

python in action

Lesson 9 - 类和对象

凤凰山 github.com/gxtrobot/pyinaction

2019-12

Lesson 9 - 类和对象

课程内容

今天的课程将介绍 Python 的类 (class)，与对象 (object) 的使用，并用新学知识重写上节课的一个程序

课程目标

- 了解类的概念
- 了解类的编写
- 了解对象的概念，以及如何创建，修改对象
- 使用类，对象的知识重写上节课的分数处理程序

```
class Classroom:
    def __init__(self, name):
        self.name = name
        self.students = []

    def __str__(self):
        return '班级: {}'.format(self.name)

    def add_student(self, no, student_name):
        self.students.append((no, student_name))

    def print_details(self):
        print(self)
        print('学号: 姓名 ')
        for no, student_name in self.students:
            print('{:3}: {:>5}'.format(no, no))
```

类的概念和语法

- 类是对象的模版，用来创建不同的对象
- 类也是组织代码的一种方式（语句，语句块，函数，模块）
- 类里面可以包含数据和方法（就是函数）
- 类一般用来将密切相关的数据和方法组织在一起，方便使用
- 关键字 `class` 后面跟类名，然后是语句块
- 方法第一个参数为 `self`，用来引用要操作的对象，其他同函数
- 必须有一个 `__init__` 方法，用来初始化对象
- 还有很多双下划线开头的特殊方法，用在一些特殊用途，比如 `__str__`，在 `print` 时候自动调用
- 还有很多特性，比如类继承，类变量等不做介绍，可以后续参考相关的书

对象的概念和使用

```
myclass = Classroom('五一班')  
myclass.name  
print(myclass)  
myclass.size = 30
```

- 调用相关的类名，类似调用函数，传入对象初始化参数
- 使用对象名． 属性名语法可以引用对象的属性，或修改属性
- 使用对象名． 方法名可以调用方法，并传入相关参数
- 操作符 `is` 可以比较两个变量是否为同一个对象
- 函数 `isinstance(obj, class)` 可以检测一个对象是否由某个类创建
- `object` 为内置的基类
- 对于任何对象 `isinstance(obj, object)`，永远返回 `True`，证明一些都是对象
- `id(obj)` 函数返回任意对象的 `id` 编号，不同的对象 `id` 不同

IDLE 玩一玩

```
jack_class = Classroom('五一班')
sally_class = Classroom('五二班')
mike_class = jack_class
jack_class is sally_class
jack_class is mike_class
isinstance(jack_class, Classroom)
isinstance(jack_class, object)
isinstance(sally_class, object)
isinstance(1, object)
id(mike_class), id(jack_class), id(sally_class)
```

写一个小程序

`reporting_scores_class.py` 基于第 8 课的 `reporting_scores.py` 使用类和对象的概念改写程序, 创建班级类 (ClassRoom), 课程类 (Lesson), 功能保持不变

- 将 `scores.txt` 扩展为以下格式, 包括标题行, 以及多门课程

学 号 , 姓 名 , 语 文 , 数 学 , 英 语

1, 王 小 明 , 100, 88, 90

2, 李 思 思 , 90, 70, 85

3, 刘 畅 , 88, 78, 92

- 将 `reporting_scores_class.py` 扩展为处理一上格式的文件, 功能不变

