

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж



Лабораторна робота №7  
з дисципліни Спеціалізовані мови програмування  
на тему  
Робота з API та веб-сервісами

Виконав:  
студент групи РІ-21сп  
Владислав ДМИТРЕНКО

Львів – 2024

**Мета:** Створення консольного об'єктно - орієнтованого додатка з використанням API та патернів проектування

### **План роботи**

**Завдання 1:** Виберіть надійний API, який надає через HTTP необхідні дані для віддаленого зберігання, вивантаження або реалізуйте свій. Для прикладу це може бути [jsonplaceholder.org](https://jsonplaceholder.org/). Крім того, оберіть 2-3 патерна проектування для реалізації імплементації цієї лабораторної роботи. Для прикладу, це може бути патерн Unit of Work та Repository

**Завдання 2:** Виберіть бібліотеку для роботи з API та обробки HTTP запитів (для прикладу це може бути бібліотека Requests). Інтегруйте обраний API в ваш консольний додаток на Python. Ознайомтеся з документацією API та налаштуйте необхідний API-ключ чи облікові дані.

**Завдання 3:** Розробіть користувацький інтерфейс, який дозволяє користувачам візуалізувати всі доступні дані в табличному вигляді та у вигляді списку. Реалізуйте механізм для збору та перевірки введеного даних користувачем.

**Завдання 4:** Створіть розбірник для видобування та інтерпретації виразів користувача на основі регулярних виразів, наприклад, для візуалізації дат, телефонів, тощо. Переконайтеся, що розбірник обробляє різні формати введення та надає зворотний зв'язок про помилки.

**Завдання 5:** Реалізуйте логіку для візуалізації даних через API в консолі. Обробляйте відповіді API для отримання даних у вигляді таблиць, списків. Заголовки таблиць, списків мають виділятися кольором та шрифтом, які задається користувачем

**Завдання 6:** Реалізуйте можливості збереження даних у чіткому та читабельному форматі JSON, CSV та TXT

**Завдання 7:** Розробіть надійний механізм обробки помилок для керування помилками API, некоректним введенням користувача та іншими можливими проблемами. Надавайте інформативні повідомлення про помилки.

**Завдання 8:** Включіть функцію, яка реєструє запити користувача, включаючи введені запити та відповідні результати. Дозвольте користувачам переглядати та рецензувати історію своїх запитів.

**Завдання 9:** Напишіть юніт-тести для перевірки функціональності вашого додатку. Тестуйте різні операції, граничні випадки та сценарії помилок.

## Результати тестування:

```
PS C:\Users\Blxxd\Documents\GitHub\DegraCalc> python .\main.py
Available resources: /posts, /comments, /albums, /photos, /todos, /users
Enter a resource name (e.g., 'posts') or type a phone number: todos
Data
```

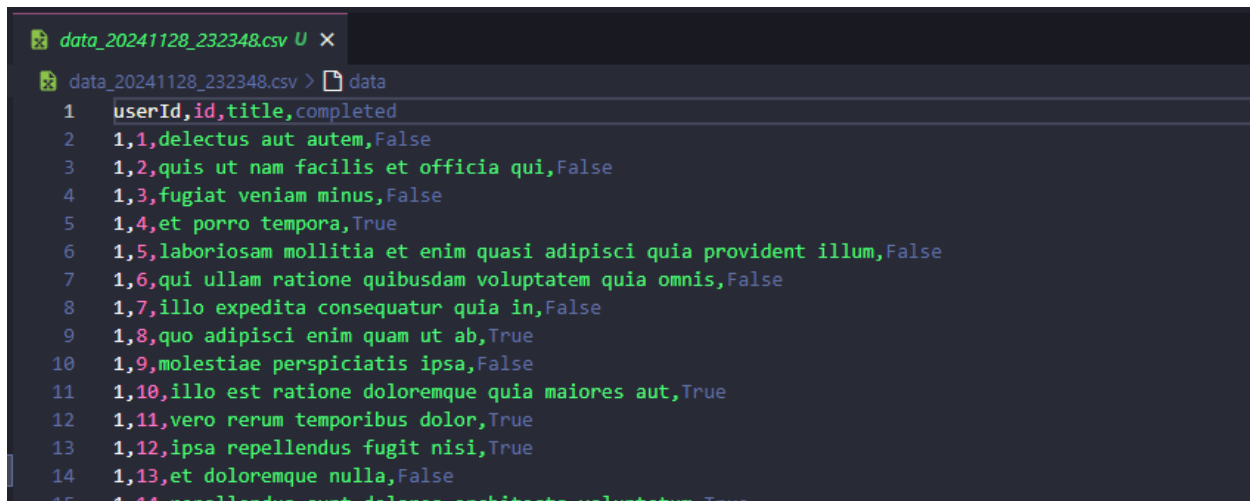
Userid	Id	Title	Completed
1	1	delectus aut autem	False
1	2	quis ut nam facilis et officia qui	False
1	3	fugiat veniam minus	False
1	4	et porro tempora	True
1	5	laboriosam mollitia et enim quasi adipisci quia provident illum	False
1	6	qui ullam ratione quibusdam voluptatem quia	False

