# 第五章 循环

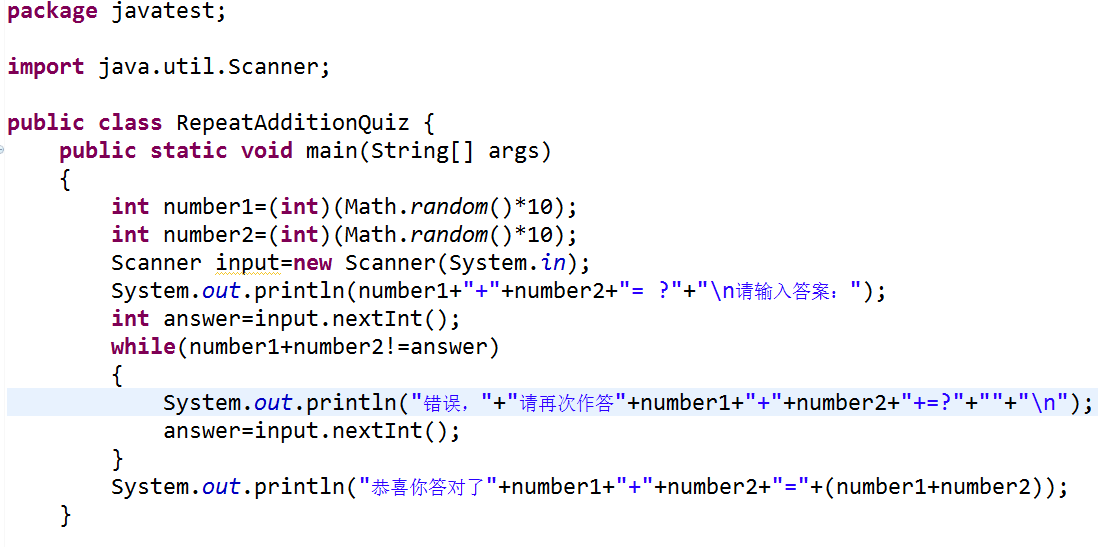
## 5.1引言

要点提示：循环可以用于让一个程序重复的执行语句。

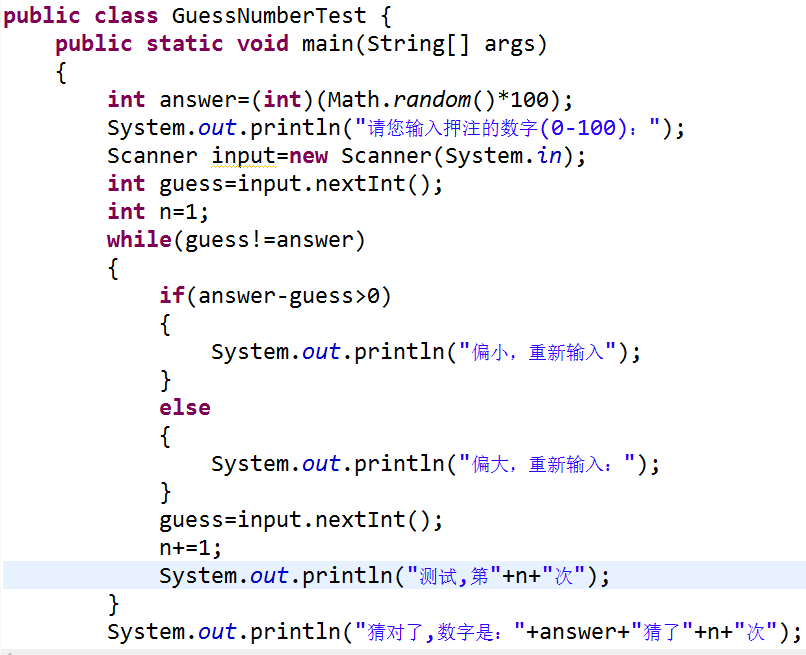
## 5.2 while循环

1.一个常见的错误错误是无限循环，程序会无限循环下去。

2.程序设计，任意两个数的加法，直至给出正确答案为止，循环结束。



### 5.2.1示例学习 猜数字



该例子，自己没有调试课本上两个例子，自己做的答案。其中没加括号，导致随机数一直为0，并且答案中要求包括0和100，



### 5.2.2 循环设计策略

第一步：确定需要重复的语句；

第二步：将这些语句放在一个循环中；

第三步：为循环继续条件编码，并为控制循环添加合适的语句。

