

Львівський національний університет імені Івана Франка

Факультет електроніки та комп'ютерних технологій

ЗВІТ

Про виконання лабораторної роботи №1

Опис класів у Python

Виконав:

Студент групи ФЕП-11

Сворень Ярослав

Перевірив:

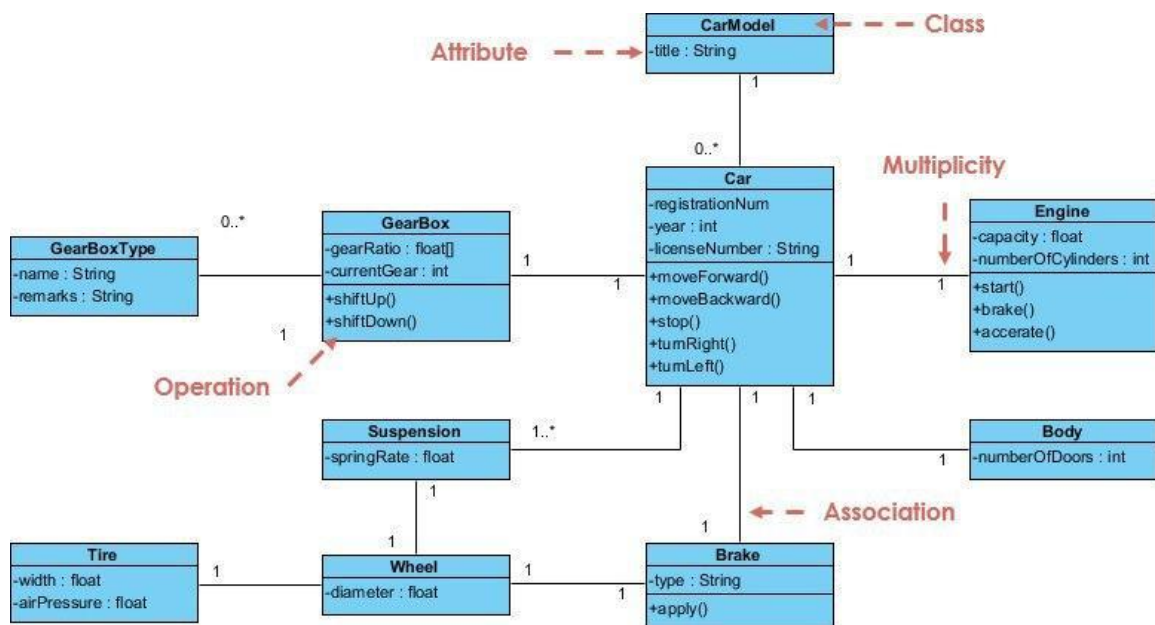
Ас. Сінькевич О. О.

Львів 2020

Практичне Завдання: 1. Розібрати UML-діаграму, щоб правильно оформити зв'язки між класами. 2. Описати цю діаграму на мові Python, включно з наданням правильних прав `private` чи `public` для полів та методів.

Хід Роботи:

1. Ознайомлюємось з діаграмою.



2. Встановлюємо інтерпретатор Python, після чого відкриваємо текстовий редактор Visual Studio Code, де потрібно встановити розширення для підтримки синтаксису мови, і починаємо писати код.



3. Перепишемо діаграму на мові Python.

```
LabPY1.py > ...
1 class CarModel():
2     def __init__(self,car,title = "Model S"):
3         self.car = Car
4         self.__title = title
5
6 class Car():
7     def __init__(self,engine,gear_box,body,suspension,brake,registrationNum,year = 2010,licenseNumber = "BC5678AC"):
8         self.engine = Engine
9         self.gear_box = GearBox
10        self.body = Body
11        self.suspension = Suspension
12        self.brake = Brake
13        self.__registrationNum = registrationNum
14        self.__year = year
15        self.__licenseNumber = licenseNumber
16    def move_forward(self):
17        print("Moving forward...")
18    def move_backward(self):
19        print("Moving backwards...")
20    def stop(self):
21        print("Stopping up...")
22    def turn_right(self):
23        print("Turning right...")
24    def turn_left(self):
25        print("Turning left...")
26
27 class Engine():
28     def __init__(self,car,capacity = 10.0,numberOfCylinders = 4):
29         self.car = Car
30         self.__capacity = capacity
31         self.__numberOfCylinders = numberOfCylinders
32    def start(self):
33        print("Starting up...")
34    def brake(self):
35        print("Breaking up...")
36    def accelerate(self):
37        print("Accelerating...")
```

```
39 class GearBox():
40     def __init__(self,gear_box_type,gearRatio = [1.0,3.0,5,0], currentGear = 4):
41         self.gear_box_type = GearBoxType
42         self.__gearRatio = gearRatio
43         self.__currentGear = currentGear
44    def shift_up(self):
45        print("Shifting up...")
46    def shift_down(self):
47        print("Shifting down...")
48
49 class GearBoxType():
50     def __init__(self,name = "Manual",remarks = "Special"):
51         self.__name = name
52         self.__remarks = remarks
53
54 class Body():
55     def __init__(self,numberOfDoors = 4):
56         self.__NumberOfDoors = numberOfDoors
57
58 class Suspension():
59     def __init__(self,wheel,springRate = 10.0):
60         self.wheel = Wheel
61         self.__springRate = springRate
62
63 class Brake():
64     def __init__(self,wheel,type = "Manual"):
65         self.wheel = Wheel
66         self.__type = type
67
68 class Wheel():
69     def __init__(self,tire,diameter = 20.0):
70         self.tire = Tire
71         self.__diameter = diameter
72
73 class Tire():
74     def __init__(self,width = 10.0 ,airPressure = 5.0):
75         self.__width = width
76         self.__airPressure = airPressure
```

Висновок: на цій лабораторній роботі я ознайомився з азами писання коду на Python, навчився розбирати UML-діаграми.