**Java Number类**

一般地，当需要使用数字的时候，我们通常使用内置数据类型，如：byte、int、long、double等。

**实例**

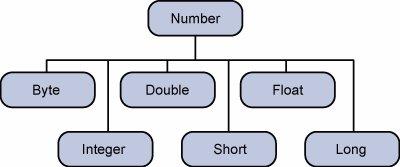
int a = 5000;

float b = 13.65;

byte c = 0x4a;

然而，在实际开发过程中，我们经常会遇到需要使用对象，而不是内置数据类型的情形。为了解决这个问题，Java语言为每一个内置数据类型提供了对应的包装类。

所有的包装类（Integer、Long、Byte、Double、Float、Short）都是抽象类Number的子类。



这种由编译器特别支持的包装称为装箱，所以当内置数据类型被当作对象使用的时候，编译器会把内置类型装箱为包装类。相似的，编译器也可以把一个对象拆箱为内置类型。Number类属于java.lang包。

下面是一个使用 Integer 对象的实例：

public class Test{

public static void main(String args[]){

Integer x = 5;

x = x + 10;

System.out.println(x);

}

}

以上实例编译运行结果如下：

15

当x被赋为整型值时，由于x是一个对象，所以编译器要对x进行装箱。然后，为了使x能进行加运算，所以要对x进行拆箱。

**Number类的成员方法**

下面的表中列出的是Number类的方法：

|  |  |
| --- | --- |
| **序号** | **方法与描述** |
| 1 | xxxValue() 将number对象转换为xxx数据类型的值并返回。 |
| 2 | compareTo() 将number对象与参数比较。 |
| 3 | equals() 判断number对象是否与参数相等。 |
| 4 | valueOf() 返回一个Integer对象指定的内置数据类型 |
| 5 | toString() 以字符串形式返回值。 |
| 6 | parseInt() 将字符串解析为int类型。 |
| 7 | abs() 返回参数的绝对值。 |
| 8 | ceil() 对整形变量向左取整，返回类型为double型。 |
| 9 | floor() 对整型变量向右取整。返回类型为double类型。 |
| 10 | rint() 返回与参数最接近的整数。返回类型为double。 |
| 11 | round() 返回一个最接近的int、long型值。 |
| 12 | min() 返回两个参数中的最小值。 |
| 13 | max() 返回两个参数中的最大值。 |
| 14 | exp() 返回自然数底数e的参数次方。 |
| 15 | log() 返回参数的自然数底数的对数值。 |
| 16 | pow() 返回第一个参数的第二个参数次方。 |
| 17 | sqrt() 求参数的算术平方根。 |
| 18 | sin() 求指定double类型参数的正弦值。 |
| 19 | cos() 求指定double类型参数的余弦值。 |
| 20 | tan() 求指定double类型参数的正切值。 |
| 21 | asin() 求指定double类型参数的反正弦值。 |
| 22 | acos() 求指定double类型参数的反余弦值。 |
| 23 | atan() 求指定double类型参数的反正切值。 |
| 24 | atan2() 将笛卡尔坐标转换为极坐标，并返回极坐标的角度值。 |
| 25 | toDegrees() 将参数转化为角度。 |
| 26 | toRadians() 将角度转换为弧度。 |
| 27 | random() 返回一个随机数。 |