювати дані, подані за допомогою електронних таблиць.

Електронна таблиця (англ. *spreadsheet*) — це комп'ютерний варіант звичайної таблиці. Вона складається з рядків і стовпців, на перетині яких розташовуються клітинки. У клітинках електронної таблиці можуть зберігатись текстові, числові дані та формули, за допомогою яких виконуються розрахунки. Електронна таблиця може містити також зображення та інші мультимедійні об'єкти.

Особливістю електронних таблиць є автоматизація обчислень: при зміні даних, що містяться в ній, автоматично оновлюються і результати, які одержуються за відповідними формулами, що використовують ці дані. Крім того, в електронних таблицях можна швидко знаходити необхідні дані за заданими критеріями, будувати діаграми для наочного подання даних тощо.

Для розв'язування задач із даними, які можна подати у вигляді електронних таблиць, розроблено спеціальні пакети програм, що називають **табличними процесорами**.

Для роботи з електронними таблицями існують різні табличні процесори: *Lotus 1-2-3, Multiplan, QuattroPro, LibreOffice Calc, Microsoft Excel, OpenOffice.org Calc* та інші. Кожна програма має подібні функції та відповідний значок (мал. 13.1).



Мал. 13.1

Далі будемо розглядати приклади опрацювання даних у середовищі двох програм: *Microsoft Excel* і *LibreOffice Calc*, які часто використовують на практиці.

Файли, створені за допомогою табличного процесора *Microsoft Excel*, мають розширення *xls,xlsx*, а *LibreOffice Calc* — розширення *ods*.

Запустити табличний процесор можна різними способами:

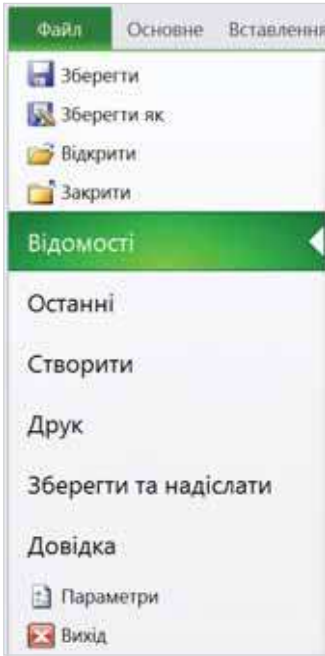
- обрати у списку програм *Головного меню*;



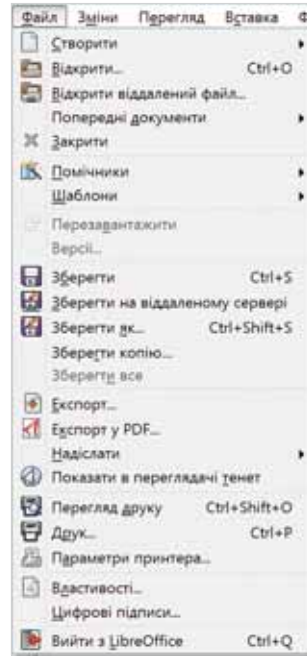
- за допомогою ярлика на *Робочому столі* ;
- відкривши документ, що має один зі значків:



Електронні таблиці зберігаються у файлах, створених у середовищі табличного процесора. З ними можна виконувати такі само дії, як і при опрацюванні документів у середовищі інших програм, що належать до відповідного пакета. Спільним є створення, відкриття, збереження документа та деякі інші дії, що виконуються за допомогою вказівок з меню вкладки *Файл* пакета *Microsoft Office* (мал. 13.2, а) або меню *Файл* пакета *LibreOffice* (мал. 13.2, б).



Мал. 13.2, а



Мал. 13.2, б

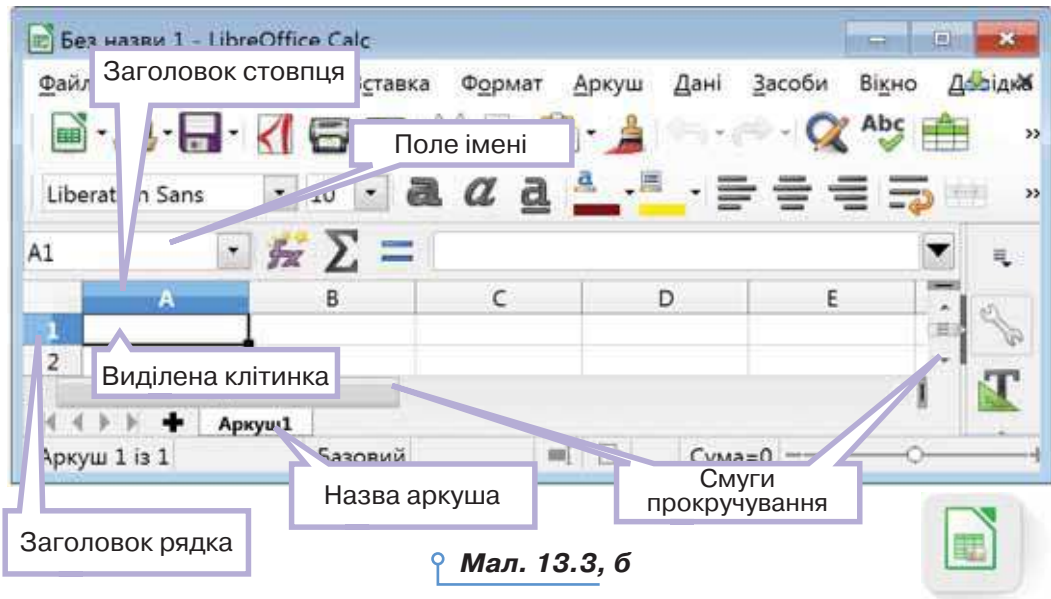
Вікно табличного процесора має елементи, схожі з елементами вікна редактора презентацій і текстового процесора. Але має і особливі елементи: робоче поле вікна табличного процесора (мал. 13.3, а і б) містить у верхній частині **заголовки стовпців**, які позначаються латинськими літерами *A, B, C...*, і ліворуч — **заголовки рядків**, що позначаються цифрами *1, 2, 3...*

Мал. 13.3, а

Компанія	Продаж у 2018р. (млн. штук)
Samsung	71,6
Huawei	54,2
Apple	41,3
Xiaomi	33
OpPO	29,6
vivo	26
LG	10,2
Інші	10,2

3D pie chart data (млн. штук):

- Samsung: 71,6
- Huawei: 54,2
- Apple: 41,3
- Xiaomi: 33
- OpPO: 29,6
- vivo: 26
- LG: 10,2
- Інші: 10,2



Мал. 13.3, б

2. З якими основними об'єктами можна працювати в середовищі табличного процесора?

Електронна таблиця може складатись із декількох аркушів, кожний з яких поділений на рядки та стовпці.

На перетині рядків і стовпців розташовані клітинки, до яких можна вводити дані. Кожна клітинка має свою адресу. Наприклад, клітинка, що розташована на перетині рядка 1 і стовпця А, має адресу А1.



Це цікаво

Деякі об'єкти в різних табличних процесорах можуть бути подані синонімами. Наприклад, у табличному процесорі *Microsoft Excel* вам трапляються назви **клітинка** та **стовпець**, а в LibreOffice Calc — відповідно **комірка** та **стовпчик**. Електронна таблиця, створена за допомогою табличного процесора *Microsoft Excel*, називається **книгою**.

При роботі з електронною таблицею одна з клітинок є виділеною. Навколо виділеної клітинки з'являється рамка, яка відрізняється від обрамлення клітинки. Щоб виділити потрібну клітинку, досить клацнути на ній лівою кнопкою миші. Адреса виділеної клітинки відображається в полі імені (мал. 13.3, а і б).

На кожному аркуші електронної таблиці, крім даних, розміщених у клітинках, можуть розміщуватись діаграми, зображення або інші об'єкти (мал. 13.3, а).

Кожний об'єкт електронної таблиці має ім'я та набір властивостей. Визначення і зміна значень властивостей у табличному процесорі


відбувається так само, як і в середовищі інших програм з відповідного пакета: *Microsoft Office* або *LibreOffice*. Наприклад, для визначення чи зміни значень властивостей об'єктів можна використовувати вказівки контекстного меню.

3. Як переміщуватись по електронній таблиці?

Для переміщення в межах електронної таблиці можна використовувати клавіші управління курсором чи виділяти потрібну клітинку клацанням миші.

Якщо в робочому полі вікна табличного процесора не відображається потрібна клітинка чи інший об'єкт, переглянути їх можна за допомогою горизонтальної або вертикальної смуг прокручування (мал. 13.3, а і б).

У нижній частині робочого поля розміщені вкладки з назвами аркушів, за допомогою яких можна переходити від одного аркуша

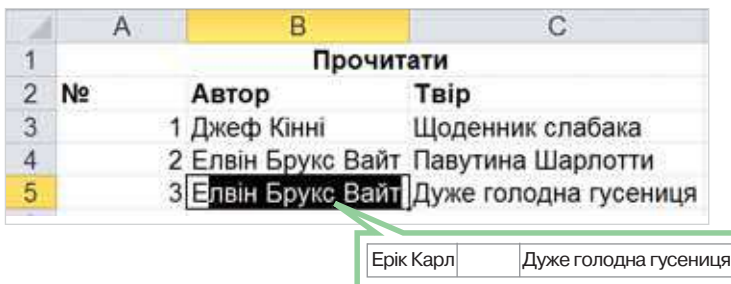
до іншого, та кнопки , за допомогою яких можна відображати вкладки аркушів, якщо не всі вони можуть розміститись у відведеній області для відображення на екрані.

4. Що відбувається при введенні нових даних до клітинки?

Внесення змін до клітинок з даними може спричинити зміни в інших об'єктах, наприклад, зміни в діаграмах або клітинках, що містять формули.

Для зміни вмісту виділеної клітинки можна видалити дані, що містяться у клітинці, та ввести з клавіатури нові. Для завершення введення даних у клітинку треба натиснути клавішу *Enter* або одну з клавіш зі стрілками. При цьому виділеною стане одна із сусідніх клітинок. Для введення тексту можна використовувати як символи латинського алфавіту, так і символи кирилиці. Якщо дані, які вводяться в клітинку, складаються лише із цифр, вони розглядаються як число.

Табличний процесор завершить введення текстових записів у стовпці даних, якщо введені перші букви збігаються з наявним записом у цьому стовпці, при цьому можна скористатись автозавершенням введення тексту — натиснути клавішу *Enter*. Якщо збіг у подальшому не передбачений, то можна продовжувати введення потрібного тексту (мал. 13.4)



	A	B	C
1		Прочитати	
2	№	Автор	Твір
3	1	Джеф Кінні	Щоденник слабака
4	2	Елвін Брукс Вайт	Павутина Шарлотти
5	3	Елвін Брукс Вайт	Дуже голодна гусениця

Ерік Карл	Дуже голодна гусениця
-----------	-----------------------

 Мал. 13.4



ДІЄМО

Вправа 1. Продаж техніки.

Завдання. Визначте призначення інструментів, розміщених на різних вкладках вікна табличного процесора та на рядку стану, інструментів навігації та панелі швидкого доступу при роботі з файлом *Продаж.xlsx* (*Продаж.ods*), що зберігається в папці *Електронні таблиці*. Дослідіть вміст контекстних меню таких об'єктів електронної таблиці: клітинки, стовпця, рядка, аркуша, діаграми.

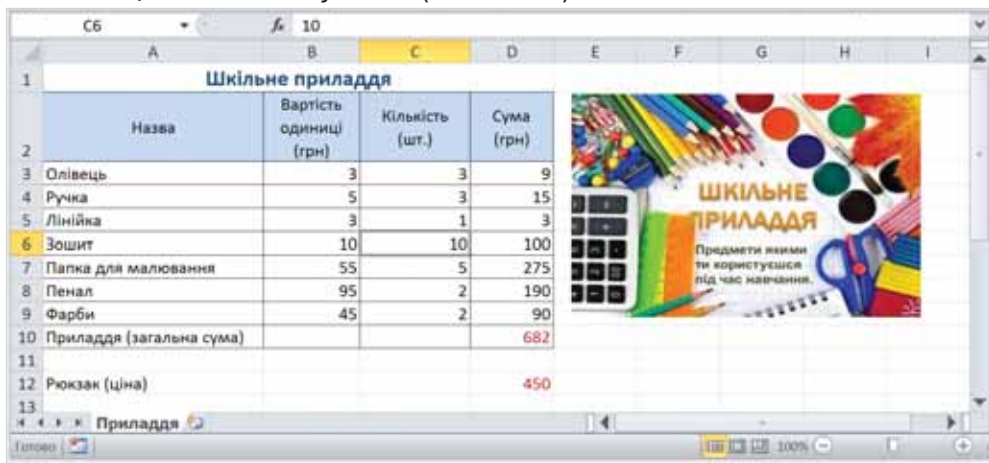
1. Відкрийте файл *Продаж.xlsx* (*Продаж.ods*) (мал. 13.3, а) у папці *Електронні таблиці* на *Робочому столі*.
2. Знайдіть на екрані елементи вікна табличного процесора, спільні для програм пакета *Microsoft Office* (*LibreOffice*): вкладку *Файл* (меню *Файл*), панель швидкого доступу, заголовок програми, кнопки управління вікном, вкладки або панель для розміщення груп інструментів, робочу область програми, рядок стану, смуги прокручування.
3. Знайдіть на екрані елементи вікна табличного процесора: заголовки рядків і стовпців, поле імені.
4. Визначте, які дії можуть виконуватись із клітинкою, стовпцем, рядком, аркушем, діаграмою. Для цього викличте контекстне меню клітинки *A1*, заголовка стовпця *A*, заголовка рядка *1*.
5. Викличте контекстне меню вкладки з назвою одного з аркушів, наприклад, *Телефони* або *Діаграми*. Дізнайтеся, чи є спільні групи вказівок для кожного з них. Кожна нова книга *Microsoft Excel* за замовчуванням містить три аркуші з іменами *Аркуш1*, *Аркуш2* і *Аркуш3*, а в *LibreOffice Calc* — *Аркуш1* (мал. 13.3, б).
Надання аркушу більш змістовного імені, як і надання імен файлів, допомагає описати вміст кожного з аркушів (мал. 13.3, а).
6. Виділіть клітинку *B4* і замість значення *54,2* введіть із клавіатури значення *65*. Зверніть увагу, як змінився вигляд діаграми.
7. Визначте, чи працювали нещодавно з електронними таблицями на вашому комп'ютері, якщо так — то з якими? Для цього скористайтесь переліком документів, що розміщені в меню вкладки *Файл/Останні* (*Microsoft Office*) або в меню *Файл/Попередні документи* (*LibreOffice*).
8. Закрийте документ, не зберігаючи змін.

Вправа 2. Шкільне приладдя.

Завдання. Визначте вартість покупки шкільного приладдя, вносячи зміни в їхню кількість. Сформуйте таку покупку приладдя, вартість якої менша від вартості шкільного рюкзака.

1. Створіть папку *Таблиці* у своїй структурі папок.

2. Відкрийте файл *Приладдя.xlsx* (*Приладдя.ods*) у папці *Електронні таблиці* на *Робочому столі* (мал. 13.5).



Назва	Вартість одиниці (грн)	Кількість (шт.)	Сума (грн)
Олівець	3	3	9
Ручка	5	3	15
Лінійка	3	1	3
Зошит	10	10	100
Папка для малювання	55	5	275
Пенал	95	2	190
Фарби	45	2	90
Приладдя (загальна сума)			682
Рюкзак (ціна)			450

Мал. 13.5

- Визначте адреси клітинок, у яких міститься текст, число. Для цього виділіть клітинку та перегляньте в полі імені її адресу. Наприклад, у клітинці *C6* міститься число, а в *A6* — текст.
- Переконайтеся, що при виділенні зображення в полі імені відображається ім'я *Рисунок 1*.
- Виділіть клітинку *C6* і замість значення *7* введіть із клавіатури значення *10*. Зверніть увагу, як змінився вміст клітинок *D6* і *D10*.
- Подумайте, яке приладдя вам потрібне для навчання і в якій кількості. Змінійте значення клітинок стовпця *C* таблиці *Шкільне приладдя* так, щоб загальна сума покупки була меншою від вартості рюкзака. Для цього порівняйте вміст клітинок *D10* і *D12*.
- Результати роботи збережіть з тим само іменем у папці *Таблиці* своєї структури папок.



Досліджуємо

Вправа 3. Довідкова система.

Завдання. За допомогою довідки операційної системи з'ясуйте, як знаходити на комп'ютері файли з електронними таблицями. Знайдіть на комп'ютері, за яким працюєте, такі файли. Порівняйте свої результати з результатами, одержаними однокласниками й однокласницями.



Обговорюємо

- Поясніть, що таке електронна таблиця. Чим електронні таблиці відрізняються від друкованих, наприклад, табеля успішності?



Працюємо самостійно

- 11.** Сплануйте і створіть презентацію про розробників першого табличного процесора, його об'єкти й особливості роботи. Знайдіть в інтернеті потрібні відомості.
- ★ **12.** Розподіліть вказані дії у групи:
- виділити клітинку *D5*;
 - перейти до аркуша з іменем *Аркуш2*;
 - клацнути на клітинці *D5*;
 - із клавіатури ввести значення *40*;
 - клацнути мишею на вкладці *Аркуш2*;
 - клацнути мишею на перетині стовпця *D* і рядка *5*.
- Придумайте назву для кожної групи.
- ★ **13.** Укажіть, люди яких професій могли б застосовувати електронну таблицю, подану на малюнку 13.7.

