

河海大学

HOHAI UNIVERSITY

---

## 《面向对象程序设计》课程设计报告

---

专业 2017 级信息管理与信息系统

学号 1763710232

姓名 王晨辉

2019 年 5 月 10 日

## 第 1 章 功能设计

本课程设计题目为《校园内二手交易系统》，供多用户使用，功能包括用户注册登录、发布商品、查询商城商品、购买商城商品、用户余额管理与实时更新。

### 1.1 账户管理

负责账户信息的添加、验证与修改，包括用户登陆、用户注册与用户密码修改。

#### 1.1.1 用户登录

用户输入账号密码，进行密码核实后，登入或报错“**密码错误，请尝试重新输入**”

#### 1.1.2 用户注册

用户注册新账号，设置密码，二次确认密码后，注册成功或报错“**改用户名已存在**”

#### 1.1.3 账号注销

注销当前用户，重新进入用户登陆界面

### 1.2 商品管理

#### 1.2.1 商品查询

用户可以根据商品不同分类来查询所需商品，商品分类包括

- **图书课本** 各种教科书、科普读物、小说文学
- **美妆用品** 化妆用品
- **手机数码** 手机、相机、电脑零件等
- **衣物服饰** 四季服饰
- **美食生鲜** 小吃零食等
- **虚拟产品** 服务性商品，例如代拿快递
- **生活用品** 其他各种未分类用品

### 1.2.2 商品发布

用户在商品中发布商品，选择填写商品名、商品数量、商品价格、商品描述，并上传商品图片，上传后用户能够在商城中看到自己发布的商品，也能够用户在用户主页查询到在售商品。

上传图片时可以进行本地图库浏览，也可以直接输入图片在计算机中的绝对路径，例如

"/Users/wangchenhui/Desktop/icon1.jpg "

### 1.2.3 商品购买

用户通过在商城点击选中商品，选择数量后进行购买。

## 1.3 交易管理

负责用户之间发生交易行为后，系统中数据的改变，包括

- **余额信息变动** 用户 A 从用户 B 处购买价值  $\alpha$  的商品，A 用户余额减少  $\alpha$ ,B 用户余额增加  $\alpha$
- **商城信息变动** 用户购买一件商品后，该商品应从商城下架
- **用户主页信息变动** 某用户商品被购买后，该用户主页中**我的在售**一栏发生改变；同时用户交易记录在买卖双方主页中增加。

