**Java运算符优先级**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序列号** | **符号** | **名称** | **结合性(与操作数)** | **目数** | **说明** |
| **1** | **.** | **点** | **从左到右** | **双目** |  |
| **( )** | **圆括号** | **从左到右** |  |  |
| **[ ]** | **方括号** | **从左到右** |  |  |
| **2** | **+** | **正号** | **从右到左** | **单目** |  |
| - | **负号** | **从右到左** | **单目** |  |
| **++** | **自增** | **从右到左** | **单目** | 前缀增，后缀增 |
| - - | **自减** | **从右到左** | 前缀减，后缀减 |
| **~** | **按位非/取补运算** | **从右到左** | **单目** |  |
| **！** | **逻辑非** | **从右到左** | **单目** | “[!”不可以与“=”联用](D:/Temp/Java%E8%BF%90%E7%AE%97%E7%AC%A6+(%E4%BC%98%E5%85%88%E7%BA%A7%E3%80%81%E7%9B%AE%E6%95%B0)+.doc" \l "非不可以与等号联用) |
| **3** | **\*** | **乘** | **从左到右** | **双目** |  |
| **/** | **除** | **从左到右** | **双目** | 整数除法：取商的整数部分，小数部分去掉，不四舍五入 |
| **%** | **取余** | **从左到右** | **双目** |  |
| **4** | **+** | **加** | **从左到右** | **双目** |  |
| **-** | **减** | **从左到右** | **双目** |  |
| **5** | **<<** | **左移位运算符** | **从左到右** | **双目** |  |
| **>>** | **带符号右移位运算符** | **从左到右** | **双目** |  |
| **>>>** | **无符号右移** | **从左到右** | **双目** |  |
| **6** | **<** | **小于** | **从左到右** | **双目** | [关系运算符“大于”说明](D:/Temp/Java%E8%BF%90%E7%AE%97%E7%AC%A6+(%E4%BC%98%E5%85%88%E7%BA%A7%E3%80%81%E7%9B%AE%E6%95%B0)+.doc" \l "关系运算符“大于”说明) |
| **<=** | **小于或等于** | **从左到右** | **双目** |  |
| **>** | **大于** | **从左到右** | **双目** |  |
| **>=** | **大于或等于** | **从左到右** | **双目** |  |
| **instanceof** | **确定某对象是否属于指定的类** | **从左到右** | **双目** |  |
| **7** | **==** | **等于** | **从左到右** | **双目** | [关系运算符“==”说明](D:/Temp/Java%E8%BF%90%E7%AE%97%E7%AC%A6+(%E4%BC%98%E5%85%88%E7%BA%A7%E3%80%81%E7%9B%AE%E6%95%B0)+.doc" \l "关系运算符“等号”说明) |
| **!=** | **不等于** | **从左到右** | **双目** |  |
| **8** | **&** | **按位与** | **从左到右** | **双目** |  |
| **9** | **|** | **按位或** | **从左到右** | **双目** |  |
| **10** | **^** | **按位异或** | **从左到右** | **双目** |  |
| **11** | **&&** | **短路与** | **从左到右** | **双目** |  |
| **12** | **||** | **短路或** | **从左到右** | **双目** |  |
| **13** | **? :** | **条件运算符** | **从右到左** | **三目** |  |
| **14** | **=** | **赋值运算符** | **从右到左** | **双目** |  |
| **+=** | **混合赋值运算符** |  |
| **-=** |  |
| **\*=** |  |
| **/=** |  |
| **%=** |  |
| **&=** |  |
| **|=** |  |
| **^=** |  |
| **<<=** |  |
| **>>=** |  |
| **>>>=** |  |

**说明：**

**1、算数运算符**

　　　　+ :加法

　　　　- ：减法

　　　　\* ：乘法

　　　　/ ：除法

　　　　% ：取余运算

**2、关系运算符**

　　　　< ：只能比较基本类型数据之间的关系，不能比较对象之间的关系。

　　　　> : (同关系运算符“<”)

　　　　<=: (同关系运算符“<”)

　　　　>=: (同关系运算符“<”)

　　　　== ：若使用该运算符比较两个对象的引用（变量），则实质上是比较两个变量是否引用了相同的对象。所谓相同的对象是指，是否是在堆栈（Heap）中开辟的同一块儿内存单元中存放的对象。

　　若比较两个对象的引用(变量)所引用的对象的内容是否相同，则应该使用equals()方法，该方法的返回值类型是布尔值。需要注意的是：若用类库中的类创建对象，则对象的引用调用equals()方法比较的是对象的内容；若用自定义的类来创建对象，则对象的引用调用equals()方法比较的是两个引用是否引用了同一个对象，因为第二种情况equals()方法默认的是比较引用。

　　　　!= ：(同关系运算符“==”)

**3、逻辑运算符 （操作符只能是布尔类型的）**

　　　　&&

　　　　||

　　　　!

[IMG_256](http://www.cnblogs.com/gw811/archive/2012/10/13/javascript:void(0);" \o "复制代码)

1 public class Demo { 2 public static void main(String[] args) { 3 // System.out.println((!'1'||'1')+5);//编译错误 4 // System.out.println(!5);//编译错误 5 // System.out.println(('1'||'1')+5);//编译错误 6 // System.out.println(1||2);//编译错误 7 // System.out.println(5-3||4-2);//编译错误 8 System.out.println(5<3||4>3);//true 9 }10 }

[IMG_257](http://www.cnblogs.com/gw811/archive/2012/10/13/javascript:void(0);" \o "复制代码)

**4、位运算符**

　　　　&

　　　　|

　　　　^

　　　　! ：不可以与=联用，因为!是一元操作符；不可以对布尔类型的数据进行按位非运算

**5、移位运算符（只能处理整数运算符）**

　　Char、byte、short类型，在进行移位之前，都将被转换成int类型，移位后的结果也是int类型；移位符号右边的操作数只截取其二进制的后5位（目的是防止因为移位操作而超出int类型的表示范围：2的5次方是32，int类型的最大范围是32位）；对long类型进行移位，结果仍然是long类型，移位符号右边的操作符只截取其二进制的后6位。

　　<< ：

　　>> ：若符号位为正，则在最高位插入0；若符号位为负，则在最高位插入1

　　>>> ：无论正负，都在最高位插入0

[IMG_258](http://www.cnblogs.com/gw811/archive/2012/10/13/javascript:void(0);" \o "复制代码)

1 public class Demo { 2 public static void main(String[] args) { 3 // 0000 0000 0000 0101 4 // 0000 0000 0000 0010 5 System.out.println(5>>1);//2 6 // 0000 0000 0000 1000 7 // 0000 0000 0000 0100 8 System.out.println(8>>1);//4 9 System.out.println(3+5>>1);//410 System.out.println(3+(5>>1));//511 }12 }