全部代码：<https://github.com/vintanda/JavaTest2018/tree/master/ProgramPractice/reflect>



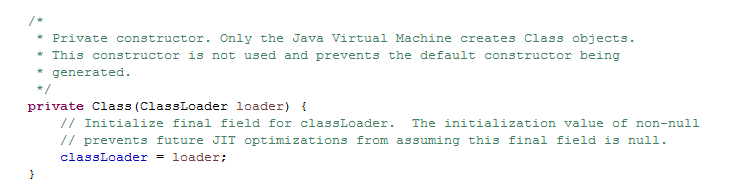
**一.Class类的使用**

java中万事万物皆对象，但除了两种：基本类型和静态的成员，基本类型也有包装类对其不是对象做了弥补。

1.那么什么是Class类呢？

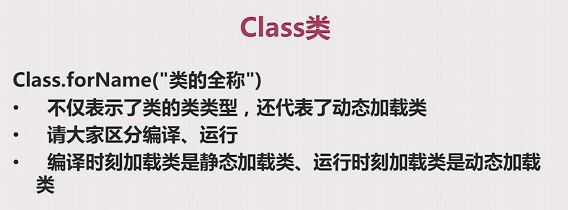
类是对象，类是java.lang.Class类的实例对象。我们写的类，都是Class类的实例对象

Class类的构造方法：



完整代码见：reflect/classDemo1.java

**2.如何Class动态一个类**



什么是动态加载，什么是静态加载？

new创建对象是类的静态加载，在编译时刻就要加载需要用到的类

通过动态加载可以解决该问题

因为有的类可能此时是不会使用的，所以暂时还不需要实现；

如果要使用其他的功能就无法实现，所以使用动态加载类，

在需要的时候再进行加载，再补充需要的部分

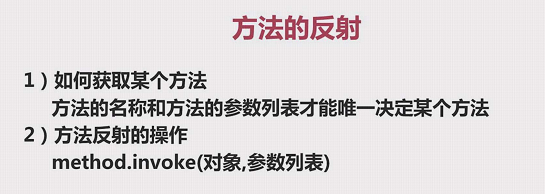
完整代码见：Office.java OfficeAble.java OfficeBetter.java Word.java

**3.Class类获取方法的信息、成员变量的信息和构造方法的信息**

要想获取一个类的信息，先获取该类的类类型，就可以获取该类的信息了

完整代码见：reflect/ClassUtil.java 和 reflect/ClassDemo3.java

**二.方法的反射**



完整代码见：reflect/MethodDemo1.java

**三.通过反射了解集合泛型的本质**

**通过Class、Method来认识泛型的本质**

完整代码见：reflect/MethodDemo2.java

从反射和泛型的例子我们可以看出：

反射的操作、Class的操作、Method的操作、Field的操作都是绕过编译，在运行时候执行的