

今天主要掌握的内容是:

什么是面向对象与对象的封装性

面向过程:

我们是执行者

面向对象:

将我们从执行者变成了指挥者,让对象去执行

面向对象开发

就是不断的创建对象,使用对象,指挥对象做事情。

面向对象特征

- * 封装(encapsulation)

- * 继承(inheritance)

- * 多态(polymorphism)

我们如何描述现实世界事物

- * 属性 就是该事物的描述信息(事物身上的名词)

- * 行为 就是该事物能够做什么(事物身上的动词)

Java中最基本的单位是类,Java中用class描述事物也是如此

- * 成员变量 就是事物的属性

- * 成员方法 就是事物的行为

成员变量和成员方法

- * a:成员变量 和以前定义变量是一样的,只不过位置发生了改变。在类中,方法外。

- * b:成员方法 和以前定义方法是一样的,只不过把static去掉。

类和对象的概念

- * a:类:是一组相关的属性和行为的集合-----一个对这类事物的描述

- * b:对象:是该类事物的具体体现-----这类事物的具体实物

P1:学生类的定义

P2:手机类的定义

对象的使用:

- * A:文件名问题

- * 在一个java文件中写两个类:一个基本的类,一个测试类。

- * 文件名称和测试类名称一致,因为我们执行的是和文件名一样的class文件,并在里面找main方法。

- * B:使用对象

- * 创建对象并使用

- * 格式:类名 对象名 = new 类名();

- * D:使用成员变量

- * 对象名.变量名

- * E:使用成员方法

- * 对象名.方法名(...)

对象的堆栈图

在栈里放的是变量名,存的是地址值

在堆里放的是具体对象

在方法区里放的是class字节和成员变量

ps:这里要能说出程序的每一步流程和堆栈做了什么事

成员变量和局部变量的区别

* A:在类中的位置不同

* 成员变量：在类中方法外

* 局部变量：在方法定义中或者方法声明上

* B:在内存中的位置不同

* 成员变量：在堆内存(成员变量属于对象,对象进堆内存)

* 局部变量：在栈内存(局部变量属于方法,方法进栈内存)

* C:生命周期不同

* 成员变量：随着对象的创建而存在，随着对象的消失而消失

* 局部变量：随着方法的调用而存在，随着方法的调用完毕而消失

* D:初始化值不同

* 成员变量：有默认初始化值

* 局部变量：没有默认初始化值，必须定义，赋值，然后才能使用。

*注意: 局部变量名称可以和成员变量名称一样，在方法中使用的时候，采用的是就近原则。

方法的形式参数是类名的时候,传过去的是个地址值,在方法里用这个地址值在堆内存中找到你要传过来的对象.

匿名对象

new出来对象后,不用变量去接收它的地址值,这个对象在用过后,因为没有地址值指向它,它就被回收了.

面向对象封装

* 是指隐藏对象的属性和实现细节，仅对外提供公共访问方式。

封装好处

* 隐藏实现细节，提供公共的访问方式

* 提高了代码的复用性

* 提高安全性。

封装原则

* 将不需要对外提供的内容都隐藏起来。

* 把属性隐藏，提供公共方法对其访问。

private关键字(私有)

* a:是一个权限修饰符

* b:可以修饰成员变量和成员方法

* c:被其修饰的成员只能在本类中被访问

this关键字

this代表其所在函数所属对象的引用，或者说this代本类对象的引用说的明白一点，

哪个对象调用了this所在的函数，this就代表这个对象。

*this用途：

解决局部变量隐藏成员变量的问题