

实验 04：流程控制

一、实验目的

1. 掌握变量的声明、赋值及命名规范，理解变量的作用。
2. 熟悉 Java 标识符的命名规则和关键字的使用。
3. 掌握各种运算符的使用，包括算术运算符、赋值运算符、关系运算符和逻辑运算符。
4. 理解选择结构（`if`、`if-else`、`switch`）的使用场景及实现方法。
5. 掌握循环结构（`while`、`do-while`、`for`、增强 `for` 循环）的使用，以及 `break` 和 `continue` 语句的作用。

二、实验学时

2 学时

三、实验类型

验证性

四、实验需求

1、硬件

每人配备计算机 1 台，建议优先使用个人计算机开展实验。

2、软件

Windows 10/11 操作系统

IntelliJ IDEA 版本号：2024.03

安装 IntelliJ IDEA，以及 Java 运行所需要的相关基础环境。

3、网络

本地主机能够访问互联网和实验中心网络。

4、工具

无。

五、实验任务

5.1 变量与运算符：实现患者基本信息的存储和计算。

5.2 选择结构：根据患者的体温判断是否需要隔离。

5.3 循环结构：使用 `while` 循环模拟患者排队挂号。

5.4 增强 `for` 循环：遍历患者列表并输出信息。

5.5 综合任务：结合选择结构和循环结构，实现一个简单的患者分诊系统。

六、实验内容及步骤

6.1、任务 1：变量与运算符

步骤：

(1) 声明并初始化患者的姓名、年龄、体温等变量。

(2) 使用算术运算符计算患者的平均体温，运行结果如图 1 所示。

示例代码：

```
1 public class PatientInfo {
2     public static void main(String[] args) {
3         // 患者基本信息
4         String patientName = "张三";
5         int patientAge = 35;
6         double temperature1 = 36.5;
7         double temperature2 = 37.2;
8         double temperature3 = 36.8;
9
10        // 计算平均体温
11        double averageTemperature = (temperature1 + temperature2 + tem
perature3) / 3;
12
13        // 输出患者信息
14        System.out.println("患者姓名: " + patientName);
15        System.out.println("患者年龄: " + patientAge + "岁");
16        System.out.println("平均体温: " + averageTemperature + "°C");
17
18        //保留两位小数的例子
19        //一
20        double number = 3.1415926;
21
22        DecimalFormat df = new DecimalFormat("#.00");
23        String formattedNumber = df.format(number);
24        double result = Double.parseDouble(formattedNumber);
25
26        System.out.println(result); // 输出: 3.14
27
28        //二
29        double number2 = 3.1415926;
30
31        String formattedNumber2 = String.format("%.2f", number2);
32        double result2 = Double.parseDouble(formattedNumber2);
33
34        System.out.println(result2); // 输出: 3.14
35
36        //三
37        double number3 = 3.1415926;
38
39        double result3 = Math.round(number3 * 100.0) / 100.0;
40
41        System.out.println(result3); // 输出: 3.14
```

```

42
43     //四
44     double number4 = 3.1415926;
45
46     BigDecimal bd = new BigDecimal(number4);
47     bd = bd.setScale(2, RoundingMode.HALF_UP);
48     double result4 = bd.doubleValue();
49
50     System.out.println(result4); // 输出: 3.14
51 }
52 }

```

The screenshot shows the IntelliJ IDEA IDE with the file `PatientInfo.java` open. The code in the editor is as follows:

```

19 // 输出患者信息
20 System.out.println("患者姓名: " + patientName);
21 System.out.println("患者年龄: " + patientAge + "岁");
22 System.out.println("平均体温: " + averageTemperature + "°C");
23
24 // 保留两位小数的例子
25 // --
26 double number = 3.1415926;
27
28 DecimalFormat df = new DecimalFormat( pattern: "#.00");
29 String formattedNumber = df.format(number);
30 double result = Double.parseDouble(formattedNumber);
31
32 System.out.println(result); // 输出: 3.14
33

```

The Run window at the bottom shows the execution of the program. The command executed is `C:\jdk1.8.0_291\bin\java.exe ...`. The output is:

```

患者姓名: 张三
患者年龄: 35岁
平均体温: 36.833333333333336°C
3.14
3.14
3.14
3.14

```

The process finished with exit code 0.