### 5.2.3 示例学习：多个减法测试题

**要加强自己解决问题的能力**

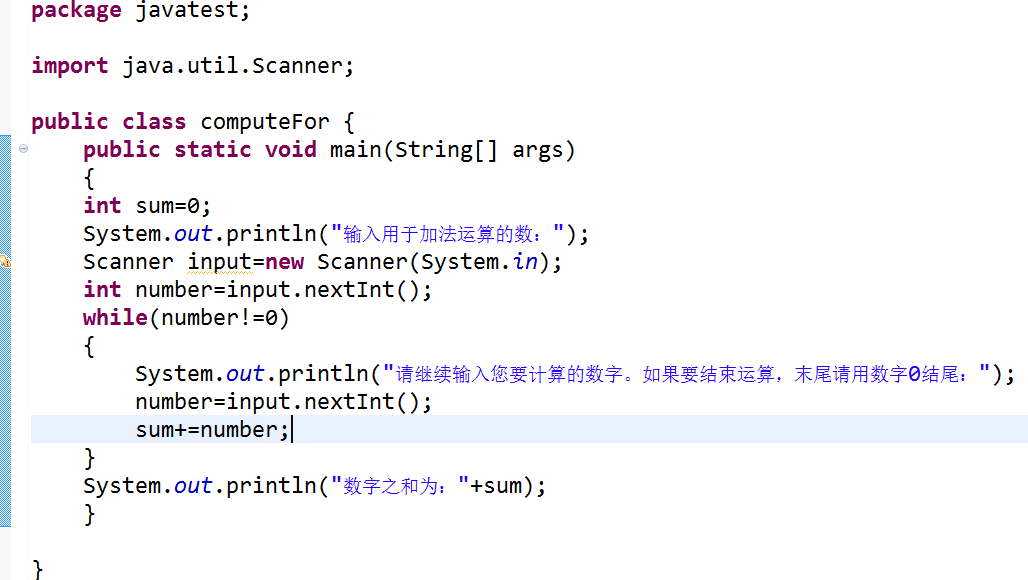


自己的不足有三处：第一是忘记添加计时功能；第二是忘记加括号；第三是没有显示出所有的题目

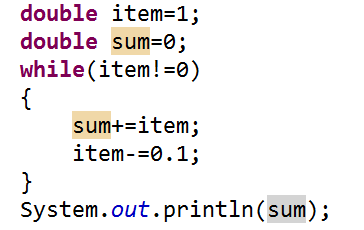
### 5.2.4 使用标记值控制循环

1. 标记值：用于表明循环的结束。

2. 该程序用于计算连续输入的几个数的和，若要结尾则输入0



3. 警告：在循环控制中，不要使用浮点值来比较值是否相等。因为浮点值都是某些值的近似值，使用它们可能导致不精确的循环次数和不精确的结果。例如：



变量item从1开始，每执行一次就减去0.1。当item变为0时循环应该中止。但是，因为浮点数在算数上是近似的，所以不可能确保item会真正成0。从表面上看，这个循环似乎没问题，但实际上是一个无限循环。