## 第 2 章 数据库设计

### 2.1 ER 图

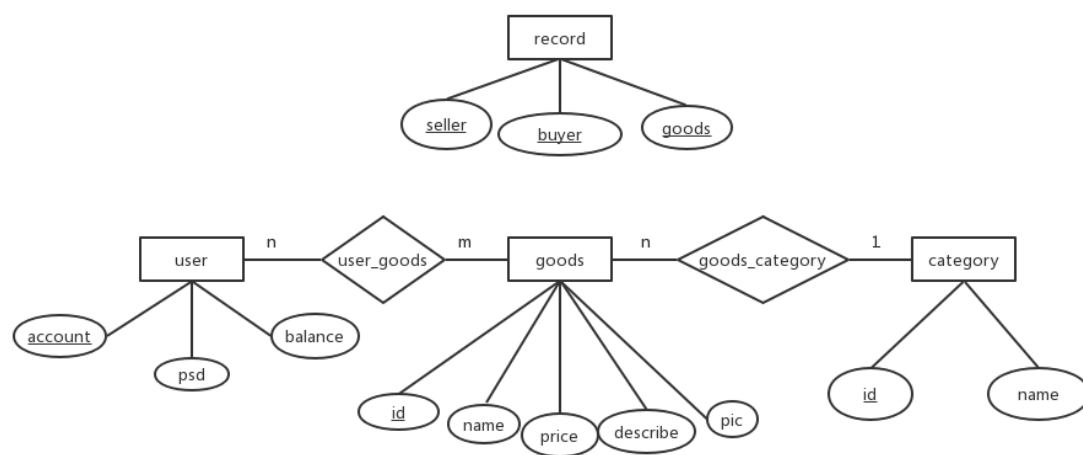


图 2.1: ER 图

## 2.2 关系模型

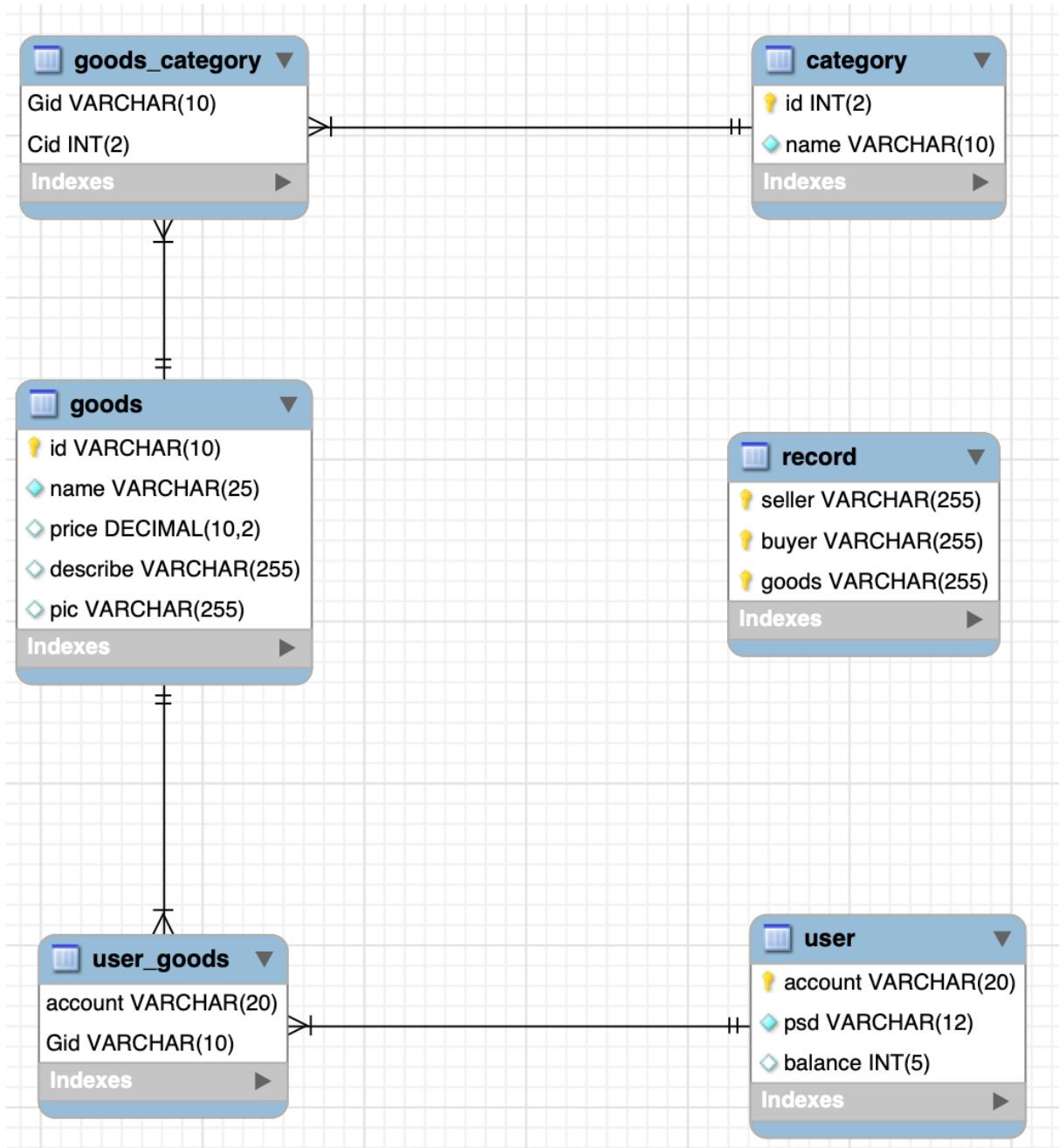


图 2.2: 关系模型

### 2.2.1 用户表

字段	类型	属性
account	varchar(20)	主键
psd	varchar(12)	
balance	int(5)	

### 2.2.2 商品表

字段	类型	属性
id	varchar(10)	主键
name	varchar(25)	
price	decimal(10,2)	
describe	varchar(255)	
pic	varchar(255)	

### 2.2.3 用户商品表

字段	类型	属性
account	varchar(20)	主键
Gid	varchar(10)	主键

#### 2.2.4 分类表

字段	类型	属性
id	int(2)	主键
name	varchar(10)	