图1 任务1：变量与运算符的运行结果

6.2、任务 2：选择结构

步骤：

(1) 根据患者的体温判断是否需要隔离（体温 $\geq 37.3^{\circ}\text{C}$ 需要隔离），运行结果如图 2 所示。

示例代码：

Java

```
1 import java.util.Scanner;
2
3 public class TemperatureCheck {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         // 输入患者体温
8         System.out.print("请输入患者体温: ");
9         double temperature = scanner.nextDouble();
10
11        // 判断是否需要隔离
12        if (temperature >= 37.3) {
13            System.out.println("体温异常, 需要隔离!");
14        } else {
15            System.out.println("体温正常, 无需隔离。");
16        }
17    }
18 }
```

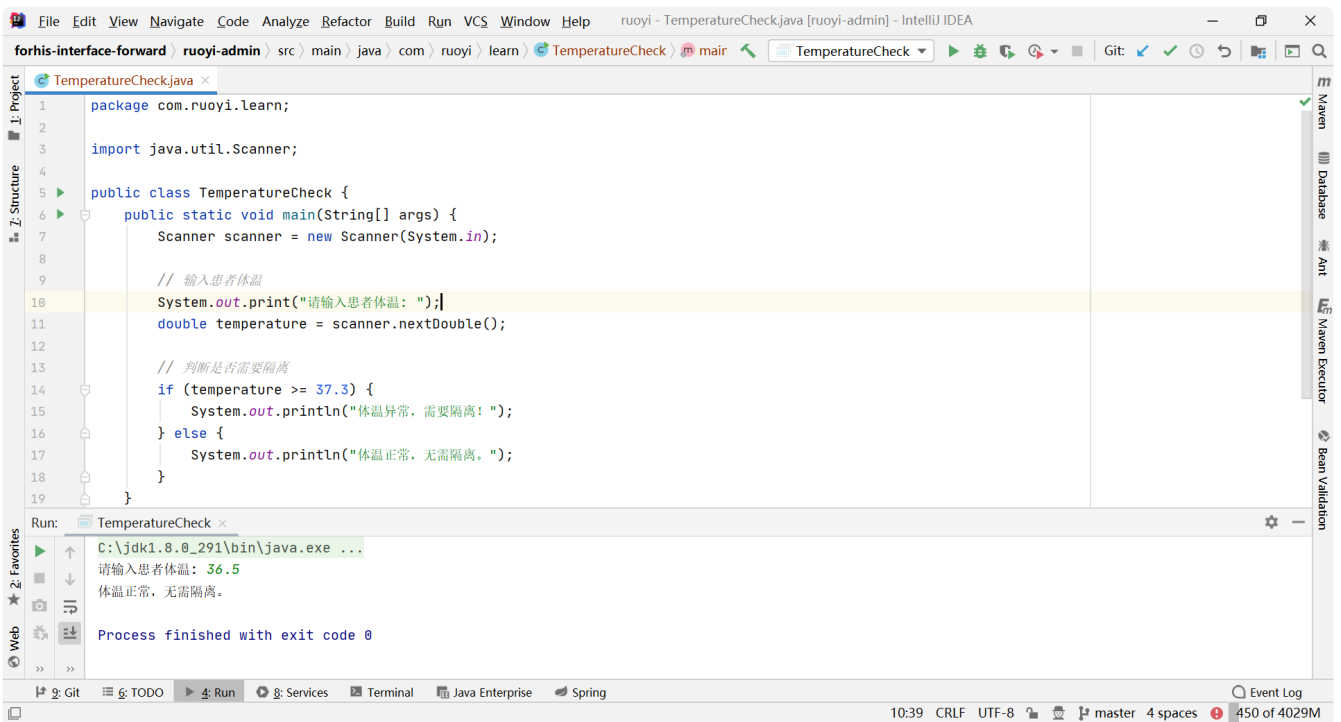


图2 任务2的运行结果

6.3、任务 3：循环结构

步骤：

(1) 使用 `while` 循环模拟患者排队挂号，直到挂满 10 个号为止，运行结果如图 3 所示。

示例代码：

Java

```
1 public class PatientQueue {
2     public static void main(String[] args) {
3         int queueNumber = 1;
4         int maxQueue = 10;
5
6         // 模拟患者排队挂号
7         while (queueNumber <= maxQueue) {
8             System.out.println("当前挂号患者编号: " + queueNumber);
9             queueNumber++;
10        }
11        System.out.println("今日挂号已满, 请明日再来。");
12    }
13 }
```

