

CHANGZHOU INSTITUTE OF TECHNOLOGY

**课程作业**

|  |
| --- |
| **课程名：** 面向对象程序设计 |
| 二级学院： 计算机信息工程学院 |
| 班 级： |
| 班 级： 23软二 |
| 学 号： 23030433 |
| 姓 名： 赵新航 |
|  |
|  |

班级： 23软二 学号：23030433 姓名：赵新航

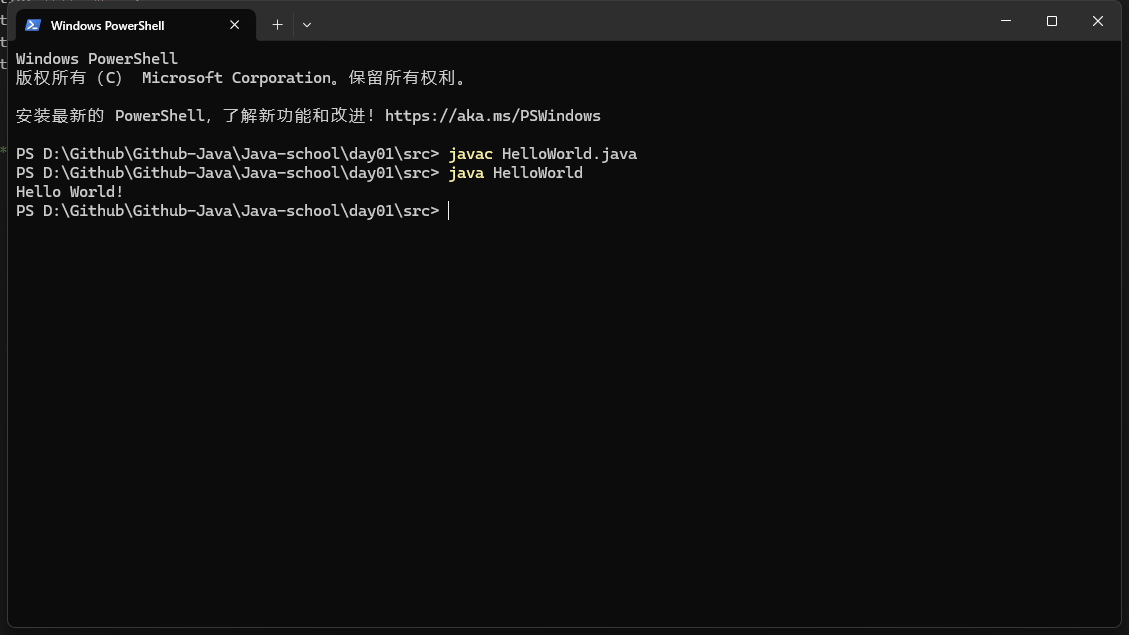
第一次作业

1. 下载jdk和api文档，进行环境变量设置，用记事本编写HelloWorld程序，在dos命令下进行编译和执行成功，提交编译界面

源码：

public class HelloWorld {  
 public static void main(String[] args) {  
 System.*out*.println("Hello World!");  
 }  
}

运行截图

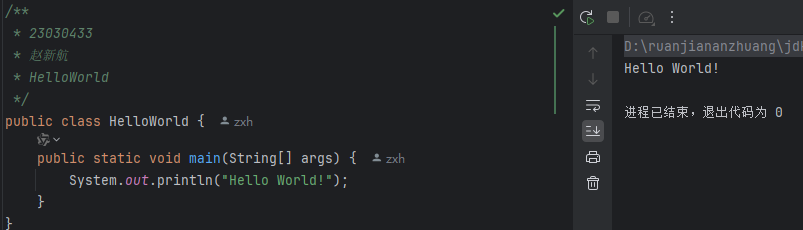
****

1. 下载eclipse或其他IDE，编写HelloWorld程序，进行编译和执行成功，提交编译界面

源码：

public class HelloWorld {  
 public static void main(String[] args) {  
 System.*out*.println("Hello World!");  
 }  
}

