Java 重写(Override)与重载(Overload)

重写是子类对父类的允许访问的方法的实现过程进行重新编写, 返回值和形参都不能改变。即外壳不变，核心重写！

**一个类中可以有多个构造函数，这多个构造函数是用重载实现的**

**重载与重写最大的区别就是，重载的参数不一样，重写参数必须一样**





方法的重写规则：

1. 参数列表与被重写方法的参数列表必须完全相同。
2. 返回类型与被重写方法的返回类型可以不相同，但是必须是父类返回值的派生类
3. 访问权限不能比父类中被重写的方法的访问权限更低。
4. 父类的成员方法只能被它的子类重写。
5. 声明为 final 的方法不能被重写。
6. 声明为 static 的方法不能被重写，但是能够被再次声明。
7. 子类和父类在同一个包中，那么子类可以重写父类所有方法，除了声明为 private 和 final 的 方法。
8. 子类和父类不在同一个包中，那么子类只能够重写父类的声明为 public 和 protected 的非 final 方法。
9. 重写的方法不能抛出新的强制性异常，或者比被重写方法声明的更广泛的强制性异常
10. 构造方法不能被重写。
11. 如果不能继承一个类，则不能重写该类的方法。

重载(Overload)

重载(overloading) 是在一个类里面，方法名字相同，而参数不同。返回类型可以相同也能不同。

每个重载的方法（或者构造函数）都必须有一个独一无二的参数类型列表。

最常用的地方就是构造器的重载。

