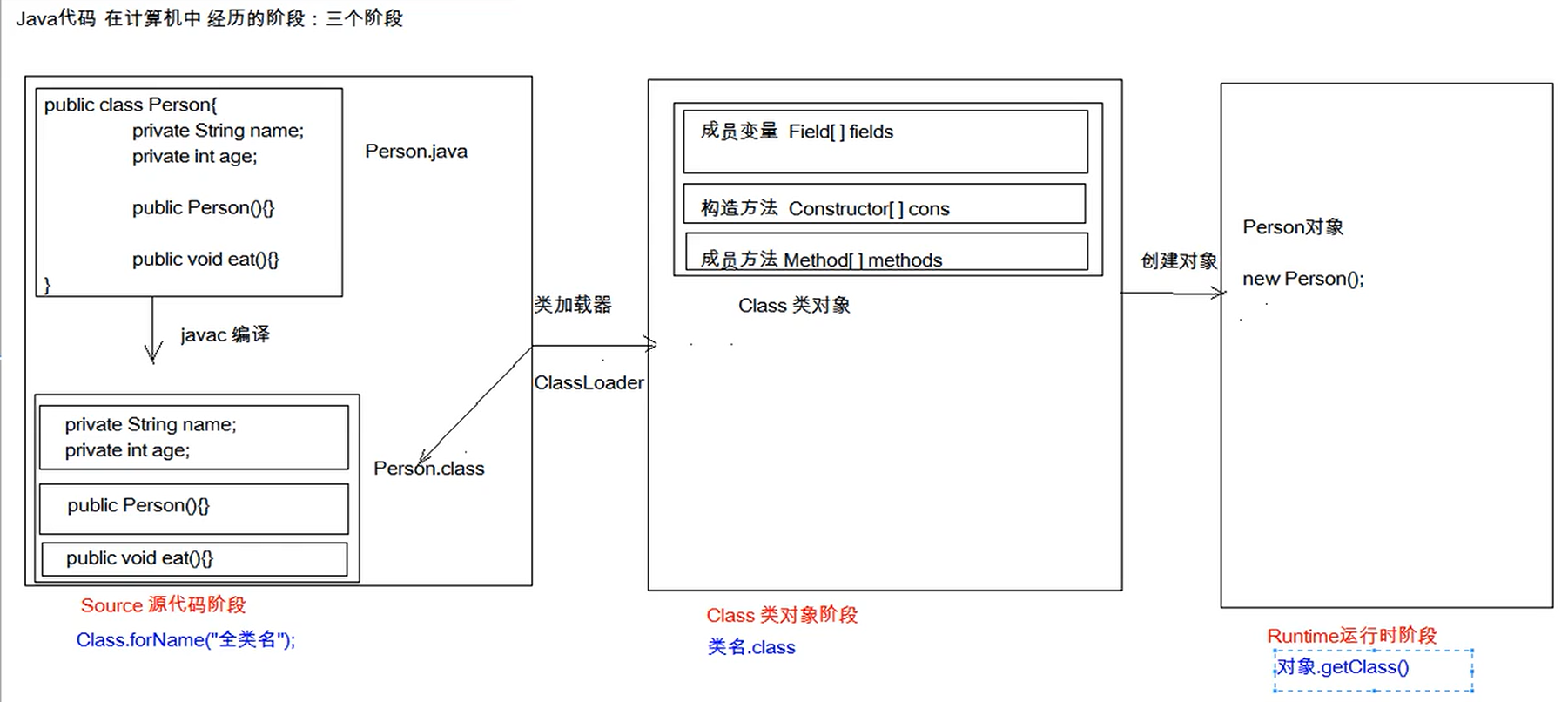
# 反射reflect（框架设计的灵魂）

1. 概念：**将类的各个组成部分封装为其他对象**，这就是反射。
2. 好处：①可以在程序运行过程中，操作这些对象。②可以解耦，提高程序的可扩展性。
3. 图示java代码运行阶段



