



Scala核心编程

-运算符

尚硅谷研究院

运算符介绍

- 运算符介绍

运算符介绍

运算符是一种特殊的符号，用以表示数据的**运算**、**赋值**和**比较**等。

- 1) 算术运算符
- 2) 赋值运算符
- 3) 比较运算符(关系运算符)
- 4) 逻辑运算符
- 5) 位运算符



- 算术运算符

介绍

算术运算符(arithmetic)是对数值类型的变量进行运算的，在Scala程序中使用的非常多。

- 算术运算符

算术运算符一览

运算符	运算	范例	结果
+	正号	+3	3
-	负号	b=4; -b	-4
+	加	5+5	10
-	减	6-4	2
*	乘	3*4	12
/	除	5/5	1
%	取模(取余)	7%5	2
+	字符串相加	"He"+"llo"	"Hello"



● 算术运算符

案例演示

案例演示算术运算符的使用(Operator.scala)。

- 1) +, -, *, /, % 重点讲解 /、%
- 2) +、-、* 是一个道理，完全可以类推。
- 3) 算数运算符的运算规则和Java一样

```
var r1 : Int = 10 / 3
println("r1=" + r1)
var r2 : Double = 10 / 3
println("r2=" + r2)
var r3 : Double = 10.0 / 3
println("r3=" + r3 )
println("r3=" + r3.formatted("%.2f") ) //
```

● 算术运算符

细节说明

- 1) 对于除号 “/”，它的整数除和小数除是有区别的：整数之间做除法时，只保留整数部分而舍弃小数部分。例如：`var x : Int = 10/3` ,结果是 3
- 2) 当对一个数取模时，可以等价 $a \% b = a - a/b * b$ ，这样我们可以看到取模的一个本质运算(和java 的取模规则一样)。

3) 注意：Scala中没有++、--操作符，需要通过+=、-=来实现同样的效果

```
println(10 % 3) //1
println(-10 % 3) //-1
println(10 % -3) //.
println(-10 % -3) //
```



- 算术运算符

课堂练习

- 1) 假如还有97天放假，问：xx个星期零xx天
- 2) 定义一个变量保存华氏温度，华氏温度转换摄氏温度的公式为： $5/9 * (\text{华氏温度} - 100)$ ，请求出华氏温度对应的摄氏温度。[测试：232.5]



- 关系运算符(比较运算符)

基本介绍

- 1) 关系运算符的结果都是boolean型，也就是要么是true，要么是false
- 2) 关系表达式 经常用在 **if结构**的条件中或**循环结构**的条件中
- 3) 关系运算符的使用和java一样



- 关系运算符(比较运算符)

关系运算符一览

运算符	运算	范例	结果
<code>==</code>	相等于	<code>4==3</code>	false
<code>!=</code>	不等于	<code>4!=3</code>	true
<code><</code>	小于	<code>4<3</code>	false
<code>></code>	大于	<code>4>3</code>	true
<code><=</code>	小于等于	<code>4<=3</code>	false
<code>>=</code>	大于等于	<code>4>=3</code>	true

- 关系运算符(比较运算符)

案例演示

案例演示关系运算符的使用(Operator02.scala)。

```
var a = 9
var b = 8
println(a>b) // true
println(a>=b) // true
println(a<=b) // false
println(a<b) // false
println(a==b) // false
println(a!=b) // true
var flag : Boolean = a>b // true
```

- 关系运算符

细节说明

- 1) 关系运算符的结果都是Boolean型，也就是要么是true，要么是false。
- 2) 关系运算符组成的表达式，我们称为关系表达式。 $a > b$
- 3) 比较运算符 “==” 不能误写成 “=”
- 4) 使用陷阱: 如果两个浮点数进行比较，应当保证数据类型一致。

- 逻辑运算符

介绍

用于连接多个条件（一般来讲就是关系表达式），最终的结果也是一个Boolean值。

- 逻辑运算符

逻辑运算符一览

假定变量 A 为 true, B 为 false

运算符	描述	实例
&&	逻辑与	(A && B) 运算结果为 false
	逻辑或	(A B) 运算结果为 true
!	逻辑非	!(A && B) 运算结果为 true

案例演示:

```
var a = true
var b = false
println("a && b = " + (a && b)) // false
println("a || b = " + (a || b)) // true
println("!(a && b) = " + !(a && b)) // true
```



● 赋值运算符

介绍

赋值运算符就是将某个运算后的值，赋给指定的变量。

赋值运算符的分类

运算符	描述	实例
=	简单的赋值运算符，将一个表达式的值赋给一个左值	$C = A + B$ 将 $A + B$ 表达式结果赋值给 C
+=	相加后再赋值	$C += A$ 等于 $C = C + A$
-=	相减后再赋值	$C -= A$ 等于 $C = C - A$
*=	相乘后再赋值	$C *= A$ 等于 $C = C * A$
/=	相除后再赋值	$C /= A$ 等于 $C = C / A$
%=	求余后再赋值	$C \% = A$ 等于 $C = C \% A$

