

Посилання на посібник

<https://colab.research.google.com/?hl=uk#scrollTo=-Rh3-Vt9Nev9>

3. ПОСТАНОВКА ЗАДАЧІ

1. Увійти до середовища Google Colab через власний Google-акаунт.
2. Створити новий notebook у середовищі Google Colab.
3. Підключити Google Drive та створити робочу папку для лабораторної роботи.
4. За допомогою мови Markdown та HTML-тегів для країни, відповідно до свого номера варіанту (див. табл. 3.1), подати інформацію щодо:

- площі території;
- чисельності населення;
- столиці;
- державного устрою;
- офіційної мови;
- видатних історичних постатей;
- пам'яток культури;
- природних об'єктів.

При оформленні інформації обов'язково використати:

- заголовки різних рівнів;
- курсив, напівжирний шрифт;
- кольоровий текст;
- таблиці;
- вкладені списки;
- гіперпосилання;
- зовнішні зображення через URL;
- вбудовані зображення;
- HTML-тег `<svg>` для побудови прапора країни.

5. Реалізувати Python-функцію відповідно до номера варіанту (див. табл. 3.2).
6. Для створеної функції реалізувати не менше 3 тестових прикладів.
7. Передбачити контроль правильності введення даних.
8. Побудувати графічну візуалізацію результатів роботи алгоритму (за допомогою matplotlib).
9. Зберегти notebook у Google Drive.
10. Експортувати notebook у форматі `.ipynb` та розмістити на GitHub.

Таблиця 3.1 – Варіанти країн

Варіант	Країна
1	Україна
2	Польща
3	Канада
4	Японія
5	Норвегія
6	Бразилія
7	Іспанія
8	Швейцарія
9	Австралія
10	Італія
11	Франція
12	Швеція
13	Аргентина
14	Казахстан
15	Португалія
16	Фінляндія
17	Грузія
18	Ісландія
19	Чилі
20	Нова Зеландія

Таблиця 3.2 – Індивідуальні алгоритмічні завдання

Варіант	Завдання
1	Знайти всі прості числа до N
2	Реалізувати алгоритм Евкліда
3	Знайти медіану списку
4	Bubble Sort + підрахунок перестановок
5	k-те просте число
6	Прості числа на відрізку
7	Переведення десяткового числа в двійкову систему
8	Gnome sort
9	Вибрати непрості числа і відсортувати
10	Подання числа як суми квадратів
11	Обчислення факторіала
12	Побудова таблиці множення
13	Перевірка паліндрома
14	Підрахунок частоти символів у рядку
15	Генерація чисел Фібоначчі
16	Знаходження найбільшого елемента матриці
17	Обчислення середнього арифметичного списку
18	Транспонування матриці
19	Пошук дубльованих елементів списку

Додатково для Google Colab

Для кожного варіанту обов'язково використати:

```
from google.colab import drive
drive.mount('/content/drive')
```

Для графічної візуалізації

```
import matplotlib.pyplot as plt
```

5. ЗМІСТ ЗВІТУ

1. Титульний аркуш
2. Тема роботи
3. Мета роботи
4. Код notebook
5. Markdown-оформлення країни
6. Алгоритм
7. Результати тестування
8. Графік
9. Посилання на GitHub
10. Висновок

Контрольні запитання

1. Що таке Google Colab?
2. Чим Google Colab відрізняється від Jupyter Notebook?
3. Як підключити Google Drive?
4. Що таке Markdown?
5. Як вставити HTML у notebook?
6. Як створити SVG?
7. Як експортувати notebook?
8. Як працює matplotlib?
9. Як зберегти notebook на GitHub?
10. Які переваги хмарного середовища?

Приклад виконання індивідуального завдання №21

Варіант 21

Країна: Німеччина

Алгоритмічне завдання: визначити, чи є введений рядок паліндромом, якщо не враховувати пробіли, розділові знаки та регістр символів.

