

Лабораторная работа 8

ТЕМА 8. Введение в ООП

Лабораторные задания:

- 1) Создайте класс “Car” с атрибутами производитель и модель. Создайте объект этого класса. Напишите комментарии для кода, объясняющие его работу. Результатом выполнения задания будет листинг кода с комментариями.

```
main.py x
1 class Car:
2     def __init__(self, make, model):
3         self.make = make
4         self.model = model
5
6
7 my_car = Car("Toyota", "Corolla")
8
```

- 2) Дополните код из первого задания, добавив в него атрибуты и методы класса, заставьте машину “поехать”. Напишите комментарии для кода, объясняющие его работу. Результатом выполнения задания будет листинг кода с комментариями и получившийся вывод в консоль.

```
main.py x
1 usage
1 class Car:
2     def __init__(self, make, model):
3         self.make = make
4         self.model = model
5
6     1 usage
7     def drive(self):
8         print(f"Driving the {self.make} {self.model}")
9
10 my_car = Car("Toyota", "Corolla")
11 my_car.drive()
12
```

```
/Users/mihail/PycharmProjects/lab_1/venv/bin/python /Users/mihail/PycharmProjects/
Driving the Toyota Corolla

Process finished with exit code 0
```

- 3) Создайте новый класс “ElectricCar” с методом “charge” и атрибутом емкость батареи. Реализуйте его наследование от класса, созданного в первом задании. Заставьте машину поехать, а потом заряжаться.

Напишите комментарии для кода, объясняющие его работу.

Результатом выполнения задания будет листинг кода с комментариями и получившийся вывод в консоль.

```
main.py ×
1 usage
14 class ElectricCar(Car):
15     def __init__(self, make, model, battery_capacity):
16         super().__init__(make, model)
17         self.battery_capacity = battery_capacity
18
19     1 usage
20     def charge(self):
21         print(f"Charging the {self.make} {self.model} with {self.battery_capacity} kWh")
22
23 my_electric_car = ElectricCar("Tesla", "Model S", 75)
24 my_electric_car.drive()
25 my_electric_car.charge()

/Users/mihail/PycharmProjects/lab_1/venv/bin/python /Users/mihail/PycharmProjects/lab_1/main.py
Driving the Toyota Corolla
Driving the Tesla Model S
Charging the Tesla Model S with 75 kWh

Process finished with exit code 0
```

- 4) Реализуйте инкапсуляцию для класса, созданного в первом задании. Создайте защищенный атрибут производителя и приватный атрибут модели. Вызовите защищенный атрибут и заставьте машину поехать. Напишите комментарии для кода, объясняющие его работу. Результатом выполнения задания будет листинг кода с комментариями и получившийся вывод в консоль.

```
main.py ×
1 usage
1 class Car:
2     def __init__(self, make, model):
3         self._make = make # Защищенный атрибут
4         self.__model = model # Приватный атрибут
5
6     1 usage
7     def drive(self):
8         print(f"Driving the {self._make} {self.__model}")
9
10 my_car = Car("Toyota", "Corolla")
11 print(my_car._make) # Доступ к защищенному атрибуту
12 # print(my_car.__model) # Ошибка! Приватный атрибут не доступен
13 my_car.drive()
14
```

```
/Users/mihail/PycharmProjects/lab_1/venv/bin/python /Users/mihail/PycharmProjects/lab_1/venv/bin/python
Toyota
Driving the Toyota Corolla

Process finished with exit code 0
```

- 5) Реализуйте полиморфизм создав основной (общий) класс “Shape”, а также еще два класса “Rectangle” и “Circle”. Внутри последних двух классов реализуйте методы для подсчета площади фигуры. После этого создайте массив с фигурами, поместите туда круг и прямоугольник, затем при помощи цикла выведите их площади. Напишите комментарии для кода, объясняющие его работу. Результатом выполнения задания будет листинг кода с комментариями и получившийся вывод в консоль.

```
main.py x
2 usages
1 class Shape:
2     def area(self):
3         pass
4
5
1 usage
6 class Rectangle(Shape):
7     def __init__(self, width, height):
8         self.width = width
9         self.height = height
10
11     def area(self):
12         return self.width * self.height
13
14
1 usage
15 class Circle(Shape):
16     def __init__(self, radius):
17         self.radius = radius
18
19     def area(self):
20         return 3.14 * self.radius * self.radius
```

```
/Users/mihail/PycharmProjects/lab_1/venv/bin/python /Users/mihail/PycharmProjects/lab_1/venv/bin/python
20
28.259999999999998

Process finished with exit code 0
```