### 5.2.5 输入和输出重定向

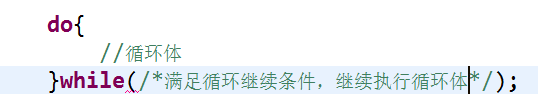
**读取/存储数据，TXT文件**

## 5.3 do-while循环

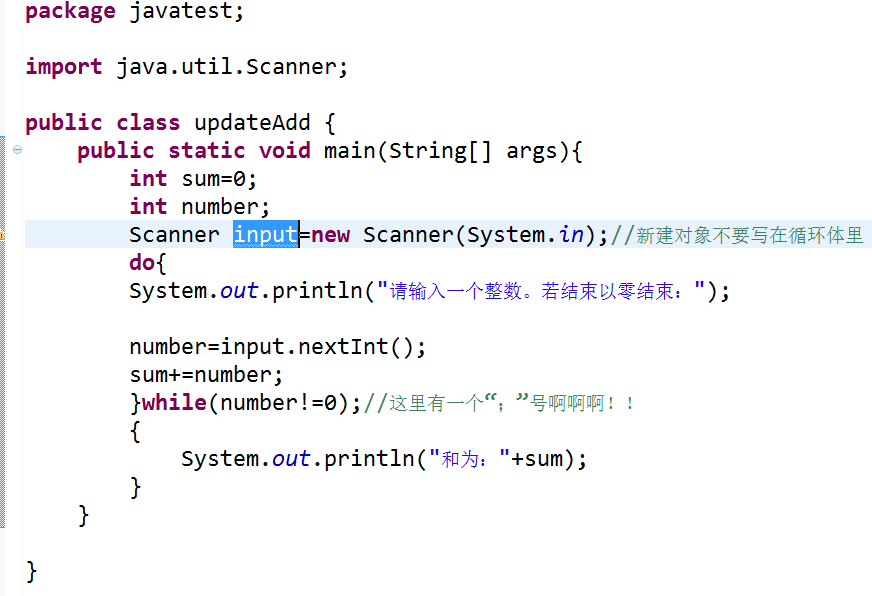
**特别注意（循环继续条件）：满足条件，继续执行。**

1.要点提示：do-while循环和while循环基本一样，不同的是它先执行循环体一次，然后判断（**循环继续条件**），继续执行。

2.do-while循环的语法



3.对计算机个数的和，末尾以零结束。改进版



有两个问题：

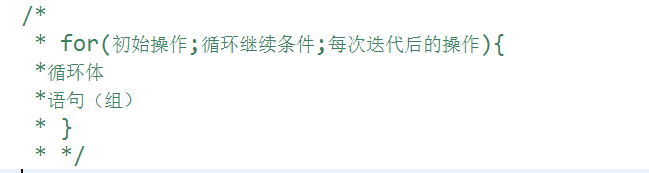
**第一新建对象一开始写在循环体里是什么鬼？**

**第二这里有一个“分号”**

## 5.4 for循环

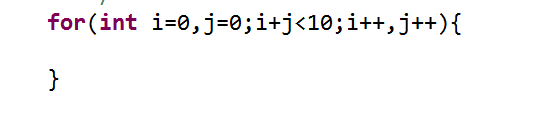
1.要点提示：for循环具有编写循环的简明语法。

2.for循环的语法如下：



3.控制变量：一般情况下，for循环使用一个变量来控制循环体的执行次数，以及什么时候循环终止。这个变量称为“控制变量”。

4.注意：for循环中的初始动作可以是0个或者是多个以逗号隔开的变量声明的语句或赋值表达式。例如：



## 5.5 采用哪种循环

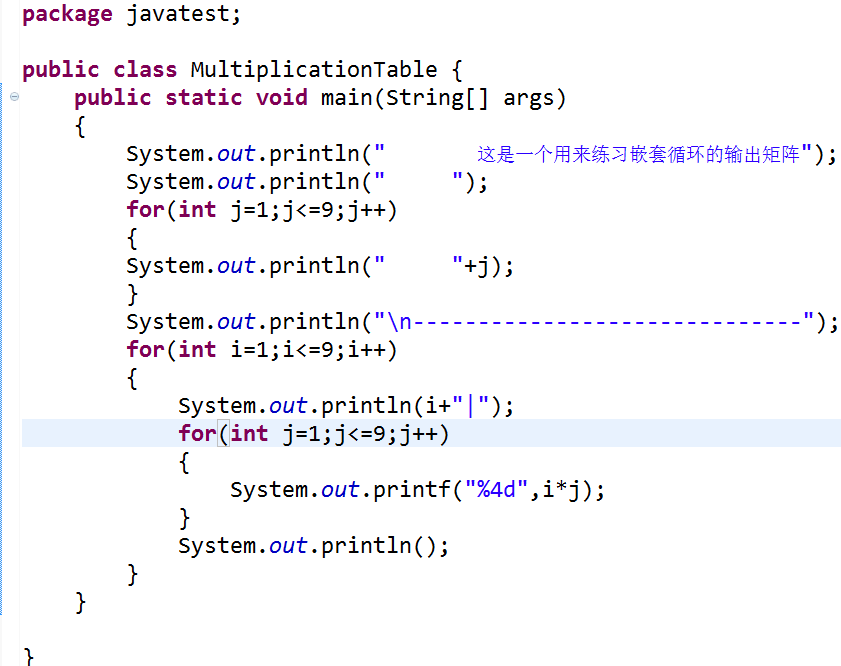
1.要点提示：可以根据哪个更方便，来使用for循环、while循环，或者do-while循环。建议使用自己觉得最自然最方便的最舒服的循环语句，避免“**分号**“错误。