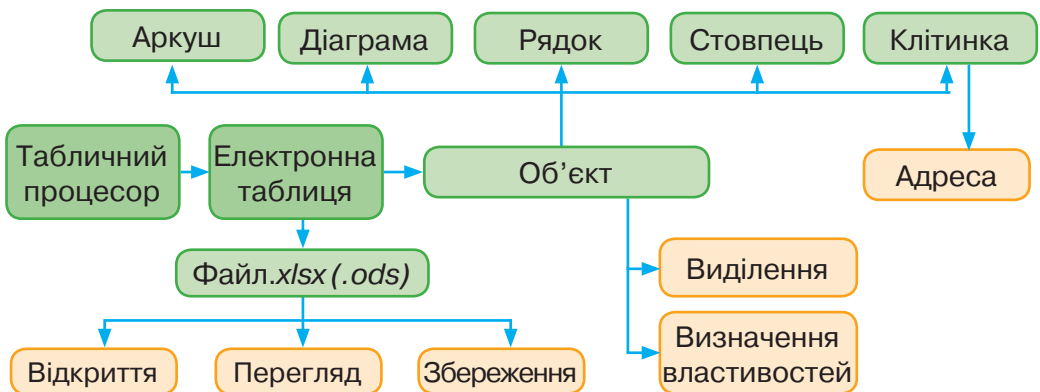
Прайс-лист на навчальні посібники							
№	Номенклатурна назва	Автор	Клас	Номенклатурна група	Мова	Стандарт улаховки	Ціна
Початкова школа 1-4 класи							
1	Перевірка предметних компетентностей, Українська мова, 2 кл.	Пономарьова К.І.	2 кл.	Зошит для перевірки навчальних досягнень	Укр.	30	26,00
2	Перевірка предметних компетентностей, Українська мова, 3 кл.	Пономарьова К.І.	3 кл.	Зошит для перевірки навчальних досягнень	Укр.	30	26,00
3	Перевірка предметних компетентностей, Українська мова, 4 кл.	Пономарьова К.І.	4 кл.	Зошит для перевірки навчальних досягнень	Укр.	30	26,00
4	Перевірка предметних компетентностей, Математика, 2 кл.	Листопад Н.П.	2 кл.	Зошит для перевірки навчальних досягнень	Укр.	30	26,00
5	Перевірка предметних компетентностей, Математика, 3 кл.	Листопад Н.П.	3 кл.	Зошит для перевірки навчальних досягнень	Укр.	30	26,00

Мал. 13.7

Повторюємо



Розгляньте схему та поясніть зв'язок між вказаними поняттями.





Редагування електронних таблиць

ВИ ДІЗНАЄТЕСЬ:

- ▶ як працювати з електронною таблицею;
- ▶ як застосовувати засіб автозаповнення для введення даних;
- ▶ як копіювати та переміщувати клітинки й діапазони клітинок;
- ▶ які особливості редагування структури таблиці.



Вивчаємо

1. Як працювати з електронною таблицею?

У табличному процесорі, як і в текстовому, можна створювати новий файл або опрацювати створений раніше.

Щоб створити новий файл чи відкрити вже існуючий, використовують вказівки з меню вкладки *Файл* табличного процесора *Microsoft Excel* або меню *Файл LibreOffice Calc*.

При створенні нової таблиці доцільно дотримуватися таких рекомендацій (мал. 14.1):

The screenshot shows an Excel spreadsheet with a table titled "КРАЇНИ". The table has 5 columns: "№", "Країна", "Площа, тис. кв. км.", "Населення, млн. чол.", and "Столиця". The rows are numbered 1 to 8. Callouts point to various elements: "Заголовок рядків" points to row 2; "Заголовок стовпців" points to column 3; "Заголовок таблиці" points to the title "КРАЇНИ"; "Виділений діапазон" points to the range A3:B8; "Вирівнювання чисел" points to the numerical values in column 3; "Вирівнювання тексту" points to the text values in column 5.

КРАЇНИ				
№	Країна	Площа, тис. кв. км.	Населення, млн. чол.	Столиця
1	ФРН	356	80	Берлін
2	Франція	552	56,5	Париж
3	Великобританія	244	57	Лондон
4	Ірландія	70	3,5	Дублін
5	Нідерланди	41	15	Амстердам
6	Бельгія	31	10	Брюссель

Мал. 14.1

1) заповнення даних починати з лівого верхнього кута — із клітинки *A1*;

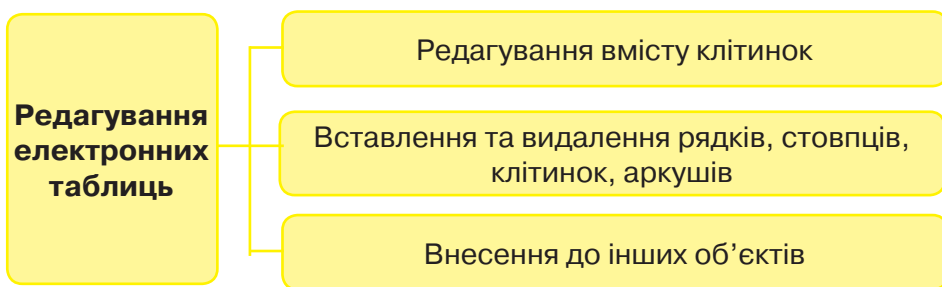
2) заголовок таблиці вводити в першу зліва клітинку рядка, розташованого над таблицею даних;

3) відводити перший лівий стовпець для заголовків рядків, а рядок після заголовка таблиці — для заголовків стовпців.

До клітинок таблиці можна вводити дані різного типу: числа, текст, дати тощо. При цьому автоматично числа вирівнюються за правою межею клітинки, текст — за лівою (мал. 14.1). Іноді потрібно, щоб вміст клітинки, який складається лише із цифр, сприймався як текст, а не як число. Наприклад, поштовий індекс 01054 у форматі числа буде перетворений на 1054. У такому разі перед введенням необхідних символів вводять знак апосторфа «'». У клітинці він не відобразатиметься, а вміст клітинки буде вирівнюватися за лівою межею — як текст.

При опрацюванні даних таблиці може виникнути потреба в її редагуванні. Редагування таблиці передбачає внесення змін до вмісту клітинок і структури таблиці (мал. 14.2).

 **Мал. 14.2**



Редагування електронної таблиці — внесення змін до вмісту та структури таблиці: заміна, видалення, переміщення, копіювання, вставлення даних, рядків і стовпців, а також внесення змін до інших об'єктів.

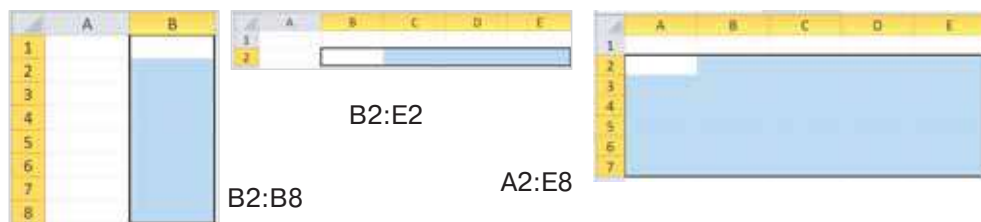
Можна вносити зміни як до окремих клітинок, так і до діапазону клітинок. Для цього має бути виділеним діапазон клітинок.

Діапазон клітинок — деяка сукупність клітинок аркуша електронної таблиці.

Виділити прямокутний діапазон клітинок можна протягуванням при натисненій лівій кнопці миші від однієї кутової клітинки діапазону до протилежної. Щоб записати такий діапазон, треба вказати адреси лівої верхньої та правої нижньої клітинок, які розділяються двокрапкою, наприклад, *B2:E2* або *A2:E7*. Щоб виділити несуміжні

клітинки, виділяють декілька прямокутних діапазонів чи клітинок, утримуючи при цьому клавішу *Ctrl*. У записі такого діапазону всі несуміжні діапазони відділяють між собою знаком «;».

При виділенні більш ніж однієї клітинки в табличному процесорі усі клітинки, крім першої, будуть відображені деяким кольором, наприклад блакитним (мал. 14.3).



Мал. 14.3

Щоб зняти виділення з діапазону клітинок, треба клацнути мишею на будь-якій клітинці, що перебуває за межами цього діапазону.

2. Як застосовувати засіб автозаповнення для введення даних?

Якщо потрібно ввести дані, що послідовно змінюються, або створити список (порядкові числа, дні тижня, місяці року), то можна скористатися засобом автозаповнення. Для цього потрібно:

- 1) ввести одне або два перші значення;
- 2) виділити клітинку або діапазон клітинок (мал. 14.4);
- 3) навести вказівник миші на маркер автозаповнення — чорний квадратик, розташований у правому нижньому куті виділеної клітинки чи діапазону клітинок;
- 5) виконати протягування мишею на потрібний для заповнення даними діапазон клітинок. Протягування можна здійснювати як уздовж рядків, так і стовпців таблиці.

	A	B	C	D
		Розклад уроків у 7 -А класі		
2	Урок	Понеділок	Вівторок	Середа
3	1	Укр. мова	Геометрія	Фізика
4	2	Алгебра	Фізична культ.	Алгебра
5		Історія	Інформатика	Англ. мова
6		Зар. література	Англ. мова	Укр. мова
7		Фізика	ОБЖ	Труд. навч.
8		Музичне мист.	Укр. література	Труд. навч.

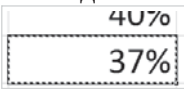
Маркер автозаповнення

Мал. 14.4

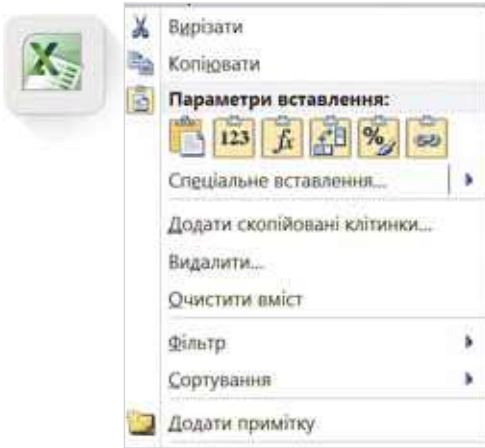
Наприклад, якщо потрібно продублювати дані (*Труд. навч.*), слід виділити клітинку *D7* та виконати протягування за маркер автозаповнення; для введення назв днів тижня достатньо ввести до однієї з клітинок *понеділок* (або будь-який інший день) і використати автозаповнення; автоматизувати введення порядкових номерів можна, якщо ввести два перші значення, виділити відповідні дві клітинки та використати засіб автозаповнення (мал. 14.4).

3. Як копіювати та переміщувати клітинки й діапазони клітинок?

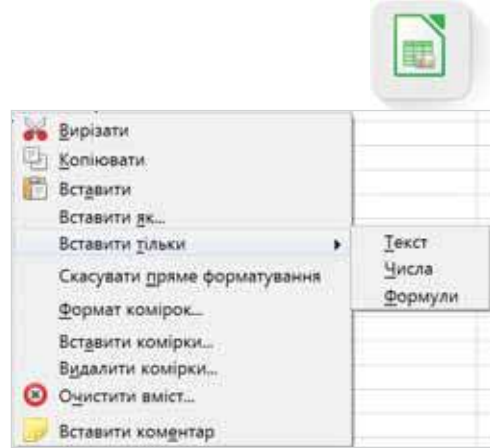
Для копіювання чи переміщення даних можна скористатися вказівками *Вирізати*, *Копіювати*, *Вставити* контекстного меню виділеної клітинки чи діапазону клітинок (мал. 14.5, *a* і *б*). Після копіювання виділена клітинка чи діапазон обрамлюється блимаючою рамкою



, яка після вставлення зникає.





Мал. 14.5, а



Мал. 14.5, б

У середовищі табличного процесора *Microsoft Excel* можна також скористатись інструментами з групи *Буфер обміну* вкладки *Основне* чи застосовувати прийоми для копіювання та переміщення введених даних за допомогою миші. Для переміщення виділеної клітинки або діапазону за допомогою миші слід навести вказівник миші

на межу виділення, і коли вказівник миші набуває вигляду , виконати перетягування в інше місце таблиці. Копіювання за допомогою миші виконується аналогічно, але при натисненій клавіші *Ctrl*, при

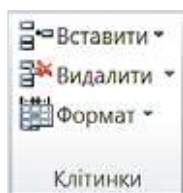
цьому вказівник миші набуває вигляду . Для одержання відомостей про виконання цих дій слід скористатись довідкою, яку можна викликати, натиснувши клавішу *F1*.

4. Які особливості редагування структури таблиці?

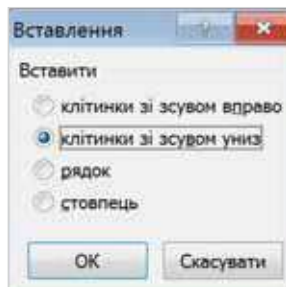
Редагування структури таблиці передбачає зміну кількості її стовпців або/та рядків і видалення чи додавання окремих клітинок таблиці.



Якщо потрібно вставити кілька рядків чи стовпців, треба виділити стільки рядків або стовпців, скільки треба додати, та вибрати вказівку контекстного меню *Додати клітинки* або вибрати інструмент *Вставити* з групи *Клітинки* вкладки *Основне* (мал. 14.6).



Мал. 14.6



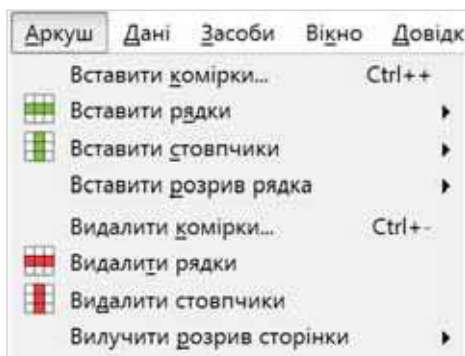
Мал. 14.7

Також можна додавати або видаляти окремі клітинки чи діапазони клітинок. У цьому разі існуючі дані можуть зміщуватись в одному з напрямків, запропонованих у відповідному діалоговому вікні *Вставлення* (мал. 14.7).

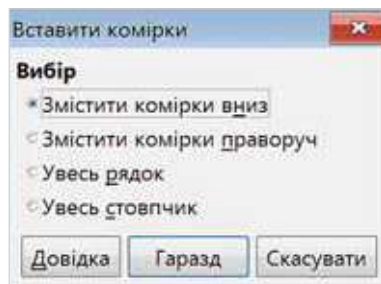
Для видалення рядків або стовпців слід скористатися вказівкою *Видалити*.



Аналогічні дії з клітинками чи їх діапазонами виконують у табличному процесорі *LibreOffice Calc* за допомогою вказівок меню *Аркуш* (мал. 14.8) або вікна *Вставити комірки* (мал. 14.9).



Мал. 14.8



Мал. 14.9



ДІЄМО

Вправа 1. Путівки.

Завдання. Введіть за зразком текстові та числові дані до таблиці *Путівки.xlsx* (*Путівки.ods*), що зберігається в папці *Електронні таблиці*, для обліку кількості путівок, проданих туристичною фірмою. Деякі числові дані слід копіювати з відповідних аркушів таблиці.

1. Відкрийте файл *Путівки.xlsx* (*Путівки.ods*), що зберігається в папці *Електронні таблиці*.
2. Перейдіть на аркуш *Путівки*. Для цього клацніть мишею на вкладці з відповідною назвою аркуша.
3. У клітинку *A4* введіть текст *Україна* та натисніть клавішу зі стрілкою вниз. У клітинки *A5* і *A6* введіть той само текст, для цього скористайтесь можливістю автоматичної підстановки — після введення літери *У* буде запропоновано підставити слово *Україна*. При цьому досить натиснути клавішу *Enter* або клавішу зі стрілкою вниз.
4. Аналогічно в клітинки *A7* і *A8* введіть *Єгипет*, у клітинки *A9* і *A10* — слово *Туреччина* (мал. 14.10).

	A	B	C	D	E
1	Продаж туристичних путівок				
2					
3	Країна	Місто	квітень	травень	червень
4	Україна	Одеса			
5	Україна	Херсон			
6	Україна	Бердянськ			
7	Єгипет	Хургада			
8	Єгипет	Шарм-ель-Шейх			
9	Туреччина	Анталія			
10	Туреччина	Кемер			
11					


Мал. 14.10

5. До клітинки *C3* введіть слово *квітень*. Скористайтесь засобом автозаповнення та виконайте протягування мишею до клітинки *E3*, щоб заповнити діапазон *C3:E3* назвами місяців.
6. У діапазон клітинок *C4:C10* введіть послідовно числа: 12, 10, 3, 128, 135, 57, 62 (мал. 14.11).

	A	B	C	D	E
1	Продаж туристичних путівок				
2					
3	Країна	Місто	квітень	травень	червень
4	Україна	Одеса	12	54	110
5	Україна	Херсон	10	67	127
6	Україна	Бердянськ	3	25	58
7	Єгипет	Хургада	128	135	76
8	Єгипет	Шарм-ель-Шейх	135	156	82
9	Туреччина	Анталія	57	187	215
10	Туреччина	Кемер	62	192	238

 Мал. 14.11

7. Перейдіть на аркуш *Травень*. Виділіть діапазон клітинок *C4:C10* і

скопійуйте у буфер обміну за допомогою кнопки  *Копіювати* з групи інструментів вкладки *Основне (Microsoft Excel)* чи кнопки



Копіювати з панелі інструментів *Стандартна (LibreOffice Calc)*.

Перейдіть на аркуш *Путівки*. Виділіть клітинку *D4*. Натисніть кнопку



(у *Microsoft Excel*) чи



(у *LibreOffice Calc*). Аналогічно

скопійуйте дані з діапазону клітинок *C4:C10* аркуша *Червень* на аркуш *Путівки*, починаючи з клітинки *E4*.

