

廈門大學



信息学院软件工程系

《JAVA 程序设计》实验报告

实验十

姓名：庾晓萍

学号：20420192201952

学院：信息学院

专业：软件工程

完成时间：2022/4/28

一、实验目的及要求

（一）实验目的

- 1、熟悉字符串及规则表达式

（二）实验要求

- 1、按照题目要求写代码和实验报告，并上传到 FTP

二、实验题目及实现过程

一、基本题目：

题目 1：输出以下新闻片段中所有出现的单词（每个单词只输出一次）。

（一）实验环境

操作系统：Windows 10;

IDE：Eclipse Java 2018-12

编程语言：Java;

（二）实现过程

（1）设计思路

建立一个哈希表，键值对为 String 和 Integer。使用 Pattern.compile 函数实现对指定字符串的截取，生成 Pattern 对象并编译正则表达式。

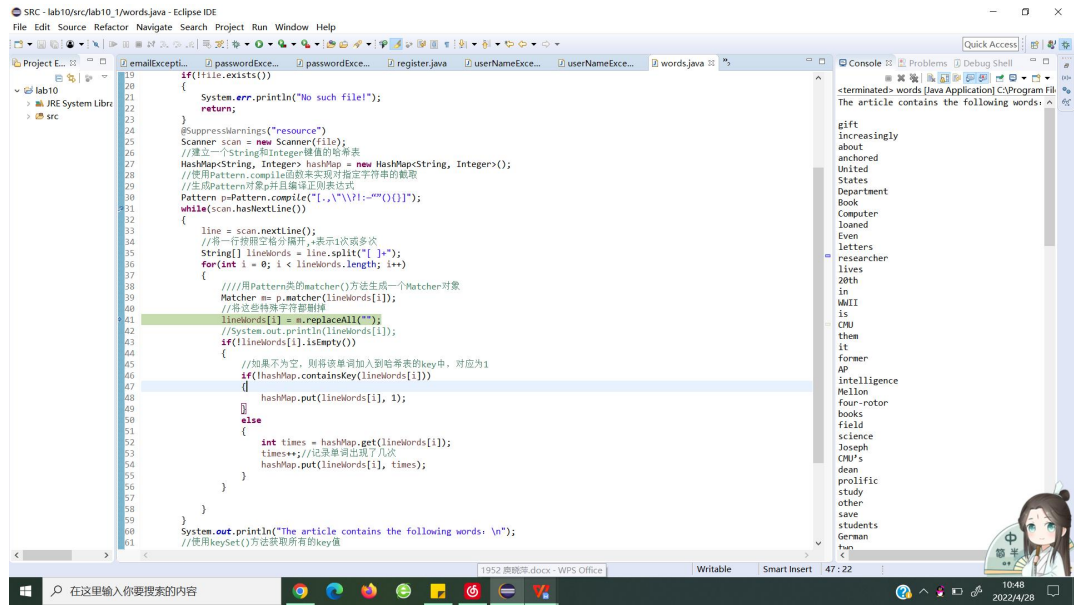
之后在循环中读取文件的每一行，以 `split` 提取出被一个或多个空格分隔开的字符段。接着用 `Pattern` 类的 `matcher()` 方法生成一个 `p` 的 `Matcher` 对象，用 `replaceAll` 将这些特殊字符全部删掉。最后如果元素不为空，则将该单词加入哈希表 `key` 中，`value` 对应出现次数。

