**实验1 回文数**

**1.相关知识点**

1)分支语句

if-else语句是Java中的一条语句，由关键字if、else和两个复合语句（分别称为if分支操作和else分支操作）按一定格式构成，if-else语句的格式如下：

if(表达式){

若干语句（if分支操作部分）

}

else {

若干语句（else分支操作部分）

}

一条if-else语句的作用是根据一个条件选择两个分支操作中的一个，执行法则是：if 后面()内的表达式的值为true,则执行紧跟着的复合语句，即执行if分支操作；如果表达式的值为false，则执行else后面的复合语句，即执行else分支操作。

if-else if-dse语句称为多条件分支语句，其作用是根据多条件来选择某一操作。语句的格式如下：

if(表达式1) {

若干语句

}

else if {表达式2) {

若干语句

}

else {

若干语句

}

2)将字符串转化为数值

执行 “int m=lnteger.parselnt(“6789”);” 可以将数字型字符串，如 “6789”、“123”，转化为int型数据。

**2.实验目的**

本实验的目的是让学生掌握使用if-else if-else多分支语句。

**3.实验要求**

**编写一个Java应用程序程序。用户从键盘输入一个1〜9999之间的数，程序将判断这个数是几位数，并判断这个数是否是回文数。回文数是指将该数含有的数字逆序排列后得到的数和原数相同，例如12121、3223都是回文数。**

**4. 程序模板**

请按模板要求，将【代码】替换为Java程序代码。

**Number.java**

import java.util.Scanner;

public class Number {

public static void main(String args[]) {

int number=0,d5,d4,d3,d2,d1;

Scanner reader = new Scanner(System.in);

System.out.println("输入一个1至99999之间的数");

number = reader.nextInt();

if(【代码1】) //判断number在1至99999之间的条件

{ 【代码2】 //计算number的最高位（万位）d5

【代码3】 //计算number的千位d4

【代码4】 //计算number的百位d3

d2=number%100/10;

d1=number%10;

if(【代码5】) //判断number是5位数的条件

{ System.out.println(number+"是5位数");

if(【代码6】) //判断number是回文数的条件

{ System.out.println(number+"是回文数");

}

else

{ System.out.println(number+"不是回文数");

}

}

else if(【代码7】) //判断number是4位数的条件

{ System.out.println(number+"是4位数");

if(【代码8】) //判断number是回文数的条件码

{ System.out.println(number+"是回文数");

}

else

{ System.out.println(number+"不是回文数");

}

}

else if(【代码9】) //判断number是3位数的条件

{ System.out.println(number+"是3位数");

if(【代码10】) //判断number是回文数的条件

{ System.out.println(number+"是回文数");

}

else

{ System.out.println(number+"不是回文数");

}

}

else if(d2!=0)

{ System.out.println(number+"是2位数");

if(d1==d2)

{ System.out.println(number+"是回文数");

}

else

{ System.out.println(number+"不是回文数");

}

}

else if(d1!=0)

{ System.out.println(number+"是1位数");

System.out.println(number+"是回文数");

}

}

else

{ System.out.printf("\n%d不在1至99999之间",number);

}

}

}

import java.util.Scanner;  
public class Number {  
 public static void main(String args[]) {  
 int number=0,d5,d4,d3,d2,d1;  
 Scanner reader = new Scanner(System.*in*);  
 System.*out*.println("输入一个1至99999之间的数");  
 number = reader.nextInt();  
 /\*if(【代码1】) //判断number在1至99999之间的条件  
 { 【代码2】 //计算number的最高位（万位）d5  
 【代码3】 //计算number的千位d4  
 【代码4】 //计算number的百位d3  
   
 \*/  
 if(number>=1&&number<=99999) //判断number在1至99999之间的条件  
 {   
 d5=number/10000;//计算number的最高位（万位）d5  
 d4=number/1000%10; //计算number的千位d4  
 d3=number/100%10; //计算number的百位d3  
  
 d2=number%100/10;  
 d1=number%10;  
 //if(【代码5】) //判断number是5位数的条件  
 if(d5>=1&&d5<=9)  
 { System.*out*.println(number+"是5位数");  
 if(d5==d1&&d2==d4) //判断number是回文数的条件  
 { System.*out*.println(number+"是回文数");  
 }  
 else  
 { System.*out*.println(number+"不是回文数");  
 }  
 }  
 else if(d4>=1&&d4<=9) //判断number是4位数的条件  
 { System.*out*.println(number+"是4位数");  
 if(d4==d1&&d2==d3) //判断number是回文数的条件码  
 { System.*out*.println(number+"是回文数");  
 }  
 else  
 { System.*out*.println(number+"不是回文数");  
 }  
 }  
 else if(d3>=1&&d1<=9) //判断number是3位数的条件  
 { System.*out*.println(number+"是3位数");  
 if(d3==d1) //判断number是回文数的条件  
 { System.*out*.println(number+"是回文数");  
 }  
 else  
 { System.*out*.println(number+"不是回文数");  
 }  
 }  
 else if(d2!=0)  
 { System.*out*.println(number+"是2位数");  
 if(d1==d2)  
 { System.*out*.println(number+"是回文数");  
 }  
 else  
 { System.*out*.println(number+"不是回文数");  
 }  
 }  
 else if(d1!=0)  
 { System.*out*.println(number+"是1位数");  
 System.*out*.println(number+"是回文数");  
 }  
 }  
 else  
 { System.*out*.printf("\n%d不在1至99999之间",number);  
 }  
 }  
}