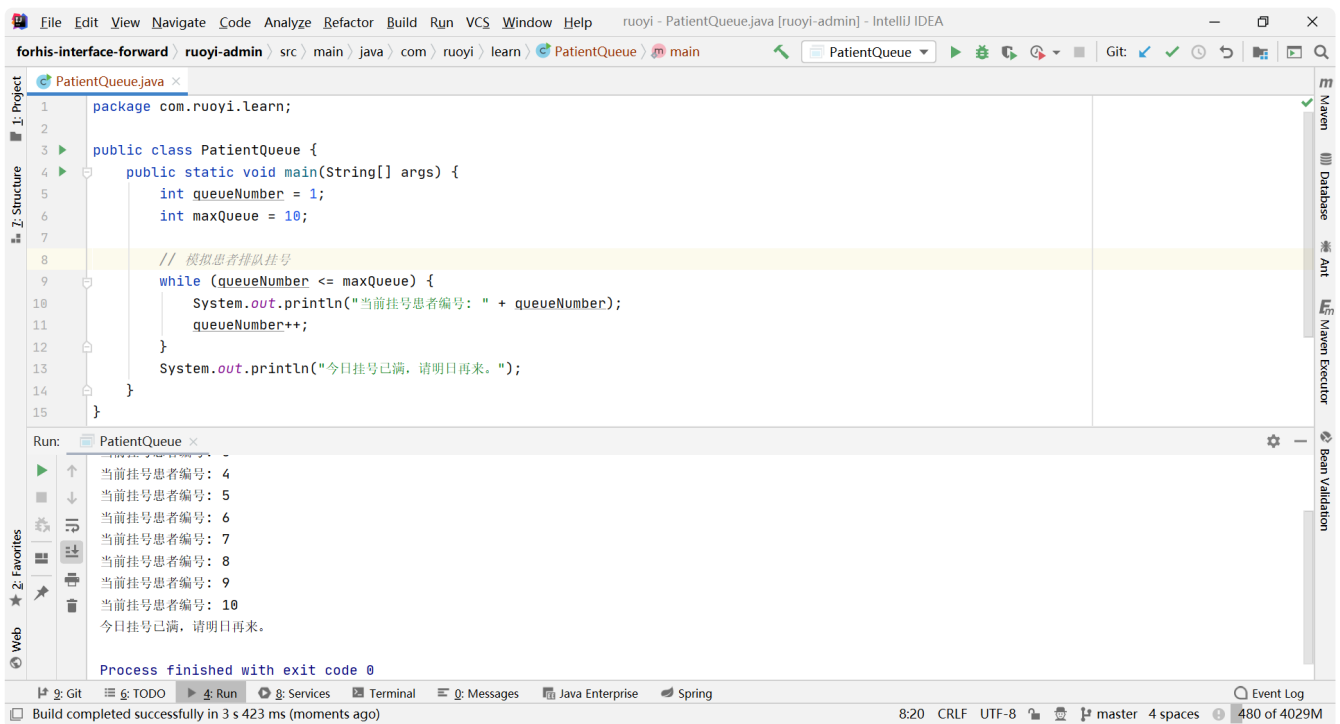


图3 任务三的运行结果

6.4、任务 4：增强 for 循环

步骤：

(1) 使用增强 `for` 循环遍历患者列表并输出信息。

示例代码：

Java

```
1 public class PatientList {
2     public static void main(String[] args) {
3         // 患者姓名列表
4         String[] patients = {"王会青", "李振东", "王风学", "周辰承"};
5
6         // 遍历并输出患者信息
7         System.out.println("今日就诊患者列表:");
8         for (String patient : patients) {
9             System.out.println("患者: " + patient);
10        }
11    }
12 }
```

6.5、任务 5：综合任务 - 患者分诊系统

步骤：

- (1) 根据患者的症状（发热、咳嗽、头痛）进行分诊。
- (2) 使用 `switch` 结构分配科室（发热 → 发热门诊，咳嗽 → 呼吸科，头痛 → 神经科）。

