二级JAVA第四章辅导：流程控制循环语句 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__E4_BA_8C_E7_BA_A7JAVA_c97_138822.htm §4.2循环语句 循环语句的作用是反复执行一段代码，直到满足终止循环条件为止，一个循环一般应包括四部分内容： 1.初始化部分(initialization)：用来设置循环的一些初始条件，计数器清零等。 2.循环体部分(body)：这是反复循环的一段代码，可以是单一一条语句，也可以是复合语句。 3.迭代部分(iteration)：这是在当前循环结束，下一次循环开始执行的语句，常常用来使计数器加1或减1。 4.终止部分(termination)：通常是一个布尔表达式，每一次循环要对该表达式求值，以验证是否满足循环终止条件。 Java中提供的循环语句有：while语句，do-while语句和for语句，下面分别介绍： 4.2.1while语句 while语句实现"当型"循环，它的一般格式为： [initialization] while (termination){ body. [iteration.] } 1.当布尔表达式(termination)的值为true时，循环执行大括号中语句。并且初始化部分和迭代部分是任选的。 2.while语句首先计算终止条件，当条件满足时，才去执行循环中的语句。这是"当型"循环的特点。 4.2.2do-while语句 do-while语句实现"直到型"循环，它的一般格式为： [initialization] do{ body. [iteration.] }while (termination). 1.do-while语句首先执行循环体，然后计算终止条件，若结果为true，则循环执行大括号中的语句，直到布尔表达式的结果为false。 2.与while语句不同的是，do-while语句的循环体至少执行一次，是"直到型"循环的特点。 4.2.3for语句 for语句也用来实现"当型"循环，它的一般格式为： for

(initialization.termination.iteration){ body. } 1.for语句执行时，首先执行初始化操作，然后判断终止条件是否满足，如果满足，则执行循环体中的语句，最后执行迭代部分。完成一次循环后，重新判断终止条件。 2.可以在for语句的初始化部分声明一个变量，它的作用域为一个for语句。 3.for语句通常用来执行循环次数确定的情况(如对数组元素进行操作)，也可以根据循环结束条件执行循环次数不确定的情况。 4.在初始化部分和迭代部分可以使用逗号语句，来进行多个动作。逗号语句是用逗号分隔的语句序列。例如： for(i=0，j=10.i＜j.i，j--){ …… } 5.初始化、终止以及迭代部分都可以为空语句(但分号不能)，三者均为空的时候，相当于一个无限循环。 4.2.4continue语句 1.continue语句用来结束本次循环，跳过循环体中下面尚未执的语句，接着进行终止条件的判断，以决定是否继续循环。对于for语句，在进行终止条件判断前，还要先执行迭代语句。它的格式为： continue. 2.也可以用continue跳转到括号指明的外层循环中，这时的格式为： continue outerLable. 例如： outer:for(int i=0.i＜10.i){ // 外层循环 for(int j=0.j＜20.j){ // 内层循环 if(j＞i){ …… continueouter. } …… } …… } 该例中，当满足j＞i的条件时，程序执行完相应的语句后跳转至外层循环，执行外层循环的迭代语句i ；然后开始下一次循环。 4.2.5举例 例4.4下例分别用while、do-while和for语句实现累计求和。 public class Sum{ public static void main(String args[]){ System.out.println("\n**while statement**"). int n=10，sum=0. // initialization while(n＞0){ // termination sum =n. // body n--. // iteration } System.out.println("sumis" sum). System.out.println("\n**do_whilestatement**"). n=0. // initialization

sum=0. do{ sum =n. // body n . // iteration }while(n<=10).
// termination System.out.println("sumis" sum).
System.out.println("\n**forstatement**"). sum=0. for(inti=1.i
<=10.i){ // initialization， termination， iteration sum =i. }
System.out.println("sumis" sum). } } 运行结果为： C:\>java Sum
whilestatement sum is 55 **do_while statement** sum is 55 **for
statement** sum is 55 可以从中来比较这三种循环语句，从而在
不同的场合选择合适的语句。 例4.5求100～200间的所有素数
public class PrimeNumber{ public static void main(String args[]){
System.out.println("**prime numbers between 100 and 200**").
intn=0. outer:for(inti=101.i<200.i =2){ // outer loop int k=15.
// 0select for convinence for(int j=2.j<=k.j){ // innerlop if(i%j==0)
continue outer. } System.out.print("" i). n . // outputanewline if(n
<10) // after10numbers continue. System.out.println(). n=0. }
System.out.println(). } } 运行结果为： C:\>java PrimeNumber
prime numbers between 100 and 200 101 103 107 109 113 127
131 137 139 149 151 157 163 167 173 179 181 191 193 197 199 该例
通过一个嵌套的for语句来实现。 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