

java学习理解面向对象程序设计 PDF转换可能丢失图片或格式
, 建议阅读原文

https://www.100test.com/kao_ti2020/273/2021_2022_java_E5_AD_A6_E4_B9_A0_c104_273477.htm 1.编程模型 所有计算机均由两种元素组成:代码和数据.精确的说,有些程序是围绕着"什么正在发生"而编写,有些则是围绕"谁正在受影响"而编写的. 第一种编程方式叫做"面向过程的模型",按这种模型编写的程序以一系列的线性步骤(代码)为特征,可被理解为作用于数据的代码.如 C 等过程化语言. 第二种编程方式叫做"面向对象的模型",按这种模型编写的程序围绕着程序的数据(对象)和针对该对象而严格定义的接口来组织程序,它的特点是数据控制代码的访问.通过把控制权转移到数据上,面向对象的模型在组织方式上有:抽象,封装,继承和多态的好处. 2.抽象 面向对象程序设计的基本要素是抽象,程序员通过抽象来管理复杂性. 管理抽象的有效方法是使用层次式的分类特性,这种方法允许用户根据物理含义分解一个复杂的系统,把它划分成更容易管理的块. 例如,一个计算机系统是一个独立的对象.而在计算机系统内部由几个子系统组成:显示器,键盘,硬盘驱动器,DVD-ROM,软盘,音响等,这些子系统每个又由专门的部件组成.关键是需要使用层次抽象来管理计算机系统(或其他任何复杂系统)的复杂性. 面向对象程序设计的本质:这些抽象的对象可以被看作具体的实体,这些实体对用来告诉我们作什么的消息进行响应. /* (我的理解) *计算机是一个实体,我要输入字符,显示器显示出来,那么 *计算机(对象).输入(键盘属性).显示(显示方法) *使用分层来引用,操作.而不用管计算机内部如何处理. *只要有计算机对象,它就能响应我的操作,而我敲键盘, *计算机对象就把这个

消息传给屏幕,屏幕显示. */ 计算机对象包含了它所有的属性,以及操作,这就是面向对象程序设计的三大原则之一:封装. 3.封装 封装是一种把代码和代码所操作的数据捆绑在一起,使这两者不受外界干扰和误用的机制.封装可被理解为一种用做保护的包装器,以防止代码和数据被包装器外部所定义的其他代码任意访问.对包装器内部代码与数据的访问通过一个明确定义的接口来控制.封装代码的好处是每个人都知道怎样访问代码,进而无需考虑实现细节就能直接使用它,同时不用担心不可预料的副作用. 在JAVA中,最基本的封装单元是类,一个类定义着将由一组对象所共享的行为(数据和代码).一个类的每个对象均包含它所定义的结构与行为,这些对象就好像是一个模子铸造出来的.所以对象也叫做类的实例. 在定义一个类时,需要指定构成该类的代码与数据.特别是,类所定义的对象叫做成员变量或实例变量.操作数据的代码叫做成员方法.方法定义怎样使用成员变量,这意味着类的行为和接口要由操作实例数据的方法来定义. 由于类的用途是封装复杂性,所以类的内部有隐藏实现复杂性的机制.所以JAVA中提供了私有和公有的访问模式,类的公有接口代表外部的用户应该知道或可以知道的每件事.私有的方法数据只能通过该类的成员代码来访问.这就可以确保不会发生不希望的事情. 4.继承 继承是指一个对象从另一个对象中获得属性的过程.是面向对象程序设计的三大原则之二,它支持按层次分类的概念.例如,波斯猫是猫的一种,猫又是哺乳动物的一种,哺乳动物又是动物的一种.如果不使用层次的概念,每个对象需要明确定义各自的全部特征.通过层次分类方式,一个对象只需要在它的类中定义是它成为唯一的各个属性,然后从父类中继承它的通用属性.因此,正是由于继承机制,

才使得一个对象可以成为一个通用类的一个特定实例.一个深度继承的子类将继承它在类层次中的每个祖先的所有属性.继承与封装可以互相作用.如果一个给定的类封装了某些属性,它的任何子类将会含有同样得属性,另加各个子类所有得属性.这是面向对象程序在复杂性上呈线性而非几何增长的一个重要概念.新的子类继承其所有祖先的所有属性.子类和系统中的其他代码不会产生无法预料的交互作用.

5.多态 多态是指一个方法只能有一个名称,但可以有許多形态,也就是程序中可以定义多个同名的方法,用"一个接口,多个方法"来描述.可以通过方法的参数和类型引用.

6.封装,继承,多态的组合使用 在由封装,继承,多态所组成的环境中,程序员可以编写出比面向过程模型更健壮,更具扩展性的程序.经过仔细设计的类层次结构是重用代码的基础.封装能让程序员不必修改公有接口的代码即可实现程序的移植.多态能使程序员开发出简洁,易懂,易修改的代码.例如:汽车 从继承的角度看,驾驶员都依靠继承性来驾驶不同类型(子类)的汽车,无论这辆车是轿车还是卡车,是奔驰牌还是菲亚特牌,驾驶员都能找到方向盘,手刹,换档器.经过一段时间驾驶后,都能知道手动档与自动档之间的差别,因为他们实际上都知道这两者的共同超类:传动装置.从封装的角度看,驾驶员总是看到封装好的特性.刹车隐藏了许多复杂性,其外观如此简单,用脚就能操作它.发动机,手刹,轮胎大小的实现对与刹车类的定义没有影响.从多态的角度看,刹车系统有正锁反锁之分,驾驶员只用脚踩刹车停车,同样的接口可以用来控制若干种不同的实现(正锁或反锁).这样各个独立的构件才被转换为汽车这个对象的.同样,通过使用面向对象的设计原则,程序员可以把一个复杂程序的各个构件组合在一起,形成一个一致,健

壮,可维护的程序 100Test 下载频道开通 , 各类考试题目直接
下载。详细请访问 www.100test.com