一，方法

是一段用来完成特定功能的代码片段。方法在其他语言中也叫函数。

语法 [修饰符]+[修饰符]+返回值类型+方法名(形式参数 1，形式参数 2…………){ //方法体 return null;//返回值 }

return :如果方法的返回值类型为 void 则可以不使用 return 如果方法需要返回值，则要返回相应的数据类型 1、形式参数(形参)：在方法被调用时用于接收外界输入的数据。形参中的变量，在方法中是 以局部变量的形式在使用。 2、实际参数(实参)：调用方法时实际传给方法的数据。 3、返回值类型：方法运行结束之后，一般会有运行结果来返回给调用者，返回值类型就是 运行结果的数据类型。若一个方法没有返回值，必须给出返回值类型 void 4、返回值：方法在执行完毕后返还给调用者的具体数据，该数据的数据类型必须与返回值 类型匹配。 5、return 语句终止方法的运行并指定要返回的数据。如果返回值类型为 void，则可以直接 写 return 而无需返回值，也可以省略 return 语句。

二, 构造方法

特点：

1） 名称必须和类名一致，与类名相同

2） 没有返回类型|void

3） 构造器可以存在 return，return 有选择的跳出构造器

4） 不能随便调用，和 new 在一起使用，其他时候不能调用

作用： 初始化对象信息，创建对象的 空构造： 没有参数的构造器、无参构造 2） 一个类中，如果没有显示|手动 加入 任意构造器 javac 编译后 自动加入空构造

一旦加入构造器， javac 不会加入空构造器

三，方法的重载

构成方法重载的必要条件是 方法名相同 参数列表不同（参数类型\当有两个不同 类型参数时，顺序不同\个数不同） 1.定义在同一个类中

四 ,方法重写

为什么子类要重写方法？ 由于父类中的方法功能过于简单，或者不适用于子类，这种情况需要对父类方法在子类中重 写。 方法重写规则 方法重写规则：两同两小一大 两同：方法名相同，参数列表相同 两小： 被重写的方法在子类中的返回值类型的范围比父类小或者与父类相同(返回值类型与父类一 致) 被覆盖的方法在子类中抛出的异常范围比父类小或者与父类相同 一大： 被覆盖的方法在子类中访问控制权限比父类的大或者相等 提示： 1. 可以在方法上加@Override 测试是否是重写的方法，如果不是，加了@Override 后编译会 出错 2. 在 Eclipse 中可以观察方法前面是否是绿色三角箭头，如果有代表是重写的方法。 方法重载和方法重写的区别

No. 比较点 重载 重写

1 单词 Overload Override

2 方法的声明 方法名相同，参数列表不同 方法名相同，参数列表相同

3 返回值 没有要求 返回值类型与父类一致

4 访问权限 没有要求 子类不能比父类更加严格

5 范围 同一个类中 继承关系中

方法中的参数传递 基本数据类型传递的是值 引用数据类型传递的对象的引用[内存地址]

形参：当方法被调用时用来接收用户输入的数据 实参：当方法被调用时用户传入的数据（传给方法的数据）

使用 super 调用父类构造器的语句必须是子类构造器的第一条语句 如果子类构造器没有显式地调用父类的构造器，则将自动调用父类的默认（没有参数） 的构造器。如果父类没有不带参数的构造器，并且在子类的构造器中又没有显式地调用父类 的构造器，则 java 编译器将报告错误