运行截图

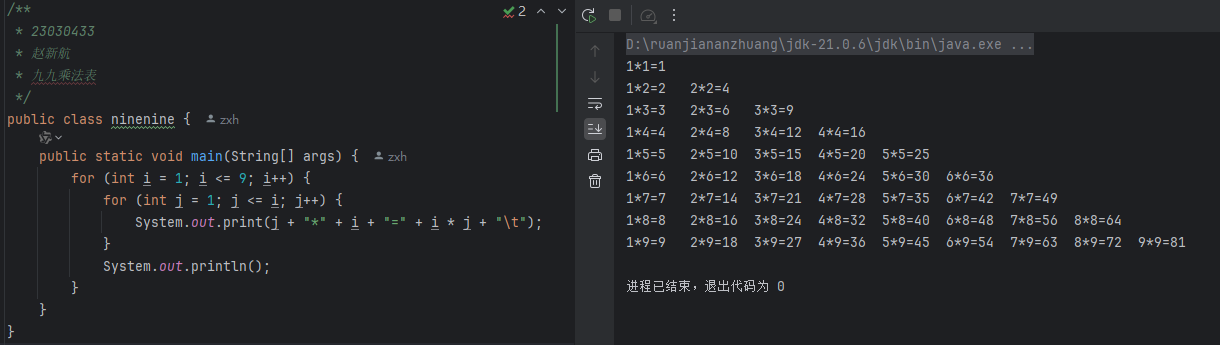


3、编写九九乘法表程序，要求每一列对齐；

源码：

public class ninenine {  
 public static void main(String[] args) {  
 for (int i = 1; i <= 9; i++) {  
 for (int j = 1; j <= i; j++) {  
 System.*out*.print(j + "\*" + i + "=" + i \* j + "\t");  
 }  
 System.*out*.println();  
 }  
 }  
}