1. Підключення Google Drive

```
from google.colab import drive
drive.mount('/content/drive')
```

2. Оформлення інформації про країну за допомогою Markdown та HTML

```
# <span style="color:darkblue;">Німеччина</span>

## Загальна інформація

**Німеччина** – одна з провідних держав Європи, яка має значний вплив на економічний, політичний та культурний розвиток світу.

<div style="border:2px solid gray; padding:10px; border-radius:8px;">
<b>Офіційна назва:</b> Федеративна Республіка Німеччина <br>
<b>Столиця:</b> Берлін <br>
<b>Площа:</b> 357 588 км2 <br>
<b>Населення:</b> понад 84 млн осіб <br>
<b>Державний устрій:</b> федеративна парламентська республіка <br>
<b>Офіційна мова:</b> німецька
</div>

## Видатні постаті

- **Альберт Ейнштейн**
- **Йоганн Вольфганг фон Гете**
- **Людвіг ван Бетховен**
- **Отто фон Бісмарк**

## Пам'ятки культури та природи

1. Бранденбурзькі ворота
2. Кельнський собор
```

3. Замок Нойшванштайн

4. Чорний ліс

Цікаві факти

- Німеччина є однією з найбільших економік світу.
- Має розвинену транспортну інфраструктуру.
- Відома своєю інженерною школою та автомобільною промисловістю.

Вкладений список

- Культура
 - музика
 - література
 - архітектура
- Природа
 - ліси
 - озера
 - гори

Гіперпосилання

[Офіційний туристичний портал Німеччини] (<https://www.germany.travel/en/home.html>)

Зображення за посиланням

![Germany] (https://upload.wikimedia.org/wikipedia/commons/b/ba/Flag_of_Germany.svg)

SVG-прапор

```
<svg width="300" height="180">
  <rect width="300" height="60" y="0" style="fill:black;" />
  <rect width="300" height="60" y="60" style="fill:red;" />
  <rect width="300" height="60" y="120" style="fill:gold;" />
</svg>
```

Таблиця основних характеристик

Показник	Значення
---	---
Столиця	Берлін
Площа	357 588 км ²
Населення	понад 84 млн
Мова	німецька
Валюта	євро

3. Реалізація алгоритмічного завдання

```
import re

def task_21(text):
    if not isinstance(text, str):
```

```
        raise TypeError("Потрібно ввести рядок.")

cleaned = re.sub(r'^[а-яА-Яа-зА-З0-9]', '', text).lower()

if len(cleaned) == 0:
    return "Помилка: рядок не містить літер або цифр."

if cleaned == cleaned[::-1]:
    return f'Рядок "{text}" є паліндромом.'
else:
    return f'Рядок "{text}" не є паліндромом.'
```

4. Тестові приклади

```
print(task_21("Козак з казок"))
print(task_21("А роза упала на лапу Азора"))
print(task_21("Google Colab"))
```

Очікуваний результат:

Рядок "Козак з казок" є паліндромом.

Рядок "А роза упала на лапу Азора" є паліндромом.

Рядок "Google Colab" не є паліндромом.

5. Приклад контролю введення

```
try:
    print(task_21(12345))
except Exception as e:
    print("Помилка:", e)
```

Очікуваний результат:

Помилка: Потрібно ввести рядок.

6. Візуалізація результатів

```
import matplotlib.pyplot as plt

examples = ["Козак з казок", "А роза упала на лапу Азора", "Google Colab", "Level", "Python"]
results = []

for item in examples:
    cleaned = re.sub(r'^[а-яА-Яа-зА-З0-9]', '', item).lower()
    results.append(1 if cleaned == cleaned[::-1] else 0)

plt.figure(figsize=(8,4))
plt.bar(examples, results)
plt.title("Перевірка рядків на паліндром")
plt.ylabel("Результат")
plt.xticks(rotation=20)
plt.show()
```

7. Висновок

У ході виконання індивідуального завдання №21 було створено notebook у Google Colab, оформлено інформаційний блок про Німеччину засобами Markdown та HTML, а також реалізовано функцію перевірки рядка на паліндром. Програма враховує коректність введених даних, ігнорує пробіли, розділові знаки та регістр символів. Для перевірки роботи функції наведено тестові приклади та побудовано графічну візуалізацію результатів.