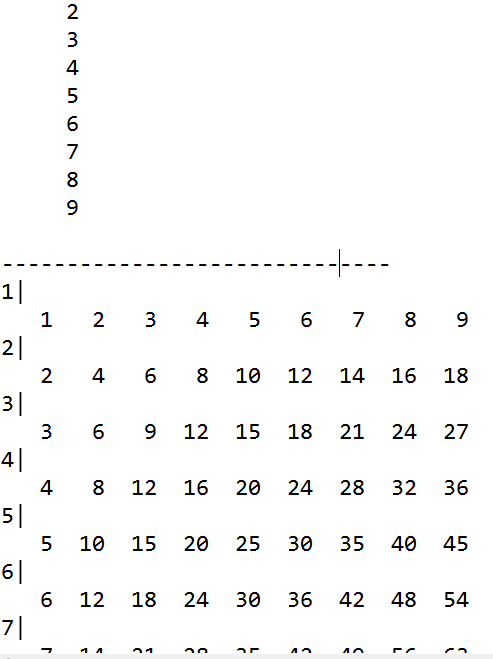
## 5.6 嵌套循环

1.要点提示：一个循环可以嵌套在另一个循环里。

嵌套循环是由一个外层循环和一个或者多个内层循环组成的。每当重复执行一次外层循环时将再次进入内部循环，然后重新开始。

2.循环输出示例





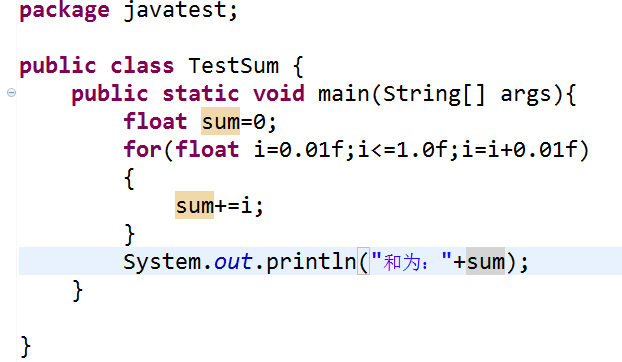
什么鬼。。。。

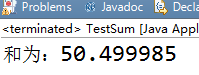
## 5.7 最小化数值错误

1.要点提示：在循环继续条件中使用**浮点数**将导致数值错误。

涉及浮点数的数值误差是不可避免的，因为浮点数在计算机中本身就是近似表示的。

2.以下示例将最小化这种误差。



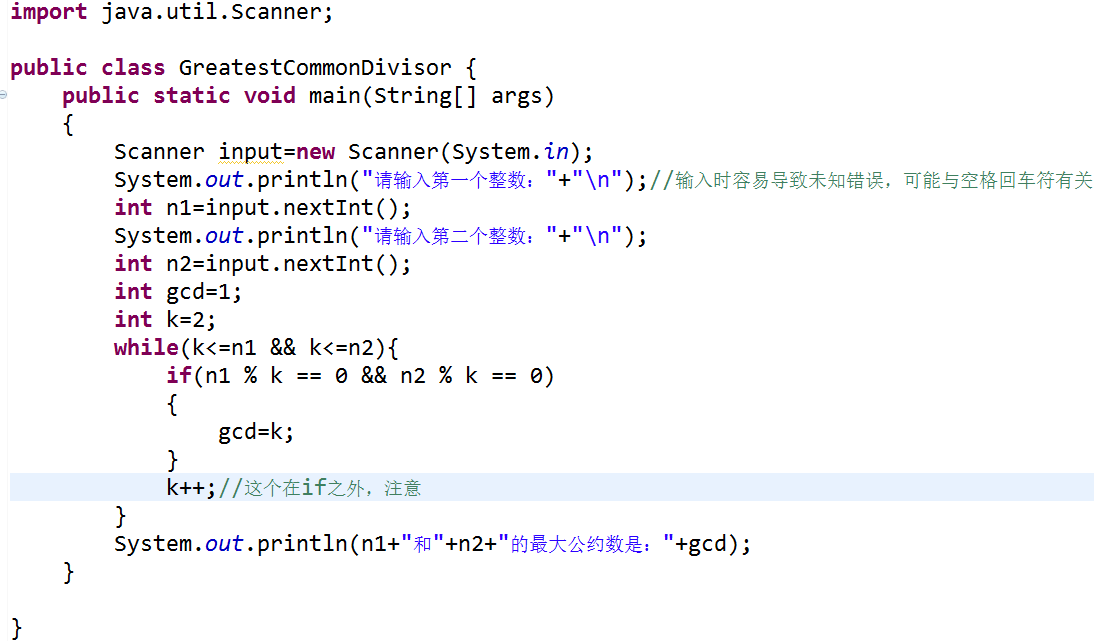


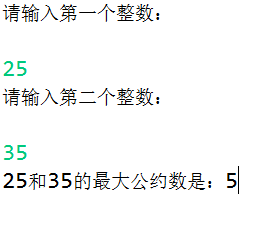
Sum的精确结果应该是50.50，但是实际答案是（上图所示），这个结果是不精确的，因为计算机使用固定的位数表示浮点数，因此，它就不能精确表示某些浮点数。

## 5.8 示例学习

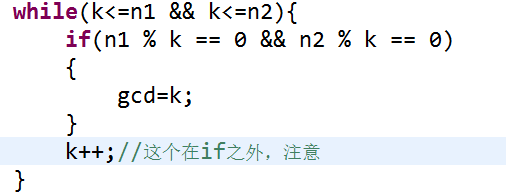
1.要点提示：循环对于编程来说非常关键。编写循环的能力在学习Java编程中是非常重要的。如果能掌握循环编程，对日后的学习大有裨益。

