



Python

2-месяц 1-урок

Тема:

ООП 1: Суть ООП, Создание первых классов, Атрибуты и Методы классов



Теория

Объектно-ориентированное программирование - методология (парадигма) программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования. В программе при этом в качестве основных логических конструктивных элементов используются объекты, а не алгоритмы.

До настоящей темы объекты так или иначе использовались в коде, однако сам стиль программирования был императивным (процедурным). Написание программы в исключительно таком стиле вполне возможно, однако с увеличением объема кода и усложнения логики обработки данных, объектно-ориентированный стиль дает значительные преимущества.



Объекты в Python

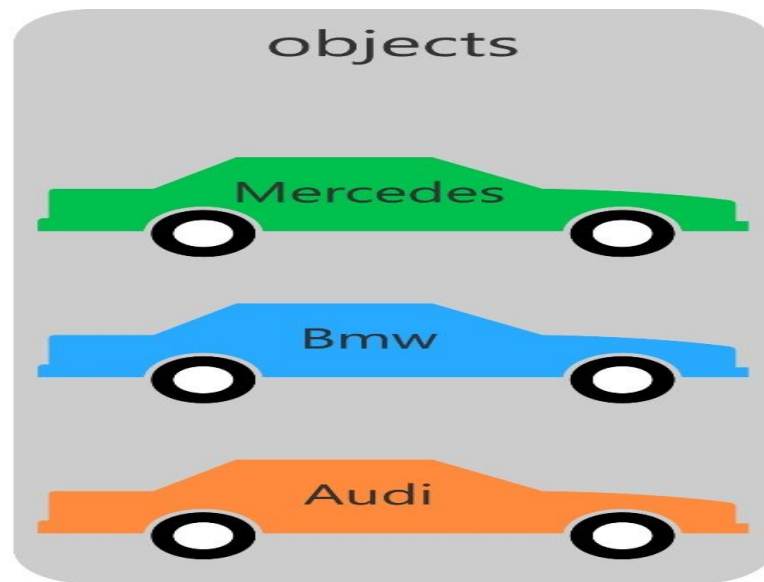
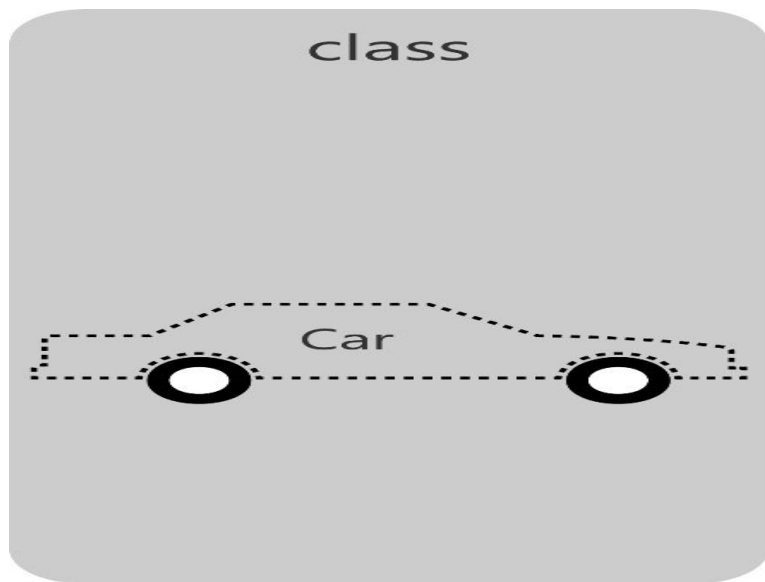
Объект - любой материальный предмет, который можно встретить в повседневной жизни: дом, телефон, машина, книга и т.д.

Python - полностью объектно-ориентированный язык, где любое значение является **объектом**, т.е. экземпляром конкретного класса. Например, число 5 или строка "python" являются объектами, экземплярами классов `int` и `str` соответственно.

Класс и объект, поля и методы

В ООП центральными являются понятия **класса** и **объекта**:

1. Класс (*англ.* Class): абстракция реального мира (обобщенный шаблон), специальный тип данных; класс описывает свойства и методы, которые могут быть доступны у подобных объектов;
2. Объект (*англ.* Object) (экземпляр класса, *англ.* Class Instance): частный случай класса



Конструктор класса – метод `__init__()`



```
class Person:
```

```
    def __init__(self, n, s):
```

```
        self.name = n
```

```
        self.surname = s
```

```
p1 = Person("Sam", "Baker")
```

```
print(p1.name, p1.surname)
```



Переменные класса

```
class Date():  
    month = ['январь', 'февраль', ..]  
    def __init__(self, day, month, year):  
        self.day = day  
        self.month = month  
        self.year = year  
        self.month_name = Date.month[month-1]
```



Домашнее Задание

Задание :

1. Создать суперкласс Animal и его классы (пример : Млекопитающие , Пресмыкающиеся) они должны быть связаны через наследование
2. Класс Animal имеет общие параметры животных
3. Класс Млекопитающих и других имеют свои особенные атрибуты и методы
4. Создать объекты животных и его классов

Дополнительные указания к дз:

1. Создать класс объекта Zoo_show
2. До выбора билета , должно высвечиваться подробная информация о видах шоу
3. У этого класса будет метод для выбора билета , при выборе определенного шоу должно выходить их цена и как проходит само шоу