(2) 核心代码

```
@SuppressWarnings("resource")
Scanner scan = new Scanner(file);
//建立一个String和Integer键值的哈希表
HashMap<String, Integer> hashMap = new HashMap<String, Integer>();
//使用Pattern.compile函数来实现对指定字符串的截取
//生成Pattern对象p并且编译正则表达式
Pattern p=Pattern.compile("[. ,\\\"\\? !: -\"\"(){} ]");
while(scan.hasNextLine())
{
    line = scan.nextLine();
    //将一行按照空格分隔开,+表示1次或多次
    String[] lineWords = line.split(" [ ]+");
    for(int i = 0; i < lineWords.length; i++)
    {
        ////用Pattern类的matcher()方法生成一个Matcher对象
        Matcher m= p.matcher(lineWords[i]);
        //将这些特殊字符都删掉
        lineWords[i] = m.replaceAll("");
        //System.out.println(lineWords[i]);
        if(!lineWords[i].isEmpty())
        {
            //如果不为空，则将该单词加入到哈希表的key中，对应为1
            if(!hashMap.containsKey(lineWords[i]))
            {
                hashMap.put(lineWords[i], 1);
            }
            else
            {
                int times = hashMap.get(lineWords[i]);
                times++;//记录单词出现了几次
                hashMap.put(lineWords[i], times);
            }
        }
    }
}
```

(三) 过程截图

(1) 全屏截图



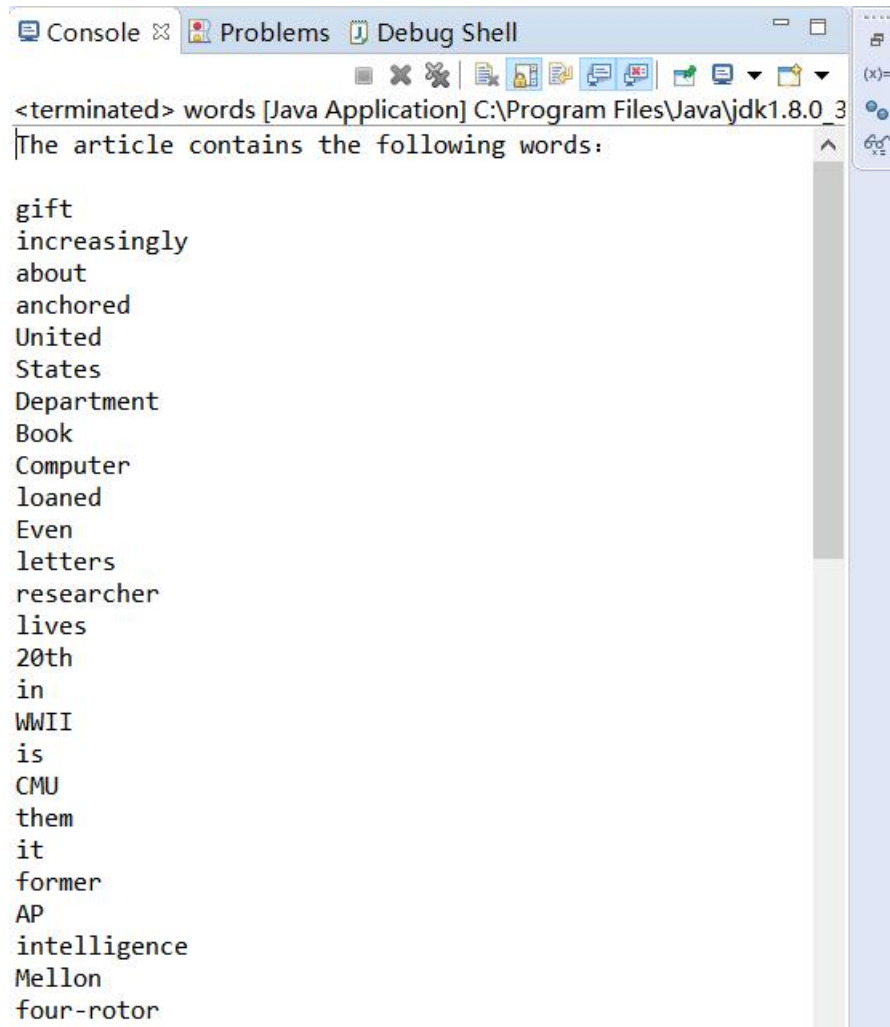
```
10 if(!file.exists())
11 {
12     System.err.println("No such file!");
13     return;
14 }
15 @SuppressWarnings("resource")
16 Scanner scan = new Scanner(file);
17 //建立一个String和Integer键值的哈希表
18 HashMap<String, Integer> hashMap = new HashMap<String, Integer>();
19 //使用Pattern.compile函数来匹配指定字符串的匹配
20 //生成Pattern对象并编译正则表达式
21 Pattern p = Pattern.compile("[a-zA-Z]+");
22 while(scan.hasNextLine())
23 {
24     line = scan.nextLine();
25     //将一行字符串按空格分割开，表示1次或多次
26     String[] lineWords = line.split(" ");
27     for(int i = 0; i < lineWords.length; i++)
28     {
29         //用Pattern类的matcher()方法生成一个Matcher对象
30         Matcher m = p.matcher(lineWords[i]);
31         //将字符串分割成单词
32         lineWords[i] = m.replaceAll("");
33         //System.out.println(lineWords[i]);
34         if(!lineWords[i].isEmpty())
35         {
36             //如果不为空，则将单词加入到哈希表的key中，对应为1
37             if(!hashMap.containsKey(lineWords[i]))
38             {
39                 hashMap.put(lineWords[i], 1);
40             }
41             else
42             {
43                 int times = hashMap.get(lineWords[i]);
44                 times++; //记录单词出现了几次
45                 hashMap.put(lineWords[i], times);
46             }
47         }
48     }
49 }
50 System.out.println("The article contains the following words: \n");
51 //使用keySet()方法获取所有的key值
```