### 5.8.1 求最大公约数



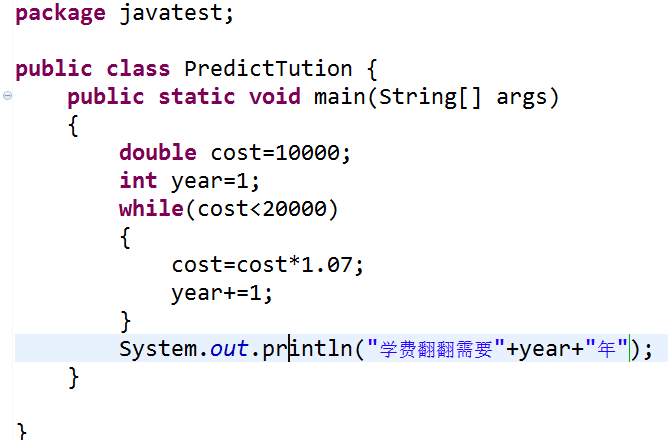


**输入字符读取错误很烦人，可能是回车符造成的，可以尝试在输出语句句尾自动加上回车符**

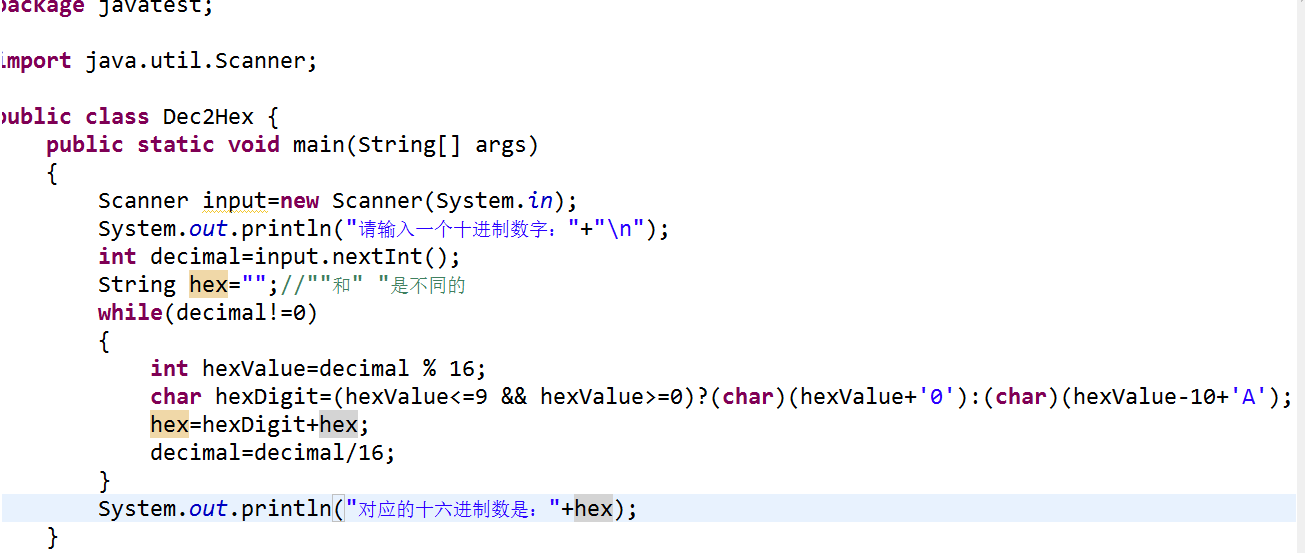


上图是最大公约数算法。

### 5.8.2预测未来学费



### 5.8.3十进制转换成十六进制



这个不是很懂啊。