Source源代码阶段，并没有进内存，还是在硬盘上，然后通过类加载器，加载进内存。

**4．获取calss类对象的方式（分别对应三个阶段）**

①class.forName(全类名):将字节码文件加载进内存，返回calss对象。（多用于配置文件）

②类名.class：通过类名的属性class获取（多用于参数传递）

1. 对象.getclass()：getclass()方法在Object类中定义着，即每个对象都有。



**\*Class对象的功能：**

**修饰符等级：public、protect、无、private**

**Ⅰ获取功能：无“Declared”获取public修饰的 有Declared获取所有，不考虑修饰符**

1. **获取成员变量们**

**Field[] getFields()，Field getfield(String name)**

**Field[] getDeclaredFields()，Field getDeclaredField(String name)**

1. **获取构造方法们**

**getConstructors()，getConstructor (String name)**

**getDeclaredConstructors()，getDeclaredConstructor (String name)**

1. **获取成员方法们**

**getMethods()，getMethod (String name)**

**getDeclaredMethods()，getDeclaredMethod (String name)**

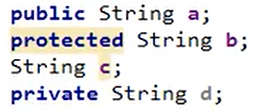
1. **获取类名**

**String getName()**

1. **获取包名**

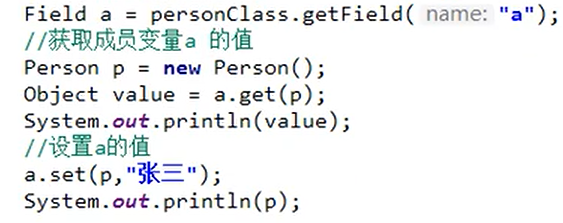
**String getpackage()**

**假如person类中定义以下成员属性：**



**拿到field对象后的操作：获取get、设置set**

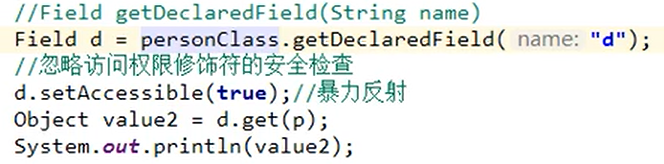




**反射面前可以说没有私有、公有这一说，我的是我的，你的也是我的，**

**d.setAccessible(true)暴力反射**

**例：**

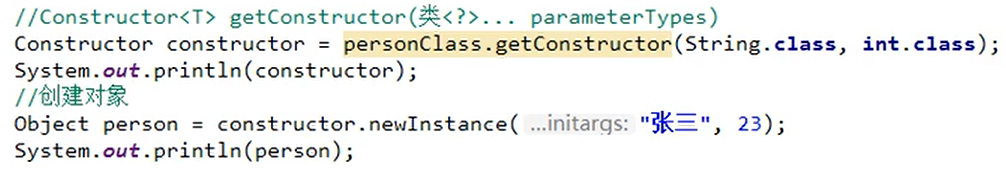


**d是私有的，但通过反射这里也可以被获取或者修改。**

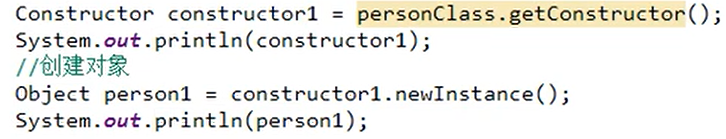
**拿到constructor对象后的操作：创建对象newInstance**

**例：（说明）String.clss和int.class那的参数是为了区分拿哪个构造器 相当于field(name)里的name参数**





**空参的构造创建对象：**



**空参的话这样也行，比较简单，用的多**

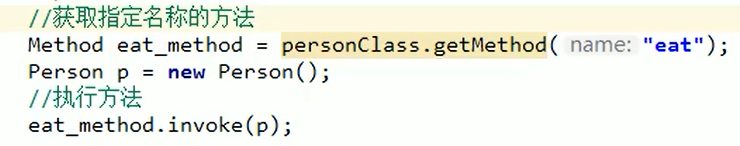


**拿到Method对象后的操作：执行方法invoke**

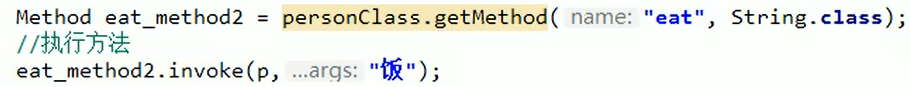


**例：（说明）getMethod（）里的参数有①方法名②方法参数（因为确定一个方法一看方法名，方法名相同看参数）。也是相当于Field（name）里的name**

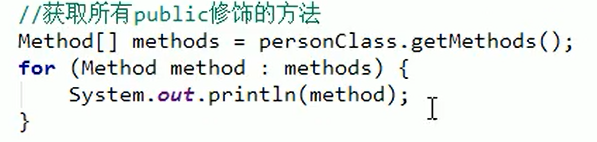
**无参数的：**

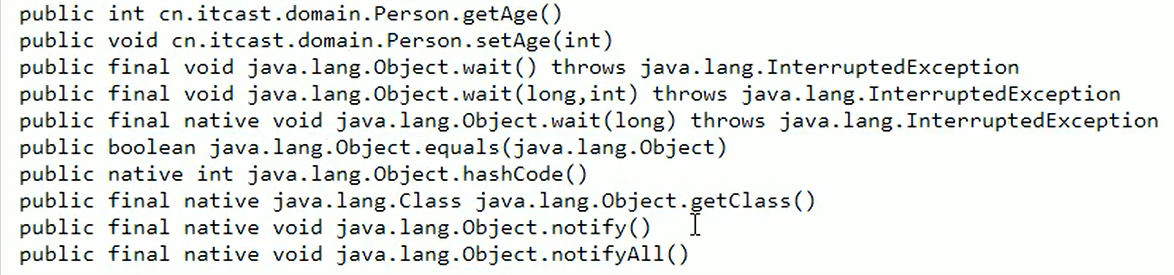


**有参数的：**



**获取所有方法：**

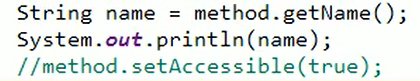




**打印结果说明：**

**自己定义的类中不仅有自己定义的方法，还有隐藏的从Object中继承过来的方法。**

**获取方法名：**



**获取类名：**