8. Збережіть результати роботи у файлі з тим само ім'ям у папці *Таблиці* своєї структури папок за допомогою вказівки



Зберегти як

з меню вкладки *Файл (у Microsoft Excel)* чи



Зберегти як...



з меню *Файл (у LibreOffice Calc)*.

Вправа 2. Продаж путівок.

Завдання. Внесіть до таблиці про продаж туристичних путівок, що зберігається у файлі *Путівки.xlsx (Путівки.ods)*, зміни щодо продажу путівок у квітні місяці: до Хургади — на 125, до Анталії — на 67, до Херсона — на 23.

1. Відкрийте файл *Путівки.xlsx (Путівки.ods)*.
2. Виділіть клітинку *C7*, де міститься значення 128, та клацніть лівою

кнопкою миші в рядку формул  таким

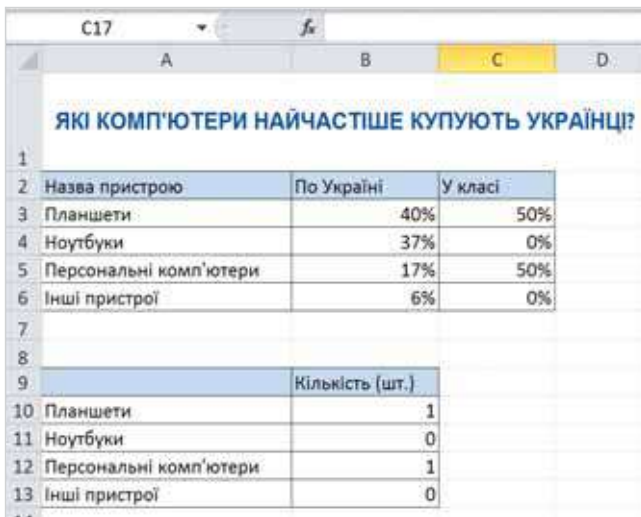
чином, щоб курсор введення був розташований після останньої цифри. Натисніть клавішу *BackSpace*, введіть число 5 і натисніть кнопку *Ввід*  (у *Microsoft Excel*) чи *Прийняти*  (у *LibreOffice Calc*). Вміст клітинки *C7* зміниться на 125.

3. Двічі клацніть лівою кнопкою миші на клітинці *C9*, що містить значення 57. За допомогою миші або клавіш зі стрілками встановіть курсор введення перед першою цифрою числа. Натисніть клавішу *Delete*, введіть число 6 і натисніть клавішу *Enter*. Вміст клітинки *C9* зміниться на 67.
4. Виділіть клітинку *C5*, де міститься значення 10, та введіть нове значення 23.
5. Клацніть правою кнопкою миші на заголовку стовпця *A* та виберіть вказівку *Додати клітинки*. Усі дані таблиці будуть переміщені на один стовпець праворуч, у стовпці *A* буде розміщено новий порожній стовпець.
6. У клітинку *A3* введіть текст *№*.
7. У клітинку *A4* введіть 1, в *A5* — 2. За допомогою засобу автозаповнення введіть нумерацію назв країн.
8. Збережіть внесені зміни у файлі з тим само іменем.

Вправа 3. Пристрої.

Завдання. Відредагуйте за зразком дані таблиці *Пристрої.xlsx* (*Пристрої.ods*), що зберігається в папці *Електронні таблиці*, для визначення кількості й типів пристроїв, якими користуються учні вашого класу, та порівняння з даними по Україні.

1. Відкрийте файл *Пристрої.xlsx* (*Пристрої.ods*), що зберігається в папці *Електронні таблиці* (мал. 14.12).



ЯКІ КОМП'ЮТЕРИ НАЙЧАСТІШЕ КУПУЮТЬ УКРАЇНЦІ?		
1		
2	Назва пристрою	По Україні
3	Планшети	40%
4	Ноутбуки	37%
5	Персональні комп'ютери	17%
6	Інші пристрої	6%
7		
8		
9		Кількість (шт.)
10	Планшети	1
11	Ноутбуки	0
12	Персональні комп'ютери	1
13	Інші пристрої	0

 Мал. 14.12

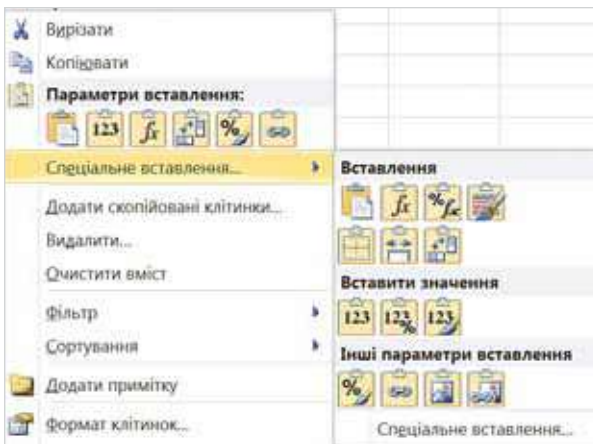
2. Виділіть клітинку A7 і з клавіатури введіть текст *Пристрої, якими користуються учні класу*. Цей текст — назва таблиці, що розміщується в діапазоні клітинок A8:B12 (мал. 14.12).
3. Клацніть мишею на рядку з номером 7 і за допомогою контекстного меню додайте один рядок для встановлення відступу між таблицями даних.
4. За допомогою протягування миші виділіть діапазон клітинок A2:A6 і скопіюйте дані до буфера обміну.
5. Виділіть клітинку A9 і за допомогою контекстного меню вставте дані з буфера обміну.
6. Проведіть опитування учнів класу та визначте, якою технікою вони користуються. Дані опитування — кількість пристроїв кожного із вказаних типів — введіть у клітинки діапазону B10:B13. Перегляньте, як змінились дані у клітинках діапазону C3:C6 і на діаграмі.
7. Збережіть внесені зміни у файлі з тим само іменем у папці *Таблиці* своєї структури папок.



Досліджуємо

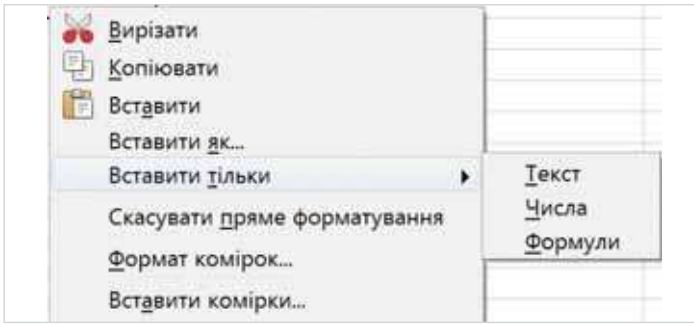
Вправа 4. Параметри вставлення.

Завдання. Копіюючи вміст клітинок за допомогою буфера обміну, можна в контекстному меню використовувати додаткові параметри вставлення (мал. 14.13, а і б). Дослідіть, як змінюються дані при використанні різних параметрів для копіювання клітинок, що містять текст, числа та формули. Поекспериментуйте з копіюванням даних у файлі *Приладдя.xlsx* (*Приладдя.ods*), що зберігається в папці *Електронні таблиці*. Для цього скопіюйте дані з клітинки, яка містить формулу, використовуючи різні способи вставлення — як число та як формули. За результатами зробіть висновки.



Мал. 14.13, а





Мал. 14.13, б



Обговорюємо

1. Дані якого типу можна вводити до клітинок електронної таблиці?
2. Як виділити діапазон клітинок у середовищі табличного процесора?
3. Для чого використовують засіб автозаповнення? Наведіть приклади завдань.
4. Які операції редагування можна виконувати в середовищі табличного процесора?
5. Як переміщувати та копіювати дані в середовищі табличного процесора? Які є способи виконання цих дій?
6. Які дії можна виконувати з аркушами робочої книги?

Працюємо в парах

7. Висловте припущення, у яких випадках може виникнути потреба внесення змін до електронної таблиці. Яких саме змін? Назвіть дві причини. Обговоріть у парі.
8. Обговоріть, як можна використовувати табличний процесор під час навчання. Наведіть приклади таблиць, які використовують при навчанні історії, географії чи інших предметів. Які переваги створення та використання таких таблиць у середовищі табличного процесора?
9. Порівняйте способи редагування таблиць у табличному й текстовому процесорах. Визначте спільне та відмінності. Зробіть висновок, у якому процесорі зручніше редагувати таблиці.

Працюємо самостійно

10. У папці *Таблиці* своєї структури папок створіть таблицю *Наш клас.xlsx* (*Наш клас.ods*), що містить відомості про учнів класу. У подальшому ви зможете доповнювати дані таблиці, проводити розрахунки та будувати діаграми. Приклад таблиці показано на малюнку 14.14. Але самостійно можна змінити структуру

таблиці — додати стовпці, змінити заголовки тощо. Головне — спланувати, які дані про учнів класу можна подати у вигляді електронної таблиці та як їх використовувати.

	A	B	C	D
1	Наш 7-А клас			
2	№	Прізвище та ім'я	Дата народження	Телефон
3	1	Градова Ангеліна	15.01.2008	0686965079
4	2	Грушевська Наталія	16.03.2008	0984839153
5	3	Ільчишин Михайло	18.09.2007	0982713227
6	4	Паньків Микола	05.02.2008	0998058730

Мал. 14.14

- ★11. Перегляньте таблиці, які ви створюєте й заповнюєте під час виконання завдань з фізики, математики тощо. Визначте три спільні ознаки та дві відмінності між електронними таблицями й друкованими. Як приклад, розгляньте правила переведення довжини з одних одиниць вимірювання в інші (мал. 14.15) і таблицю у файлі *Довжини.xlsx* (*Довжини.ods*) — як на малюнку 14.16.

ОДИНИЦІ ВИМІРЮВАННЯ ДОВЖИНИ

ММ — МІЛІМЕТР
СМ — САНТИМЕТР
ДМ — ДЕЦИМЕТР
М — МЕТР
КМ — КІЛОМЕТР

1 см = 10 мм
 1 дм = 10 см = 100 мм
 1 м = 10 дм = 100 см
 1 км = 1000 м = 1000000 см

46 м 30 см + 5 м 12 см = 51 м 42 см

46 м 30 см	4630 см
+ 5 м 12 см	+ 512 см
51 м 42 см	5142 см

Мал. 14.15

Довжина	м	см	мм
Олівець	0,2	20	200
Кімната	5	500	5000

Мал. 14.16

Запишіть висновок щодо переваг використання електронних таблиць.

12. У папці *Таблиці* своєї структури папок створіть таблицю *Курс валют.xlsx* (*Курс валют.ods*), що містить відомості про курси валют. Структуру таблиці розробіть самостійно. Потрібні дані можна знайти, наприклад, на офіційному сайті *Національного банку України* або визначити, який курс установив той чи інший банк (<http://kurs.com.ua/ua>).

Повторюємо



Опишіть алгоритми редагування електронної таблиці для користувача, який уперше виконує дані операції. У чому будуть їх особливості?



Формати даних та форматування таблиць

ВИ ДІЗНАЄТЕСЬ:

- ▶ як встановлювати формати подання даних у клітинках;
- ▶ як змінити форматування таблиці;
- ▶ як відобразити вміст клітинки повністю на екрані.



Вивчаємо

1. Як встановлювати формати подання даних у клітинках?

У клітинки таблиці можна вводити різні дані. Вигляд даних в електронній таблиці визначається їхнім форматом (мал. 15.1).

ПРОДАЖ КВИТКІВ У КІНОТЕАТРІ						
Код	Назва фільму	Ціна квитка	Кількість	Всього виторг	Продаж за замовленням	Дата сеансу
458	Пекельна Хоругва, або Різдво Козацьке	100,00€	75	7 500,00€	35%	07.01.2020
784	Крижане серце 2	95,00€	50	4 750,00€	27%	08.01.2020

Текстовий грошовий Числовий Відсотковий Дата

Мал. 15.1

Формат даних визначає дії, які можна виконувати з ними.

Формат даних визначає тип даних, які введено до клітинки, та особливості їхнього відображення у клітинці.



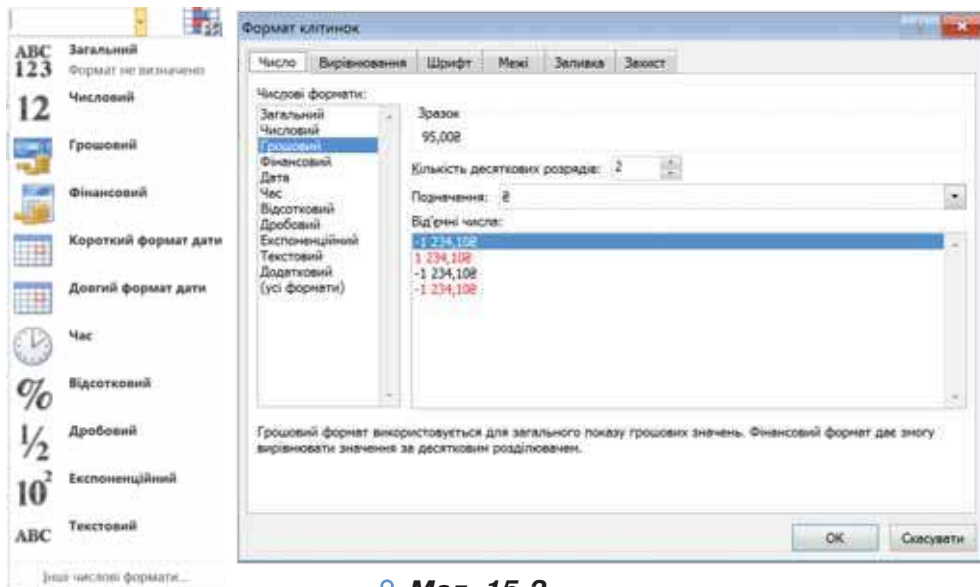
За замовчуванням у *Microsoft Excel* формат даних клітинки визначається як *Загальний*, що передбачає відображення числових і текстових даних без спеціального форматування. У *Microsoft Excel* для зміни формату можна скористатись інструментами групи *Число* вкладки *Основне*. Зокрема, зі списку числових форматів (мал. 15.2) обрати потрібний.

Змінити формати даних можна також у вікні *Формат клітинок*. Для цього слід виділити клітинку чи діапазон та вибрати вказівку



Формат клітинок...

у контекстному меню. Далі на вкладці *Число* вікна *Формат клітинок* (мал. 15.2), що відкривається, обрати відповідний формат.

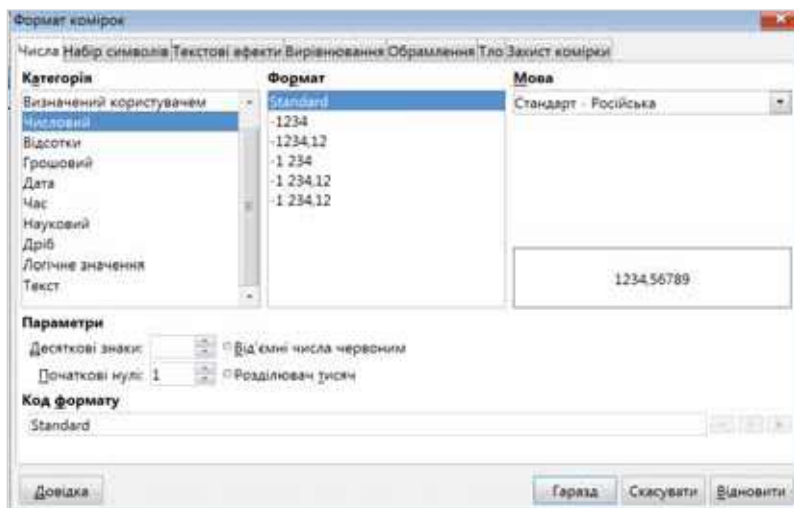


Мал. 15.2

За замовчуванням, у *LibreOffice Calc* формат даних клітинки визначається як *Числовий стандарт*.



У *LibreOffice Calc* формат даних встановлюють у вкладці *Числа* вікна *Формат комірок*, яке викликається вказівкою меню *Формат/Комірки* (мал. 15.3).



Мал. 15.3


2. Як змінити форматування таблиці?

Для зовнішнього оформлення електронної таблиці змінюють її форматування.

Форматування електронної таблиці — зміна зовнішнього вигляду таблиці чи окремих її клітинок: шрифту, кольору, накреслення символів; вирівнювання; кольору заливки, розмірів і меж клітинок тощо.




Форматування клітинок електронних таблиць здійснюється засобами різних вкладок вікна *Формат клітинок* та інструментів стрічки:

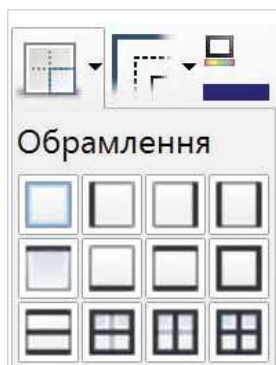
- вирівнювання вмісту можна змінити за допомогою відповідних кнопок  із групи *Вирівнювання* вкладки *Основне*;
- форматування тексту — вкладка *Шрифт*: шрифт, розмір, колір тексту, накреслення; пропонувані вказівки дублюють інструменти групи *Шрифт* вкладки *Основне*;
- встановлення меж таблиці чи окремих її об'єктів — вкладка *Межі*: зміна значень для кольору, типу ліній; пропонувані засоби дублюють вказівки інструмента *Межі* (мал. 15.4) групи *Шрифт* вкладки *Основне*;
- встановлення кольору тла клітинки чи діапазону клітинок таблиці — вкладка *Заливка*; пропонувані засоби дублюють вказівки інструмента *Колір заливки* групи *Шрифт* вкладки *Основне*.