## 5.9 关键字 break 和 continue（难以理解，本节跳过，只写了一点概念，程序未调试，内容未深究）

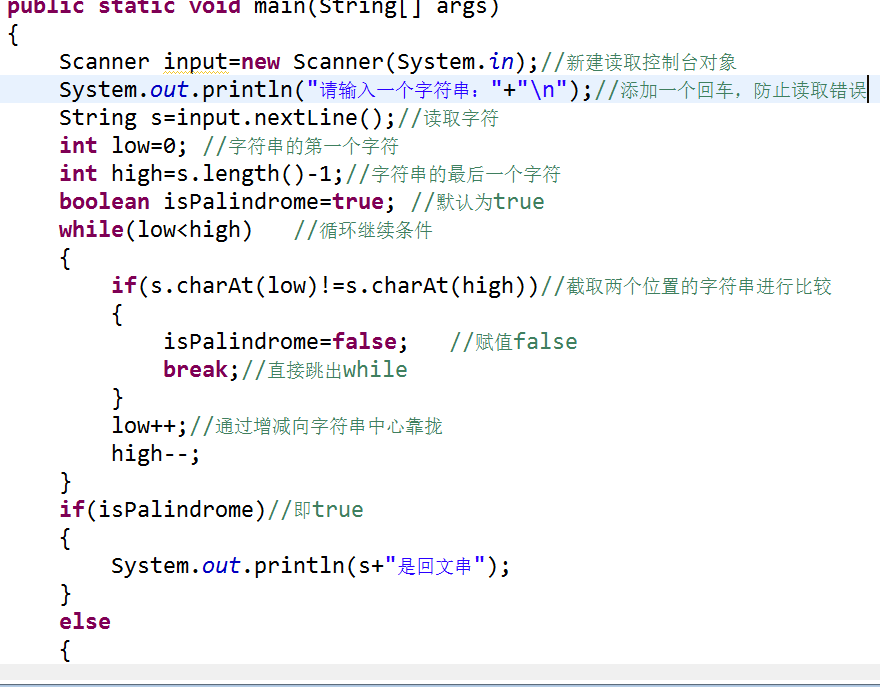
1.要点提示：关键词break和continue在循环中提供了额外的控制。

2.教学注意：关键字break和continue都可以在循环语句中使用，为循环提供额外的控制。在某些情况下，使用break和continue可以简化程序设计。但是，使用过度或者不正确使用都会使程序难以调试。