Console Output:

```
<terminated> words [Java Application] C:\Program Fi
The article contains the following words:
gift
increasingly
about
anchored
United
States
Department
Book
Computer
loaned
Even
letters
researcher
lives
20th
in
WWII
is
CNN
them
it
former
AP
intelligence
Mellon
four-rotor
books
field
science
Joseph
CNN's
dean
prolific
study
other
save
students
German
the
```

(2) 运行结果

输出以下新闻片段中所有出现的单词（每个单词只输出一次）



```
<terminated> words [Java Application] C:\Program Files\Java\jdk1.8.0_3
The article contains the following words:

gift
increasingly
about
anchored
United
States
Department
Book
Computer
loaned
Even
letters
researcher
lives
20th
in
WWII
is
CMU
them
it
former
AP
intelligence
Mellon
four-rotor
```

题目 2：对用户输入的用户名、密码、邮箱进行判断，若不满足输入要求则提示出错类型。

（一）实验环境

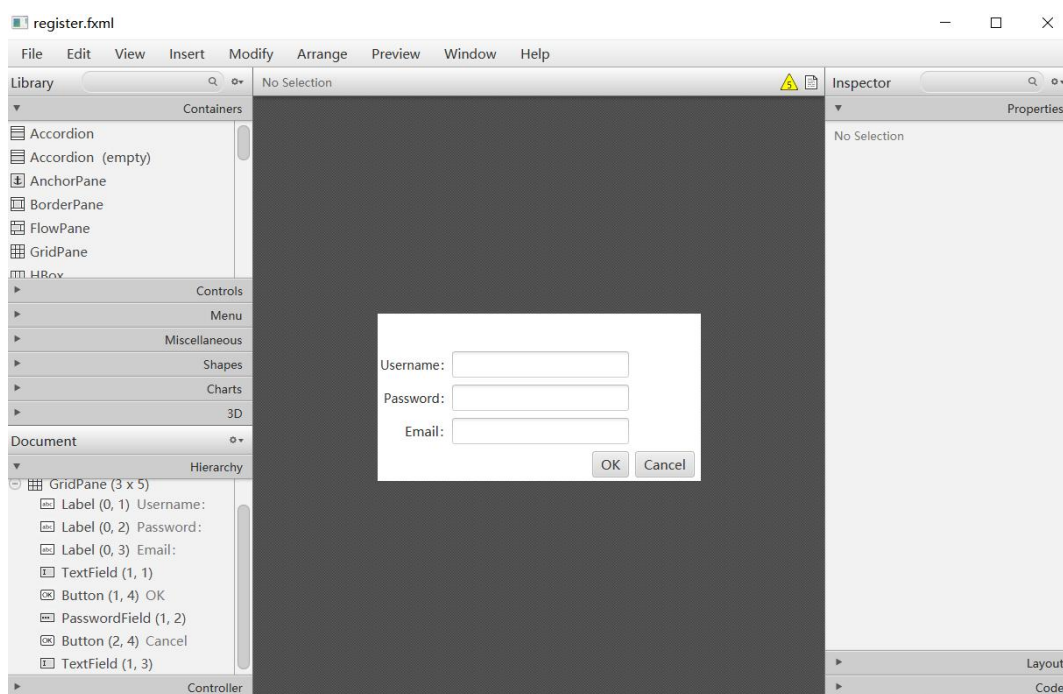
操作系统：Windows 10;