运行截图

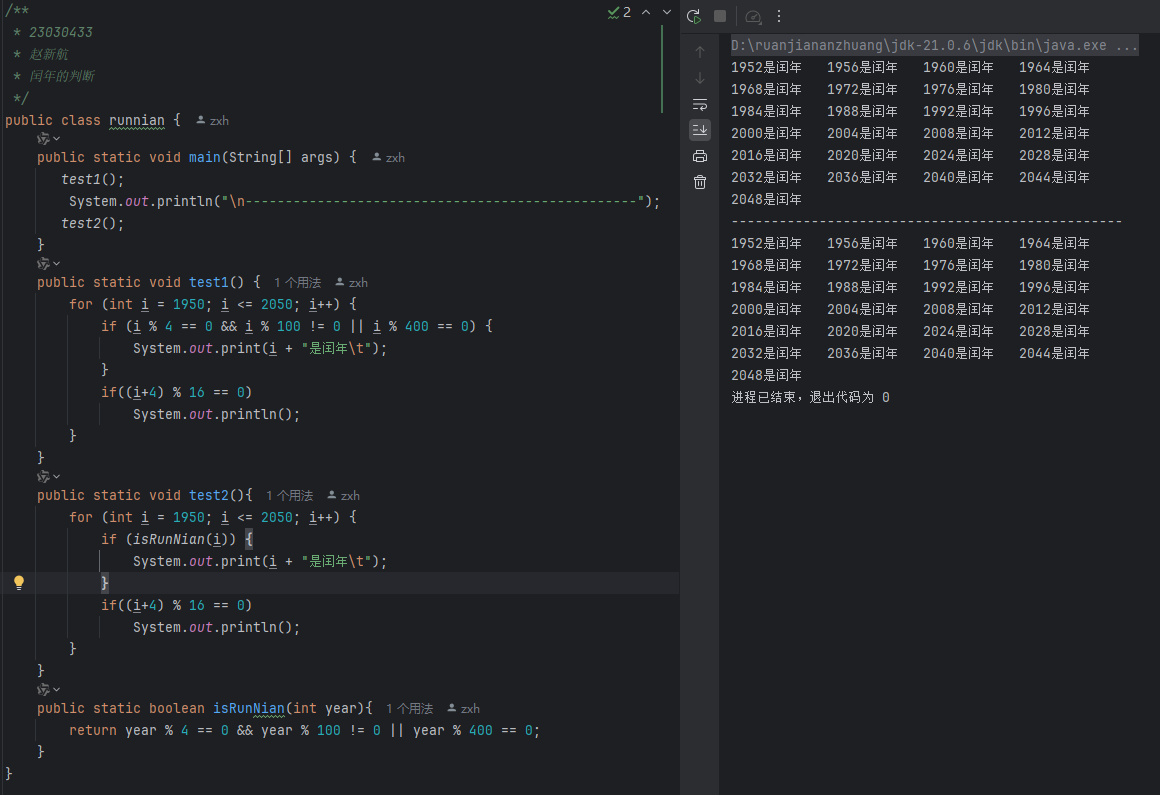


4、提交判断1950-2050年中的闰年（两种方法）

源码：

public class runnian {  
 public static void main(String[] args) {  
 *test1*();  
 System.*out*.println("\n-------------------------------------------------");  
 *test2*();  
 }  
 public static void test1() {  
 for (int i = 1950; i <= 2050; i++) {  
 if (i % 4 == 0 && i % 100 != 0 || i % 400 == 0) {  
 System.*out*.print(i + "是闰年\t");  
 }  
 if((i+4) % 16 == 0)  
 System.*out*.println();  
 }  
 }  
 public static void test2(){  
 for (int i = 1950; i <= 2050; i++) {  
 if (*isRunNian*(i)) {  
 System.*out*.print(i + "是闰年\t");  
 }  
 if((i+4) % 16 == 0)  
 System.*out*.println();  
 }  
 }  
 public static boolean isRunNian(int year){  
 return year % 4 == 0 && year % 100 != 0 || year % 400 == 0;  
 }  
}

