

Python Введение в программирование

Gmail: vitalijfisenko5@gmail.com

Telegram: @vitaliifisenko

ВОПРОСЫ?)

ЗМЕЧАНИЯ

ВОПРОСЫ?)

ООП часть 1

- ООП - методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определённого класса, а классы образуют иерархию наследования.
- Класс - шаблон, описывающий пользовательский тип данных, на основе которого в программе создаются однородные объекты.
- Объект - экземпляр класса, его реальное воплощение.

ЦЕЛЬ ООП

- Улучшить восприятие поставленной задачи при работе над проектом
- Сократить количество кода
- Упростить написания кода

Всё вокруг нас является
объектом.



У объекта есть
свойства
(еще называют параметры)



У объекта есть
методы
(методы – это действия,
то есть что может
делать объект)

Например, машина – это объект.



У любой машины есть такие
свойства: модель, цвет, размер и т.д.

Методы машины:
затормозить ();
нажатьНаГаз ();
открытьДверь ();
закрытьДверь ();
и т.д.

Например, котёнок – это объект.



У любого котенка есть свойства:
порода, имя, цвет, длина шерсти,
возраст и т.д.

Методы котенка:
спать();
кушать();
играть ();
шкодить ();
и т.д.

Car Class



Red

Ford

Mustang



Blue

Toyota

Prius



Green

Volkswagen

Golf

Class



Building

Objects



Home



Church



Skyscraper

Class

С точки зрения кода класс объединяет набор функций и переменных, которые выполняют определенную задачу. Функции класса называются **методами** и они определяют поведение класса. А переменные класса называются **атрибутами** - они хранят состояние класса.

Class

```
class НазваниеКласса:  
    атрибуты, методы
```


объект класса

```
название_объекта = класс (параметры)
```

Основные принципы ООП

- Абстракция выполняет основную задачу ООП, позволяя программисту формировать объекты определенного типа с различными характеристиками и поведением, благодаря применению классов.
- Инкапсуляция помогает скрыть детали реализации конкретных объектов и защитить их свойства от постороннего вмешательства.
- Наследование дает возможность расширять уже существующие классы за счет автоматического переноса их параметров и методов к новым структурам данных.
- Полиморфизм используется для создания единого интерфейса, который имеет множество различных реализаций, зависящих от класса применяемого объекта.

Инкапсуляция



Наследование



Полиморфизм

