# 一、构造函数、静态代码块

## 1、构造函数

（1）函数名与类名相同

（2）没有返回值，也没有void

（3）可以有多个构造函数（函数重载）

（4）没有显示的构造函数，java编译器会提供一个默认的无参构造函数；如果有，则不提供；

（5）创建对象时，jvm会主动的调用对应的构造函数

## 2、构造函数、代码块、静态代码块的执行顺序

（1）构造函数

* + 对象一建立，jvm虚拟机就会立即调用构造函数，每创建一个对象，构造函数就会被调用一次。
  + 构造函数是用于给指定对象进行初始化的（为当前构建的对象进行初始化），**因为构造函数可以有多个，也就是可以创建多个不一样的对象。**
  + 构造函数初始化成员变量，不需要代码书写顺序，可引用全局变量。

（2）代码块

* 对象一建立就运行构造代码块，并且优先于构造函数。
* **构造代码块是对所有对象进行统一的初始化（通过该类创建的所有对象），构造函数是对当前对象就行初始化。**
* 构造代码块初始化成员变量，被初始化的成员变量书写必须在构造代码块之前。

（3）静态代码块

* **静态代码块是被类调用的，一旦类被加载，静态代码块就被执行，并且只被执行一次。**
* **静态代码块优于主函数，静态代码块是给类初始化的，而构造代码块是给对象初始化的**。
* 一个类中可以有多个静态代码块。按代码顺序从上到下顺序执行。

# 二、反编译

* + Jdk给我们提供了一个开发工具（javap.exe），给我们进行反编译的。

### 1、javap.exe反编译工具的使用格式

* + javap –c –l –private 类名

# 三、排序

1.冒泡排序

2.直接排序

3.排序工具方法

# 四、Instanceof、Typeof

## 1、Instanceof

* + Instanceof是java中的二元运算符，左边是对象，右边是类。当左边对象是右边类或子类所创建的对象时，返回true；否则返回false；
  + instanceof左边显示声明的对象类型，与右边的类必须是同种类或存在继承关系，也就是说要位于同一继承树，否则编译会报错

1. Instanceof用法

* 左边的对象实例不能是基本数据类型
* 左边对象实例和右边的 类必须在同一继承树上