运行截图



班级： 23软二 学号：23030433 姓名：赵新航

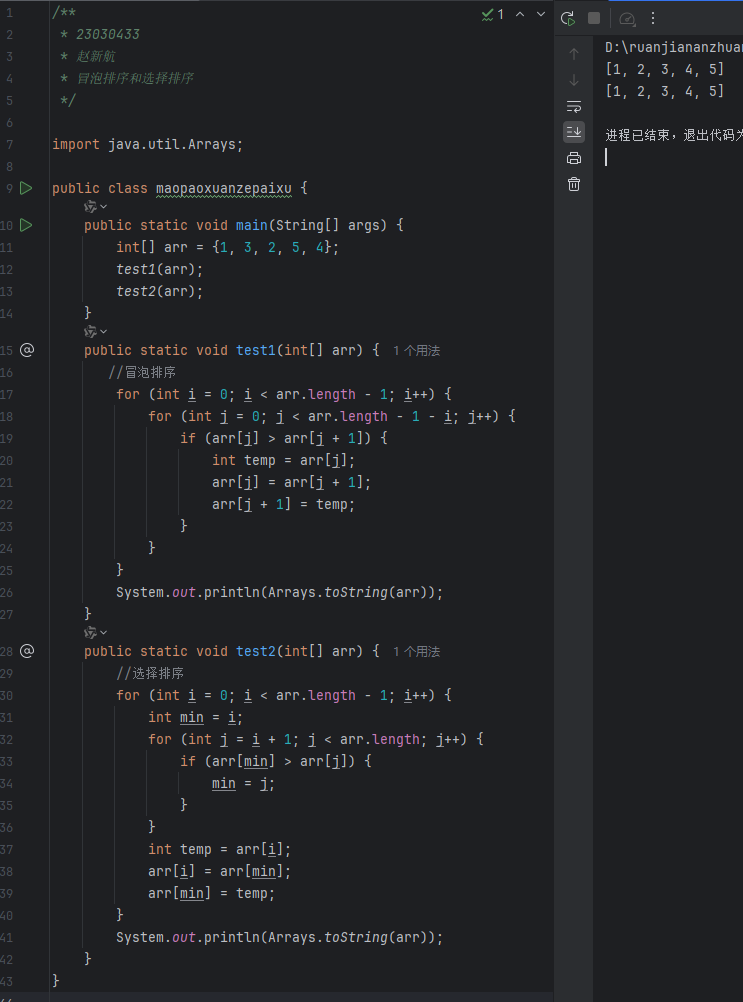
第二次作业

1. 编写冒泡排序和选择排序

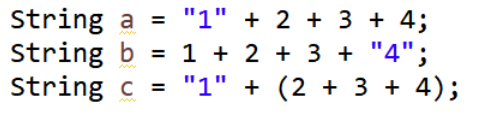
源码：

import java.util.Arrays;  
  
public class maopaoxuanzepaixu {  
 public static void main(String[] args) {  
 int[] arr = {1, 3, 2, 5, 4};  
 *test1*(arr);  
 *test2*(arr);  
 }  
 public static void test1(int[] arr) {  
 //冒泡排序  
 for (int i = 0; i < arr.length - 1; i++) {  
 for (int j = 0; j < arr.length - 1 - i; j++) {  
 if (arr[j] > arr[j + 1]) {  
 int temp = arr[j];  
 arr[j] = arr[j + 1];  
 arr[j + 1] = temp;  
 }  
 }  
 }  
 System.*out*.println(Arrays.*toString*(arr));  
 }  
 public static void test2(int[] arr) {  
 //选择排序  
 for (int i = 0; i < arr.length - 1; i++) {  
 int min = i;  
 for (int j = i + 1; j < arr.length; j++) {  
 if (arr[min] > arr[j]) {  
 min = j;  
 }  
 }  
 int temp = arr[i];  
 arr[i] = arr[min];  
 arr[min] = temp;  
 }  
 System.*out*.println(Arrays.*toString*(arr));  
 }  
}

运行截图：



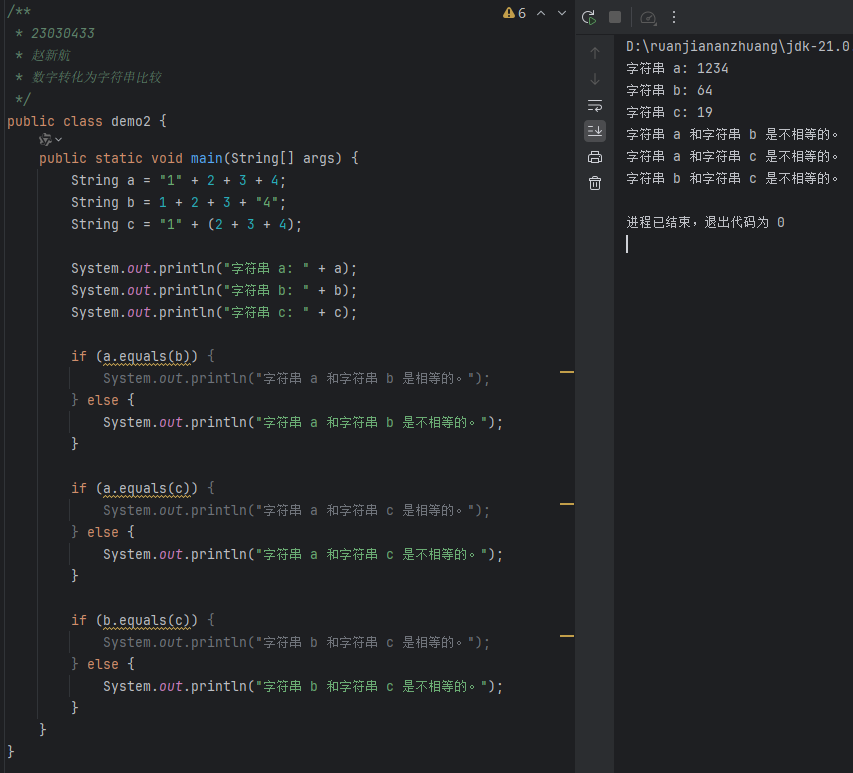
1. 编写程序实现数字转换为字符串，比较结果并分析



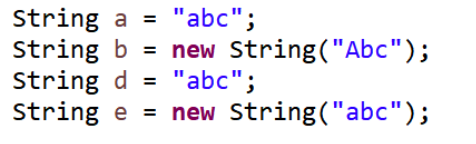
源码：

public class demo2 {  
 public static void main(String[] args) {  
 String a = "1" + 2 + 3 + 4;   
 String b = 1 + 2 + 3 + "4";   
 String c = "1" + (2 + 3 + 4);  
  