Мал. 15.4

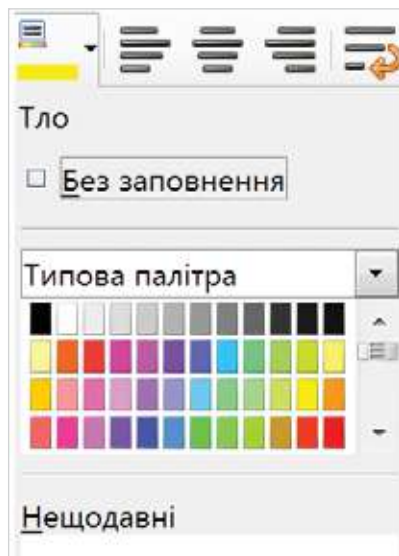


У *LibreOffice Calc* змінювати параметри форматування можна засобами діалогового вікна *Формат комірок* та інструментів на панелі *Форматування*:

- вирівнювання вмісту — за допомогою відповідних інструментів  панелі інструментів *Форматування*;
- форматування тексту — *Текстові ефекти* чи панелі інструментів *Форматування тексту*;
- межі таблиці чи окремих її об'єктів — за допомогою інструмента *Обрамлення* (мал. 15.5), а колір тла — за допомогою інструмента *Тло* (мал. 15.6).



Мал. 15.5



Мал. 15.6

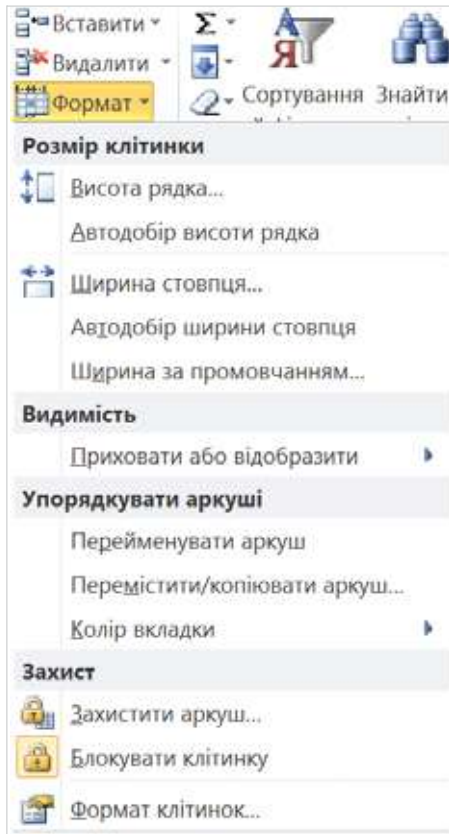
3. Як відобразити вміст клітинки повністю на екрані?

За замовчуванням, в електронній книзі всі клітинки мають однакову ширину й висоту. Але в деяких випадках довжина тексту або числа більша, ніж ширина стовпця, у якому розташована клітинка. Якщо сусідні праворуч клітинки порожні, вміст клітинки буде відображено повністю з поширенням його на вільні клітинки. Якщо ж сусідня праворуч клітинка містить дані, то текстові дані будуть відображені частково — стільки символів, скільки вміститься в клітинку поточної ширини, а дані деяких числових форматів (зокрема, типу дата і час) відображаються як послідовність знаків фунта (### ##). Але це не означає, що дані втрачено — вони зберігаються в пам'яті й при збільшенні ширини стовпця будуть відображені повністю.

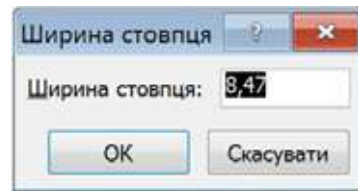


Змінити ширину стовпця можна різними способами:

- за допомогою вказівки *Ширина стовпця* зі списку *Формат* групи інструментів *Клітинки* (мал. 15.7), при цьому треба в діалоговому вікні ввести потрібну ширину в символах і натиснути кнопку *ОК* (мал. 15.8);



Мал. 15.7



Мал. 15.8

- установити вказівник миші на праву межу заголовка стовпця (коли він набуде вигляду двонапрямленої стрілки) та виконати протягування вправо чи вліво відповідно для збільшення або зменшення ширини стовпця (мал 15.9);

	A	B	C	D	E	F
1	<i>Календар погоди з 15.01 по 15.02.2020</i>					
2						
3		Температура повітря (°C)				
4	Дата	Середня	анок (8 год	нь (14 год	ечір (20 год	Опади
5	#####	-6	-6	-6	-7	0,94
6	#####	-5	-5	-5	-6	0,4

Мал. 15.9

- можна також автоматично встановити ширину стовпця так, щоб вона відповідала довжині найдовших даних у цьому стовпці. Для цього необхідно двічі клацнути лівою кнопкою миші на правій межі заголовка стовпця.

Іноді немає потреби збільшувати ширину стовпця, оскільки лише в одній клітинці (наприклад, у заголовку), довжина даних перевищує ширину стовпця. У цьому разі можна дати вказівку перенести текст на кілька рядків і збільшити висоту рядка. Для цього треба виділити потрібну клітинку (або діапазон клітинок) та виконати вказівку *Формат клітинок (Формат комірок)* із контекстного меню, перейти на вкладку *Вирівнювання* та встановити прапорець *переносити по словах*.

Висота рядка в цьому разі може бути дібрана автоматично, в іншому випадку — змінити її можна аналогічно до зміни ширини стовпця — за допомогою вказівки *Формат/Розмір клітинки/Висота рядка* або з використанням нижньої межі заголовка рядка (мал. 15.10).

	A	B	C	D	E	F
1	<i>Календар погоди з 15.01 по 15.02.2020</i>					
2						
3		Температура повітря (°C)				
4	Дата	Середня	Ранок (8 год)	День (14 год)	Вечір (20 год)	Опади
5	15.01.20	-6	-6	-6	-7	0,94
6	16.01.20	-5	-5	-5	-6	0,4

 **Мал. 15.10**

Для того щоб дані були відображені на кілька клітинок, можна об'єднувати клітинки таблиці. При об'єднанні двох або кількох клітинок утворюється одна клітинка, що спрощує форматування тексту або виразу у клітинках. Наприклад, може знадобитись об'єднання кількох клітинок рядка, відведеного для запису заголовка таблиці, щоб заголовок розмістився в одній клітинці. Після цього можна вирівняти текст заголовка, щоб він був розташований в об'єднаній клітинці по центру.



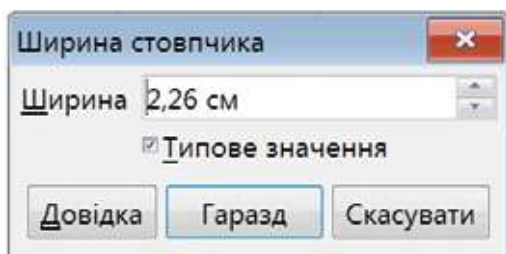
Для об'єднання клітинок їх слід виділити, обрати інструмент

Об'єднати та розмістити в центрі  .




Дії в *LibreOffice Calc* дуже схожі на дії в *Microsoft Excel*.

Для зміни ширини стовпця можна використати вказівку меню *Формат/Стовпчик/Ширина*. У діалоговому вікні *Ширина стовпчика* слід ввести ширину в сантиметрах і натиснути кнопку *Гаразд* (мал. 15.11).



Мал. 15.11

Аналогічно можна змінювати висоту рядка за допомогою вказівки меню *Формат/Рядок/Висота*. Для об'єднання клітинок використовують інструмент *Об'єднати й центрувати комірки*  панелі інструментів *Форматування*.



ДІЄМО

Вправа 1. Календар погоди.

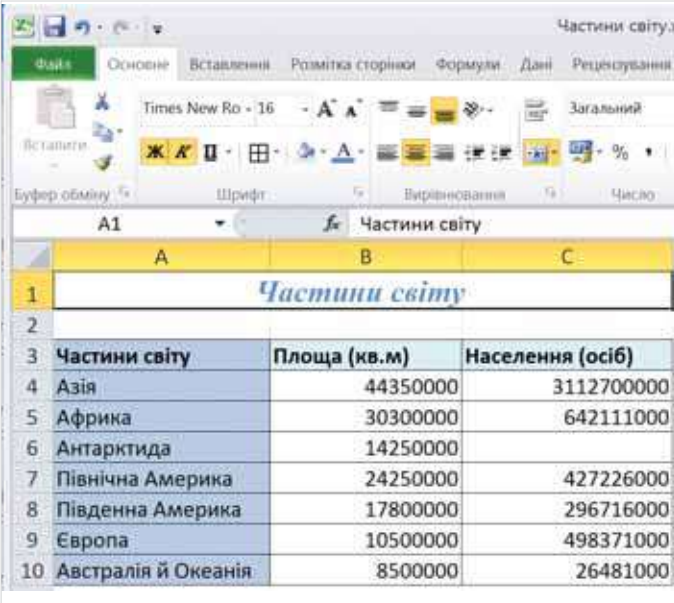
Завдання. Змініть параметри форматування клітинок таблиці, що зберігається у файлі *Календар погоди.xlsx* (*Календар погоди.ods*) у папці *Електронні таблиці*, так, щоб усі дані таблиці відображалися на екрані.

1. Відкрийте файл *Календар погоди*, що зберігається в папці *Електронні таблиці* (мал. 15.10).
2. Виділіть діапазон клітинок *A4:F4*. Виберіть з контекстного меню вказівку *Формат клітинок (Формат комірок)*, перейдіть на вкладку *Вирівнювання*, установіть прапорець *переносити по словах* і натисніть кнопку *ОК*. За необхідності збільшіть висоту четвертого рядка аркуша.
3. Установіть вказівник миші на праву межу заголовка стовпця *A* (коли він набуде вигляду двонапрямленої стрілки) та двічі клацніть лівою кнопкою миші. Розмір стовпця автоматично збільшиться так, що всі дані, розміщені в цьому стовпці, будуть відображені на екрані. За потреби змініть ширину інших стовпців.
4. Виділіть діапазон *B5:B36*, установіть формат даних *Числовий*, кількість десяткових знаків — 1 (дробова частина — 1 знак).

- Виділіть діапазон $F5:F36$, установіть формат даних *Відсотковий (Відсотки)*, кількість десяткових знаків — 0 (дробова частина — 0 знаків).
- Збережіть результати у файлі з тим само іменем у папці *Таблиці* своєї структури папок.



Вправа 2. Частини світу.




Завдання. Змініть параметри форматування клітинок таблиці, що зберігається у файлі *Частини світу.xlsx (Частини світу.ods)* у папці *Електронні таблиці*, за зразком (мал. 15.12).



Частини світу		
Частини світу	Площа (кв.м)	Населення (осіб)
Азія	44350000	3112700000
Африка	30300000	642111000
Антарктида	14250000	
Північна Америка	24250000	427226000
Південна Америка	17800000	296716000
Європа	10500000	498371000
Австралія й Океанія	8500000	26481000

Мал. 15.12

- Відкрийте файл *Частини світу.xlsx (Частини світу.ods)*, що зберігається в папці *Електронні таблиці*, та збережіть його в папці *Таблиці* своєї структури папок.
- Змініть ширину стовпців, об'єднайте клітинки $A1:C1$, розмістіть у центрі та встановіть такі параметри форматування заголовка таблиці *Частини світу*: шрифт — *Times New Roman*, розмір шрифту — 16, стиль накреслення — *курсив*, колір — *синій* (мал. 15.12).
- Виділіть діапазон клітинок $A3:C10$. Для виділеного діапазону встановіть межі — у списку інструмента *Усі межі*  з групи *Шрифт* вкладки *Основне* виберіть  *Усі межі*, якщо працюєте в *Microsoft Excel*. Якщо ви працюєте в *LibreOffice Calc*, то скористайтеся тим само інструментом зі списку *Обрамлення* на панелі інструментів *Форматування*.

- Виділіть діапазон клітинок A3:C3. Для виділеного діапазону застосуйте колір тла — *блакитний*. Для цього скористайтеся інструментом *Колір заливки*  або *Тло*.
- За допомогою інструмента *Формат за зразком*  з групи *Буфер обміну (Копіювання формату)*  панелі інструментів *Стандартна* установіть таку само заливку для діапазону клітинок A3:A10. У разі потреби для виконання завдання скористайтеся довідкою.
- Збережіть внесені зміни.

Вправа 3. Результати успішності.

Завдання. Змініть формат даних і параметри форматування клітинок таблиці, що зберігається у файлі *Результати успішності.xlsx* (*Результати успішності.ods*) у папці *Електронні таблиці*, за зразком (мал. 15.13).

- Відкрийте файл *Результати успішності*, що зберігається в папці *Електронні таблиці*, та перегляньте її вміст.
- Змініть формат даних діапазону клітинок C3:I3, установивши формат *дата*.
- Відформатуйте таблицю так, як показано на малюнку 15.13.

	A	B	C	D	E	F	G	H	I
1	<i>Результати успішності учнів 7-А класу з історії за III чверть</i>								
2									
3	Прізвище	Ім'я	14.Січ	18.Січ	22.Січ	02.Лют	19.Лют	05.Бер	20.Бер
4	Владимиренко	Федір		9	5	6	8	9	
5	Обрусняк	Софія	10		9	11		11	
6	Павх	Вероніка	9		11	11	10		
7	Паньків	Микола		8			11		
8	Пізнюк	Мар'ян	8	9			6		
9	Польник	Павло	7	10		9		8	

 **Мал. 15.13**

- Збережіть результати роботи у файлі з тим само іменем у папці *Таблиці* своєї структури папок.
- Сформулюйте послідовність дій, які необхідно виконати, та які параметри форматування застосувати, щоб отримати результат, як на зразку. Відправте на навчальну електронну скриньку електронного листа, у якому запишіть алгоритм виконання завдання. Файл *Результати успішності* додайте як вкладення.

Досліджуємо

Вправа 4. Параметри додавання.

Завдання. Визначте дії, що відбуваються при застосуванні засобу *Параметри додавання* в середовищі табличного процесора *Microsoft Excel*, та застосуйте до файла *Країни.xlsx*, що зберігається в папці *Електронні таблиці*.

1. Прочитайте теоретичні відомості, які містяться у файлі *Параметри додавання.docx*, що зберігається в папці *Електронні таблиці*.
2. До таблиці даних *Населення найбільших міст* із файла *Країни.xlsx*, що зберігається в папці *Електронні таблиці*, додайте відомості про столицю Китаю, де проживає 21 150 000 осіб. Оскільки дані в таблиці упорядковано за алфавітом, визначте позицію додавання даних.
3. Застосуйте параметри додавання за зразком (мал. 15.14). Скільки існує способів виконання завдання? Відповідь поясніть.

6			
7	Індія		17 838 842
8	Індія		17 060 307
9	Індія		14 160 467
10	Індія		12 700 800

Мал. 15.14

Обговорюємо

1. У чому полягає різниця між форматом даних і форматуванням клітинок?
2. Які формати належать до числових? Як змінити подання числових даних у середовищі табличного процесора?
3. Чи є різниця між форматуванням усієї таблиці та окремих клітинок у середовищі табличного процесора?
4. Які параметри можна застосувати для форматування клітинки чи діапазону клітинок? Які інструменти табличного процесора можна використати для зміни параметрів?
5. У яких випадках є потреба об'єднувати клітинки електронної таблиці, а в яких достатньо застосувати перенесення по словах? Відповідь поясніть на прикладах.

Працюємо в парах


6. Чи подібні дії форматування клітинок електронної таблиці, текстових написів у презентаціях і тексту, створеного засобами текстового процесора? Дані подайте у вигляді діаграми Венна. Обговоріть у парі.
- ★ 7. Перегляньте таблицю, подану на малюнку 15.1, і визначте можливі параметри форматування її об'єктів. Складіть і запропонуйте одне одному завдання на форматування: заголовка таблиці, стовпців *Код*, *Ціна квитка*, *Кількість*, *Дата сеансу*. Хтось із пари формулює завдання, а хтось його виконує. Проведіть взаємооцінювання.
- ★★ 8. За схемою *Форматування електронної таблиці* з рубрики *Повторюємо* складіть правила для зміни параметрів форматування клітинок електронної таблиці. Обговоріть у парі. У разі потреби скористайтесь довідкою табличного процесора.

Працюємо самостійно

9. Зазначте засоби форматування, які використано для форматування електронної таблиці у вправі за посиланням <https://learningapps.org/view8768649>.
10. Укажіть, які формати використано для форматування даних електронної таблиці у вправі за посиланням <https://learningapps.org/view8769007>.
- ★ 11. У таблиці, що міститься у файлі *Наш клас.xlsx* (*Наш клас. ods*), яку ви створювали раніше, змініть формат даних і встановіть параметри форматування. Параметри форматування доберіть на власний розсуд.
- ★★ 12. У таблиці, що міститься у файлі *Кросворд*, виділіть відповідні клітинки та задайте колір заливки для заданих діапазонів клітинок за зразком (мал. 15.15). У клітинки *D6*, *D8* і *D10* введіть із клавіатури літери, щоб по вертикалі одержати прізвище одного з розробників першого табличного процесора. Потрібні відомості знайдіть в інтернеті.



Мал. 15.15

-  13. Знайдіть у Вікіпедії список найбільших озер України, скопіюйте таблицю в середовище табличного процесора (мал. 15.16), змініть формат даних і параметри форматування так, щоб усі дані були відображені на екрані, а таблиця мала привабливий вигляд.

