**八种基本数据类型：**

**byte、short、int、long、float、**

**double、boolean、char**

概念

**byte**

8位、有符号的以二进制补码表示的整数

min :  -128（-2^7）

max:   127(2^7-1)

default: 0

对应包装类：Byte

一般不用，但是如果计算确定在在-128-127之间且结果不超出范围的可以用来节省内存。

**short**

16位、有符号的以二进制补码表示的整数

min :    -32768（-2^15）

max:   32767（2^15 - 1）

default: 0

对应包装类：Short

一般不用，但是如果计算确定在在-32768-32767之间且结果不超出范围的可以用来节省内存。

**int**

32位、有符号的以二进制补码表示的整数

min :    -2,147,483,648（-2^31）

max:   2,147,483,647（2^31 - 1）

default: 0

对应包装类：Integer

常用数据类型，适用于一般整数运算中，遇到小数直接凉凉使用（double类型），阶乘超过10！直接数据溢出（使用double或者long类型）。

**long**

64位、有符号的以二进制补码表示的整数

min :    -9,223,372,036,854,775,808（-2^63）

max:   9,223,372,036,854,775,807（2^63 -1）

default: 0

对应的包装类：Long

是int的翻倍容量，常见在计算阶乘时使用，但是还是不能用于小数，小数左转double类型。

**float**

单精度、32位、符合IEEE 754标准的浮点数float 在储存大型浮点数组的时候可节省内存空间

浮点数不能用来表示精确的值，如货币

default: 0.0f

对应的包装类：Float

一般用来计算浮点数，应用范围没有double的双精度广。

**double**

双精度、64位、符合IEEE 754标准的浮点数

浮点数的默认类型为double类型

double类型同样不能表示精确的值，如货币

default: 0.0d

对应的包装类：Double

应用最为广泛的数据类型，可用于小数、浮点数的计算，双精度，存储容量很大，缺点：太占内存、不能表示开关结果。

**char**

char类型是一个单一的 16 位 Unicode 字符

最小值是 \u0000（即为0）

最大值是 \uffff（即为65,535）

char 数据类型可以储存任何字符

对应的包装类：Character

换行最常用的方法，“\n”,用计算机对应的Unicode码表示，可以表示基本上所有的字符，缺点也很明显，没有Unicode 字符表基本上寸步难行。

**boolean**

boolean数据类型表示一位的信息

只有两个取值：true 和 false

这种类型只作为一种标志来记录 true/false 情况

对应的包装类：Boolean

用于开关的制作，反馈是和否的结果，相当于一个判断门。