

§ 8. 类和对象的特性

8.1. 面向对象程序设计方法概述

8.1.1. 面向对象的基本概念

8.1.1.1. 对象

定义：对象是一组信息及其上操作行为的描述，是具有相同状态的一组操作的集合

对象的属性与行为：

属性：对象自身所要维护的知识，可用值表示

行为：通过对象的行为所表示的特征，用执行步骤来表示

8.1.1.2. 类和抽象

类的定义：具有相同或相似行为(属性和操作)的一组对象的共同描述

元类：在若干具有相同/相似行为的类的基础上抽象出的类，称为元类

基类和派生类：

基类：已有的类（父类）

派生类：在已有类的基础上继承特性而产生的新类（子类、导出类）

类的特征：

★ 继承性：可以把子类看作是继承了父类特征，并包含了新特征的一类

★ 层次性：父类可以抽象出更高层次的类，子类也可以派生出其它类，类的逐层抽象关系形成类的层次结构

§ 8. 类和对象的特性

8. 1. 面向对象程序设计方法概述

8. 1. 1. 面向对象的基本概念

8. 1. 1. 3. 实例

定义：某个特定类所描述的一个具体的对象，称为类的实例对象

对象和类的关系：

- ★ 类是一组对象的抽象，是一个抽象的概念（现实世界中是没有类存在的）
- ★ 对象是指一个属于某个类的有意义的具体实体

8. 1. 1. 4. 消息

定义：在对象之间传递的信息，是要求某个对象执行特定操作的规格说明

消息的组成：接收对象、消息的名称、消息的变元

消息的特征：只包含了请求，不包含实现的具体指示

8. 1. 1. 5. 封装与信息隐蔽

定义：将某个对象的信息隐藏起来，使外界无法知道该对象实现的细节内容

特征：

- ★ 清晰的边界：所有私有数据和操作的实现细节都在边界内
- ★ 明确的接口：说明了对象可以接受那些消息，需要什么具体的参数
- ★ 受保护的内部实现：所有内部数据及操作对外不可访问

§ 8. 类和对象的特性

8.1. 面向对象程序设计方法概述

8.1.1. 面向对象的基本概念

8.1.1.6. 继承与重用

定义：继承是派生类自动共享基类中定义的数据和方法的机制

继承性的两个含义：

- ★ 同一类的实例对象共享所属类的特征
- ★ 类层次中子类自动继承全部父类(直接/间接)的特征

单继承与多继承：

单继承：一个子类只允许继承一个父类

多继承：一个子类允许继承多个父类

8.1.1.7. 多态性

广义定义：一个事物具有多种形态

面向对象方法学定义：在不同的类的继承层次中，以同一方法使用不同类中的相同操作，称为多态性

- ★ 父类子类都具有接受某个消息的能力
- ★ 父类子类处理这个消息的方法相互独立，互不相同
- ★ 不同类的实例对象在收到这个消息后，根据所属类动态选用该操作的实现算法

§ 8. 类和对象的特性

8.1. 面向对象程序设计方法概述

8.1.1. 面向对象的基本概念

8.1.1.8. 重载

定义：在同一作用域中，不同操作可以采用相同的名称

重载的类别：

函数重载：在同一作用域中若干函数可以使用相同的名称（要求参数的个数/类型有所不同）

操作符重载：不同类型的操作数可以使用相同的运算符

§ 8. 类和对象的特性

8.1. 面向对象程序设计方法概述

8.1.2. 面向过程与面向对象的软件开发

面向过程的软件开发：将软件按其功能分解为若干模块采用结构化技术完成软件开发的各项任务，使软件的开发与解决问题的过程一致

面向对象的软件开发：将软件按相同或相似的特性分解为若干对象，对象是融合了数据以及在其上的操作的软件构件，对象之间相互独立，彼此仅通过消息相互联系，消息中仅包含了功能的请求，而没有具体的实现指示，从而做到了相对封闭，使内部的调整不影响系统的全局

8.1.3. 面向对象的软件开发

含义：将面向对象方法学应用于整个软件开发过程中

(软件生命周期)

★ 面向对象分析(Object Oriented Analysis)

★ 面向对象设计(Object Oriented Design)

★ 面向对象编程(Object Oriented Programming)

用一种具体的面向对象程序设计语言(C++)来实现面向对象的设计要求

★ 面向对象测试(Object Oriented Test)

★ 面向对象维护(Object Oriented Soft Maintenance)