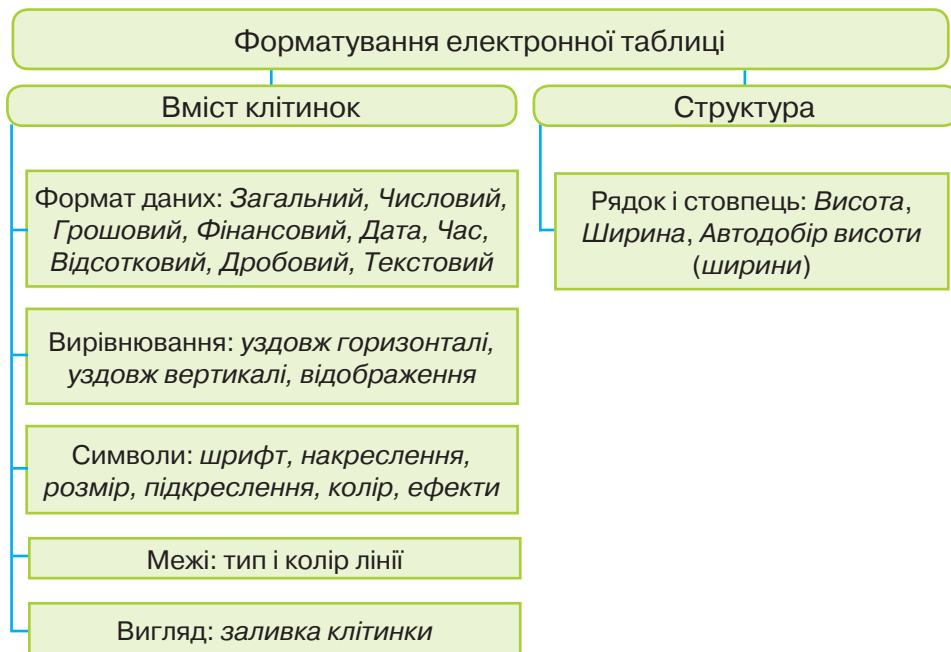
	A	B	C	D	E	F
1	Список найбільших озер України					
2						
3	Озера і лимани	Розташування	Площа, км²	Довжина, км	Максимальна ширина, км	Дата заповнення
4	Сасик (Кундук) (озеро)	Узбережжя Чорного моря	210.0	39.0	12,0	#####
5	Тилігульський лиман	Узбережжя Чорного моря	150 — 1780.0		3,5	
6	Молочний лиман (Молочне озеро)	Узбережжя Азовського моря	170.0	32.0	8,0	

 Мал. 15.16

Повторюємо



Розгляньте схему та поясніть зв'язок між вказаними поняттями.





Адресація. Формули. Обчислення в електронних таблицях

ВИ ДІЗНАЄТЕСЬ:

- ▶ як виконувати обчислення із числовими даними електронної таблиці;
- ▶ як здійснювати обчислення за допомогою вбудованих функцій;
- ▶ що відбувається при копіюванні формул;
- ▶ які помилки можуть виникати при використанні формул.



Вивчаємо

1. Як виконувати обчислення із числовими даними електронної таблиці?

У клітинки таблиці можна вводити не лише конкретні значення (числові та текстові дані), а й формули. За допомогою формул виконують обчислення з використанням значень, що розміщуються в інших клітинках таблиці.

Формула — це вираз, що може містити числові дані, адреси клітинок чи їх діапазонів, функції та знаки арифметичних операцій.

Запис формули в середовищі табличного процесора завжди починається зі знака «=» для того, щоб можна було відрізнити формули від даних.

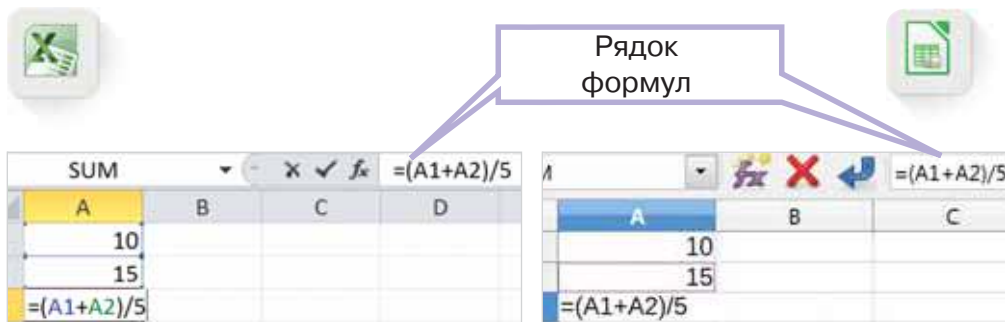
Наприклад, формула може мати такий вигляд: $=(A1 + A2)/5$, де $A1$, $A2$ — адреси клітинок, 5 — число, $+$ і $/$ — знаки арифметичних операцій.



Це цікаво

Зверніть увагу, що, на відміну від запису формул у математиці, формули в клітинках електронних таблиць, так само, як мовою програмування, записують у рядок.

Формула відображається в рядку формул (мал. 16.1, а і б). А її результат по завершенні введення — у відповідній клітинці, наприклад $A3$. Для перевірки правильності написання формули при її введенні або редагуванні посилання на різні клітинки позначають різними кольорами, і навколо таких клітинок на екрані відображаються рамки відповідного кольору.



Мал. 16.1, а

Мал. 16.1, б

Порядок виконання арифметичних дій у формулах визначається пріоритетом операцій за правилами математики. Піднесення до степеня виконується в першу чергу, потім — множення і ділення. Останніми виконуються операції додавання і віднімання. Усі операції виконуються в порядку запису послідовно зліва направо. За допомогою використання дужок можна змінити порядок виконання арифметичних операцій: вираз, записаний у дужках, обчислюється в першу чергу (табл. 16.1).

Таблиця 16.1

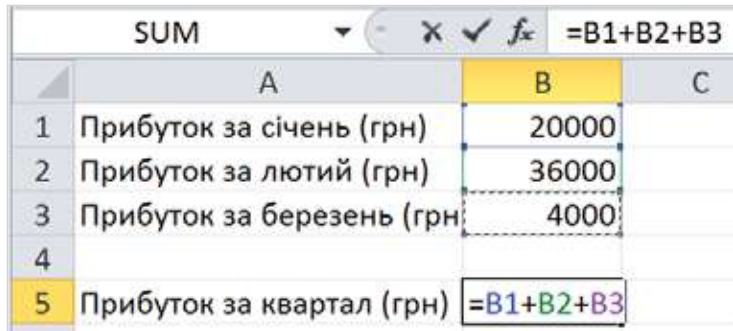
Знак арифметичної операції	Арифметична операція
^	Піднесення до степеня
*	Множення
/	Ділення
+	Додавання
-	Віднімання

Результат обчислення за формулою пов'язаний з даними, що містяться в інших клітинках. Якщо дані змінюються, то автоматично змінюватиметься результат, пов'язаний з ними.

Адреси клітинок при створенні формул можна задавати різними способами. Найзручніше клацнути мишею на відповідній клітинці чи виділити діапазон клітинок. Якщо ж вводити адреси клітинок із клавіатури, то важливо слідкувати, щоб вводилися символи латинського алфавіту.

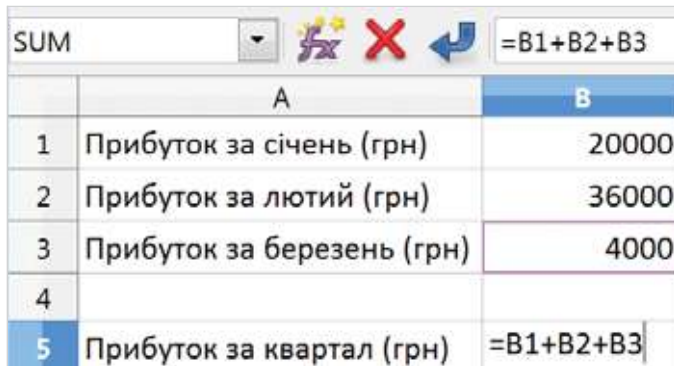
Для редагування формули слід помістити курсор у рядок формул і внести необхідні зміни (мал. 16.2, а і б). Також можна переглядати й редагувати формули безпосередньо в клітинці. Для цього потрібно двічі клацнути на ній мишею.

Мал. 16.2, а



	A	B	C
1	Прибуток за січень (грн)	20000	
2	Прибуток за лютий (грн)	36000	
3	Прибуток за березень (грн)	4000	
4			
5	Прибуток за квартал (грн)	=B1+B2+B3	

Мал. 16.2, б




	A	B
1	Прибуток за січень (грн)	20000
2	Прибуток за лютий (грн)	36000
3	Прибуток за березень (грн)	4000
4		
5	Прибуток за квартал (грн)	=B1+B2+B3

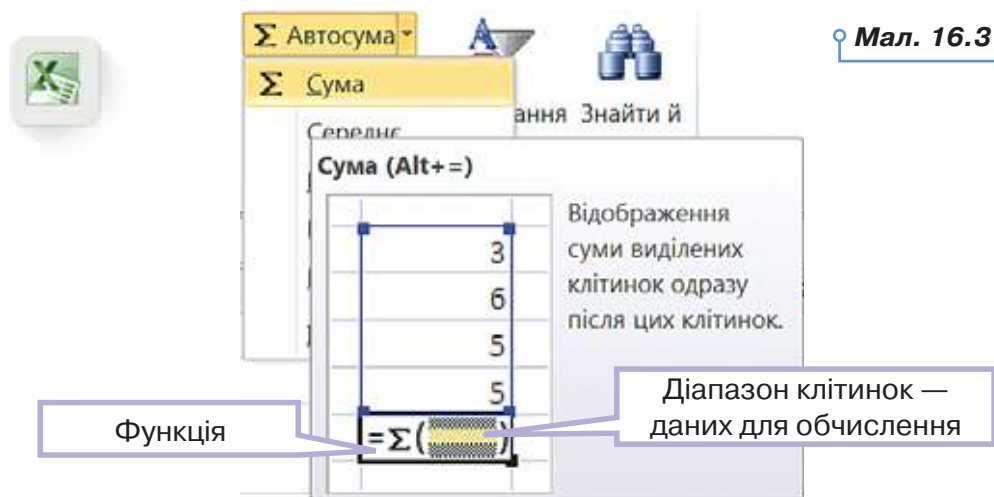
2. Як здійснювати обчислення за допомогою вбудованих функцій?

Табличний процесор містить набір вбудованих функцій, які можуть використовуватись у формулах. До основних функцій належать: знаходження суми, кількості, середнього значення, мінімального та максимального значень у діапазоні. Ці функції використовують для аналізу даних в електронних таблицях. Наприклад, для знаходження найбільшої вартості покупки, найнижчої оцінки в таблиці успішності учнів, середньої температури розчину під час проведення досліду, кількості учасників змагань тощо.




Для використання вбудованих функцій у табличному процесорі *Microsoft Excel* можна скористатись інструментом *Сума*  (мал. 16.3) з групи *Редагування*, що розміщена на вкладці *Основне* на стрічці.

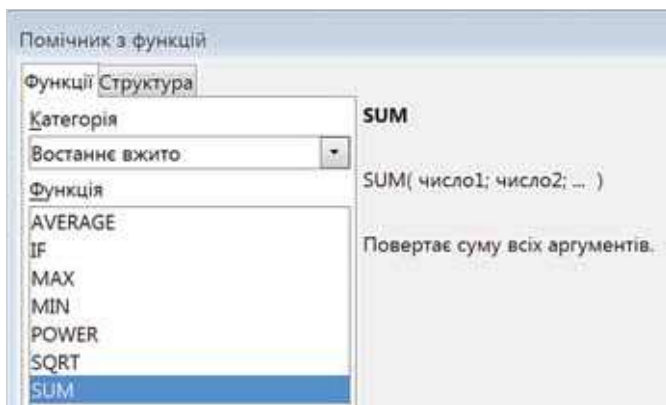
При виборі зі списку потрібної функції автоматично до рядка формул вводиться назва функції, і для її аргументів пропонується орієнтовний діапазон клітинок, де містяться дані, необхідні для обчислення. Тому застосування цих функцій часто називають **автообчисленням**.



За замовчуванням пропонується діапазон, клітинки якого містять числові дані та розташовані зазвичай або зверху, або ліворуч від клітинки, до якої вводиться формула. Можна підтвердити цей вибір, натиснувши клавішу *Enter*, або виділити новий діапазон.



У табличному процесорі *LibreOffice Calc* інструмент , який міститься в рядку формул, призначено тільки для знаходження суми даних у виділеному чи запропонованому автоматично діапазоні. Решту функцій можна обрати за допомогою вказівки *Вставка/Функція*, яку використовують для відкриття вікна зі списком функцій для вибору (мал. 16.4).



Мал. 16.4

Для спрощення використання у формулах діапазонів клітинок і пояснення даних, що містяться у клітинках, їм можна задати ім'я. Наприклад, при обчисленні вартості канцелярського приладдя для школяра діапазону клітинок *D3:D9* можна надати ім'я *Вартість_товарів* (табл. 16. 2).

Таблиця. 16.2

Приклад формули без іменування діапазону	Приклад формули з використанням імені діапазону
=SUM (D3:D9)	=SUM (Вартість_товарів)

Для іменування діапазону клітинок потрібно його виділити, у полі *Ім'я* замість адреси ввести з клавіатури ім'я, наприклад, *Вартість_товарів* (мал. 16.5), та натиснути *Enter*.

Вартість_товарів				
fx =B3*C3				
	A	B	C	D
1	Шкільне приладдя			
2	Назва	Вартість одиниці (грн)	Кількість (шт.)	Сума (грн)
3	Олівець	3	3	9
4	Ручка	5	3	15
5	Лінійка	3	1	3
6	Зошит	10	10	100
7	Папка для малювання	25	5	125
8	Пенал	30	2	60
9	Фарби	30	2	60
10	Приладдя (загальна сума)			372

Мал. 16.5

3. Що відбувається при копіюванні формул?

Часто в суміжних клітинках таблиць виконуються однотипні розрахунки. Для автоматизації однотипних розрахунків можна скопіювати формулу до клітинок відповідного діапазону. При цьому достатньо скористатися засобом автозаповнення. Наприклад, якщо клітинка *D3* містить формулу $=B3*C3$, то при копіюванні її до клітинки *D4* отримуємо $=B4*C4$ (мал. 16.6). Тобто при копіюванні формули в нову клітинку змінюються адреси клітинок, що використовуються у формулі, залежно від напрямку копіювання формули, та значення результату обчислення, а сама формула інтерпретується так: знайти добуток значень двох сусідніх зліва клітинок, що розташовані в тому самому рядку, що і клітинка, яка містить формулу.

Шкільне приладдя			
Назва	Вартість одиниці (грн)	Кількість (шт.)	Сума (грн)
Олівець	3	3	9
Ручка	5	3	15
Лінійка	3	1	3
Зошит	10	10	
Папка для малювання	25	5	
Пенал	30	2	
Фарби	30	2	
Приладдя (загальна сума)			27

D3 fx =B3*C3

D4 fx =B4*C4

Маркер
автозаповнення

Мал. 16.6

4. Які помилки можуть виникати при використанні формул?

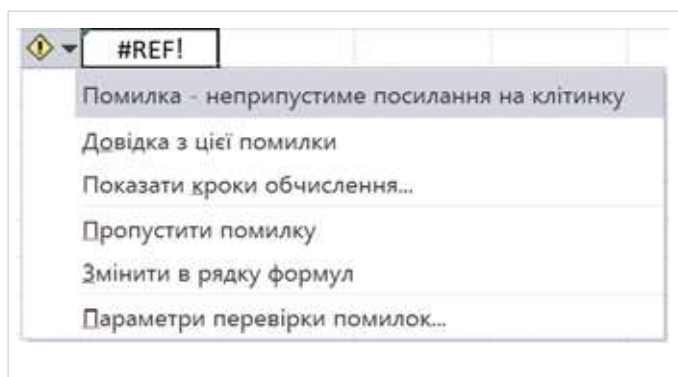
При використанні формул можуть виникати різні помилки, які допускають користувачі. Про них можна дізнатися з відповідних повідомлень, що автоматично з'являються у клітинці з формулою та починаються зі знака #.

Формули та повідомлення в табличному процесорі *Microsoft Excel* з україномовним й англomовним інтерфейсом відображаються скороченнями від слів англійською мовою, з російськомовним інтерфейсом — відповідно російською. Наприклад:

- **#REF! (#ССЫЛКА!)** — означає неправильне використання посилань, які є у формулі. Наприклад, у формулі $=A2(B3+C4)$ пропущено знак множення перед дужкою, що відкривається. Перевіряйте парність дужок у формулі та не пропускайте знаки арифметичних операцій;
- **#DIV/0! (#ДЕЛ/0!)** — спроба ділення на нуль. Така ситуація найчастіше виникає при використанні як дільника посилання на порожню клітинку або клітинку, яка містить нульове значення. Уважно перевіряйте дані, записані у формулі, чи результати копіювання формул;
- **#NAME? (#ИМЯ?)** — неможливість розпізнати ім'я, що використовується. Ця помилка виникає, наприклад, коли неправильно записано назву вбудованої функції чи в адресах клітинок використано літери кирилиці. Перевірте правильний запис назви функції та обрану мову введення.



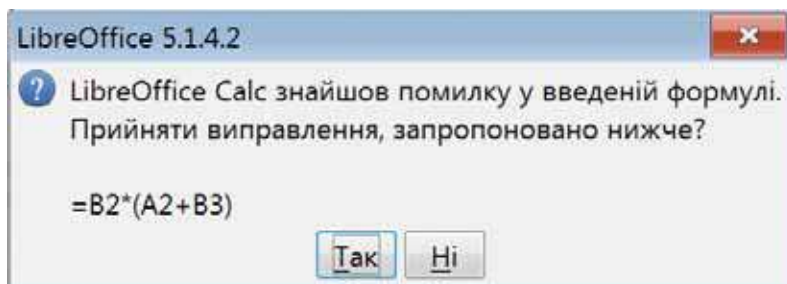
При виникненні помилки у верхньому лівому куті клітинки з'являється позначка у вигляді трикутника. Помилку слід виправити, вибравши один із запропонованих варіантів (мал. 16.7).