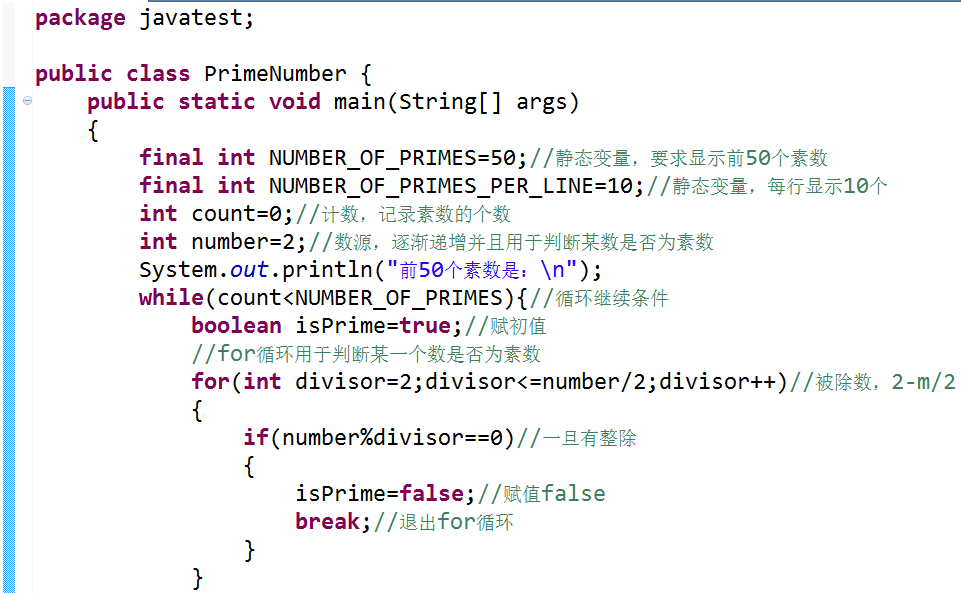
## 5.10 示例学习：判断回文串

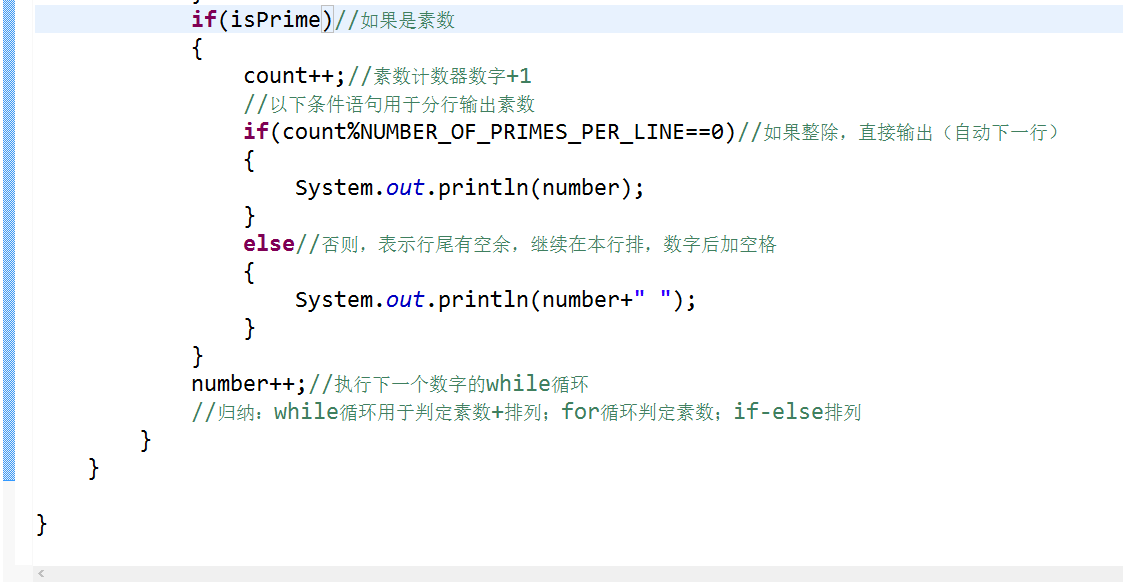
1.回文：如果一个字符串从前往后，以及从后往前都是一样的，那么它就是一个回文。例如：mom dad noon

2.以下程序用于判断一个字符是否回文



## 5.11 示例学习：显示素数





**阵列显示素数有问题，未解决。**