#### 2.2.5 商品分类表

字段	类型	属性
Gid	varchar(10)	主键
Cid	int(2)	主键

#### 2.2.6 交易记录表

字段	类型	属性
seller	varchar(255)	主键
buyer	varchar(255)	主键
goods	varchar(255)	主键

## 第 3 章 类设计

这里仅给出与 java 文件同名的公共类，其他的类包括用于存储各种中间数据类型的中间类，已经 swing 事件的内部类。主要的类分为两种：

- **xxx.class**            用于构造各种功能的界面
- **db\_xxx.class**        用于该界面与数据库的交互工作，包括对数据库的增删查改

类名	内容
db.class	数据库的连接
db_log.class	获得登陆界面所需数据
log.class	登陆界面
db_register.class	注册界面所需数据
register.class	注册界面
db_mainWindow.class	商店主界面所需数据
mainWindow.class	商店主界面
db_myPage.class	用户主页所需数据
myPage.class	用户主界面
db_mySale.class	我的在售界面所需数据
mySale.class	我的在售界面
db_addGoods.class	发布商品界面所需数据
addGoods.java	发布商品界面

表 3.1: 类内容



## 第 4 章 代码及展现

### 4.1 代码实现

#### 4.1.1 db.java(数据库链接)

```
1
2 import java.sql.*;
3 import java.util.InputMismatchException;
4 import java.util.Scanner;
5
6 public class db{//用于数据库的链接
7     // JDBC 驱动名及数据库 URL
8     static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
9     static final String DB_URL = "jdbc:mysql://localhost:3306/javaCourseTask?useSSL
                                =false";
10
11     // 数据库的用户名与密码
12     static final String USER = "root";
13     static final String PASS = "root";
14
15
16     public static Connection getConn() {//连接数据库
17         Connection conn = null;
18         try {
19             // 注册 JDBC 驱动
20             Class.forName("com.mysql.cj.jdbc.Driver");
21
22             // 打开链接
23             System.out.println("连接数据库...");
24             conn = DriverManager.getConnection(DB_URL, USER, PASS);
25         } catch (SQLException e) {
26             e.printStackTrace();
27         } catch (Exception e) {
28             e.printStackTrace();
29         }
30         return conn;
31     }
32
33     static ResultSet sqlEnquire(Connection conn,String keyword){//留作子类重写
34         String sql = "";
35         PreparedStatement pstmt;
36         ResultSet rs = null;
37         try {
38             pstmt = conn.prepareStatement(sql);
39             rs = pstmt.executeQuery();
40             pstmt.close();
41             conn.close();
```

```
42         } catch (Exception e) {
43             e.printStackTrace();
44         }
45         return rs;
46     }
47 }
```

#### 4.1.2 db\_log.java (读取登陆界面所需数据库的数据)

```
1
2
3 import java.sql.*;
4 import java.util.InputMismatchException;
5 import java.util.Scanner;
6
7 public class db_log extends db{
8     public static ResultSet sqlEnquire(Connection conn,String account){//登陆时获取
                                数据库中密码
9         String sql = "select psd from user where account = ?";
10        PreparedStatement pstmt;
11        ResultSet rs = null;
12        try {
13            pstmt = conn.prepareStatement(sql);
14            pstmt.setString(1, account);
15            rs = pstmt.executeQuery();
16
17        } catch (Exception e) {
18            e.printStackTrace();
19        }
20
21        return rs;
22    }
23 }
```

