МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра інформаційних систем та мереж



Лабораторна робота № 9

з дисципліни Спеціалізовані мови програмування

на тему

Створення та рефакторінг програмно-інформаційного продукту засобами Python

Виконав:

студент групи РІ-21сп

Владислав ДМИТРЕНКО

Львів – 2024

**Мета:** Розробка програмно-інформаційного продукту засобами Python

**План роботи**

**Завдання 1.** Створити скрипт запуску лабораторних робіт 1-8 (Runner) з єдиним меню для управління додатками використовуючи патерн FACADE https://refactoring.guru/uk/design-patterns/facade

**Завдання 2.** Зробити рефакторінг додатків, які були зроблені в лб 1-8, для підтримки можливості запуску через Runner

**Завдання 3.** Зробити рефакторинг додатків, які були зроблені в лб 1-8, використовуючи багаторівневу архітектуру додатків (див. приклад нижче) та всі принципи об’єктно-орієнтованого підходу

**Завдання 4.** Створити бібліотеку класів, які повторно використовуються у всіх лабораторних роботах та зробити рефакторінг додатків для підтримки цієї бібліотеки. Таких класів в бібліотеці має буде як найменш 5

**Завдання 5.** Додати логування функцій в класи бібліотеки програмного продукту використовуючи https://docs.python.org/uk/3/howto/logging.html

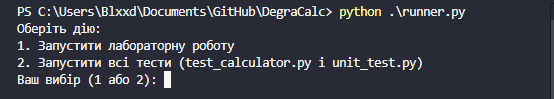
**Завдання 6.** Додати коментарі до програмного коду та сформувати документацію програмного продукту засобами pydoc. Документація має бути представлена у вигляді сторінок тексту на консолі, подана у веб-браузері та збережена у файлах HTML

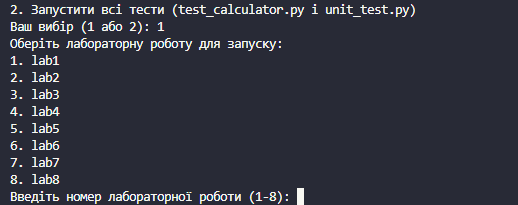
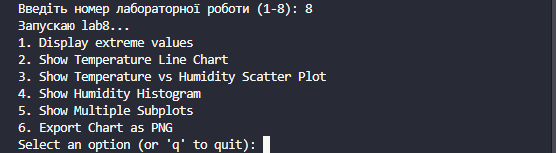
**Завдання 7.** Документація та код програмного продукту має бути розміщено в GIT repo

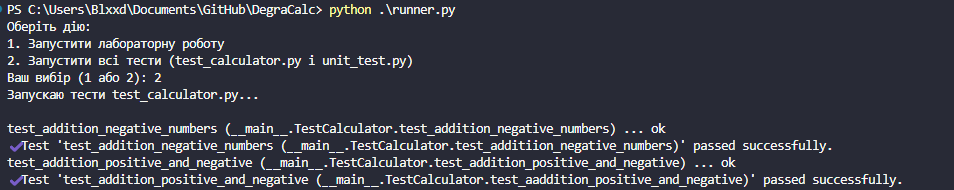
**Завдання 8.** Проведіть статичний аналіз коду продукту засобами PYLINT https://pylint.readthedocs.io/en/stable/ та виправте помилки, які були ідентифіковані. Первинний репорт з помилками додайте до звіту лабораторної роботи

**Завдання 9.** Підготуйте звіт до лабораторной роботи

**Результати тестування:**

  
Рис. 1. Користувацький інтерфейс

  
Рис. 2. Вивід списку лабораторних робіт  
  
Рис 3. Запуск лабораторної роботи

  
Рис. 4. Вигляд всіх діаграм разом

Текст ui інтеракції

def get\_lab\_choice():

print("Оберіть лабораторну роботу для запуску:")

lab\_dirs = [f"lab{i}" for i in range(1, 9)]

for i, lab in enumerate(lab\_dirs, 1):

print(f"{i}. {lab}")

choice = input("Введіть номер лабораторної роботи (1-8): ")

return int(choice) if choice.isdigit() and 1 <= int(choice) <= 8 else None

def run\_lab(lab\_number):

base\_dir = os.path.dirname(os.path.abspath(\_\_file\_\_))

lab\_name = f"lab{lab\_number}"

lab\_path = os.path.join(base\_dir, "..", lab\_name, "main.py")

if os.path.exists(lab\_path):

print(f"Запускаю {lab\_name}...")

try:

subprocess.run(["python", lab\_path], check=True)

except subprocess.CalledProcessError as e:

print(f"Помилка при виконанні {lab\_path}: {e}")

else:

print(f"Файл {lab\_path} не знайдено!")

def interact():

print("Оберіть дію:")

print("1. Запустити лабораторну роботу")

print("2. Запустити всі тести (test\_calculator.py і unit\_test.py)")

choice = input("Ваш вибір (1 або 2): ")

if choice == "1":

lab\_choice = get\_lab\_choice()

if lab\_choice:

run\_lab(lab\_choice)

else:

print("Невірний вибір. Будь ласка, введіть номер від 1 до 8.")

elif choice == "2":

run\_tests()

else:

print("Невірний вибір. Введіть 1 або 2.")

def run\_tests():

base\_dir = os.path.dirname(os.path.abspath(\_\_file\_\_))

test\_calculator\_path = os.path.join(base\_dir, "..", "lab6", "tests", "test\_calculator.py")

unit\_test\_path = os.path.join(base\_dir, "..", "lab7", "tests", "unit\_test.py")

for test\_path in [test\_calculator\_path, unit\_test\_path]:

if os.path.exists(test\_path):

print(f"Запускаю тести {os.path.basename(test\_path)}...\n")

try:

subprocess.run(["python", test\_path], check=True)

except subprocess.CalledProcessError as e:

print(f"Помилка під час виконання тестів у {test\_path}: {e}")

else:

print(f"Файл {test\_path} не знайдено!")

**Висновки:** В ході виконання лабораторної роботи було створено проєкт, який об’єднує всі лабораторні роботи, та дозволяє запустити їх з одного раннера, також виконано рефакторинг та описана документація коду.