НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж

Лабораторна робота №7

з дисципліни

СПЕЦІАЛІЗОВАНІ МОВИ ПРОГРАМУВАННЯ

Виконав:

ст. гр. РІ-31

Назар МРАКА

Прийняв:

Сергій ЩЕРБАК

Львів-2024

**Мета:** Створення консольного об’єктно - орієнтованого додатка з використанням API

**Завдання:**

Завдання 1: Вибір провайдера API Виберіть надійний API, який надає через HTTP необхідні дані для віддаленого зберігання, вивантаження або реалізуйте свій. Для прикладу це може бути jsonplaceholder.org

Завдання 2: Інтеграція API 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: Введення користувача Розробіть користувальницький 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: Розбір введення користувача 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: Відображення результатів Реалізуйте логіку для візуалізації даних через API в консолі. Обробляйте відповіді API для отримання даних у вигляді таблиць, списків. Заголовки таблиць, списків мають виділяться кольором та шрифтом, які задається користувачем.

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

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

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

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

Код програми:

*class APIRequestFactory:*

*def \_\_init\_\_(self, settings):*

*self.settings = settings*

*@staticmethod*

*def create\_request(self):*

*if self.settings.method.lower() == 'get':*

*return GetRequest(self.settings.url)*

*elif self.settings.method.lower() == 'post':*

*return 0*

*else:*

*raise ValueError("Unknown method")*

**Висновок:**

У рамках лабораторної роботи було створено консольний об’єктно-орієнтований додаток з використанням API на мові Python. Програма реалізована з використанням об'єктно-орієнтованого підходу, що забезпечує гнучкість, модульність і масштабованість додатка. Використання API дозволило інтегрувати зовнішні сервіси та отримувати дані для виконання функцій додатка.