IDE：Eclipse Java 2018-12

编程语言：Java;

(二) 实现过程

(1) 设计 fxml



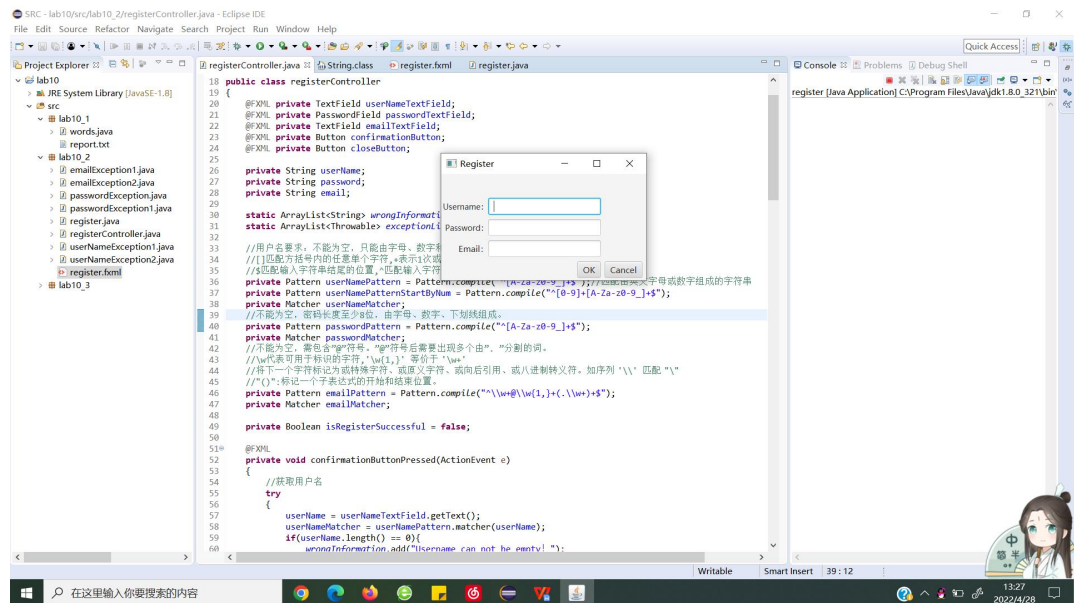
(2) 设计 Controller

对用户输入的用户名、密码、邮箱进行判断，若不满足输入要求则提示出错类型。用户名不能为空，只能由字母、数字和_组成，第一位不能为数字。可以用`"^[A-Za-z0-9_]+$"`来匹配由英文字母或数字或下划线组成的字符串。用`"^[0-9]+[A-Za-z0-9_]+$"`来匹配第一位是数字的由英文字母或数字或下划线组成的字符串（从而剔除）。密码不能为空，密码长度至少 8 位，由字母、数字、下划线组成。同样可以用`"^[A-Za-z0-9_]+$"`来匹配由英文字母或数字或下划线组成的字符串。最后增加一个`password.length() < 8`用来剔除不符合要求的输入。邮箱要求：不能为空，需包含`"@"`符号。`"@"`符号后需要出现多个由`"."`分割的词。可以通过`"^\\w+@\\w{1,}+(\\.\\w+)+$"`来进行匹配。