#### 4.1.3 log.java (登陆界面)

```
1
2
3 import java.sql.*;
4 import java.util.InputMismatchException;
5 import java.util.Scanner;
6 import java.awt.*;
7 import java.awt.event.ActionEvent;
8 import java.awt.event.ActionListener;
9
10 import javax.swing.*;
```

```
11 import javax.swing.border.EmptyBorder;
12
13
14 class log_info{
15     public static String correctPsd;
16
17     static public boolean testPassword(String psd, ResultSet rs) { // 检验密码与数据库
        库中密码是否匹配
18
19         try {
20             while(rs.next()){
21                 correctPsd = rs.getString(1);
22                 correctPsd.trim();
23             }
24         } catch (Exception e) {
25             e.printStackTrace();
26         }
27         return psd.equals(correctPsd);
28     }
29 }
30
31 public class log extends JFrame {
32
33     private static JPanel contentPane;
34     private JTextField textField;
35     private JPasswordField passwordField;
36     private JLabel label;
37     private JLabel label_1;
38     private final JLabel lblNewLabel_2 = new JLabel("");
39     private JLabel label_3 = new JLabel("");
40     public static String account;
41     public static String psd;
42
43     public log() {
44         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
45         setBounds(100, 100, 548*2, 316*2);
46         contentPane = new JPanel();
47         contentPane.setBackground(Color.WHITE);
48         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
49         setContentPane(contentPane);
50         contentPane.setLayout(null);
51
52         JLabel lblNewLabel = new JLabel("");
53         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
54         image.setImage(image.getImage().getScaledInstance(159*2, 294*2, Image.
            SCALE_DEFAULT));
55
56         lblNewLabel.setIcon(image);
57         lblNewLabel.setBounds(0, 0, 159*2, 294*2);
```

```
57 contentPane.add(lblNewLabel);
58
59 JLabel lblNewLabel_1 = new JLabel("");
60 lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
61 lblNewLabel_1.setBounds(317, 0, 84, 588);
62 contentPane.add(lblNewLabel_1);
63
64 JPanel panel = new JPanel();
65 panel.setBorder(null);
66 panel.setOpaque(false);
67 panel.setBounds(479, 57, 584, 502);
68 contentPane.add(panel);
69 panel.setLayout(null);
70
71 textField = new JTextField();
72 textField.setBounds(266, 158, 240, 43);
73 panel.add(textField);
74 textField.setColumns(25);
75
76 passwordField = new JPasswordField();
77 passwordField.setBounds(266, 226, 240, 43);
78 panel.add(passwordField);
79
80 label = new JLabel("用户名");
81 label.setFont(new Font("Xingkai TC", Font.PLAIN, 25));
82 label.setBounds(117, 154, 114, 43);
83 panel.add(label);
84
85 label_1 = new JLabel("密码");
86 label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 25));
87 label_1.setBounds(117, 222, 114, 43);
88 panel.add(label_1);
89
90 JRadioButton rdbtnNewRadioButton = new JRadioButton("记住密码");
91 rdbtnNewRadioButton.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
92 rdbtnNewRadioButton.setBounds(143, 300, 141, 23);
93 panel.add(rdbtnNewRadioButton);
94
95 JRadioButton radioButton = new JRadioButton("自动登录");
96 radioButton.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
97 radioButton.setBounds(313, 300, 141, 23);
98 panel.add(radioButton);
99
100 JButton button = new JButton("登录");
101
102
103 button.addActionListener(new ActionListener() { // 登陆（密码错误重新输入/跳
                                                转主页面）
```

```
104     public void actionPerformed(ActionEvent e) {
105         account = textField.getText();
106         account.trim();
107         psd = passwordField.getText();
108         psd.trim();
109
110         Connection conn = db.getConn();
111         ResultSet rs = db_log.sqlEnquire(conn, account);
112         boolean tempt = log_info.testPassword(psd, rs);
113         String test = log_info.correctPsd;
114         if (tempt == true) {
115             label_3.setText("密码正确");
116
117             mainWindow frame_1 = new mainWindow();
118             frame_1.setVisible(true);
119             log.this.dispose();
120
121         } else {
122             label_3.setText("密码错误，请重新输入");
123             button.setEnabled(true);
124             passwordField.setText("");
125         }
126
127     }
128 });
129
130
131 button.setFont(new Font("Xingkai TC", Font.PLAIN, 34));
132 button.setBackground(new Color(30, 144, 255));
133 button.setBounds(167, 370, 308, 54);
134 button.setOpaque(true);
135 button.setBorderPainted(false);
136 panel.add(button);
137
138 JLabel label_2 = new JLabel("淘 客");
139 label_2.setFont(new Font("Xingkai TC", Font.PLAIN, 49));
140 label_2.setBounds(214, 6, 261, 92);
141 panel.add(label_2);
142
143 JButton button_1 = new JButton("去注册");
144 button_1.addActionListener(new ActionListener() {
145     public void actionPerformed(ActionEvent e) {
146         register frame_2 = new register();
147         frame_2.setVisible(true);
148         log.this.dispose();
149     }
150 });
151
```

```
152    button_1.setBorder(null);
153    button_1.setForeground(new Color(65, 105, 225));
154    button_1.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
155    button_1.setBounds(132, 438, 117, 29);
156    panel.add(button_1);
157
158
159    label_3.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
160    label_3.setForeground(Color.RED);
161    label_3.setBounds(170, 342, 191, 23);
162    panel.add(label_3);
163
164    lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
165    lblNewLabel_2.setBounds(413, 0, 683, 622);
166    contentPane.add(lblNewLabel_2);
167 }
168 }
```