 System.*out*.println("字符串 a: " + a);  
 System.*out*.println("字符串 b: " + b);  
 System.*out*.println("字符串 c: " + c);  
  
 if (a.equals(b)) {  
 System.*out*.println("字符串 a 和字符串 b 是相等的。");  
 } else {  
 System.*out*.println("字符串 a 和字符串 b 是不相等的。");  
 }  
  
 if (a.equals(c)) {  
 System.*out*.println("字符串 a 和字符串 c 是相等的。");  
 } else {  
 System.*out*.println("字符串 a 和字符串 c 是不相等的。");  
 }  
  
 if (b.equals(c)) {  
 System.*out*.println("字符串 b 和字符串 c 是相等的。");  
 } else {  
 System.*out*.println("字符串 b 和字符串 c 是不相等的。");  
 }  
 }  
}

运行截图



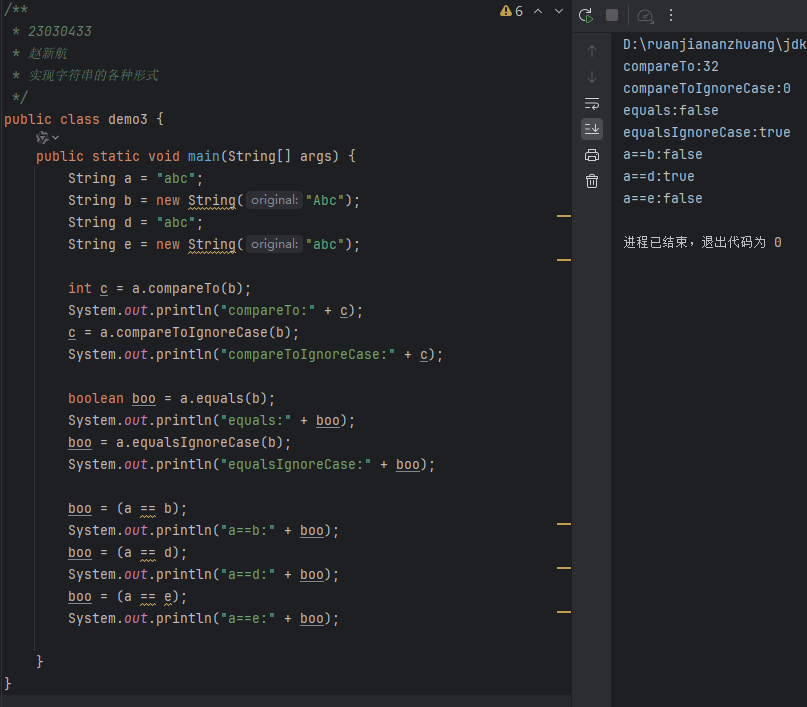
1. 编写程序实现字符串的各种方法，比较结果并分析



源码：

public class demo3 {  
 public static void main(String[] args) {  
 String a = "abc";  
 String b = new String("Abc");  
 String d = "abc";  
 String e = new String("abc");  
  
 int c = a.compareTo(b);  
 System.*out*.println("compareTo:" + c);  
 c = a.compareToIgnoreCase(b);  
 System.*out*.println("compareToIgnoreCase:" + c);  
  
 boolean boo = a.equals(b);  
 System.*out*.println("equals:" + boo);  
 boo = a.equalsIgnoreCase(b);  
 System.*out*.println("equalsIgnoreCase:" + boo);  
  
 boo = (a == b);  
 System.*out*.println("a==b:" + boo);  
 boo = (a == d);  
 System.*out*.println("a==d:" + boo);  
 boo = (a == e);  
 System.*out*.println("a==e:" + boo);  
  