Мал. 16.7



У деяких випадках пропонується інформаційне вікно з повідомленням про помилку та можливим її виправленням (мал. 16.8).




Мал. 16.8



ДІЄМО

Вправа 1. Путівки.



Завдання. Знайдіть для даних таблиці *Путівки.xlsx* (*Путівки.ods*) загальну кількість проданих туристичних путівок, найбільшу та найменшу кількість проданих путівок у квітні, використовуючи вбудовані функції.

1. Відкрийте файл *Путівки.xlsx* (*Путівки.ods*), що зберігається в папці *Електронні таблиці* (мал. 16.9).
2. Введіть до клітинки *C11* текст *Усього*.
3. Виділіть клітинку *D11*, натисніть кнопку *Сума*  та виберіть функцію *Сума*.
4. Переконайтесь, що у формулі, яка з'явилась у клітинці *D11*, зазначено діапазон клітинок *D4:D10*, та натисніть клавішу *Enter*.

За потреби впишіть або виділіть потрібний діапазон.

SUM						
A	B	C	D	E	F	
1	Продаж туристичних путівок					
2						
3	№	Країна	Місто	квітень	травень	червень
4	1	Україна	Одеса	12	54	110
5	2	Україна	Херсон	3	67	127
6	3	Україна	Бердянськ	4	25	58
7	4	Єгипет	Хургада	112	135	76
8	5	Єгипет	Шарм-ель-Шейх	126	156	82
9	6	Туреччина	Анталія	68	187	215
10	7	Туреччина	Кемер	45	192	238
11		Усього		=SUM(D4:D10)		
12						
13						

Мал. 16.9

- У клітинку C12 введіть текст *Максимум*.
- Виділіть клітинку D12, натисніть кнопку *Сума*  та виберіть функцію *Максимум*.
- Проаналізуйте, який діапазон автоматично запропонований для знаходження максимального значення. Виділіть діапазон D4:D10, оскільки в запропонований діапазон включено також клітинку D11, що містить результат обчислення суми. Натисніть клавішу *Enter*.
- У клітинку C13 введіть текст *Мінімум*.
- Виділіть клітинку D13, натисніть кнопку *Сума* , виберіть функцію *Мінімум*, виділіть діапазон D4:D10 і натисніть клавішу *Enter* (мал. 16.10).

SUM						
A	B	C	D	E	F	
1	Продаж туристичних путівок					
2						
3	№	Країна	Місто	квітень	травень	червень
4	1	Україна	Одеса	12	54	110
5	2	Україна	Херсон	3	67	127
6	3	Україна	Бердянськ	4	25	58
7	4	Єгипет	Хургада	112	135	76
8	5	Єгипет	Шарм-ель-Шейх	126	156	82
9	6	Туреччина	Анталія	68	187	215
10	7	Туреччина	Кемер	45	192	238
11		Усього		370		
12		Максимум		126		
13		Мінімум		=MIN(D4:D10)		
14						

Мал. 16.10

10. Клацніть правою кнопкою миші на заголовку рядка з номером 11 і виберіть вказівку *Додати клітинки*. До таблиці буде вставлено порожній рядок між таблицею з даними та результатами обчислень, щоб наведені результати було зручніше переглядати.
11. Проаналізуйте, чи змінилися значення знайдених суми, максимального та мінімального значень.
12. Уведіть нове значення кількості проданих путівок у квітні до Бердянська — у клітинку *D6* введіть *11*. Перегляньте, як змінились результати обчислень за формулами.
13. Збережіть результати у файлі з тим само іменем у папці *Таблиці* своєї структури папок.

Вправа 2. Комунальні платежі.

Завдання. Розрахуйте для даних, що містяться у файлі з електронною таблицею *Послуги.xlsx* (*Послуги.ods*), у папці *Електронні таблиці*, вартість комунальних послуг за допомогою введення відповідних формул. Значення в стовпці *Спожито* розраховують як різницю *Поточних* і *Попередніх* показників лічильника, у стовпці *До сплати* — як добуток *Спожитих обсягів* і *Тарифу за одиницю обсягу*.

1. Відкрийте файл *Послуги.xlsx* (*Послуги.ods*), що зберігається в папці *Електронні таблиці*, та збережіть його з іменем *Вартість* у папці *Таблиці* своєї структури папок (мал. 16.11).

Примітка: на час виконання роботи дані щодо вартості послуг можуть бути іншими — за потреби змініть вміст відповідних клітинок.

	A	B	C	D	E	F
1	Рахунок на сплату житлово-комунальних послуг					
2						
3	Показники лічильників	Поточні	Попередні	Спожито	Тариф (грн за КВт/год або куб.м)	До сплати
4	Електроенергія	7480	7330		0,90	
5	Холодне водопостачання	708	683		16,72	
6	Гаряче водопостачання	560	541		90,54	
7						
8	Разом					

 Мал. 16.11

2. Виділіть клітинку *D4*. Щоб розпочати введення формули, введіть символ «=». Клацніть на клітинці *B4*, введіть символ «-», клацніть на клітинці *C4* (мал. 16.12) та натисніть клавішу *Enter*.
3. Виділіть клітинку *F4*. Уведіть символ «=», клацніть на клітинці *D4*, введіть символ «*» (операція множення), клацніть на клітинці *E4*. Натисніть клавішу *Enter*.
4. Збережіть результати у файлі з тим само іменем.

Рахунок на сплату житлово-комунальних послуг					
Показники лічильників	Поточні	Попередні	Спожито	Тариф (грн за КВт/год або куб.м)	До сплати
Електроенергія	7480	7330	=B4-C4	0,90	
Холодне водопостачання	708	683		16,72	

Мал. 16.12

Вправа 3. Копіювання формул.

Завдання. Завершіть розрахунок вартості комунальних послуг за допомогою копіювання створених формул. Проаналізуйте отримані результати оплат.

1. Відкрийте вікно книги *Послуги*.
2. Виділіть клітинку *D4*, наведіть вказівник миші на маркер автозаповнення та виконайте протягування вниз до клітинки *D6*. Перегляньте формули, розміщені в клітинках *D5* і *D6*.
3. Виділіть клітинку *F4* та скопіюйте її вміст за допомогою буфера обміну. Виділіть діапазон клітинок *F5:F6*, натисніть комбінацію клавіш *Ctrl+V* для вставлення. Перегляньте отримані формули (мал. 16.13).
4. Виділіть клітинку *F8* та введіть формулу $=SUM(F4:F6)$. Мал. 16.13

Рахунок на сплату житлово-комунальних послуг					
Показники лічильників	Поточні	Попередні	Спожито	Тариф (грн за КВт/год або куб.м)	До сплати
Електроенергія	7480	7330	150	0,90	135
Холодне водопостачання	708	683	25	16,72	418
Гаряче водопостачання	560	541	19	90,54	1720,26
Разом					2273,26

5. Збережіть результати у файлі з тим само ім'ям. Проаналізуйте їх. Чи варто для цього використовувати вбудовані функції? Чому?



Досліджуємо

Вправа 4. Презентація-пояснення.

Завдання. Створіть презентацію пояснення, як працювати з формулами та функціями в *Таблицях Google*.


1. У персональному *Диску Google* створіть *Google таблицю*.
2. Виконайте вправи 2–3, використовуючи засоби онлайн-ового середовища.
3. Створіть презентацію з поясненням та ілюстрацією інструментів і дій, які ви застосовували. Надішліть створену презентацію на навчальну скриньку.


Обговорюємо


1. Як визначити, що у клітинці електронної таблиці міститься формула?
2. Як визначається порядок виконання операцій у формулах електронної таблиці? Чи подібний запис формул у середовищі табличного процесора до того, який використовується при вивченні математики?
3. Які існують способи копіювання формул в електронних таблицях?
4. Чим відрізняються в середовищі табличного процесора формули від функцій?
5. Яку послідовність дій потрібно виконати, щоб засобами табличного процесора обчислити середній бал учня чи учениці за семестр?
6. Які види помилок можуть виникати під час введення даних і формул та які існують способи їх усунення?

Працюємо в парах

7. Пограйте у гру «Вбудовані функції». По черзі наведіть приклади електронних таблиць, у яких необхідно обчислити суму, середнє значення, максимальне та мінімальне значення для деяких діапазонів даних.

 8. Із чого можуть складатися формули електронних таблиць? Відобразіть за допомогою схеми. Обговоріть у парі.

 9. Визначте, як побудувати електронну таблицю для знаходження вартості покупки продуктів у супермаркеті. Обговоріть у парі.

 10. За даними таблиці *Навчальні досягнення учнів* (мал. 16.14) обговоріть, як обчислити значення середнього балу успішності кожного учня чи учениці й середній бал успішності всіх учнів з кожного предмета. Кожен виконайте завдання запропонованим способом. Порівняйте отримані результати.

 Мал. 16.14

	A	B	C	D	E	F
1	<i>Навчальні досягнення учнів</i>					
2	Учень	Біологія	Фізика	Інформатика	Укр.літ.	Історія
3	Бондар В.	11	10	12	11	11
4	Гора А.	8	9	10	8	8
5	Король І.	8	10	9	10	11
6	Мороз К.	6	8	9	7	7
7	Стецюк П.	10	8	10	8	10



Працюємо самостійно



- 11.** Заповніть таблицю використання вбудованих функцій табличного процесора у вправі за посиланням <https://learningapps.org/view8773654>.
- 12.** Розрахуйте потрібну суму на закупку розсади полуниці сорту «королева Вікторія» за таких умов:
- потрібно купити 1000 кущів;
 - саджанці продають пакетами, кожний пакет містить 10 саджанців і коштує 169 грн;
 - за першу сотню саджанців покупець платить повну вартість, за кожну наступну сотню вартість зменшується на 1 % від попередньої.

Примітка: для зручності здійснення розрахунків можна визначити ціну для кожної сотні саджанців.

- 13.** Складіть таблицю множення. Для цього створіть структуру таблиці, введіть дані — цифри від 1 до 9, формули для обчислення першого значення кожного рядка та скопіюйте їх у клітинки всього рядка. Для допомоги скористайтеся відеоматеріалами (<https://cutt.ly/sr10JpR>).



- 14.** За даними таблиці *Населення найбільших міст* за допомогою вбудованих функцій табличного процесора знайдіть найменше значення, найбільше та суму значень кількості жителів міст (мал. 16.15).

	A	B	C
1	Населення найбільших міст		
2			
3	Місто	Країна	Населення, осіб
4	Шанхай	Китай	24 150 000
5	Карачі	Пакістан	23 500 000
6	Делі	Індія	17 838 842
7	Лагос	Нігерія	17 060 307
8	Стамбул	Туреччина	14 160 467
9	Гуанчжоу	Китай	12 700 800
10	Мумбаї	Індія	12 655 220
11	Москва	Росія	12 111 194
12	Дакка	Бангладеш	12 043 977
13	Каїр	Єгипет	11 922 949
14	Сан-Паулу	Бразилія	11 895 893
15	Лахор	Пакістан	11 318 745
16	Шеньчжень	Китай	10 467 400
17	Сеул	Корея	10 388 055

Мал. 16.15

- 15.** Уміст калорій у продуктах харчування залежить від їхнього складу. Наприклад, 1 г жирів виділяє 9 ккал, 1 г вуглеводів — 3,8 ккал, 1 г білків — 4 ккал. Визначте кількість калорій, що містяться в кож-

ному продукті з таблиці *Таблиця калорійності* (мал. 16.16). Побудуйте відповідну таблицю в середовищі табличного процесора. Виконайте обчислення у клітинці E3. Виконайте автозаповнення формулами інших клітинок діапазону E4:E12.

	A	B	C	D	E
1	Таблиця калорійності				
2	Назва продукту	Білки (г)	Жири (г)	Вуглеводи (г)	Ккал
3	Молоко	2,8	3,2	4,7	
4	Хліб житній	4,7	0,7	49,8	
5	Крупа вівсяна	11,8	5,5	65,3	
6	Баклажани	0,6	0,1	5,5	
7	Картопля	2	0,1	19,5	
8	Огірки ґрунтові	0,8	0	3	
9	Яблука	0,4	0	11,5	
10	Грейпфрут	1,2	0	7,9	
11	Телятина	19,7	1,2	0	
12	Кури	20,8	8,8	0,6	

 **Мал. 16.16**

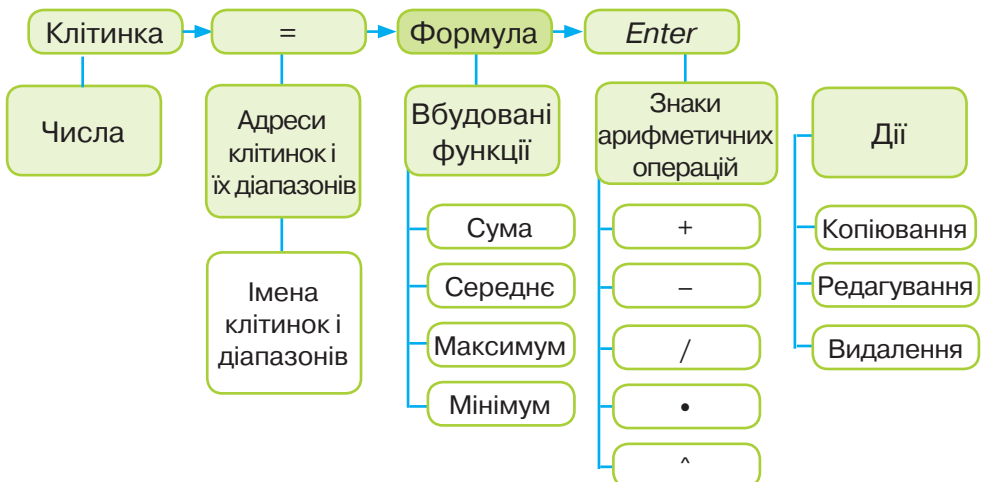


- 16.** Розробіть електронну таблицю для обчислення кількості рулонів шпалер, потрібних для обклеювання трьох однакових кімнат готелю. Потрібні дані та підказки щодо виконання завдання можна знайти у файлі *Ремонт*. Задайте клітинці B7 ім'я *Площа_кімнати*, E5 — *Площа_рулону*. При введенні у клітинку D9 формули замість адрес відповідних клітинок використайте їхні імена: *Площа_кімнати* та *Площа_рулону*.

Повторюємо



Розгляньте схему та поясніть зв'язок між вказаними поняттями.





Поняття моделі

ВИ ДІЗНАЄТЕСЬ:

- ▶ що таке предметна галузь;
- ▶ чому люди користуються моделями та створюють їх;
- ▶ як можна класифікувати моделі.



Вивчаємо

1. Що таке предметна галузь?

Ви вже знаєте, що нас оточують різні об'єкти: предмети, живі істоти, явища, процеси. Кожний об'єкт має свої властивості та їхні значення — дані різного типу. Об'єкти, що мають однакові властивості, можна об'єднувати в групи — класифікувати. З об'єктами також можна виконувати певні дії. Крім того, різні об'єкти взаємодіють між собою: планети утворюють Сонячну систему, молекули — різні сполуки, учні та вчителі взаємодіють у школі. Сукупність об'єктів, що мають однакові властивості й перебувають у певних відношеннях, утворюють предметну галузь.

Предметна галузь — сукупність усіх об'єктів, властивості яких і відношення між якими розглядаються в процесі діяльності.

Предметними галузями можуть бути, наприклад, архітектура, промислове підприємство, транспорт, освіта тощо.

Предметну галузь часто називають галуззю дослідження, областю вивчення чи колом інтересів. В англійській мові це *Field of Study*.

Одну й ту само предметну галузь можуть досліджувати люди різних професій чи віку. При цьому вони виділяють об'єкти з цієї галузі або розглядають різні властивості об'єктів залежно від завдання, яке виконується. Наприклад, об'єкти предметної галузі *біологія* можуть розглядати учні на уроках, науковці в лабораторіях чи дослідники-мандрівники в реальних умовах.

2. Чому люди користуються моделями та створюють їх?

Об'єкти, що оточують людину, належать до різних предметних галузей і є досить складними. Деякі реальні об'єкти досліджувати в повній мірі складно. Наприклад, без додаткового сучасного обладнання

неможливо провести наукові дослідження з об'єктами, що перебувають на значній відстані, зокрема з космічними, небезпечними, дуже малими або дуже великими об'єктами. При розв'язуванні кожної конкретної задачі людину можуть цікавити лише деякі з властивостей об'єктів, тоді говорять про їхні суттєві властивості.

Наприклад, якщо ви з однокласниками вирішите висадити на шкільному подвір'ї квітковий годинник (мал. 17.1), то для предметної галузі *садовий дизайн* суттєвими властивостями об'єкта *квітка* будуть: розмір, час цвітіння, час розкривання та закривання квітів. Але якщо добиратимете квіти для гербарію, то для предметної галузі *ботаніка* вас цікавитимуть зовсім інші властивості: збереження кольору квітки після сушіння, ступінь усушки тощо.



Мал. 17.1

На основі даних про властивості, що є суттєвими для конкретної задачі, можна створити інший об'єкт, який спрощено відображатиме ці властивості реального. Такі спрощені об'єкти називають **моделями** (мал. 17.2).