#### 4.1.4 db\_register.java(注册界面所需数据库数据读取)

```
1
2
3 import java.sql.*;
4 import java.util.InputMismatchException;
5 import java.util.Scanner;
6
7 public class db_register {
8     public static void insertNewAccount(String account,String psd){
9         Connection conn = db.getConn();
10        String sql = "insert into user values(?,?,0)";
11        PreparedStatement pstmt;
12        try {
13            pstmt = (PreparedStatement) conn.prepareStatement(sql);
14            pstmt.setString(1, account);
15            pstmt.setString(2, psd);
16            pstmt.executeUpdate();
17            pstmt.close();
18            conn.close();
19        } catch (SQLException e) {
20            e.printStackTrace();
21        }
22    }
23 }
```

#### 4.1.5 register.java(用户注册界面)

```
1
2
3
4 import java.sql.*;
5 import java.util.InputMismatchException;
6 import java.util.Scanner;
7 import java.awt.*;
8 import java.awt.event.ActionEvent;
9 import java.awt.event.ActionListener;
10
11 import javax.swing.*;
12 import javax.swing.border.EmptyBorder;
13
14 public class register extends JFrame {
15
16     private JPanel contentPane;
17     private final JLabel lblNewLabel_2 = new JLabel("");
18
19     private JTextField textField;
20     private JTextField textField_1;
21
22
23
24     public register() {
25         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
26         setBounds(100, 100, 548*2, 316*2);
27         contentPane = new JPanel();
28         contentPane.setBackground(Color.WHITE);
29         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
30         setContentPane(contentPane);
31         contentPane.setLayout(null);
32         JLabel lblNewLabel = new JLabel("");
33         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
34         image.setImage(image.getImage().getScaledInstance(159*2, 294*2, Image.
35                                     SCALE_DEFAULT));
36         lblNewLabel.setIcon(image);
37         lblNewLabel.setBounds(0, 0, 159*2, 294*2);
38         contentPane.add(lblNewLabel);
39         JLabel lblNewLabel_1 = new JLabel("");
40         lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
41         lblNewLabel_1.setBounds(317, 0, 84, 588);
42         contentPane.add(lblNewLabel_1);
43
44         JLabel label = new JLabel("用户名");
45         label.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
46         label.setBounds(476, 216, 99, 32);
47         contentPane.add(label);
```

```

47
48 JLabel label_1 = new JLabel("密码");
49 label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
50 label_1.setBounds(476, 279, 99, 32);
51 contentPane.add(label_1);
52
53 textField = new JTextField();
54 textField.setBounds(608, 216, 180, 32);
55 contentPane.add(textField);
56 textField.setColumns(10);
57
58 textField_1 = new JTextField();
59 textField_1.setColumns(10);
60 textField_1.setBounds(608, 285, 180, 32);
61 contentPane.add(textField_1);
62
63 JButton button = new JButton("注册");
64 button.setFont(new Font("Xingkai TC", Font.PLAIN, 28));
65 button.setBackground(new Color(30, 144, 255));
66 button.setBounds(560, 374, 171, 41);
67 button.setOpaque(true);
68 button.setBorderPainted(false);
69 button.addActionListener(new ActionListener(){
70     public void actionPerformed(ActionEvent e){
71         log.account = textField.getText();
72         log.psd = textField_1.getText();
73         db_register.insertNewAccount(log.account, log.psd);
74         myPage frame_3 = new myPage();
75         frame_3.setVisible(true);
76         register.this.dispose();
77     }
78 });
79 contentPane.add(button);
80
81
82 lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
83 lblNewLabel_2.setBounds(413, 0, 683, 622);
84 contentPane.add(lblNewLabel_2);
85
86 }
87 }

