Лабораторная работа №6

Работа с классами часть 2

Автор:

Голиков Михаил Вячеславович

Группа:

БВТ2402

**Цель лабораторной работы**

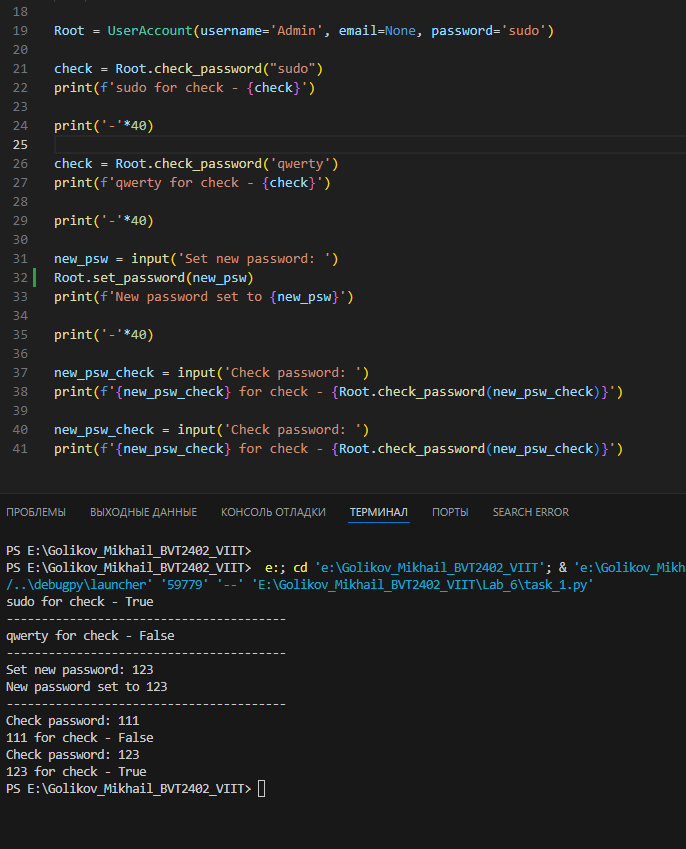
Получить практический опыт работы с ООП в Python. использование инкапсуляции, наследования.

**Ход выполнения лабораторной работы**

Выполним задание. Для этого создадим класс UserAccount со всеми необходимыми свойствами и методами:

|  |
| --- |
| class UserAccount:     def \_\_init\_\_(self, username, email, password):         self.username = username         self.email = email         self.\_\_password = password         self.\_\_password\_hash = hash(password)  ​     def set\_password(self, new\_password):         self.\_\_password = new\_password         self.\_\_password\_hash = hash(new\_password)         return True       def check\_password(self, password):         #if hash(password) == self.\_\_password\_hash:         if password == self.\_\_password:             return True         return False  ​  Root = UserAccount(username='Admin', email=None, password='sudo')  ​  check = Root.check\_password("sudo")  print(f'sudo for check - {check}')  ​  print('-'\*40)  ​  check = Root.check\_password('qwerty')  print(f'qwerty for check - {check}')  ​  print('-'\*40)  ​  new\_psw = input('Set new password: ')  Root.set\_password(new\_psw)  print(f'New password set to {new\_psw}')  ​  print('-'\*40)  ​  new\_psw\_check = input('Check password: ')  print(f'{new\_psw\_check} for check - {Root.check\_password(new\_psw\_check)}')  ​  new\_psw\_check = input('Check password: ')  print(f'{new\_psw\_check} for check - {Root.check\_password(new\_psw\_check)}') |

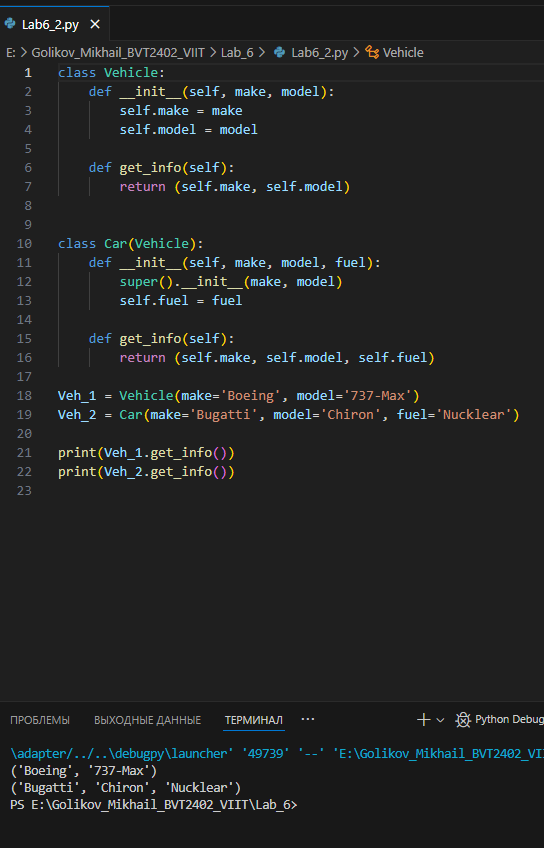
Элемент 1.1 ― Класс UserAccount (код)



Элемент 1.2 ― Класс UserAccount (вывод)

|  |
| --- |
| class Vehicle:     def \_\_init\_\_(self, make, model):         self.make = make         self.model = model  ​     def get\_info(self):         return (self.make, self.model)    ​  class Car(Vehicle):     def \_\_init\_\_(self, make, model, fuel):         super().\_\_init\_\_(make, model)         self.fuel = fuel  ​     def get\_info(self):         return (self.make, self.model, self.fuel)    Veh\_1 = Vehicle(make='Boeing', model='737-Max')  Veh\_2 = Car(make='Bugatti', model='Chiron', fuel='Nucklear')  ​  print(Veh\_1.get\_info())  print(Veh\_2.get\_info())  ​ |

Элемент 2.1 ― Задание 2 (код)



Элемент 2.2 ― Задание 2 (вывод)

**Заключение**

Была изучена подробная работа с классами в Python.