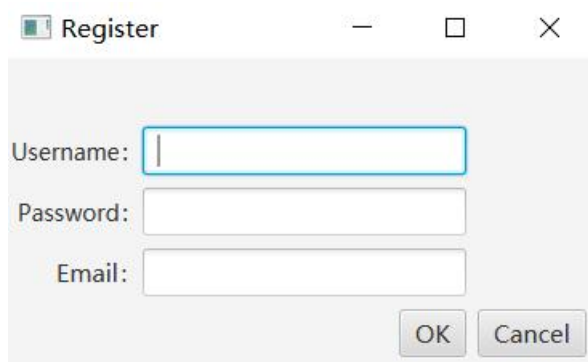
```
private Boolean isRegisterSuccessful = false;
```

（三）过程截图

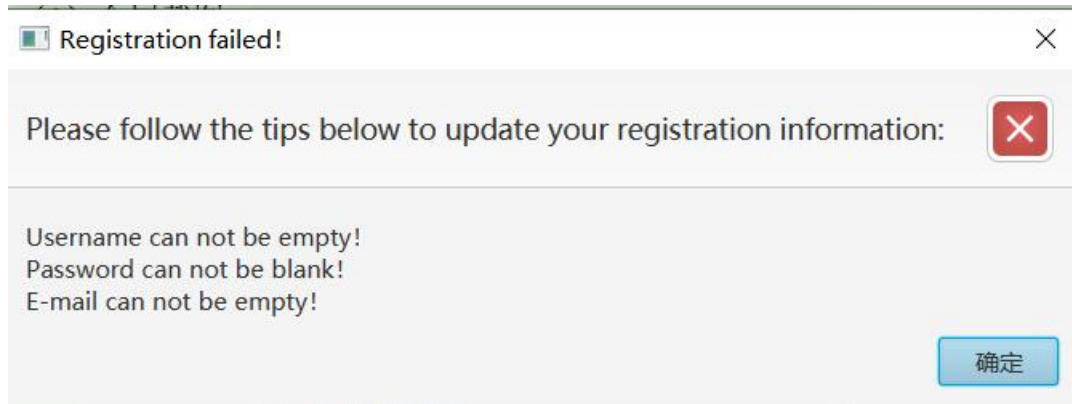
(1) 全屏截图



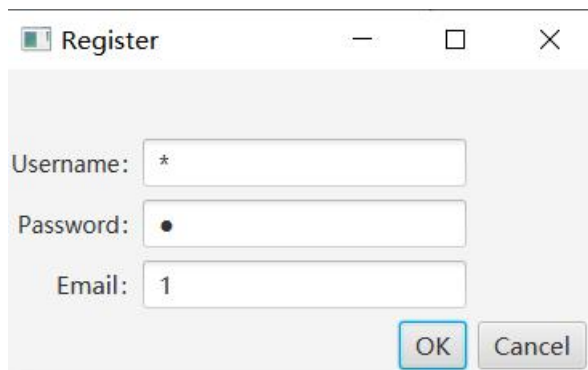
(2) 运行结果



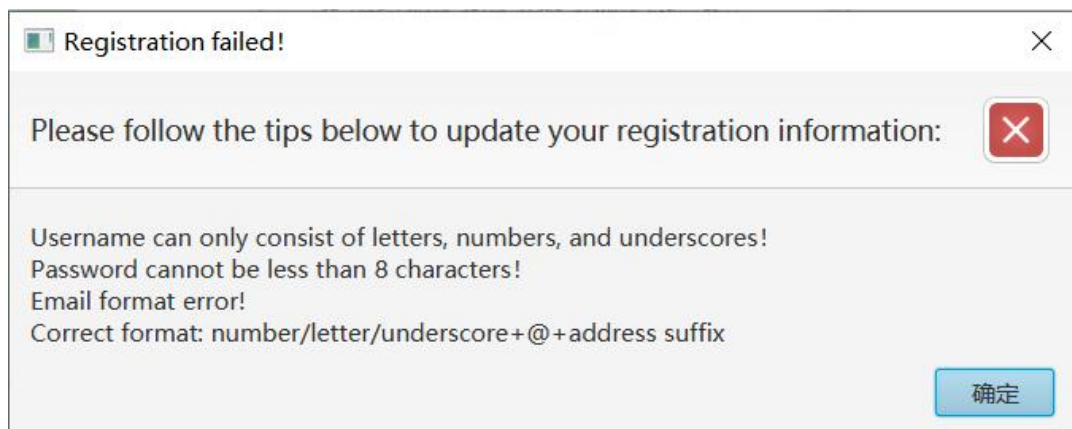
A Java Swing dialog box titled "Register" with standard window controls (minimize, maximize, close). It contains three text input fields labeled "Username:", "Password:", and "Email:". At the bottom right are "OK" and "Cancel" buttons.



