

# OOP基础

# 1

## OOP基础

### ——面向对象技术简介

# 基本概念

- 类(Class)：用来描述具有相同的属性和方法的对象的集合。它定义了该集合中每个对象所共有的属性和方法。对象是类的实例。
- 实例化：创建一个类的实例，类的具体对象。
- 方法：类中定义的函数。
- 对象：通过类定义的数据结构实例。对象包括两个数据成员（类变量和实例变量）和方法。

# 创建类

- 使用 class 语句来创建一个新类，class 之后为类的名称并以冒号结尾
- 类名建议使用驼峰形式

```
class BearToy:  
    pass
```

# 创建实例

- 类是蓝图，实例是根据蓝图创建出来的具体对象

```
tidy = BearToy()
```

# 2

OOP基础

——绑定方法

# 构造器方法

- 当实例化类的对象是，构造器方法默认自动调用
- 实例本身作为第一个参数，传递给self

```
class BearToy:
    __init__(self, size, color):
        self.size = size
        self.color = color

if __name__ == '__main__':
    tidy = BearToy('small', 'orange')
```

## 其他绑定方法

- 类中定义的方法需要绑定在具体的实例，由实例调用
- 实例方法需要明确调用

```
class BearToy:
    def __init__(self, size, color):
        self.size = size
        self.color = color

    def speak(self):
        print('hahaha')

if __name__ == '__main__':
    tidy = BearToy('small', 'orange')
    tidy.speak()
```





更多精彩...



<http://bj.linux.tedu.cn/>  
企业QQ：86198501