```

#### 4.1.6 db\_mainwindow(商店主界面所需数据)

```

1
2
3

```



```
4 import java.sql.*;
5 import java.util.ArrayList;
6 import java.util.InputMismatchException;
7 import java.util.Scanner;
8
9 public class db_mainWindow extends db{
10     public static ResultSet sqlEnquire(Connection conn,String Cname){
11         String Namesql = "select a.`name` from goods as a,goods_category as b,
                                category as c where a.id = b.Gid
                                and b.Cid = c.id and c.`name`=?";
12
13         PreparedStatement pstmt;
14         ResultSet rs = null;
15         try {
16             pstmt = conn.prepareStatement(Namesql);
17             pstmt.setString(1, Cname);
18             rs = pstmt.executeQuery();
19
20         } catch (Exception e) {
21             e.printStackTrace();
22         }
23         return rs;
24     }
25
26     public static ArrayList goods_name(ResultSet rs) throws SQLException {
27         ArrayList list = new ArrayList();
28         while(rs.next()){
29             list.add(rs.getString(1));
30         }
31         return list;
32     }
33
34     public static ResultSet pic_sqlEnquire(Connection conn,String Gname){
35         String Picsql = "select pic from goods where goods.name = ?";
36         PreparedStatement pstmt;
37         ResultSet rs = null;
38         try {
39             pstmt = conn.prepareStatement(Picsql);
40             pstmt.setString(1, Gname);
41             rs = pstmt.executeQuery();
42
43         } catch (Exception e) {
44             e.printStackTrace();
45         }
46
47         return rs;
48     }
49 }
```

```
50 public static ResultSet price_sqlEnquire(Connection conn,String Gname){
51     String Picsql = "select price from goods where goods.name = ?";
52     PreparedStatement pstmt;
53     ResultSet rs = null;
54     try {
55         pstmt = conn.prepareStatement(Picsql);
56         pstmt.setString(1, Gname);
57         rs = pstmt.executeQuery();
58
59     } catch (Exception e) {
60         e.printStackTrace();
61     }
62
63     return rs;
64 }
65
66 public static String pic(ResultSet rs)throws SQLException{
67     String pic="";
68     while(rs.next()){
69         pic = rs.getString(1);
70     }
71     return pic;
72 }
73 public static float price(ResultSet rs)throws SQLException{
74     float price = 0;
75     while(rs.next()){
76         price = Float.parseFloat(rs.getString(1));
77     }
78     return price;
79 }
80 }
81 ...
82
83 \subsubsection{ mainwindow.java(商店主界面)
84 ```java
85
86
87 import java.sql.*;
88 import java.text.DecimalFormat;
89 import java.util.ArrayList;
90 import java.util.InputMismatchException;
91 import java.util.Scanner;
92 import java.awt.*;
93 import java.awt.event.ActionEvent;
94 import java.awt.event.ActionListener;
95
96 import java.awt.event.*;
97 import javax.swing.event.*;
```

```
98
99 import javax.swing.*;
100 import javax.swing.border.EmptyBorder;
101 import javax.swing.border.LineBorder;
102
103 class mainWindow_info {
104     public static String[] getName(String categoryName) { // 初始商品名
105         ArrayList list = new ArrayList();
106         Connection conn = db.getConn();
107         // String category = (String) comboBox.getSelectedItemAt();
108         ResultSet rs = db_mainWindow.sqlEnquire(conn, categoryName);
109         try {
110             list = db_mainWindow.goods_name(rs);
111         } catch (SQLException e1) {
112             // TODO Auto-generated catch block
113             e1.printStackTrace();
114         }
115         int size = list.size();
116         String[] s = new String[size];
117         for (int i = 0; i < list.size(); i++) {
118             s[i] = (String) list.get(i);
119         }
120         return s;
121     }
122
123     public static String picRoute(String goodsName) { // 初始商品名
124         String pic = "";
125         Connection conn = db.getConn();
126         // String category = (String) comboBox.getSelectedItemAt();
127         ResultSet rs = db_mainWindow.pic_sqlEnquire(conn, goodsName);
128         try {
129             pic = db_mainWindow.pic(rs);
130         } catch (SQLException e1) {
131             // TODO Auto-generated catch block
132             e1.printStackTrace();
133         }
134
135         return pic;
136     }
137
138 }
139
140 public class mainWindow extends JFrame {
141     private JPanel contentPane;
142     private final JLabel lblNewLabel_2 = new JLabel("");
143     public static String goodsName;
144
145     String[] category = new String[] { "图书课本", "美妆用品", "手机数码", "衣物服饰" }
```

```

146         , "美食生鲜", "虚拟产品", "生活用品" };
147
148 public mainWindow() {
149     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
150     setBounds(100, 100, 548 * 2, 316 * 2);
151     contentPane = new JPanel();
152     contentPane.setBackground(Color.WHITE);
153     contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