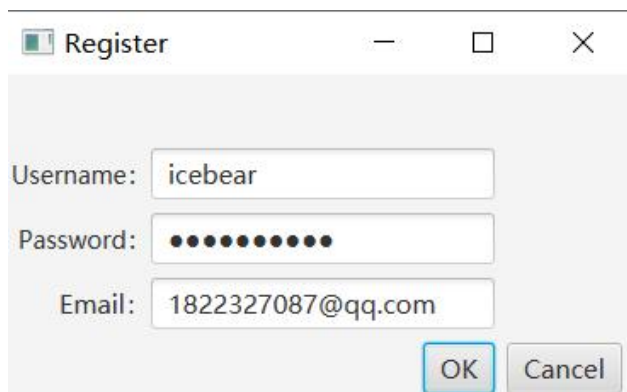
A Java Swing dialog box titled "Registration failed!" with a close button. It contains a message: "Please follow the tips below to update your registration information:" followed by a list of error messages: "Username can not be empty!", "Password can not be blank!", and "E-mail can not be empty!". A "确定" (Confirm) button is at the bottom right.



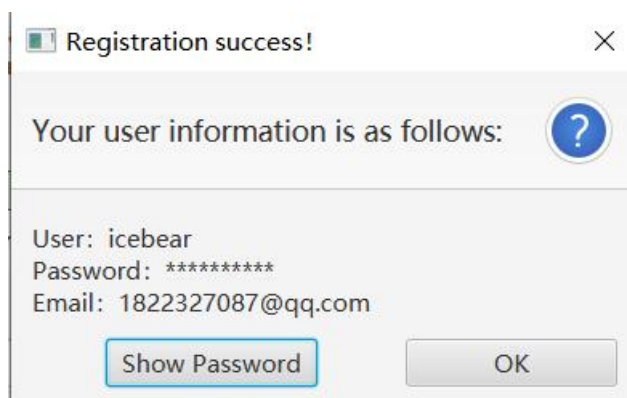
A Java Swing dialog box titled "Register" with standard window controls. It contains three text input fields labeled "Username:", "Password:", and "Email:". The "Username:" field contains an asterisk (*), the "Password:" field contains a bullet point (•), and the "Email:" field contains the number 1. At the bottom right are "OK" and "Cancel" buttons.



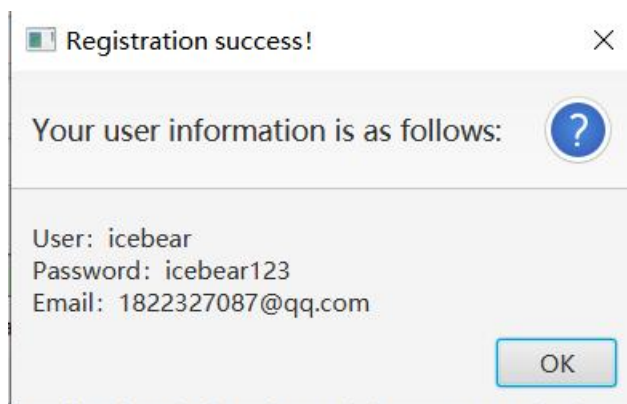
A Java Swing dialog box titled "Registration failed!" with a close button. It contains a message: "Please follow the tips below to update your registration information:" followed by a list of error messages: "Username can only consist of letters, numbers, and underscores!", "Password cannot be less than 8 characters!", "Email format error!", and "Correct format: number/letter/underscore+@+address suffix". A "确定" (Confirm) button is at the bottom right.



