在实际的项目中看到一个很奇怪的现象，Java可以直接new一个接口，然后在new里面粗暴的加入实现代码。就像下面这样。**那么问题来了，new出来的对象没有实际的类作为载体，这不是很奇怪吗？**

思考以下代码的输出是什么？

Runnable x = new Runnable() {

@Override

public void run() {

System.out.println(this.getClass());

}

};

x.run();

实际答案是出现xxxx$1这样一个类名，它是编译器给定的名称。

# 在android中的使用

**三、匿名内部类（Android运用最多）**

　　　　匿名内部类就是没有名字的局部内部类，不使用关键字class, extends, implements, 没有构造方法。

　　　　什么情况下需要使用匿名内部类？如果满足下面的一些条件，使用匿名内部类是比较合适的：

　　　　　　a·只用到类的一个实例。  
　　　　　　b·类在定义后马上用到。  
　　　　　　c·类非常小（SUN推荐是在4行代码以下）  
　　　　　　d·给类命名并不会导致你的代码更容易被理解。

　　　　**在使用匿名内部类时，要记住以下几个原则：**

　　　　　　a·匿名内部类不能有构造方法。  
　　　　　　b·匿名内部类不能定义任何静态成员、方法和类。  
　　　　　　c·匿名内部类不能是public,protected,private,static。  
　　　　　　d·只能创建匿名内部类的一个实例。  
　　　　　　e·一个匿名内部类一定是在new的后面，用其隐含实现一个接口或实现一个类。  
　　　　　　f·因匿名内部类为局部内部类，所以局部内部类的所有限制都对其生效。

以下是匿名内部类的实例：

　　实例背景：输出每个Doctor博士每天做的事情

　　　　1、白天大家都是“白天传授理论知识”；

　　　　2、晚上每个人都是不一样的

　　\*用以前继承方法需要写多个实现类，而匿名内部类只需要实现“晚上做事workInNight()”这个方法就可以了