



OOP (object oriented programming)

PYTHON

Что такое ООП?

Python проектировался как
объектно-ориентированный
язык программирования.



Основные понятия

Класс и Объект

- Класс – это абстрактный тип данных. С помощью класса описывается некоторая сущность (характеристики и возможные действия)
- Описав класс, мы можем создать его экземпляр – объект. Объект – это уже конкретный представитель класса.
- Класс – это абстрактное представление чего-либо.
- Объект - используемый экземпляр того, что представляет класс.

Создание классов, методов и объектов

**Объявление
класса**

```
class ИМЯ КЛАССА ():  
    ПЕРЕМЕННАЯ = ЗНАЧЕНИЕ  
    ...
```

**Объявление
метода**

```
def ИМЯ МЕТОДА (self, . . .):  
    self. ПЕРЕМЕННАЯ = ЗНАЧЕНИЕ  
    ...
```

**Создание
объекта**

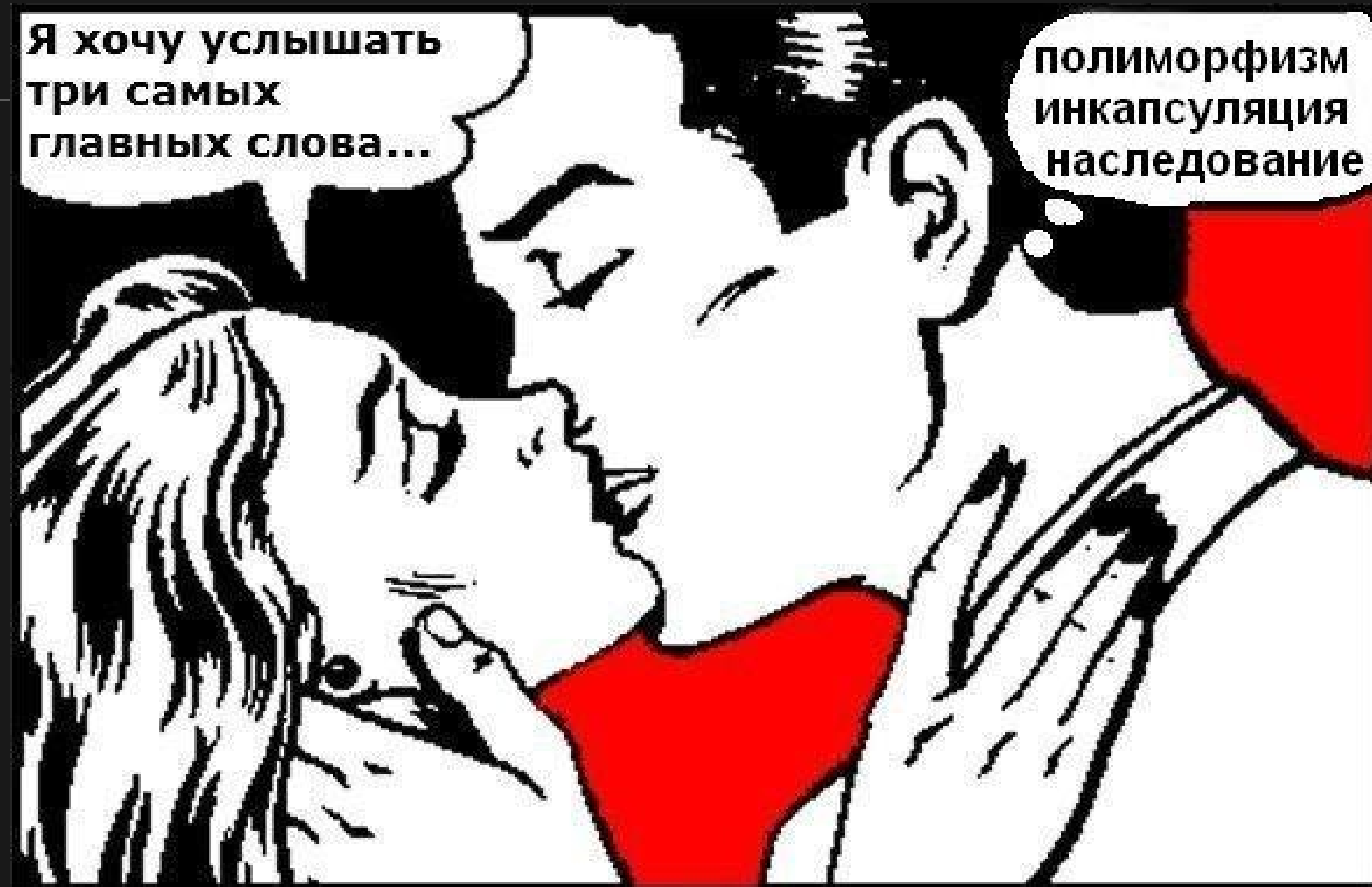
```
# Основная часть  
ПЕРЕМЕННАЯ = ИМЯ КЛАССА ()  
...
```