 }  
}

运行截图：



班级： 23软二 学号：23030433 姓名：赵新航

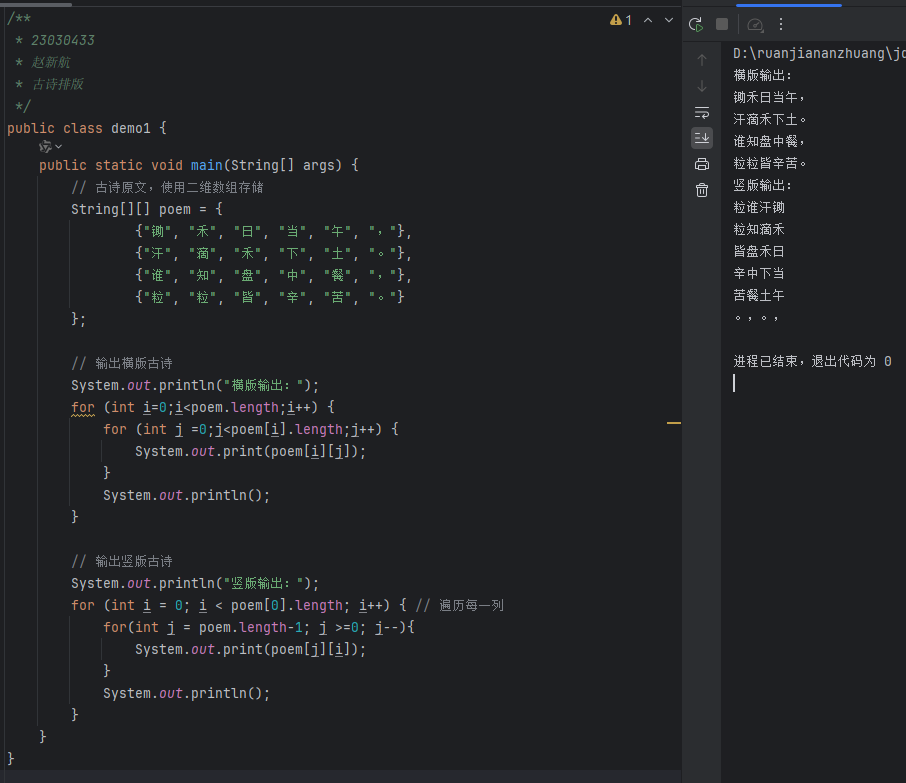
第三次作业

1. 输出横版和竖版的古诗

源码：

public class demo1 {  
 public static void main(String[] args) {   
 String[][] poem = {  
 {"锄", "禾", "日", "当", "午", "，"},  
 {"汗", "滴", "禾", "下", "土", "。"},  
 {"谁", "知", "盘", "中", "餐", "，"},  
 {"粒", "粒", "皆", "辛", "苦", "。"}  
 };   
 System.*out*.println("横版输出：");  
 for (int i=0;i<poem.length;i++) {  
 for (int j =0;j<poem[i].length;j++) {  
 System.*out*.print(poem[i][j]);  
 }  
 System.*out*.println();  
 }   
 System.*out*.println("竖版输出：");  
 for (int i = 0; i < poem[0].length; i++) { // 遍历每一列  
 for(int j = poem.length-1; j >=0; j--){  
 System.*out*.print(poem[j][i]);  
 }  
 System.*out*.println();  
 }  
 }  
}

