**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**ІНСТИТУТ КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**Кафедра ІСМ**



**Звіт**

**до лабораторної роботи №8**

**З дисципліни “Спеціалізовані мови програмування”**

**Виконав:**

**ст. гр. ІТ-32**

**Пазюк Сергій**

**Прийняв:**

**Щербак С.С.**

**Львів — 2023**

**Тема роботи:**  Створення та рефакторінг програмно-інформаційного продукту засобами Python

**Мета роботи:** 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 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

**Хід роботи**

**Реалізований код:**

Lab\_runner.py

import logging

from lab1 import lab1

from lab2and6 import lab2

from lab3 import lab3

from lab4 import lab4

from lab5 import lab5

from lab7 import lab7

from lab8 import lab8

# Налаштування логування

logging.basicConfig(filename='lab\_runner.log', level=logging.DEBUG)

logger = logging.getLogger(name)

class LabRunnerFacade:

def init(self):

# Ініціалізація об'єкта з вказанням доступних лабораторних робіт

self.lab1 = lab1

self.lab2and6 = lab2

self.lab3 = lab3

self.lab4 = lab4

self.lab5 = lab5

self.lab7 = lab7

self.lab8 = lab8

def run\_lab1(self):

# Запуск лабораторної роботи 1

logger.info("Running Lab 1")

self.lab1.run\_lab1()

logger.info("Lab 2 completed")

def run\_lab2(self):

# Запуск лабораторної роботи 2

logger.info("Running Lab 2")

self.lab2and6.run\_lab2()

logger.info("Lab 2 completed")

def run\_lab3(self):

# Запуск лабораторної роботи 3

logger.info("Running Lab 3")

self.lab3.run\_lab3()

logger.info("Lab 3 completed")

def run\_lab4(self):

# Запуск лабораторної роботи 4

logger.info("Running Lab 4")

self.lab4.run\_lab4()

logger.info("Lab 4 completed")

def run\_lab5(self):

# Запуск лабораторної роботи 5

logger.info("Running Lab 5")

self.lab5.run\_lab5()

logger.info("Lab 5 completed")

def run\_lab7(self):

# Запуск лабораторної роботи 7

logger.info("Running Lab 7")

self.lab7.run\_lab7()

logger.info("Lab 7 completed")

def run\_lab8(self):

# Запуск лабораторної роботи 8

logger.info("Running Lab 8")

self.lab8.run\_lab8()

logger.info("Lab 8 completed")

if name == "main":

lab\_runner = LabRunnerFacade()

main.py

import logging

from lab1 import lab1

from lab2and6 import lab2

from lab3 import lab3

from lab4 import lab4

from lab5 import lab5

from lab7 import lab7

from lab8 import lab8

# Налаштування логування

logging.basicConfig(filename='lab\_runner.log', level=logging.DEBUG)

logger = logging.getLogger(name)

class LabRunnerFacade:

def init(self):

# Ініціалізація об'єкта з вказанням доступних лабораторних робіт

self.lab1 = lab1

self.lab2and6 = lab2

self.lab3 = lab3

self.lab4 = lab4

self.lab5 = lab5

self.lab7 = lab7

self.lab8 = lab8

def run\_lab1(self):

# Запуск лабораторної роботи 1

logger.info("Running Lab 1")

self.lab1.run\_lab1()

logger.info("Lab 2 completed")

def run\_lab2(self):

# Запуск лабораторної роботи 2

logger.info("Running Lab 2")

self.lab2and6.run\_lab2()

logger.info("Lab 2 completed")

def run\_lab3(self):

# Запуск лабораторної роботи 3

logger.info("Running Lab 3")

self.lab3.run\_lab3()

logger.info("Lab 3 completed")

def run\_lab4(self):

# Запуск лабораторної роботи 4

logger.info("Running Lab 4")

self.lab4.run\_lab4()

logger.info("Lab 4 completed")

def run\_lab5(self):

# Запуск лабораторної роботи 5

logger.info("Running Lab 5")

self.lab5.run\_lab5()

logger.info("Lab 5 completed")

def run\_lab7(self):

# Запуск лабораторної роботи 7

logger.info("Running Lab 7")

self.lab7.run\_lab7()

logger.info("Lab 7 completed")

def run\_lab8(self):

# Запуск лабораторної роботи 8

logger.info("Running Lab 8")

self.lab8.run\_lab8()

logger.info("Lab 8 completed")

if name == "main":

lab\_runner = LabRunnerFacade()

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

**Висновки:** Виконуючи ці завдання, я розробив програмно-інформаційний продукт засобами Python