示例代码：

Java

```
1 import java.util.Scanner;
2
3 public class PatientTriage {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         // 输入患者症状
8         System.out.println("请输入患者症状（发热、咳嗽、头痛）：");
9         String symptom = scanner.next();
10
11        // 根据症状分诊
12        switch (symptom) {
13            case "发热":
14                System.out.println("请前往发热门诊就诊。");
15                break;
16            case "咳嗽":
17                System.out.println("请前往呼吸科就诊。");
18                break;
19            case "头痛":
20                System.out.println("请前往神经科就诊。");
21                break;
22            default:
23                System.out.println("症状不明，请前往急诊科就诊。");
24        }
25    }
26 }
```

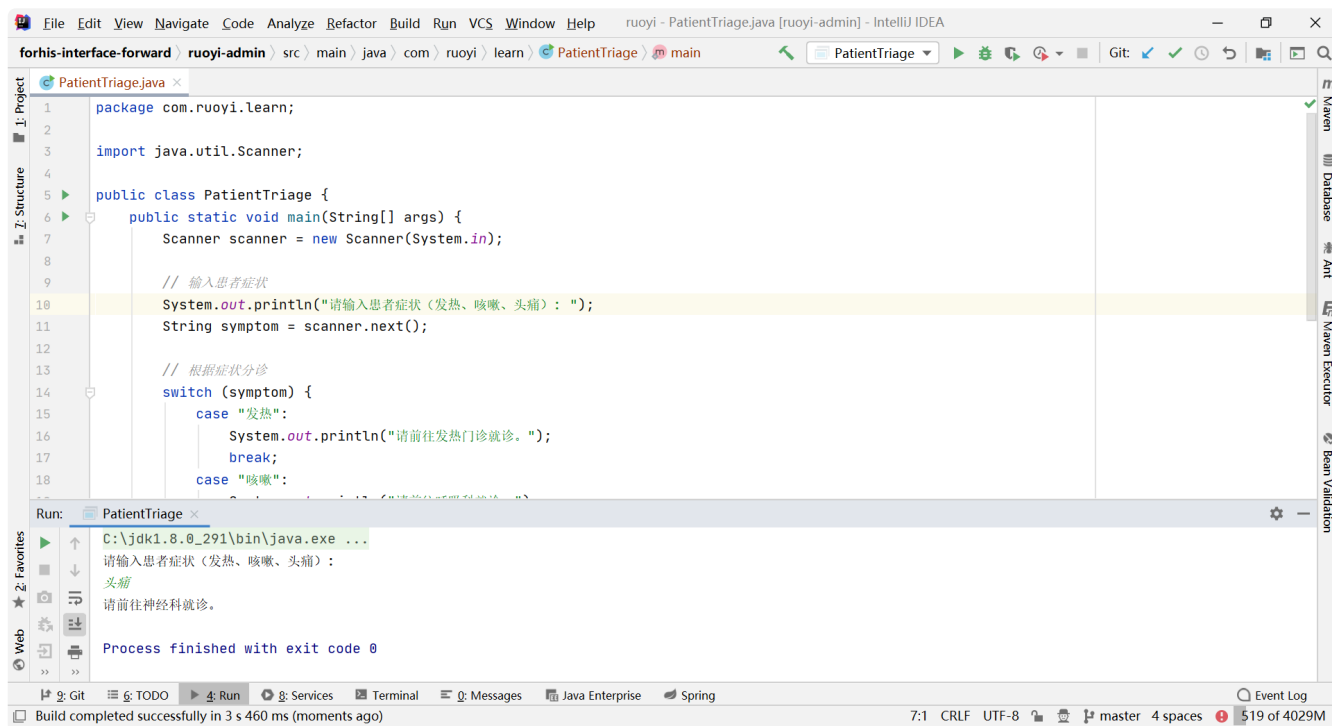


图4 任务5的运行结果

七、实验考核

本实验考核采用【实验随堂查】方式开展。

每个实验完成后，在实验课上通过现场演示的方式向实验指导教师进行汇报，并完成现场问答交流。

每个实验考核满分 100 分，其中实验成果汇报 60 分，现场提问交流 40 分。

实验考核流程：

- (1) 学生演示汇报实验内容的完成情况，实验指导老师现场打分。
- (2) 指导老师结合实验内容进行提问，每位学生提问 2-3 个问题，根据回答的情况现场打分。
- (3) 实验考核结束后，进行公布成绩。