154     setContentPane(contentPane);
155     contentPane.setLayout(null);
156
157     JButton buy = new JButton("点击购买");
158     buy.setBounds(614, 533, 125, 29);
159     buy.setBorder(null);
160     buy.setForeground(new Color(65, 105, 225));
161     buy.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
162     buy.addActionListener(new ActionListener() {
163
164         @Override
165         public void actionPerformed(ActionEvent e) {
166             buy frame = new buy();
167             frame.setVisible(true);
168             mainWindow.this.dispose();
169         }
170
171     });
172     contentPane.add(buy);
173
174     ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
175     image.setImage(image.getImage().getScaledInstance(159 * 2, 294 * 2, Image.
176         SCALE_DEFAULT));
177
178     JButton mypage = new JButton("前往我的主页");
179     mypage.setBounds(584, 40, 155, 29);
180     mypage.setForeground(new Color(65, 105, 225));
181     mypage.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
182     mypage.addActionListener(new ActionListener() {
183
184         public void actionPerformed(ActionEvent e) {
185             myPage frame_1 = new myPage();
186             frame_1.setVisible(true);
187             mainWindow.this.dispose();
188         }
189     });
190     contentPane.add(mypage);
191
```

```
192 JPanel panel = new JPanel();
193 panel.setBorder(new LineBorder(Color.LIGHT_GRAY));
194 panel.setBackground(Color.WHITE);
195 panel.setBounds(6, 145, 749, 459);
196 contentPane.add(panel);
197 panel.setLayout(null);
198
199 JLabel lblNewLabel = new JLabel("");
200 //ImageIcon img = new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg");
201 //img.setImage(img.getImage().getScaledInstance(448, 307, Image.SCALE_DEFAULT))
    ;
202 //lblNewLabel.setIcon(img);
203 lblNewLabel.setBorder(new LineBorder(Color.LIGHT_GRAY));
204 lblNewLabel.setBounds(284, 6, 448, 307);
205 panel.add(lblNewLabel);
206
207 JTextArea textArea = new JTextArea();
208 textArea.setBorder(new LineBorder(Color.LIGHT_GRAY));
209 textArea.setFont(new Font("Krungthep", Font.PLAIN, 52));
210 textArea.setBackground(Color.WHITE);
211 textArea.setText("0.00");
212 textArea.setBounds(400, 358, 229, 66);
213 textArea.setColumns(10);
214 panel.add(textArea);
215
216 JList list = new JList(mainWindow_info.getName("图书课本"));
217 list.setBorder(new LineBorder(Color.LIGHT_GRAY, 1, true));
218 list.setFont(new Font("Yuanti TC", Font.PLAIN, 16));
219 list.setBounds(6, 6, 200, 447);
220 list.addListSelectionListener(new ListSelectionListener() {
221
222     @Override
223     public void valueChanged(ListSelectionEvent e) {
224         mainWindow.goodsName = (String) list.getSelectedValue();
225         String pic = mainWindow_info.picRoute(mainWindow.goodsName);
226         //System.out.println(pic);
227         float price = 0;
228         try {
229             price = db_mainWindow.price(db_mainWindow.price_sqlEnquire(db.getConn(),
                goodsName));
230         } catch (SQLException e1) {
231             // TODO Auto-generated catch block
232             e1.printStackTrace();
233         }
234         ImageIcon image = new ImageIcon(pic);
235
236         image.setImage(image.getImage().getScaledInstance(448, 307, Image.
            SCALE_DEFAULT));
```

```
237
238     lblNewLabel.setIcon(image);
239
240     DecimalFormat df =new DecimalFormat("#0.00");
241     String result = df.format(price);
242     textArea.setText(result);
243 }
244
245 });
246 panel.add(list);
247
248 list.setBackground(Color.WHITE);
249
250 list.setFixedCellWidth(200);
251 list.setFixedCellHeight(50);
252
253 JLabel label = new JLabel("¥");
254 label.setFont(new Font("Lucida Grande", Font.BOLD, 30));
255 label.setBounds(368, 383, 61, 36);
256 panel.add(label);
257
258
259
260
261 lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
262 lblNewLabel_2.setBounds(413, 0, 683, 622);
263 contentPane.add(lblNewLabel_2);
264
265 JComboBox comboBox = new JComboBox(category);
266 comboBox.addItemListener(new ItemListener() {
267     @Override
268     public void itemStateChanged(ItemEvent e) {
269         if (e.getStateChange() == ItemEvent.SELECTED) {
270             // 选择的下拉框选项
271             String text=(String) comboBox.getSelectedItem();
272             String[] goodsName = mainWindow_info.getName(text);
273             list.setModel(new AbstractListModel() {
274                 String[] values = goodsName;
275                 public int getSize() {
276                     return values.length;
277                 }
278                 public Object getElementAt(int index) {
279                     return values[index];
280                 }
281             });
282         }
283     }
284 });
```

```
285     comboBox.setFont(new Font("Xingkai SC", Font.PLAIN, 20));
286     comboBox.setBounds(6, 21, 272, 83);
287     contentPane.add(comboBox);
288
289 }
290 }
```