● 赋值运算符

赋值运算符的分类

运算符	描述	实例
<<=	左移后赋值	$C \ll= 2$ 等于 $C = C \ll 2$
>>=	右移后赋值	$C \gg= 2$ 等于 $C = C \gg 2$
&=	按位与后赋值	$C \&= 2$ 等于 $C = C \& 2$
^=	按位异或后赋值	$C \wedge= 2$ 等于 $C = C \wedge 2$
=	按位或后赋值	$C = 2$ 等于 $C = C 2$

说明：这部分的赋值运算涉及到二进制相关知识，其运行的规则和Java一样。



● 赋值运算符

案例演示

交换两个数的值。

```
var a = 10
var b = 99
println("a="+a+"\tb="+b)
//交换
val t = a
a = b
b = t
println("a="+a+"\tb="+b)
```

赋值运算符特点

- 1) 运算顺序从右往左
- 2) 赋值运算符的左边 只能是变量,右边 可以是变量、表达式、常量值/字面量
- 3) 复合赋值运算符等价于下面的效果
比如: `a+=3` 等价于 `a=a+3`



- 赋值运算符

面试题

有两个变量，a和b，要求将其进行交换，但是不允许使用中间变量，最终打印结果：[学员做]

```
val a = 10
```

```
val b
```




● 位运算符

运算符	描述	实例
&	按位与运算符	(a & b) 输出结果 12 ， 二进制解释： 0000 1100
	按位或运算符	(a b) 输出结果 61 ， 二进制解释： 0011 1101
^	按位异或运算符	(a ^ b) 输出结果 49 ， 二进制解释： 0011 0001
~	按位取反运算符	(~a) 输出结果 -61 ， 二进制解释： 1100 0011， 在一个有符号二进制数的补码形式。
<<	左移动运算符	a << 2 输出结果 240 ， 二进制解释： 1111 0000
>>	右移动运算符	a >> 2 输出结果 15 ， 二进制解释： 0000 1111
>>>	无符号右移	A >>> 2 输出结果 15, 二进制解释: 0000 1111

说明: 位运算符的规则和Java一样

- 运算符的特别说明

Scala不支持三目运算符, 在Scala 中使用 if – else 的方式实现。

```
val num = 5 > 4 ? 5 : 4 //没有
```

```
val num = if (5>4) 5 else 4
```

课堂练习

- 1) 案例1: 求两个数的最大值
- 2) 案例2: 求三个数的最大值



Scala的设计理念:
一种事情尽量只有一种方法完成,
这样可以让代码风格更加统一.



● 运算符优先级

1) 运算符有不同的优先级，所谓优先级就是表达式运算中的**运算顺序**。如右表，上一行运算符总优先于下一行。

2) 只有单目运算符、赋值运算符是从右向左运算的。

3) 运算符的优先级和Java一样。

小结运算符的优先级

1.() []

2.单目运算

3.算术运算符

4.移位运算

5.比较运算符(关系运算符)

6.位运算

7.关系运算符

8.赋值运算符

类别	运算符	关联性
1	() []	左到右
2	! ~	右到左
3	* / %	左到右
4	+ -	左到右
5	>> >>> <<	左到右
6	> >= < <=	左到右
7	== !=	左到右
8	&	左到右
9	^	左到右
10		左到右
11	&&	左到右
12		左到右
13	= += -= *= /= %= >>= <<= &= ^= =	右到左
14	,	左到右

高



低



● 键盘输入语句

介绍

在编程中，需要接收用户输入的数据，就可以使用键盘输入语句来获取。

InputDemo.scala

步骤：

- 1) 导入该类的所在包
- 2) 创建该类对象（声明变量）
- 3) 调用里面的功能

```
System.out.println("请输入姓名");  
Scanner scanner = new Scanner(System.in);  
String name = scanner.next();  
System.out.println("请输入年龄");  
int age = scanner.nextInt();  
System.out.println("请输入薪水");  
double salary = scanner.nextDouble();
```

```
println("name")  
var name = StdIn.readLine()  
println("age")  
var age = StdIn.readInt()  
println("sal")  
var sal = StdIn.readDouble()
```

案例演示：

要求：可以从控制台接收用户信息，【姓名，年龄，薪水】。

1) 回顾Java的实现//不说明

2) **Scala的实现** 【import scala.io.StdIn】



谢谢！ 欢迎收看！