**5. 实验指导与检查**

两个int型数据做除法运算时，运算的结果是int型，因此，6234/1000的结果刚好是6234的最高位上的数字6。

为了计算出56321中千位上的数字6,首先计算56321%10000得到6321，然后 6321/1000的结果是6。

**6. 实验后的练习**

(1) 程序运行时，用户从键盘输入2332,程序提示怎样的信息。

(2) 程序运行时，用户从键盘输入654321，程序提示怎样的信息。

(3) 程序运行时，用户从键盘输入33321，程序提示怎样的信息。

**7. 填写实验报告**

**实验2猜数字游戏**

**1.相关知识点**

循环是控制结构语句中的最重要的语句之一，循环语句是根据条件反复执行同一代码块。循环语句有下列三种：

1) while 循环

while语句的一般格式：

while (表达式）{

若干语句

}

while语句的执行规则如下：

(1) 计算表达式的值，如果该值是true时，就进行(2),否则进(3)。

(2) 执行循环体，再进(1)。

(3) 结束while语句的执行。

2) for循环

for语句的一般格式：

for (表达式1;表达式2;表达式3) {

若干语句

}

for语句的执行规则如下：

(1) 计算“表达式1”，完成必要的初始化工作。

(2) 判断“表达式2”的值，若“表达式2”的值为true,则进行(3)，否则进行(4)。

(3) 执行循环体，然后计算“表达式3”，以便改变循环条件，进行(2)。

(4) 结束for语句的执行。

3) 将字符串转化为数值

执行 **“int m=lnteger.parselnt("6789”);”可以将数字型字符串**，如 “6789”、“123” 转化为int型数据.

4) Scanner类创建一个对象

**Scanner reader=new Scanner(System.in);**

**reader对象调用方法nextInt()方法读取用户输入的整数。**

**2.实验目的**

本实验的目的是让学生使用if-else分支和while循环语句解决问题。

**3，实验要求**

**编写一个Java应用程序，实现下列功能：**

**•程序随机分配给客户一个1〜100之间的整数。**

**•用户输入自己的猜测。**

**•程序返回提示信息，提示信息分别是“猜大了”、“猜小了”或“猜对了”。**

**•用户可根据提示信息再次输人猜测，直到提示信息是“猜对了”。**

**4. 程序模板**

请按模板要求，将【代码】 替换为Java程序代码。

**GuessNumber.java**

import java.util.Scanner;

import java.util.Random;

public class GuessNumber {

public static void main (String args[]) {

Scanner reader = new Scanner(System.in);

Random random = new Random();

System.out.println("给你一个1至100之间的整数,请猜测这个数");

int realNumber = random.nextInt(100)+1; //random.nextInt(100)是[0,100)中的随机整数

int yourGuess = 0;

System.out.print("输入您的猜测:");

yourGuess = reader.nextInt();

while(【代码1】) //循环条件

{

if(【代码2】) //猜大了的条件代码

{

System.out.print("猜大了,再输入你的猜测:");

yourGuess = reader.nextInt();

}

else if(【代码3】) //猜小了的条件代码

{

System.out.print("猜小了,再输入你的猜测:");

yourGuess = reader.nextInt();

}

}

System.out.println("猜对了!");

}

}

**import java.util.Scanner;**

**import java.util.Random;**

**public class GuessNumber {**

**public static void main (String args[]) {**

**Scanner reader = new Scanner(System.in);**

**Random random = new Random();**

**System.out.println("给你一个1至100之间的整数,请猜测这个数");**

**int realNumber = random.nextInt(100)+1; //random.nextInt(100)是[0,100)中的随机整数**

**int yourGuess = 0;**

**System.out.print("输入您的猜测:");**

**yourGuess = reader.nextInt();**

**while(yourGuess != realNumber) //循环条件**

**{**

**if(yourGuess>realNumber) //猜大了的条件代码**

**{**

**System.out.print("猜大了,再输入你的猜测:");**

**yourGuess = reader.nextInt();**

**}**

**else if(yourGuess<realNumber) //猜小了的条件代码**

**{**

**System.out.print("猜小了,再输入你的猜测:");**

**yourGuess = reader.nextInt();**

**}**

**}**

**System.out.println("猜对了!");**

**}**

**}**

**5. 实验指导**

人们经常使用while循环“强迫”程序重复执行一段代码，【代码1】必须是值为boolean型数据的表达式，只要【代码1】的值为true就是让用户继续输入猜测。

只要用户的输入能使得循环语句结束，就说明用户已经猜对了。

**6. 实验后的练习**

(1) 用“yourGuess>realNumber”替换【代码1】可以吗？

(2) 语句“System.out.printl(“猜对了!”);”为何要放在while循环语句之后？放在while 语句的循环体中合理吗？

**7.填写实验报告**