Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра «Електронних обчислювальних машин»



**Звіт**

з лабораторної роботи № 9

**з дисципліни:** «Кросплатформенні засоби програмування»

**на тему:** «ОСНОВИ ОБ’ЄКТНО-ОРІЄНТОВАНОГО

ПРОГРАМУВАННЯ У PYTHON»

**Виконав:**

студент групи *КІ-306*

*Ярмола Юрій*

**Прийняв:**

доцент кафедри ЕОМ

Іванов Ю. С.

**Львів – 2023**

**Мета роботи:** оволодіти навиками реалізації парадигм об’єктно-орієнтованого

програмування використовуючи засоби мови Python.

**Завдання ( Варіант 30 «Локомотив» - «Електричка»):**

1. Написати та налагодити програму на мові Python згідно варіанту. Програма має

задовольняти наступним вимогам:

• класи програми мають розміщуватися в окремих модулях в одному пакеті;

• точка входу в програму (main) має бути в окремому модулі;

• мають бути реалізовані базовий і похідний класи предметної області згідно

варіанту;

• програма має містити коментарі.

2. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.

3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її

виконання та фрагменту згенерованої документації та завантажити його у ВНС.

4. Дати відповідь на контрольні запитання.

**Вихідний код програми**

Locomotive.py

# оголошення класу Локомотив  
class Locomotive:  
 def \_\_init\_\_(self, model, max\_speed):  
 self.model = model  
 self.max\_speed = max\_speed  
 self.\_current\_speed = 0  
  
 # метод запускає локомотив  
 def start(self):  
 print(f"Start Locomotive {self.model}")  
  
 # метод зупиняє локомотив  
 def stop(self):  
 print(f'Stop Locomotive {self.model}')  
  
 # метод встановлює швидкість  
 def set\_speed(self, speed):  
 if speed < self.max\_speed:  
 self.\_current\_speed = speed  
 else:  
 print(f"Max speed for this model is {self.max\_speed} km/h")  
  
 # метод повертає поточну швидкість  
 def get\_current\_speed(self):  
 return self.\_current\_speed  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 print('Locomotive class, use it in your program')

ElectricTrain.py

# імпорт класу Локомотив  
from .locomotive import Locomotive  
  
  
# оголошення класу Електричка що наслідує клас Локомотив  
class ElectricTrain(Locomotive):  
 def \_\_init\_\_(self, model, max\_speed, passenger\_capacity):  
 super().\_\_init\_\_(model, max\_speed)  
 self.passenger\_capacity = passenger\_capacity  
 self.\_passenger\_num = 0  
  
 # запустити пасажирів  
 def load\_passenger(self, count):  
 if self.\_passenger\_num + count < self.passenger\_capacity:  
 self.\_passenger\_num += count  
 else:  
 self.\_passenger\_num = self.passenger\_capacity  
  
 # випустити пасжирів  
 def unload\_passengers(self, count):  
 if self.\_passenger\_num - count > 0:  
 self.\_passenger\_num -= count  
 else:  
 self.\_passenger\_num = 0  
  
 # повертає кількість пасажирів в поїзді  
 def get\_passenger\_count(self):  
 return self.\_passenger\_num  
  
 # повертає вмістимість поїзда  
 def get\_passenger\_capacity(self):  
 return self.passenger\_capacity  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 print('ElectricTrain class, use it in your program')

Main.py

from LAB9 import electric\_train # імпорт пакету  
  
# точка входу в програму  
if \_\_name\_\_ == "\_\_main\_\_":  
 intersiti = electric\_train.ElectricTrain('Intersiti', 200, 500)  
 intersiti.start()  
 intersiti.load\_passenger(200)  
 print(intersiti.get\_passenger\_count())  
 intersiti.load\_passenger(20)  
 print(intersiti.get\_passenger\_count())  
  
 ukrzal = electric\_train.ElectricTrain("UkrZaliznitsa", 120, 700)  
 ukrzal.set\_speed(40)  
 print(ukrzal.get\_current\_speed())  
  
 intersiti.unload\_passengers(30)  
 print(intersiti.get\_passenger\_count())

**Результат виконання програми**

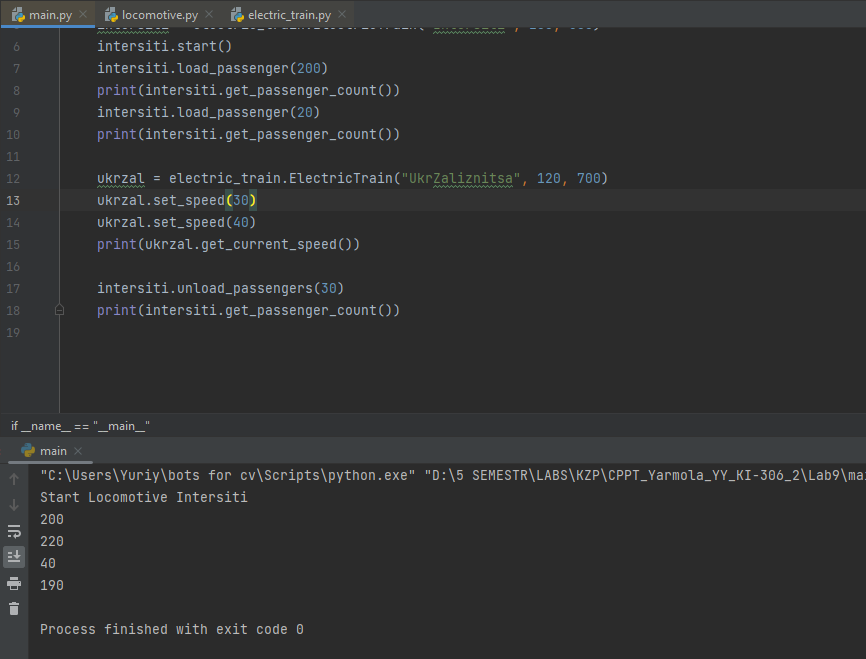


Рис.1 – Скріншот документу згенерованого програмою

**Відповіді на контрольні запитання**

1. Що таке модулі?

Python код в окремому файлі який містить методи або класи

1. Як імпортувати модуль?

Import <name>

1. Як оголосити клас?

Class <name>(<name\_superclass>)

1. Що може міститися у класі?

Методи та змінні

1. Як називається конструктор класу?

Def \_\_init\_\_(self):

1. Як здійснити спадкування?

Class <name>(<name\_superclass>)

1. Які види спадкування існують?

Одиночне і множинне

1. Які небезпеки є при множинному спадкуванні, як їх уникнути?

Невизначеність в виборі роботи методів, спадкувати один за одним

1. Що таке класи-домішки?

**Клас**-**домішка** діє як батьківський **клас**, що містить бажану функціональність. Потім підклас може успадкувати або просто повторно використовувати цю  функціональність

1. Яка роль функції super() при спадкуванні?

Доступ до батьківського класу

**Висновок**

Ознайомився з роботою класів та пакетів у мові програмування Python. Розробив клас який реалізує предметну область «Локомотив» та розширив його розробивши похідний клас «Електричка», та програму драйвер яка перевіряє роботу класу.