

J2SE(TM)5.0语言特性 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/213/2021_2022_J2SE_TM_5_c104_213768.htm

1. [C风格格式化输出] Java总算也有类似C的printf()风格的方法了，方法名同样叫作printf()，这一特性依赖于前边提到的可变长度参数。举个例子来说，我们现在可以写：`System.out.printf("%s has a value of %d.%n", someString,`

`a)`。怎么样，看上去还不错吧？需要注意的是Java为了支持多平台，新增了%n标示符，作为对\n的补充。有关Java格式化输出的具体语法，请参考java.util.Formatter的API文档。

2.[增强的for循环] 你是否已经厌倦了每次写for循环时都要写上那些机械的代码，尤其当你需要遍历数组或者Collection，如：

（假设在Collection中储存的对象是String类型的）`public void showAll (Collection c) {for (Iterator iter = c.iterator()`

`iter.hasNext().) {System.out.println((String) iter.next()).}}public void showAll (String[] sa) {for (int i = 0. i`

`System.out.println(sa[i]).}}`这样的代码不仅显得臃肿，而且容易出错，我想我们大家在刚开始接触编程时，尤其是C/C

和Java，可能多少都犯过以下类似错误的一种或几种：把for语句的三个表达式顺序弄错；第二个表达式逻辑判断不正确

（漏掉一些、多出一些、甚至死循环）；忘记移动游标；在循环体内不小心改变了游标的位置等等。为什么不能让编译器帮我们处理这些细节呢？在5.0中，我们可以这样写：`public`

`void showAll (Collection c) {for (Object obj : c)`

`{System.out.println((String) obj).}}public void showAll (String[] sa) {for (String str : sa) {System.out.println(str).}}` 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