A Java Swing dialog box titled "Register". It contains three text input fields: "Username:" with the text "icebear", "Password:" with masked characters (dots), and "Email:" with the text "1822327087@qq.com". At the bottom right, there are two buttons: "OK" and "Cancel".



A Java Swing dialog box titled "Registration success!". It contains a message "Your user information is as follows:" followed by a question mark icon. Below this, the user information is displayed: "User: icebear", "Password: *****", and "Email: 1822327087@qq.com". At the bottom, there are two buttons: "Show Password" and "OK".



A Java Swing dialog box titled "Registration success!". It contains a message "Your user information is as follows:" followed by a question mark icon. Below this, the user information is displayed: "User: icebear", "Password: icebear123", and "Email: 1822327087@qq.com". At the bottom right, there is a single button: "OK".

题目 3: Tokenizing Telephone Numbers

(一) 实验环境

操作系统: Windows 10;

IDE: Eclipse Java 2018-12

编程语言：Java；

（二）实现过程

（1）实验思路

编写应用程序，将电话号码输入为 (555) 555-5555 形式的字符串。应用程序应该通过 `String` 方法拆分提取区号作为 `token`，电话号码的前三位作为 `token`，电话号码的后四位作为 `token`。电话号码的七位数字应连接成一个字符串。应打印区号和电话号码。请记住，您必须在标记化过程中更改分隔符。