Рис. 1. Користувацький інтерфейс та запит до АПІ

10	192	ut quas possimus. Cuius exercitationem sunt voluptates	False
10	193	rerum debitis voluptatem qui eveniet tempora distinctio a	True
10	194	sed ut vero sit molestiae	False
10	195	rerum ex veniam mollitia voluptatibus pariatur	True
10	196	consequuntur aut ut fugit similique	True
10	197	dignissimos quo nobis earum saepe	True
10	198	quis eius est sint explicabo	True
10	199	numquam repellendus a magnam	True
10	200	ipsam aperiam voluptates qui	False

```
Choose a format to save data (json, csv, txt): csv
Data saved to data_20241128_232348.csv
```

Рис. 2. Вибір формату файлу збереження



```
data_20241128_232348.csv U x
data_20241128_232348.csv > data
1  userId,id,title,completed
2  1,1,delectus aut autem,False
3  1,2,quis ut nam facilis et officia qui,False
4  1,3,fugiat veniam minus,False
5  1,4,et porro tempora,True
6  1,5,laboriosam mollitia et enim quasi adipisci quia provident illum,False
7  1,6,qui ullam ratione quibusdam voluptatem quia omnis,False
8  1,7,illo expedita consequatur quia in,False
9  1,8,quo adipisci enim quam ut ab,True
10 1,9,molestiae perspiciatis ipsa,False
11 1,10,illo est ratione doloremque quia maiores aut,True
12 1,11,vero rerum temporibus dolor,True
13 1,12,ipsa repellendus fugit nisi,True
14 1,13,et doloremque nulla,False
15 1,14,repellendus sunt dolores architecto voluptatum,True
```

Рис. 3. Результат збереження

## Текст функції збереження в файл

```
class FileManager:
    @staticmethod
    def save_to_file(content, filename):
        try:
            with open(filename, 'w') as file:
                file.write(content)
        except Exception as e:
            raise IOError(f"Помилка збереження файлу: {filename}. Причина: {str(e)}")

# utils/data_saver.py
import json
import csv
from datetime import datetime

class DataSaver:
    @staticmethod
    def _generate_filename(extension):
        """Генерує унікальну назву файлу з поточною датою і часом."""
        timestamp = datetime.now().strftime("%Y%m%d_%H%M%S")
        return f"data_{timestamp}.{extension}"

    @staticmethod
    def save_to_json(data):
        filename = DataSaver._generate_filename("json")
        try:
            with open(filename, "w") as json_file:
                json.dump(data, json_file, indent=4)
            print(f"Data saved to {filename}")
            return filename
        except (IOError, OSError) as e:
            print("Failed to save data to JSON file.")
            raise e

    @staticmethod
```

```

def save_to_csv(data):
    filename = DataSaver._generate_filename("csv")
    if not data:
        print("No data to save")
        return None

    try:
        with open(filename, "w", newline='') as csv_file:
            writer = csv.writer(csv_file)
            writer.writerow(data[0].keys()) # headers
            for row in data:
                writer.writerow(row.values())
            print(f>Data saved to {filename}<
            return filename
    except (IOError, OSError) as e:
        print("Failed to save data to CSV file.")
        raise e

@staticmethod
def save_to_txt(data):
    filename = DataSaver._generate_filename("txt")
    try:
        with open(filename, "w") as txt_file:
            for row in data:
                txt_file.write(f">{row}<
            print(f>Data saved to {filename}<
            return filename
    except (IOError, OSError) as e:
        print("Failed to save data to TXT file.")
        raise e

```

**Висновки:** В ході виконання лабораторної роботи було створено проєкт, який надав цінний досвід роботи з API.