

Python程序设计

第七讲 面向对象程序设计 概述



张 华
WHU

面向对象思想概述

面向过程

- ✿ 需要求解的问题域被视为待处理的一个大过程，经分析处理后，并根据问题的复杂程度，整个系统最终被分解为一个个易于处理的子过程。
- ✿ 面向过程方法将过程（或函数）作为系统的主要构件，数据和对数据的操作（函数）分离。
- ✿ 该方法对需求变化的适应能力差，并且处理大型、复杂问题时，系统维护困难。

面向对象思想概述

面向对象

- ✿ 直接以问题域中的事物（对象）为中心来思考问题，把对象作为系统的基本构成单位，又通过将对象之间的相互作用、相互联系映射到计算机系统来模拟现实客观世界。
- ✿ 面向对象把数据和对数据的处理看成一个整体——对象，其概念和处理方法比较符合人类认识问题、分析问题和解决问题的一般规律。

面向对象思想概述

面向对象的主要优点

✿ 自然高效

- 可更客观更自然地描述现实世界。

✿ 易于重用

- 相关机制可方便继承已有或已经完成的工作，而不必一切从头开始。

✿ 便于维护

- 开发出的系统健壮，适应性和扩展性强。

面向对象的基本概念

对象

- ✿ 人们进行研究的任何事物统称为对象。
- ✿ 对象可以是有形的实体，也可以是活动或事件等。

属性和方法

- ✿ 属性是描写对象静态特性的数据元素。
- ✿ 方法是用于描写对象动态特性（行为特性）的一组操作。

封装

- ✿ 通过对象把数据以及对数据的操作结合在一起。

面向对象的基本概念

■ 类

- ✿ 类是对一组具有相同特性的对象的抽象描述。

■ 实例化

- ✿ 实例化是指在类定义的基础上构造对象的过程。

■ 继承

- ✿ 一个新类（子类）从已有的类（父类）那里获得已有特性和功能。

■ 多态

- ✿ 不同的对象可以以不同的方式响应相同的消息。

面向对象程序设计

■ 面向对象程序设计 (Object Oriented Programming, OOP)

- ✿ 主要针对大型软件设计而提出，使得软件设计更加灵活，能够很好地支持代码复用和设计复用，代码具有更好的可读性和可扩展性，大幅度降低了软件开发的难度。
- ✿ 将数据以及对数据的操作封装在一起，组成一个相互依存、不可分割的整体（对象），不同对象之间通过消息机制来通信或者同步。
- ✿ 对相同类型的对象进行分类、抽象后，得出共同的特征而形成了类，面向对象程序设计的关键就是如何合理地定义这些类并且组织多个类之间的关系。
 - 例如，绘图程序中的定义圆类，一次执行时可以创建很多圆对象。

■ Python完全支持面向对象的基本功能。