**Вызов
метода**

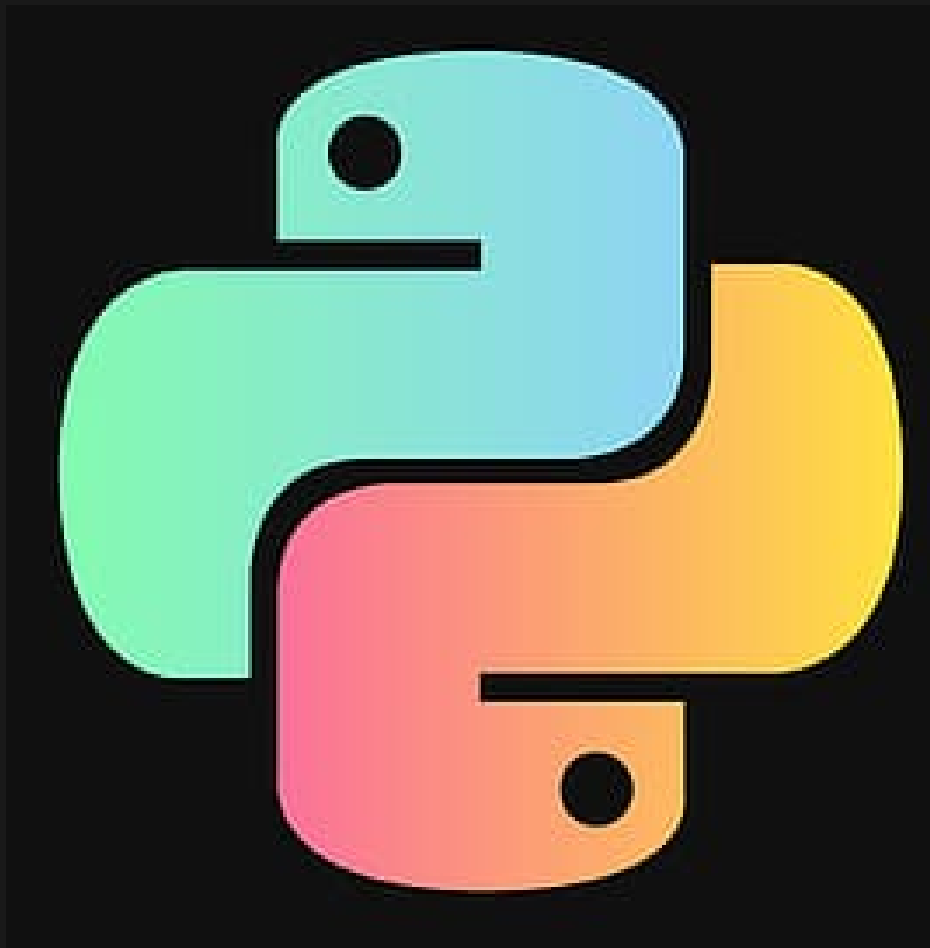
```
ОБЪЕКТ.ИМЯ МЕТОДА ()  
...
```

Основные принципы объектно-ориентированного программирования

- Инкапсуляция
- Наследование
- Полиморфизм



Инкапсуляция



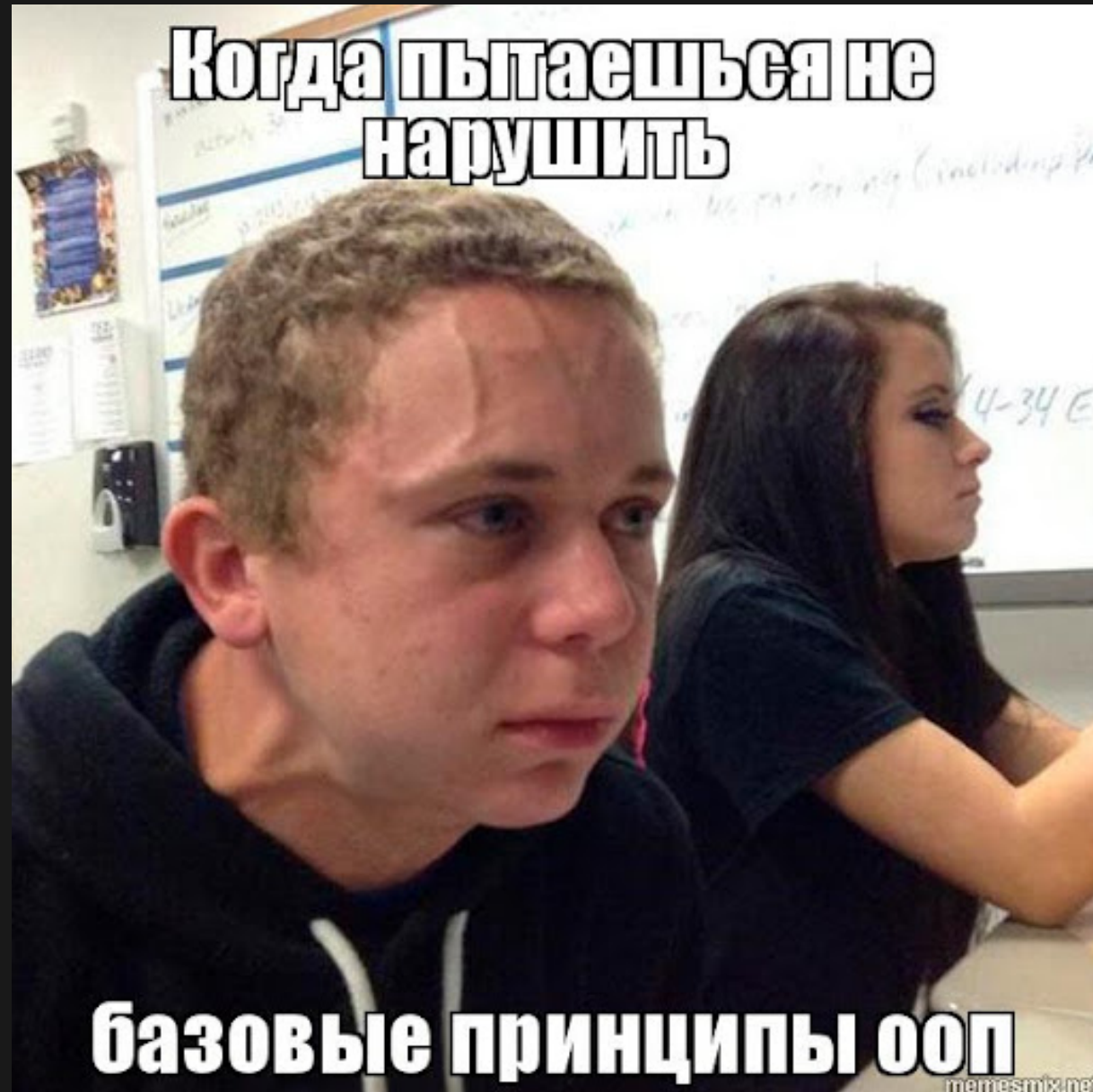
- Инкапсуляция – позволяет скрывать внутреннюю реализацию.
- В классе могут быть реализованы внутренние вспомогательные методы, поля, к которым доступ для пользователя необходимо запретить

Наследование



- Наследование – позволяет создавать новый класс на базе другого.
 - Класс, на базе которого создается новый класс, называется базовым, а базирующийся новый класс – наследником.
- Например, есть базовый класс животное. В нем описаны общие характеристики для всех животных (класс животного, вес). На базе этого класса можно создать классы наследники (Собака, Слон) со своими специфическими свойствами.
- Все свойства и методы базового класса при наследовании переходят в класс наследник

Полиморфизм



- Полиморфизм – это способность объектов с одним интерфейсом иметь различную реализацию.

- Например, есть два класса, Круг и Квадрат. У обоих классов есть метод `GetSquare()`, который считает и возвращает площадь. Но площадь круга и квадрата вычисляется по-разному, соответственно, реализация одного и того же метода различная

Особенности ООП в Python



**PYTHON OBJECT
ORIENTED
PROGRAMMING**



POOP

По сравнению со многими другими языками в Python объектно-ориентированное программирование обладает рядом особых черт. Всё является объектом – число, строка, список, функция, экземпляр класса, сам класс, модуль. Так класс – объект, способный порождать другие объекты – экземпляры. В Python нет просто типов данных. Все типы – это классы.

И напоследок

Python – это все-таки скриптовый интерпретируемый язык. Хотя на нем пишутся в том числе крупные проекты, часто он используется в веб-разработке, системном администрировании для создания небольших программ-сценариев. В этом случае обычно достаточно встроенных средств языка, "изобретать" собственные классы излишне. Однако, поскольку в Python всё – объект и всё пронизано объектно-ориентированной парадигмой, понимание ООП позволит более полно и грамотно использовать возможности языка как инструмента разработки.