Мал. 17.2



Модель об'єкта
Динозавр



Модель явища
Зміна дня і ночі
та пір року



Модель процесу
Рух електронів навколо
ядра

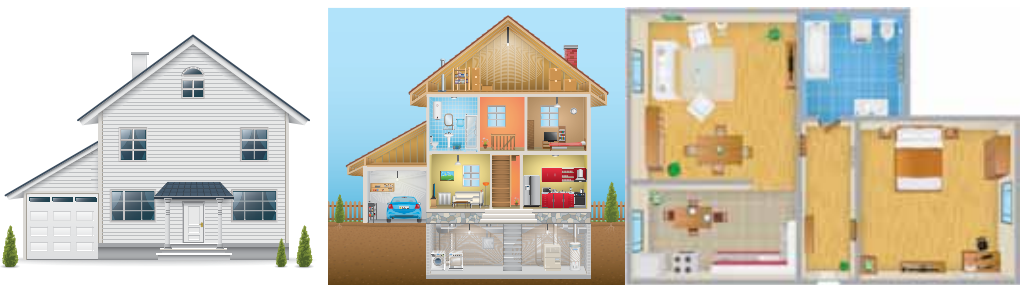
Модель — спрощене подання предмета, істоти, явища чи процесу. Слово модель походить від латинського *modulus* (міра, зразок, норма) й означає копію або образ.

Модель — це матеріальний або уявний об'єкт, який у процесі вивчення деякої предметної галузі (навчання) замінює реальний об'єкт-оригінал, зберігаючи важливі для даного дослідження суттєві властивості оригіналу. Моделями об'єктів є знайомі вам карта України та глобус у кабінеті географії, моделі органів людини в кабінеті біології, план спортивного майданчика школи, словесний опис незнайомої людини, малюнок веселки, звукозапис співу птахів, відеозапис північного саява тощо.

Люди створюють і використовують моделі з різних причин.

- Оригінал уже не існує або ще не існував. Наприклад, розробники науково-популярного фільму «Прогулянки з динозаврами» використовували під час зйомки моделі вимерлих тварин. А будівельники спочатку працюють з моделлю будинку, для цього створюють її з пластику чи креслять на папері.
- Розміри оригіналу не дають змоги ефективно досліджувати його. Наприклад, вивчати явище зміни дня і ночі, пір року на планеті Земля можна за допомогою спеціальної моделі, оскільки реальне дослідження обмежене значними розмірами Землі та Сонця.
- Дослідження оригіналу може призвести до значних матеріальних втрат і зашкодити здоров'ю людини. Наприклад, уявлення про роботу органів людини студенти-медики спочатку отримують за допомогою спеціальних моделей, а випробування літака варто здійснювати спочатку на моделі через його високу вартість.

Для опису або дослідження одного й того самого об'єкта можна використовувати декілька моделей (мал. 17.3). Це залежить від призначення моделі та визначення предметної галузі, до якої належить її об'єкт при розв'язуванні конкретної задачі.



Мал. 17.3

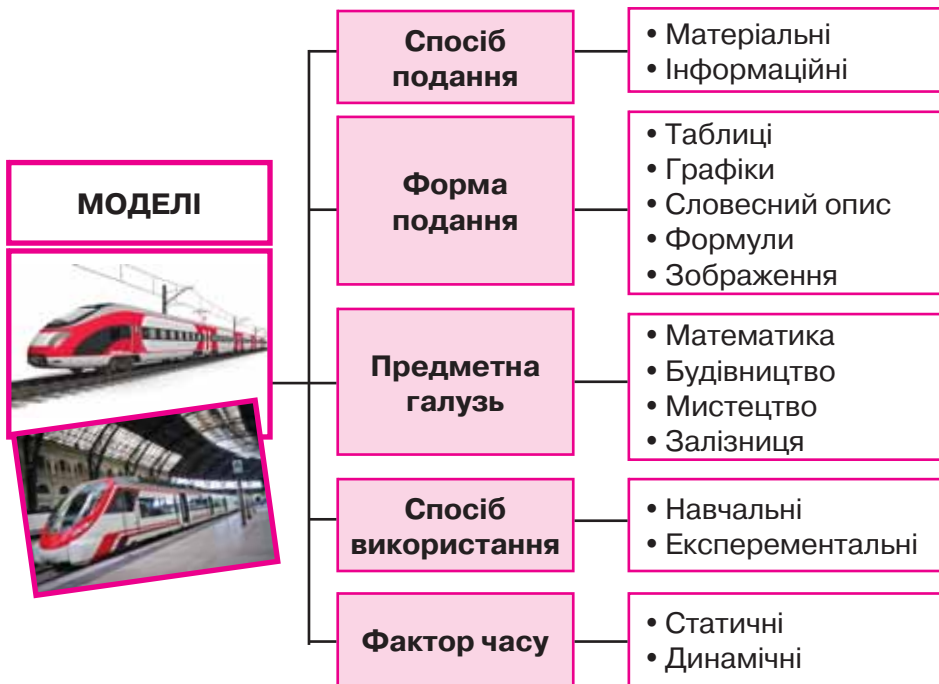
І навпаки, для опису та дослідження різних об'єктів може бути використано одну й ту саму модель. Наприклад, модель підлоги в кімнаті, вікна у тролейбусі, присадибної ділянки можна подати за допомогою геометричної фігури — прямокутника, оскільки кожна з них можна розглядати у предметній галузі *геометрія*.

Отже, у процесі пізнання та практичної діяльності людина широко застосовує різноманітні моделі для дослідження реального об'єкта в різних ситуаціях. Більше того, будь-яка наука починається з розробки моделей.

Моделювання — це процес дослідження властивостей об'єкта на основі створення відповідної моделі.

3. Як можна класифікувати моделі?

Усе розмаїття моделей, з якими має справу людина, можна класифікувати за різними ознаками: способом і формою подання, предметною галуззю, фактором часу та іншими (мал. 17.4).



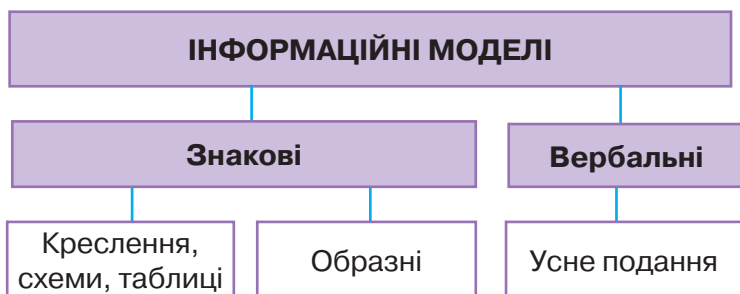
Мал. 17.4

За способом подання моделі поділяють на матеріальні й інформаційні. **Матеріальна модель** — це модель, яка відтворює геометричні та фізичні властивості об'єкта-оригіналу. Вона завжди має реальне втілення. Наприклад, дитячі іграшки, макети різноманітних споруд, технічних конструкцій — це матеріальні моделі реальних об'єктів.

Модель, яку подають за допомогою описів, формул, зображень, схем, таблиць, креслень, графіки тощо називають *інформаційною*.

Інформаційна модель — це модель, що містить опис інформаційного процесу або об'єкта, у якому вказано лише суттєві властивості, важливі для розв'язування задачі у визначеній предметній галузі.

За формою подання розрізняють вербальні та знакові інформаційні моделі (мал. 17.5).



Мал. 17.5

Знакова інформаційна модель — це модель, у якій властивості об'єкта описуються за допомогою певної системи знаків: математичних виразів і рівнянь, фізичних або хімічних формул, нотних записів тощо. Наприклад, за допомогою відомої вам формули $S = vt$ описують відстань, яку проїхав автомобіль зі швидкістю v за час t . А сімейна фотографія є прикладом образної моделі.

Оскільки інформаційні моделі доступні для опрацювання на комп'ютері, то для них у різних середовищах створюють комп'ютерну модель.

Комп'ютерна модель — це інформаційна модель, яку подано у знаковій формі та реалізовано за допомогою комп'ютера.



ДІЄМО

Вправа 1. Використання моделей людьми.

Завдання. За матеріалами файла *Розмір Всесвіту* (мал. 17.6) знайдіть підтвердження причин використання людьми моделей. Доповніть відповідну інформаційну модель у середовищі текстового процесора, користуючись шаблоном *Побудова моделей*, що збережений у папці *Моделювання*.



Мал. 17.6

1. Завантажте з папки *Моделювання* шаблон *Побудова_моделей.dotx* (*Побудова_моделей.ott*). Поміркуйте, які зображення можуть підтверджувати запропоновані причини використання людьми моделей замість реальних об'єктів.
2. Скористайтеся матеріалами програми *Розмір Всесвіту*, яку можна відкрити за допомогою браузера з папки *Моделювання*.
3. Запустіть модель шкали масштабів Всесвіту.
4. Знайдіть потрібні зображення. Скопіюйте зображення в буфер обміну та додайте їх до текстового документа. Повторюйте зазначену дію доти, доки інформаційна модель не буде побудована.
5. Збережіть файл з іменем *Модель* у своїй структурі папок.

Вправа 2. Атлас людини.

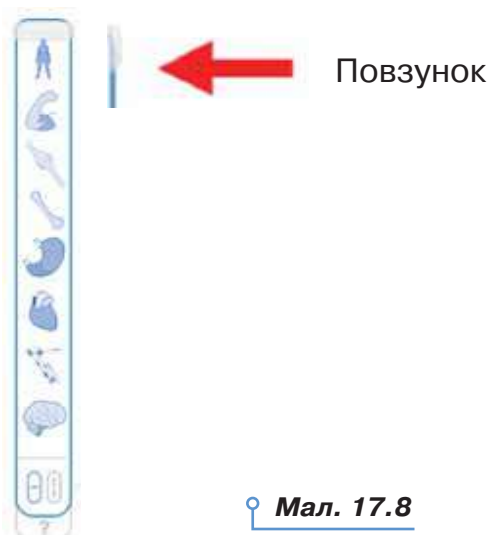
Завдання. Визначте, які засоби онлайн-сервісу *ZygotеBody* підтверджують те, що це середовище реалізує комп'ютерну модель анатомії людини.

1. У полі адреси браузера введіть <https://zygotebody.com/> та дочекайтеся завантаження онлайн-середовища.
2. У списку компонентів оберіть статтю моделі людини (мал. 17.7).



Мал. 17.7


3. Рухайте повзунком на панелі відображення систем органів, яка задає рівні глибини проникнення: від покривної системи — до нервової (мал. 17.8).



Мал. 17.8

4. Управляйте процесом перегляду моделі: повертайте модель за до-



помогою стрілок у вікні середовища або скористайтесь відповідними клавішами клавіатури; змінійте розмір моделі за допомогою інструментів масштабування  або коліщатка миші.

5. Зробіть висновок за умовою завдання та завершіть роботу із середовищем.



ДОСЛІДЖУЄМО

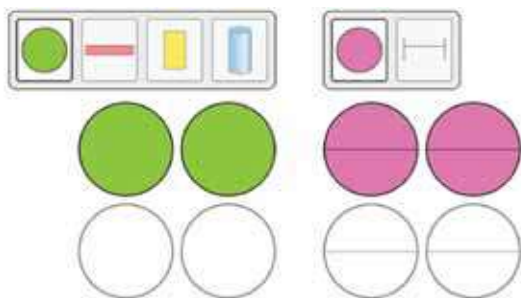
Вправа 3. Дроби.

Завдання. Дослідіть модель, яка демонструє рівність дробів за посиланням: <https://cutt.ly/Yr102Ma>. Знайдіть три підтвердження того, що ця комп'ютерна модель реалізує реальні математичні завдання.



1. Перейдіть за посиланням в умові завдання.
2. Використовуйте геометричні фігури для моделювання дробів і по-

взунки для зміни кількості фігур та їх поділів  та їх поділів  (мал. 17.9)



$$\frac{2}{1} = \frac{4}{2}$$

 Мал. 17.9

3. Зробіть висновок за умовою задачі.

Вправа 4. Місто.

Завдання. Розгляньте модель міста Тернополя, подану за адресою <https://cutt.ly/url2qgY>. Оберіть зручну для вас мову інтерфейсу за допомогою ін-

струмента **UK** ▾. Дослідіть, які зміни відбуватимуться на екрані при виборі різних моделей за допомогою списку, що з'являється при виборі інструмента *Wikimapia супутник* у правому верхньому куті екрана: супутник, мапа, вид із супутника *Google* тощо (мал. 17.10). Поміркуйте, для чого можуть бути призначені запропоновані моделі.



Мал. 17.10

Обговорюємо

1. Чи можна вважати людину об'єктом дослідження?
2. Що називають предметною галуззю?
3. Що розуміють під поняттям модель?
4. Чому для одного реального об'єкта можна створити декілька моделей?
5. Чим визначається вибір суттєвих властивостей, на основі яких буде створюватися модель?
6. Як можна класифікувати моделі?
7. Що таке інформаційна модель? Які особливості мають інформаційні моделі?
8. Запитайте в батьків, чи вони колись використовували інформаційні моделі.

Працюємо в парах

9. Обговоріть, що із запропонованих зображень може бути об'єктом, моделлю та яку модель можна поставити у відповідність визначеному об'єкту у вправі за посиланням <https://learningapps.org/view8777899>.
- ★ 10. З переліку назв різних моделей виділіть матеріальні та інформаційні:
 - глобус;
 - іграшковий автомобіль;
 - атлас з історії;
 - збірка поезій;
 - план евакуації школи;



- м'яч;
 - схема руху потягів метрополітену.
- Обговоріть у парі. Доповніть список іншими прикладами.

11. До кожної з поданих на малюнку 17.4 ознак класифікації наведіть по три приклади моделей. Обговоріть у парі.



Працюємо самостійно

12. Визначте моделі об'єктів в описаних ситуаціях і встановіть причину їх використання замість їхніх реальних аналогів.

- На уроках географії учні вивчають рельєф за допомогою фізичної карти світу.
- Конструктори АКБ «Антонов» завершили випробування комп'ютерної моделі нового літака.
- За допомогою комп'ютерних технологій науковцям вдалося відтворити вигляд та поведінку прадавніх тварин.
- Курсанти пройшли курс навчання водінню літаків на тренажерах.
- В офтальмології широко застосовується штучний кришталік ока.

13. Розподіліть моделі у групи: інформаційні та матеріальні — у вправі за посиланням <https://learningapps.org/view8777787>.

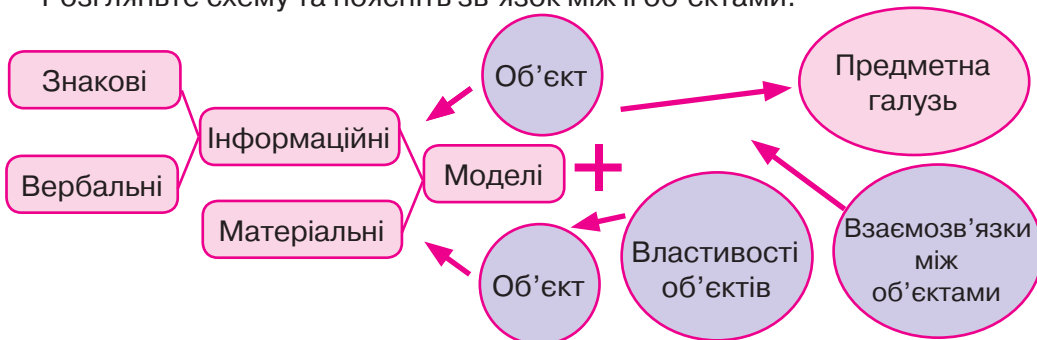


14. Наведіть приклади матеріальних моделей, які трапляються щодня; знакових моделей, які трапляються при вивченні інших предметів: географії, хімії, біології тощо; знакових інформаційних моделей, які подаються за допомогою карт, схем, креслень, графіків. Визначте, які моделі вам трапляються найчастіше. Складіть схему класифікації таких моделей. Ознаки класифікації доберіть самостійно.

Повторюємо



Розгляньте схему та поясніть зв'язок між її об'єктами.





Етапи створення інформаційної моделі

ВИ ДІЗНАЄТЕСЬ:

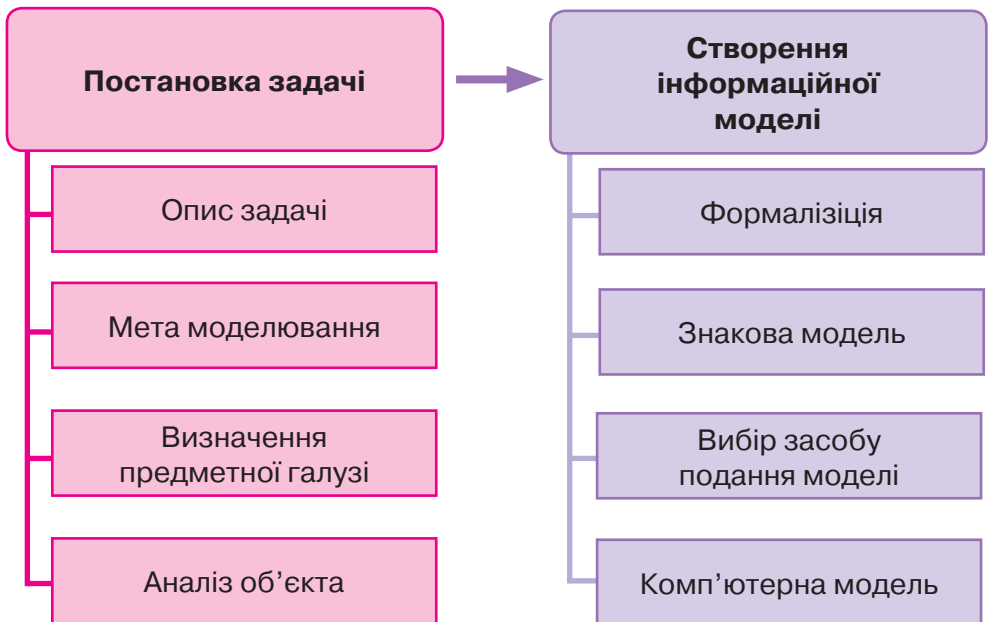
- ▶ з яких етапів складається процес створення інформаційної моделі;
- ▶ для чого використовується комп'ютерне моделювання.



