**#基本数据类型（3类8种）**

##数值类 - byte、short、int、long、float、double

##字符型（文本型） - char

##布尔型 – Boolean

**#浮点型变量/常量**

浮点数是不精确的，一定不能用于比较。需要比较请使用BigDecimal类，使用BigDecimal进行浮点数的比较

**#字符型变量/常量**

字符型在内存中占2个字节，在Java中使用单引号来表示字符常量。例如’A’是一个字符。它与“A”是不同的，“A”表示含有一个字符的字符串.

char类型用来表示在Unicode编码表中的字符。Unicode编码被设计用来设计各种语言的文字，它占两个字节，可允许有65536个字符。

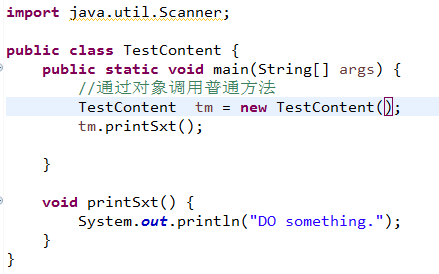
**#boolean类型变量/常量**  
boolean类型有两个常量值, true和false，在内存中占一位（不是一个字节），不可以使用0或非0的整数代替true和false，这点和C语言不同。boolean类型用来判断逻辑条件，一般用于程序流程控制。

**#语句块**

是用花括号扩起的任意数量的简单Java语句。块确定了局部变量的作用域。块中的程序代码，作为一个整体，是要被一起执行的。块可以被嵌套在另一个块中，但是不能在两个嵌套的块内声明同名的变量。语句块可以使用外部的变量，而外部不能使用语句块中定义的变量，因为语句块中定义的变量作用域只限于语句块。

**#方法的重载**

通过对象调用普通方法



**#面向过程和面向对象**

面向过程和面向对象都是对软件分析、设计和开发的一种思想，它指导着人们以不同的方法去分析、设计和开发软件。

**#面向对象和面向过程的总结**

##1 都是解决问题的四位方式，都是代码组织的方式

##2 解决简单问题可以使用面向过程

##3 解决复杂问题：宏观上使用面向对象把握，微观处理上仍然是面向过程。

#对象和类

##1 类可以看做是一个模板，或者图纸，系统根据类的定义来造出对象。

##2 类：class. 对象：Object, instance. 以后我们说某个类的对象，某个类的实例，是一样的意思。

**#面向对象的内存分析**

Java虚拟机的内存可以分为三个区域：栈stack、堆heap、方法区method area

##1 栈的特点如下：（1）栈描述的是方法执行的内存模型。每个方法被调用都会创建一个栈帧（存储局部变量、操作数、方法出口等）；（2）JVM为每个线程创建一个栈，用于存放该线程执行方法的信息（实际参数、局部变量等）；（3）栈属于线程私有，不能实现线程间的共享；（4）栈的存储特性是“先进后出，后进先出”；（5）栈是由系统自动分配，速度快！栈是一个连续的内存空间

##2 堆的特点如下：（1）堆用于存储创建好的对象和数组（数组也是对象）；（2）JVM只有一个堆，被所有线程共享；（3）堆是一个不连续的内存空间，分配灵活，速度慢

##3 方法区（又如静态区）特点如下：（1）JVM只有一个方法区，被所有线程共享；（2）方法区实际 也是堆，只有用于存储类、常量相关的信息；（3）用来存放程序中永远是不变或唯一的内容。（类信息【Class对象】、静态变量、字符串常量等）

**#构造器（构造方法）**

用于对象的初始化.要点：（1）通过new关键字调用；（2）构造器虽然有返回值，但是不能定义返回值类型（返回值的类型肯定是本类），不能在构造器里使用return返回某个值；（3）如果我们没有定义构造器，则编译器会自动定义一个无参的构造函数。如果已定义则编译器不会自动添加；（4）构造器的方法名必须和类名一致。

**#垃圾回收机制**

发现无用的对象，回收无用对象占用的内存空间。

相关算法：引用计数法；引用可达法（根搜索算法）。

**#通用的分代垃圾回收机制**

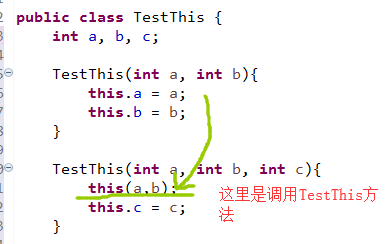
垃圾回收过程：（1）新创建的对象，绝大多数都会存储在Eden中；（2）当Eden满了（达到一定比例）不能创建新对象，则触发垃圾回收（GC），将无用对象清理掉，然后剩余对象复制到某个Survivor中，如S1，同时情况Eden区；（3）当Eden区再次满了，会将S1中的不能清空的对象，也复制到S1中，保证Eden和S1，均被清空；（4）重复多次（默认15次）Survivor中没有被清理的对象，则会复制到老年代Old（Tenured）区中；（5）当Old区满了，则会触发一个一次完整地垃圾回收（FullGC），之前新生代的垃圾回收为（minorGC）

**#对象创建的过程和this的本质**

##1 创建一个对象分为如下四步：（1）分配对象空间，并将对象成员变量初始化为0或空；（2）执行属性值的显式初始化；（3）执行构造方法；（4）返回对象的地址给相关的变量

##2 this的本质就是“创建好的对象的地址”！由于在构造方法调用前，对象已经创建。因此在构造方法中也可以使用this代表“当前对象”

##3This用法



##4 this不能用于static方法中

**#static关键字**

在类中，用static声明的成员变量为静态成员变量，也称为类变量。类变量的生命周期和类相同，在整个应用程序执行期间都有效。

static修饰的成员变量和方法，从属于类。普通变量和方法从属于对象。

**#静态初始化块**

构造方法用于对象的初始化，静态初始化块，用于类的初始化操作！在静态初始化块中不能直接访问非static成员。

注意事项：

静态初始化块执行顺序（学完继承再看见这里）：

1. 上溯到Object类，先执行Object的静态初始化块，再向下执行子类的静态初始化，直到我们的类的静态初始化块为止。
2. 构造方法执行顺序和上面顺序一样！

**#包**

包机制是Java中管理类的重要手段。开发中，我们会遇到大量同名的类，通过包我们很容易对解决类重名的问题，也可以实现对类的有效管理。包对于类，相当于文件夹对于文件的作业。

我们通过package实现对类的管理，package的使用有两个要点：

1. 通常是类的第一句非注释性语句
2. 包名：域名倒着写即可，再加上模块名，便于内部管理类。

##1注意：（1）写项目时都要加包，不要使用默认包；（2）com.go和com.gao.car，这两个没有包含关系，是两个完全独立的包，只是逻辑上看起来后者是前者的一部分