程序说明

程序功能与网络学堂上要求一致。为保证程序运行连续性，rand函数与testCircle方法的调用通过控制台输入触发，若输入rand则调用rand()方法，输入circle则调用testCircle方法。

程序代码

EX1的主程序如下：

```
//引入Math类
import java.lang.Math;
public class EX1 {

    //添加一个新函数cal，形式参数为整形n
    public static int cal(int n) {
        //函数作用为返回1-n的和
        int sum = 0;
        for (int i = 1; i <= n; i++) {
            sum += i;
        }
        return sum;
    }
    //新建一个函数rand，作用为随机生成100个0-99的整数，打印其中的最大值和最小值，并返回其中大于70的数的个数
    public static int rand() {
        int max = 0;
        int min = 100;
        int count = 0;
        for (int i = 0; i < 100; i++) {
            int temp = (int) (Math.random() * 100);
            if (temp > max) {
                max = temp;
            }
            if (temp < min) {
                min = temp;
            }
            if (temp > 70) {
                count++;
            }
        }
        System.out.println("the max is " + max);
        System.out.println("the min is " + min);
        return count;
    }
    //新建一个函数testCircle,作用为测试Circle类
    public static void testCircle() {

        //创建一个Circle对象c1，调用构造函数Circle()
        Circle c1 = new Circle();
        //创建一个Circle对象c2，调用构造函数Circle(double x, double y, double r)
```

```
        Circle c2 = new Circle(1, 2, 3);
        //创建一个Circle对象c3，调用构造函数Circle(double r)
        Circle c3 = new Circle(4);
        //创建一个Circle对象c4，调用构造函数Circle(double x, double y)
        Circle c4 = new Circle(5, 6);
        //调用成员方法setX，setY，setR，分别用于设置圆心的x、y坐标、圆半径r；
        c1.setX(1);
        c1.setY(2);
        c1.setR(3);
        //调用成员方法getX，getY，getR，分别用于获取圆心的x、y坐标、圆半径r；
        System.out.println("c1's x is " + c1.getX());
        System.out.println("c1's y is " + c1.getY());
        System.out.println("c1's r is " + c1.getR());
        //调用成员方法calArea，calPerimeter，分别用于获取圆的面积、周长；
        System.out.println("c1's area is " + c1.calArea());
        System.out.println("c1's perimeter is " + c1.calPerimeter());
        //调用成员方法display，用于输出圆的信息；
        System.out.println("c1's information is ");
        c1.display();
        System.out.println("----------------------");
        System.out.println("c2's information is ");
        c2.display();
        System.out.println("----------------------");
        System.out.println("c3's information is ");
        c3.display();
        System.out.println("----------------------");
        System.out.println("c4's information is ");
        c4.display();
        System.out.println("----------------------");
        //调用静态方法printNum，用于获取创建的圆的个数；
        System.out.println("the number of circles is " + Circle.printNum());
    }


    public static void main(String[] args) {
        //通过控制台从键盘获取输入
        System.out.println("please input a positive integer n");
        while (true) {
            java.util.Scanner input = new java.util.Scanner(System.in);
            //检查输入是否为"exit"
            while (input.hasNext()) {
                if (input.hasNext("exit")) {
                    System.out.println("system will exit now");
                    System.exit(0);
                }
                //若输入为rand调用rand函数
                if(input.hasNext("rand")){
                    int num=rand();
                    System.out.println("the number of integers greater than 70
is "+num);
                    System.out.println("please input a positive integer n");
                    break;
```

```
                }
                //若输入为circle调用testCircle函数
                if(input.hasNext("circle")){
                    testCircle();
                    System.out.println("please input a positive integer n");
                    break;
                }
                //检查输入是否为整数
                if (input.hasNextInt()) {
                    int n = input.nextInt();
                    //检查输入是否为正整数
                    if (n > 1) {
                        System.out.println("the sum of 1 to " + n + " is " +
cal(n));
                    } else {
                        System.out.println("please input a positive integer
n");
                    }
                } else {
                    //检查输入是否为正浮点数，若是则强制转换为整形并输出提示信息，否则提示
重新输入
                    if (input.hasNextDouble()) {
                        double n = input.nextDouble();
                        if (n > 1) {
                            System.out.println("you input a float, system will
convert it to int " + (int) n);
                            System.out.println("the sum of 1 to " + (int) n +
" is " + cal((int) n));
                        } else {
                            System.out.println("please input a positive
integer n");
                        }
                    } else {
                        System.out.println("please input a positive integer
n");
                    }

                }
                break;
            }
        }
    }
}
```