运行截图：

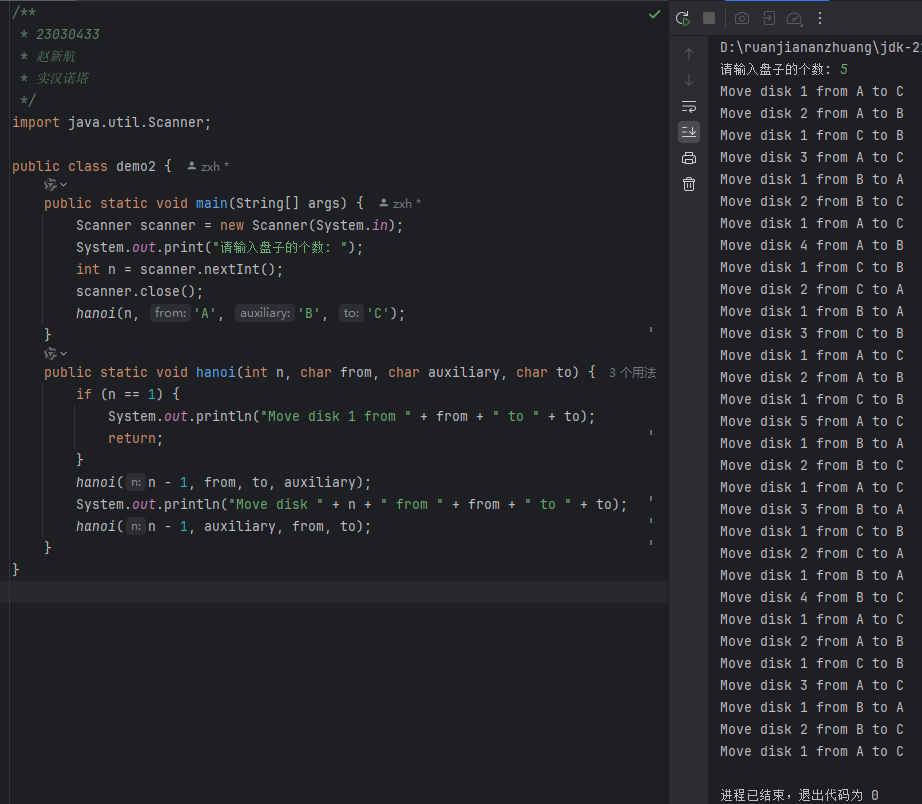


1. 编写程序实现汉诺塔，用户输入盘子个数，输出盘子移动的信息

源码：

import java.util.Scanner;  
public class demo2 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("请输入盘子的个数: ");  
 int n = scanner.nextInt();  
 scanner.close();  
 *hanoi*(n, 'A', 'B', 'C');  
 }  
 public static void hanoi(int n, char from, char auxiliary, char to) {  
 if (n == 1) {  
 System.*out*.println("Move disk 1 from " + from + " to " + to);  
 return;  
 }  
 *hanoi*(n - 1, from, to, auxiliary);  
 System.*out*.println("Move disk " + n + " from " + from + " to " + to);  
 *hanoi*(n - 1, auxiliary, from, to);  
 }  
}