Вивчаємо

1. З яких етапів складається процес створення інформаційної моделі?

Чітких правил створення моделі не існує. Однак завжди потрібно розпочинати з постановки задачі.



Мал. 18.1

Спочатку задача формулюється звичайною мовою — здійснюється її опис. На цьому етапі важливо визначити мету створення моделі та предметну галузь дослідження, детально описати об'єкт моделювання

та його властивості, з'ясувати, які з них є суттєвими для даної задачі, взаємозв'язки між значеннями вибраних властивостей, між даним об'єктом й іншими об'єктами предметної галузі, які є суттєвими для задачі, та визначити очікуваний результат (мал.18.1).

На етапі створення інформаційної моделі спочатку проводиться формалізація — етап переходу від словесного опису зв'язків між об'єктами та їхніми властивостями до побудови інформаційних моделей за допомогою деякої формальної мови кодування (мова схем, мова математики тощо).

Як правило, у результаті формалізації модель подається в знаковій формі: математична формула, схема, графік, таблиця, карта місцевості, креслення, електрична схема тощо.

Формалізація — процес створення інформаційної моделі за допомогою формальних мов, за якого здійснюється перехід від словесного опису зв'язків між виділеними властивостями об'єкта до опису, що використовує деяку мову кодування (мова схем, мова математики тощо).

Якщо між величинами, які характеризують об'єкт або процес, встановлено співвідношення у вигляді математичних формул, то говорять про створену **математичну** модель.

За допомогою математичних моделей описуються розв'язки різних логічних задач, фізичних процесів. Наприклад, модель рівномірного прямолінійного руху описується рівнянням: $x = x_0 + v \cdot t$. Оскільки знакова форма подання моделі доступна для опрацювання за допомогою комп'ютера, то можна створювати комп'ютерну модель, тобто інформаційну модель, реалізовану за допомогою комп'ютерних програм.

2. Для чого використовується комп'ютерне моделювання?

Моделі, що досліджуються за допомогою комп'ютера, можуть описувати досить різноманітні об'єкти, такі як мости, архітектурні споруди, літаки тощо, а також імітувати їх функціонування, протікання різноманітних процесів, пов'язаних із ними. Дослідження таких моделей дає змогу вивчити властивості багатьох об'єктів без безпосереднього доступу до них. Це допомагає суттєво зменшити матеріальні та часові витрати для вивчення властивостей ще не споруджених будинків, мостів, літаків, двигунів тощо.

Крім виконання числових розрахунків, комп'ютерне моделювання дає змогу відтворити явища, які в реальних земних умовах людині відтворити не під силу. Це, наприклад, рух материків, дія землетрусів, народження нової зірки, зміна напрямків морських підводних течій тощо. При вивченні цих явищ на допомогу приходять спеціальні

комп'ютерні програми, які складаються кваліфікованими програмістами разом із фахівцями: фізиками, географами, біологами тощо.

Комп'ютерне моделювання надає також унікальні переваги для опису й розрахунку небезпечних експериментів, які не бажано виконувати в реальному житті. Це, наприклад, моделі ядерного вибуху, пожежі на підприємстві, військових дій, зіткнення автомобілів чи поїздів тощо. За допомогою комп'ютерних моделей можна досить точно отримати деталі цих катастрофічних процесів.

Існують різні комп'ютерні програми для створення, опрацювання та дослідження інформаційних моделей. Одну й ту само задачу можна розв'язати, скориставшись різними програмами. Обране програмне середовище дає змогу за певним алгоритмом створювати інформаційну модель. Добір потрібних програм для створення інформаційної моделі залежить від мети її створення, знань й умінь дослідника чи дослідниці, призначення обраного комп'ютерного середовища.



ДІЄМО

Вправа 1. Економія.

Завдання. Звичайна лампа розжарювання споживає 100 Вт/год. електроенергії, а енергозберігальна — 20 Вт/год. Побудуйте математичну модель та реалізуйте її засобами табличного процесора й мови програмування для визначення економії домашнього бюджету за умов, що лампа горить 7 год на добу й вартість 1 Вт становить V грн.

1. Опишемо інформаційну модель.

Мета: визначити різницю вартості електроенергії двох споживачів.

Предметна галузь: тарифи.

Суттєві властивості: обсяг споживання, час використання, вартість.

Формалізація: використаємо такі величини: H — обсяг споживання

звичайної лампи; L — обсяг споживання енергоефективної лампи;

T — час використання; V — вартість електроенергії.

2. Побудуємо математичну модель для визначення економії R :

$$R = (H - L) \cdot T \cdot V.$$

3. Реалізуйте модель засобами табличного процесора, враховуючи переведення коштів у гривні (мал. 18.2).

	A	B	C	D	E
1	Обсяг споживання				
2	Звичайна лампа	Енергоефективна лампа	Час	Вартість 1 Ват, коп	Економія, грн
3	H	L	T	V	R
4	100	20	7	0,168	0,94

Мал. 18.2

4. Збережіть файл з іменем *Економія_Таблиця* в папці *Моделі* своєї структури папок.
5. Реалізуйте модель мовою програмування *Python* у середовищі *Thonny*.


```
H=int(input('Обсяг споживання звичайної лампи'))
L=int(input('Обсяг споживання енергоефективної лампи'))
T=int(input('Час роботи'))
V=float(input('Вартість 1 Вт'))
R=(H-L)*T*V/100
print('Економія', R)
```
6. Збережіть файл з іменем *Економія_Програма* в папці *Моделі* своєї структури папок.
7. Порівняйте обидва результати. Зробіть висновок.

Обговорюємо

1. Чи існує універсальний алгоритм створення інформаційних моделей? Відповідь поясніть.
2. З яких етапів складається процес створення інформаційної моделі? Чи можна змінювати послідовність етапів створення інформаційної моделі?
3. Від чого залежить вибір форми подання інформаційної моделі? Наведіть конкретні приклади.
4. Чи можна створити одну й ту само модель за допомогою різних комп'ютерних програм?
5. Що таке формалізація моделі? Чому цей етап створення інформаційної моделі важливий?

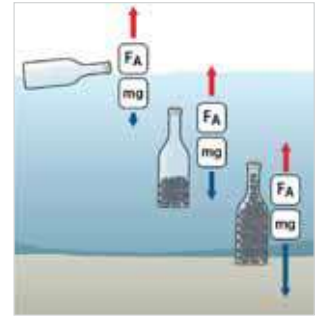
Працюємо в парах

6. Наведіть приклади задач, розв'язування яких неможливе без комп'ютерного моделювання. Обговоріть у парі.
7. Пограйте у гру «Модель — середовище створення». Хтось називає об'єкт моделі, хтось — програмне середовище для створення моделі. Перший учасник або учасниця має висловити припущення, чому вказано саме це програмне середовище. Після обговорення поміняйтесь у парі ролями.
8. У комп'ютерній моделі калькулятора місьць у літаку за посиланням <https://cutt.ly/4yPaY6R> побувайте за допомогою блоків модель формули обчислення кількості місьць, яка враховує кількість визначених рядів (мал. 18.3). Обговоріть отриманий результат.





Мал. 18.3



Мал. 18.4

- ★ 9. Дію закону Архімеда проілюстровано на графічній моделі (мал. 18.4). Пригадайте з уроків фізики, як обчислюють значення F_A або скористайтесь довідниками. Побудуйте відповідну математичну модель. Обговоріть, що краще використати для реалізації математичної моделі на комп'ютері. Створіть модель та порівняйте її з моделлю сусідів справа.
- ★★ 10. Відомо, що важіль перебуває в рівновазі, якщо добуток сили на плече обабіч опори однаковий: $F_1 * l_1 = F_2 * l_2$. Побудуйте математичну модель визначення місця розташування опори гойдалки, щоб вона перебувала в рівновазі за умов відомих F_1 , F_2 та загальної довжини дошки для гойдання A . Обговоріть складені моделі в парі. Порівняйте, чи можна побудувати різні моделі. Реалізуйте модель на комп'ютері у вибраній самостійно програмі.

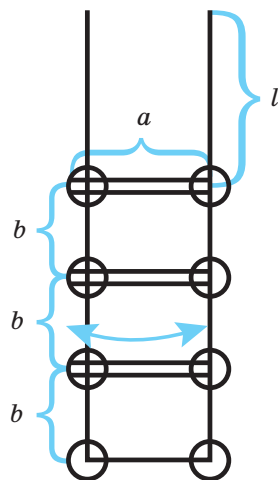


Працюємо самостійно

11. Відновіть порядок дій в алгоритмі побудови інформаційної моделі у вправі за посиланням <https://learningapps.org/view8778963>.
12. Для шкільної урочистої лінійки ви плануєте обгородити шкільний майданчик кольоровими прапорцями. Створіть математичну модель розв'язування цієї задачі. Реалізуйте її у вибраному комп'ютерному середовищі.
- ★ 13. За день машина проїжджає n кілометрів. Скільки днів потрібно, щоб проїхати маршрут m кілометрів? Створіть математичну комп'ютерну модель розв'язування цієї задачі.
- ★★ 14. Банк залучає від вкладників кошти під 14,5 % річних та віддає їх як кредити позичальникам під 1,75 % на місяць. Різниця між цими коштами — прибуток банку. Складіть математичну модель такої діяльності банку та реалізуйте її у вибраній самостійно програмі.

15. У вашого двоюрідного брата скоро день народження — йому виповниться 7 років. Його батьки звернулися до вас по допомогу у створенні кошторису для проведення дитячого свята. Брат хотів би запросити 11 друзів. Пригощати гостей заплановано вдома двома видами тістечок, соком і морозивом, які можна придбати в одному із супермаркетів. Кімнату можна прикрасити повітряними кульками власними силами або замовити оформлення в агенції «Веселий клоун», про яку батьки багато чули та хотіли б отримати її контактні дані, щоб запросити клоуна для проведення свята. Побудуйте математичну модель розрахунку кошторису свята, який би був найбільш економним для сімейного бюджету. Реалізуйте модель у табличному процесорі або середовищі програмування.

16. Взуттєва фабрика планує випуск нового чоловічого взуття. Дірки для шнурівки будуть розташовані у два ряди, відстань між рядами a , а відстань між дірками в ряді — b . Кількість дірок у кожному ряді n . Дизайнери розробили спеціальний спосіб шнурування (мал. 18.5). Крім цього передбачено, що довжина вільного кінця шнурівки становитиме l . Яка має бути довжина шнурівки?

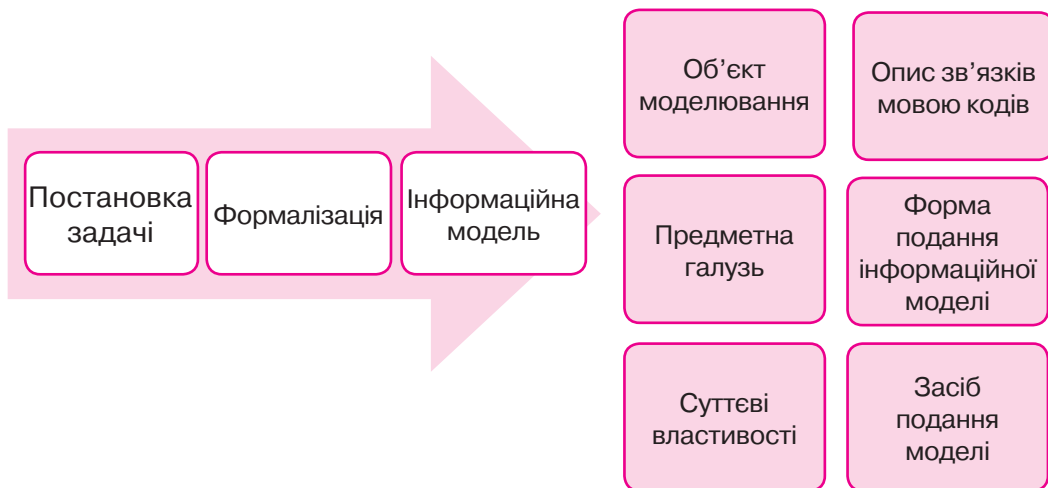


Мал. 18.5

Повторюємо

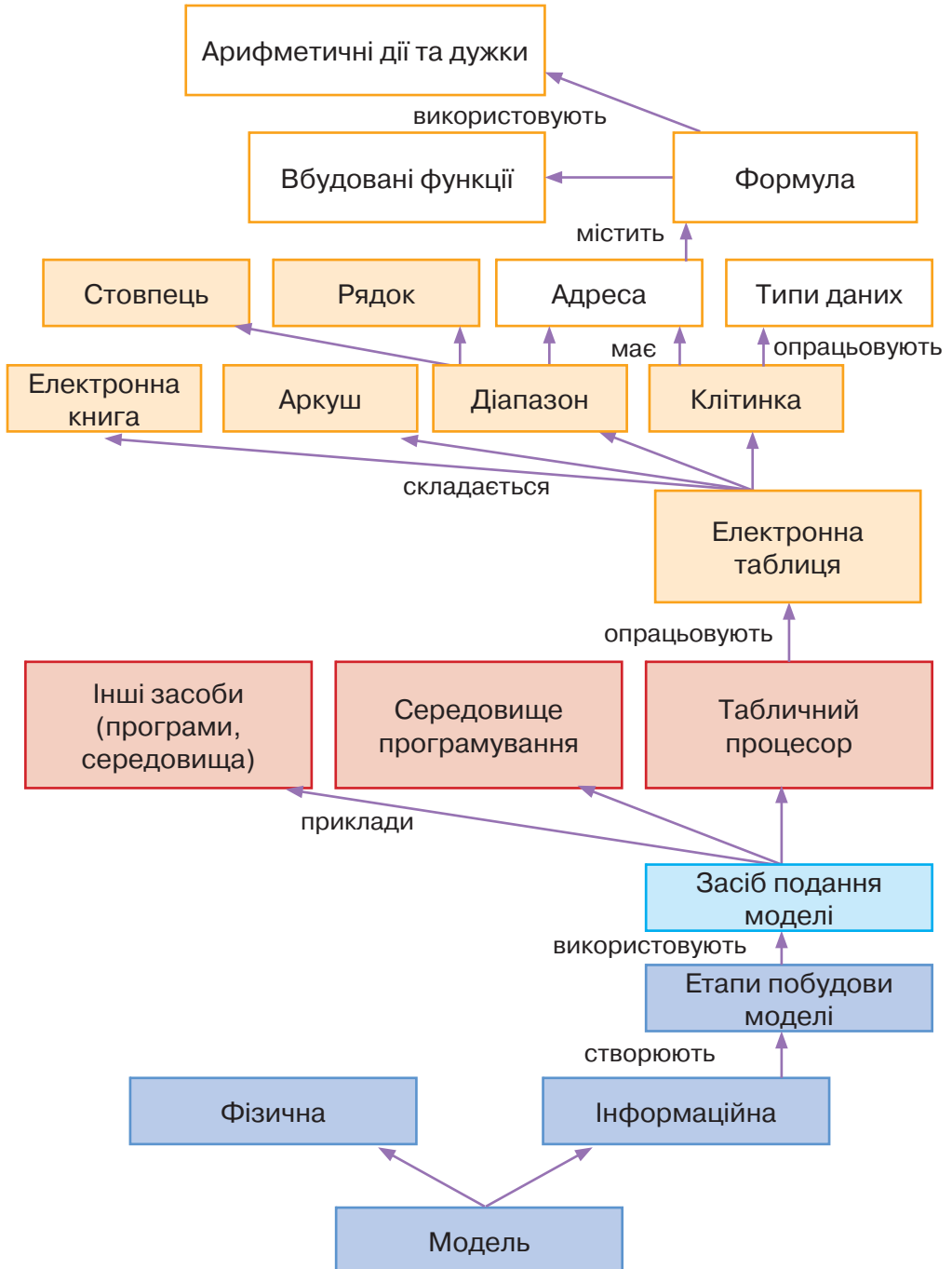


Розгляньте схему, установіть потрібні зв'язки між її елементами та складіть коротке повідомлення.





Узагальнюємо



Корисні посилання

Працюємо з електронною поштою
(<https://cutt.ly/auLjr6U>)



Безпека в Інтернеті
(<https://cutt.ly/auLkROg>)

Інтернет речей
(<http://thefuture.news/iot/>)



Інтерактивний підручник *Python*
(<https://cutt.ly/PuLzhTF>)

Путівник мовою програмування
Python (<https://cutt.ly/quLz1V7>)



Ігрове середовище для створення
програм мовою *Python*
(<https://cutt.ly/1uLxtMF>)

Табличний процесор *Gnumeric*
(<http://www.gnumeric.org/>)



ЗМІСТ

Розділ 1. Електронне листування

1. Поштові служби Інтернету. Створення електронної скриньки	4
2. Використання електронної пошти.	10
3. Безпека в інтернеті	21
4. Інтернет-ресурси для спільної роботи	29
5. Навчання в інтернеті	44
6. Інтернет речей	50

Розділ 2. Алгоритми та програми

7. Мова програмування. Середовище програмування	61
8. Величини. Змінні. Вказівка присвоювання	69
9. Додаткові модулі	78
10. Текстові величини та операції над ними	85
11. Опрацювання величин логічного типу. Команда розгалуження.	92
12. Реалізація алгоритмів повторення мовою програмування	103

Розділ 3. Опрацювання табличних даних

13. Електронні таблиці та їхні об'єкти.	114
14. Редагування електронних таблиць.	123
15. Формати даних та форматування таблиць.	134
16. Адресація. Формули. Обчислення в електронних таблицях	146
17. Поняття моделі	159
18. Етапи створення інформаційної моделі.	168
Корисні поради.	175