其中circle类代码如下：

```
class Circle {
    //包含成员变量：常量PI，私有变量x、y、r，分别为圆心的x、y坐标、圆半径r；
    private final double PI = 3.1415926;
    private double x;
    private double y;
```

```
    private double r;
    //包含静态变量num，用于记录创建的圆的个数；
    private static int num = 0;
    //包含静态方法printNum，用于获取创建的圆的个数；
    public static int printNum() {
        return num;
    }

    //包含构造函数Circle()
    public Circle() {
        this.x = 0;
        this.y = 0;
        this.r = 1;
        num++;
    }
    //包含构造函数Circle(double x, double y, double r)
    public Circle(double x, double y, double r) {
        this.x = x;
        this.y = y;
        this.r = r;
        num++;
    }
    public Circle(double r) {
        this.x = 0;
        this.y = 0;
        this.r = r;
        num++;
    }
    public Circle(double x, double y) {
        this.x = x;
        this.y = y;
        this.r = 1;
        num++;
    }
    //包含成员方法setX，setY，setR，分别用于设置圆心的x、y坐标、圆半径r；
    public void setX(double x) {
        this.x = x;
    }
    public void setY(double y) {
        this.y=y;
    }
    public void setR(double r) {
        this.r=r;
    }
    //包含成员方法getX，getY，getR，分别用于获取圆心的x、y坐标、圆半径r；
    public double getX() {
        return this.x;
    }
    public double getY() {
        return this.y;
    }
    public double getR() {
```

```
        return this.r;
    }
    //包含成员方法calArea，calPerimeter，分别用于获取圆的面积、周长；
    public double calArea() {
        return PI * r * r;
    }
    public double calPerimeter() {
        return 2 * PI * r;
    }
    //包含成员方法display，用于输出圆的信息；
    public void display() {
        System.out.println("the circle's x is " + x);
        System.out.println("the circle's y is " + y);
        System.out.println("the circle's r is " + r);
        System.out.println("the circle's area is " + calArea());
        System.out.println("the circle's perimeter is " + calPerimeter());
    }
}

```

程序运行后控制台输出如下：

```
please input an order
rand
the max is 99
the min is 2
the number of integers greater than 70 is 32
please input an order
19
the sum of 1 to 19 is 190
18.6
you input a float, system will convert it to int 18
the sum of 1 to 18 is 171
dfal
please input a positive integer n
-123
please input a positive integer n
circle
c1's x is 1.0
c1's y is 2.0
c1's r is 3.0
c1's area is 28.274333400000003
c1's perimeter is 18.849555600000002
c1's information is
the circle's x is 1.0
the circle's y is 2.0
the circle's r is 3.0
the circle's area is 28.274333400000003
the circle's perimeter is 18.849555600000002
----------------------
c2's information is

```

```
the circle's x is 1.0
the circle's y is 2.0
the circle's r is 3.0
the circle's area is 28.274333400000003
the circle's perimeter is 18.849555600000002
----------------------
c3's information is
the circle's x is 0.0
the circle's y is 0.0
the circle's r is 4.0
the circle's area is 50.2654816
the circle's perimeter is 25.1327408
----------------------
c4's information is
the circle's x is 5.0
the circle's y is 6.0
the circle's r is 1.0
the circle's area is 3.1415926
the circle's perimeter is 6.2831852
----------------------
the number of circles is 4
please input an order
exit
system will exit now

Process finished with exit code 0
```