运行截图：

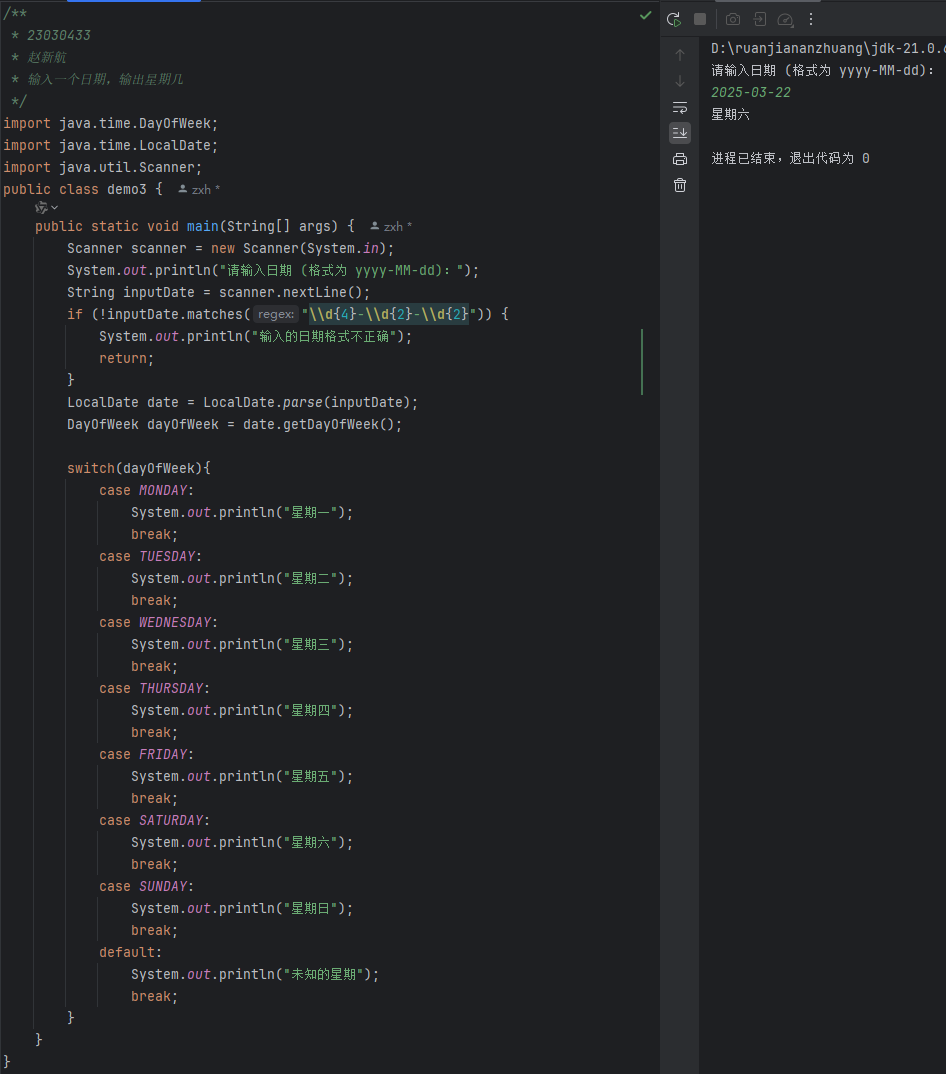


1. 程序实现输入一个日期，输出星期几的功能

源码：

import java.time.DayOfWeek;  
import java.time.LocalDate;  
import java.util.Scanner;  
public class demo3 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.println("请输入日期 (格式为 yyyy-MM-dd)：");  
 String inputDate = scanner.nextLine();  
 if (!inputDate.matches("\\d{4}-\\d{2}-\\d{2}")) {  
 System.*out*.println("输入的日期格式不正确");  
 return;  
 }  
 LocalDate date = LocalDate.*parse*(inputDate);  
 DayOfWeek dayOfWeek = date.getDayOfWeek();  
  
 switch(dayOfWeek){  
 case *MONDAY*:  
 System.*out*.println("星期一");  
 break;  
 case *TUESDAY*:  
 System.*out*.println("星期二");  
 break;  
 case *WEDNESDAY*:  
 System.*out*.println("星期三");  
 break;  
 case *THURSDAY*:  
 System.*out*.println("星期四");  
 break;  
 case *FRIDAY*:  
 System.*out*.println("星期五");  
 break;  
 case *SATURDAY*:  
 System.*out*.println("星期六");  
 break;  
 case *SUNDAY*:  
 System.*out*.println("星期日");  
 break;  
 default:  
 System.*out*.println("未知的星期");  
 break;  
 }  
 }  
}

运行截图：



1. 编号为1-100的一群人进入一个有编号1-100盏灯的房间，每个人只按电灯编号是自己编号倍数的电灯开关；初始时，电灯都为关闭状态，求最后有哪些灯是打开状态。

源码：

public class demo4 {  
 public static void main(String[] args) {  
 boolean[] data = new boolean[101];  
 for (int i = 1; i < data.length; i++) {  
 data[i] = false;  
 }  
 for(int i =1;i<data.length;i++){  
 for(int j=i;j<data.length;j+=i){  
 data[j] = !data[j];  
 }  
 }  
 for(int i =1;i<data.length;i++){  
 if(data[i]){  
 System.*out*.println(i);  
 }  
 }  
 }  
}

运行截图：

