

计算机二级Java的选择语句 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/138/2021\\_2022\\_\\_E8\\_AE\\_A1\\_E7\\_AE\\_97\\_E6\\_9C\\_BA\\_E4\\_c97\\_138713.htm](https://www.100test.com/kao_ti2020/138/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E4_c97_138713.htm)

编程语言使用控制

( control ) 语句来产生执行流，从而完成程序状态的改变，

如程序顺序执行和分支执行。Java 的程序控制语句分为以下

几类:选择，重复和跳转。根据表达式结果或变量状态选择

( Selection ) 语句来使你的程序选择不同的执行路径。重复

( Iteration ) 语句使程序能够重复执行一个或一个以上语句 (

也就是说，重复语句形成循环)。跳转 ( Jump ) 语句允许你的

程序以非线性的方式执行。下面将分析Java 的所有控制语句。

如果你熟悉C/C，那么掌握Java 的控制语句将很容易。

事实上，Java 的控制语句与C/C 中的语句几乎完全相同。当然

它们还是有一些差别的尤其是break语句与continue 语句。

Java 支持两种选择语句：if语句和switch语句。这些语句允许

你只有在程序运行时才能知道其状态的情况下，控制程序的

执行过程。如果你没有C/C 的编程背景，你将会为这两个语句

的强大功能和灵活性而感到吃惊。5.1.1 if语句 if语句曾在

第2章中介绍过，我们将在这里对它进行详细讨论。if语句

是Java 中的条件分支语句。它能将程序的执行路径分为两条

。if语句的完整格式如下：if (condition) statement1. else

statement2. 其中，if和else的对象都是单个语句 ( statement ) ，

也可以是程序块。条件condition 可以是任何返回布尔值的表达式。

else子句是可选的。if语句的执行过程如下:如果条件为真，

就执行if的对象 ( statement1 ) ；否则，执行else 的对象

( statement2 ) 。任何时候两条语句都不可能同时执行。考虑

下面的例子：`int a, b. // ...if(a < b) a = 0; else b = 0;` 本例中，如果a小于b，那么a被赋值为0；否则，b被赋值为0。任何情况下都不可能使a和b都被赋值为0。通常，用于控制if语句的表达式都包含关系运算符。当然，这在技术上并不是必要的。仅用一个布尔值来控制if语句也是可以的，如下面的程序段：`boolean dataAvailable. // ... if (dataAvailable) ProcessData(). else waitForMoreData().` 记住，直接跟在if或else语句后的语句只能有一句。如果你想包含更多的语句，你需要建一个程序块，如下面的例子：`int bytesAvailable. // ... if (bytesAvailable > 0) { ProcessData().bytesAvailable -= n.} else waitForMoreData().` 这里，如果变量bytesAvailable大于0，则if块内的所有语句都会执行。一些程序员觉得在使用if语句时在其后跟一个大括号是很方便的，甚至在只有一条语句的时候也使用大括号。这使得在日后添加别的语句变得容易，并且你也不必担心忘记括号。事实上，当需要定义块时而未对其进行定义是一个导致错误的普遍原因。例如，考虑下面的程序段：`int bytesAvailable. // ... if (bytesAvailable > 0) { ProcessData().bytesAvailable -= n. } else waitForMoreData(). bytesAvailable = n.` 由于编排的原因，看起来似乎bytesAvailable = n语句应该在else子句中执行。然而，当你调用时，空白对Java无关紧要，编译器无法知道你的意图。这段程序会通过编译，但运用时会出错。上述例子应修改如下：`int bytesAvailable. // ... if (bytesAvailable > 0) { ProcessData(). bytesAvailable -= n. } else { waitForMoreData(). bytesAvailable = n. }` 嵌套if语句 嵌套（nested）if语句是指该if语句为另一个if或者else语句的对象。在编程时经常要用到嵌套if语句。当你使用嵌套if语句时，需记住的要点就是：一个else

语句总是对应着和它在同一个块中的最近的if语句，而且该if语句没有与其他else语句相关联。下面是一个例子：`if(i == 10) {if(j < 100) c = d. // this if is else a = c. // associated with this else } else a = d. // this else refers to if(i == 10)` 如注释所示，最后一个else语句没有与if ( j < 100 )，因为它是同一个块中最近的if语句。

if-else-if 阶梯 基于嵌套if语句的通用编程结构被称为 if-else-if 阶梯。它的语法如下：`if(condition) statement. else if(condition) statement. else if(condition) statement. else statement.` 条件表达式从上到下被求值。一旦找到为真的条件，就执行与它关联的语句，该阶梯的其他部分就被忽略了。如果所有的条件都不为真，则执行最后的else语句。最后的else语句经常被作为默认的条件，即如果所有其他条件测试失败，就执行最后的else语句。如果没有最后的else语句，而且所有其他的条件都失败，那程序就不做任何动作。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)