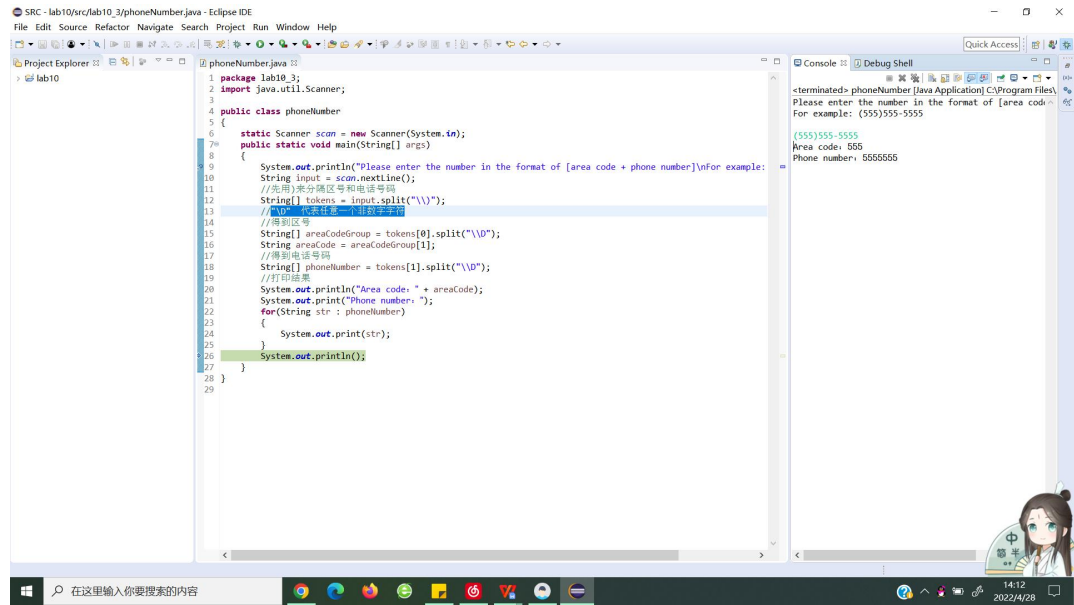
（2）Controller 代码

先用 `)` 来分割区号和电话号码，`"\D"` 代表任意一个非数字字符，用这个来完成后续的分割。

```
phoneNumber.java
1 package lab10_3;
2 import java.util.Scanner;
3
4 public class phoneNumber
5 {
6     static Scanner scan = new Scanner(System.in);
7     public static void main(String[] args)
8     {
9         System.out.println("Please enter the number in the format of [area code + phone number]\nFor example: (555)555-5555\n");
10        String input = scan.nextLine();
11        //先用)来分隔区号和电话号码
12        String[] tokens = input.split("\\)");
13        //"\\D" 代表任意一个非数字字符
14        //得到区号
15        String[] areaCodeGroup = tokens[0].split("\\D");
16        String areaCode = areaCodeGroup[1];
17        //得到电话号码
18        String[] phoneNumber = tokens[1].split("\\D");
19        //打印结果
20        System.out.println("Area code: " + areaCode);
21        System.out.print("Phone number: ");
22        for(String str : phoneNumber)
23        {
24            System.out.print(str);
25        }
26        System.out.println();
27    }
28 }
```

（三）过程截图

（1）全屏截图



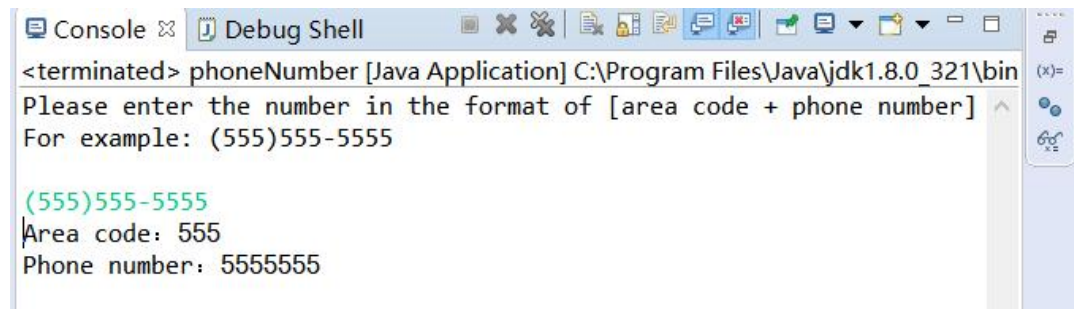
The screenshot shows the Eclipse IDE with the file `phoneNumber.java` open. The code uses a `Scanner` to read input and `split` to parse the area code and phone number. The console shows the program's execution with the input `(555)555-5555` and the resulting output.

```
1 package lab10_3;
2 import java.util.Scanner;
3
4 public class phoneNumber
5 {
6     static Scanner scan = new Scanner(System.in);
7
8     public static void main(String[] args)
9     {
10         System.out.println("Please enter the number in the format of [area code + phone number]\nFor example: (555)555-5555");
11         //先用\来分隔区号和电话号码
12         String[] tokens = input.split("\\\\");
13         //得到区号
14         String[] areaCodeGroup = tokens[0].split("\\0");
15         String areaCode = areaCodeGroup[1];
16         //得到电话号码
17         String[] phoneNumber = tokens[1].split("\\0");
18         //打印结果
19         System.out.println("Area code: " + areaCode);
20         System.out.print("Phone number: ");
21         for(String str : phoneNumber)
22         {
23             System.out.print(str);
24         }
25         System.out.println();
26     }
27 }
28
29
```

Console Output:

```
<terminated> phoneNumber [Java Application] C:\Program Files\Java\jdk1.8.0_321\bin
Please enter the number in the format of [area code + phone number]
For example: (555)555-5555
(555)555-5555
Area code: 555
Phone number: 5555555
```

(2) 运行结果



This screenshot shows the Eclipse console window. It displays the prompt for input, the user's input `(555)555-5555`, and the program's output: `Area code: 555` and `Phone number: 5555555`.

```
<terminated> phoneNumber [Java Application] C:\Program Files\Java\jdk1.8.0_321\bin
Please enter the number in the format of [area code + phone number]
For example: (555)555-5555
(555)555-5555
Area code: 555
Phone number: 5555555
```

三、实验总结与心得记录

在本次实验过程中,我熟悉了 java 的语法,熟悉了 javafx,体会到了 JAVA 语言的优点。