

数值类型的基本类型的取值范围，我们无需强制去记忆，因为它们的值都已经以常量的形式定义在对应的包装类中（`java.lang.xxx`）了。实际上，JAVA中还存在另外一种基本类型`void`，它也有对应的包装类 `java.lang.Void`，不过我们无法直接对它们进行操作。

byte :

- byte数据类型是8位、有符号的，以二进制补码表示的整数；
- 最小值是-128 (-2^7) ；
- 最大值是127 (2^7-1) ；
- 默认值是0；
- byte类型用在大型数组中节约空间，主要代替整数，因为byte变量占用的空间只有int类型的四分之一；
- 例子：byte a = 100 , byte b = -50。

short :

- short数据类型是16位、有符号的，以二进制补码表示的整数
- 最小值是-32768 (-2^{15}) ；
- 最大值是32767 ($2^{15} - 1$) ；
- Short数据类型也可以像byte那样节省空间。一个short变量是int型变量所占空间的二分之一；
- 默认值是0；
- 例子：short s = 1000 , short r = -20000。

int :

- int数据类型是32位、有符号的以二进制补码表示的整数；
- 最小值是-2,147,483,648 (-2^{31}) ；
- 最大值是2,147,483,647 ($2^{31} - 1$) ；
- 一般地整型变量默认为int类型；
- 默认值是0；
- 例子：int a = 100000, int b = -200000。

long :

- long数据类型是64位、有符号的以二进制补码表示的整数；
- 最小值是-9,223,372,036,854,775,808 (-2^{63}) ；
- 最大值是9,223,372,036,854,775,807 ($2^{63} - 1$) ；
- 这种类型主要使用在需要比较大整数的系统上；
- 默认值是0L；
- 例子：long a = 100000L , long b = -200000L。

float :

- float数据类型是单精度、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；
- 默认值是0.0f；
- 浮点数不能用来表示精确的值，如货币；
- 例子：float f1 = 234.5f。

double :

- double数据类型是双精度、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；
- 默认值是0.0d；
- 例子：double d1 = 123.4。

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；
- 默认值是false；
- 例子：boolean one = true。

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；
- 例子：char letter = 'A'。