

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。

重写的好处：子类可以根据需要，定义特定于自己的行为。

```
Father father = new Son();  
father.eat();
```

编译时，用父类的源方法（语句左侧）验证调用方法。运行时，jvm调用是特定对象（语句右侧即father真实的原类型）的方法。（动态绑定）

方法重写的规则：

- 子类方法的访问权限必须大于或等于父类方法的访问权限。
- 重写的方法能够抛出任何非强制异常，无论被重写的方法是否抛出异常。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。
- 返回值类型小于等于父类
- 声明为final的方法不能被重写。声明为static的方法不能被重写，但是能够被再次声明。构造方法（1.不会被继承的方法，则不会被重写）不能重写（2.构造方法名和类名相同）

- 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。
- 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。