#### 4.1.7 db\_myPage.java (用户主页所需数据)

```
1
2
3
4 import java.sql.*;
5 import java.util.InputMismatchException;
6 import java.util.Scanner;
7
8 public class db_myPage{
9     public static String balance(String account){
10         Connection conn = db.getConn();
11         String sql = "select balance from user where account = ?";
12         PreparedStatement pstmt;
13         String bal = "";
14         try {
15             pstmt = (PreparedStatement) conn.prepareStatement(sql);
16             pstmt.setString(1, account);
17             ResultSet rs = pstmt.executeQuery();
18             while(rs.next()){
19                 bal = rs.getString(1);
20             }
21             pstmt.close();
22             conn.close();
23         } catch (SQLException e) {
24             e.printStackTrace();
25         }
26
27         return bal;
28     }
29 }
```

#### 4.1.8 db\_myPage.java (用户主页界面)

```
1
2
3
4 import java.sql.*;
5 import java.util.InputMismatchException;
```

```
6 import java.util.Scanner;
7 import java.awt.*;
8 import java.awt.event.ActionEvent;
9 import java.awt.event.ActionListener;
10
11 import javax.swing.*;
12 import javax.swing.border.EmptyBorder;
13
14 public class myPage extends JFrame {
15
16     private JPanel contentPane;
17     private final JLabel lblNewLabel_2 = new JLabel("");
18
19     private JTextField textField;
20     private JTextField textField_1;
21     private JTextField textField_2;
22
23     public myPage() {
24         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
25         setBounds(100, 100, 548 * 2, 316 * 2);
26         contentPane = new JPanel();
27         contentPane.setBackground(Color.WHITE);
28         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
29         setContentPane(contentPane);
30         contentPane.setLayout(null);
31         JLabel lblNewLabel = new JLabel("");
32         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
33         image.setImage(image.getImage().getScaledInstance(159 * 2, 294 * 2, Image.
34                                     SCALE_DEFAULT));
35         lblNewLabel.setIcon(image);
36         lblNewLabel.setBounds(0, 0, 159 * 2, 294 * 2);
37         contentPane.add(lblNewLabel);
38         JLabel lblNewLabel_1 = new JLabel("");
39         lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png")
40                                     );
41         lblNewLabel_1.setBounds(317, 0, 84, 588);
42         contentPane.add(lblNewLabel_1);
43
44         JLabel label = new JLabel("用户名");
45         label.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
46         label.setBounds(476, 216, 99, 32);
47         contentPane.add(label);
48
49         JLabel label_1 = new JLabel("密码");
50         label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
51         label_1.setBounds(476, 279, 99, 32);
52         contentPane.add(label_1);
53     }
54 }
```



```
52     JLabel label_2 = new JLabel("余额");
53     label_2.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
54     label_2.setBounds(476, 346, 99, 32);
55     contentPane.add(label_2);
56
57     textField = new JTextField();
58     textField.setText(log.account);
59     textField.setBounds(608, 216, 180, 32);
60     contentPane.add(textField);
61     textField.setColumns(10);
62
63     textField_1 = new JTextField();
64     textField_1.setColumns(10);
65     textField_1.setText(log.psd);
66     textField_1.setBounds(608, 285, 180, 32);
67     contentPane.add(textField_1);
68
69     JButton alter = new JButton("修改密码");
70     alter.addActionListener(new ActionListener() {
71         public void actionPerformed(ActionEvent e) {
72             alterPsd frame = new alterPsd();
73             frame.setVisible(true);
74             myPage.this.dispose();
75         }
76     });
77     alter.setBounds(800, 285, 104, 32);
78     alter.setForeground(new Color(65, 105, 225));
79     alter.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
80     contentPane.add(alter);
81
82     JButton add = new JButton("充值");
83     add.addActionListener(new ActionListener() {
84         public void actionPerformed(ActionEvent e) {
85             charge frame = new charge();
86             frame.setVisible(true);
87             myPage.this.dispose();
88         }
89     });
90     add.setBounds(800, 346, 104, 32);
91     add.setForeground(new Color(65, 105, 225));
92     add.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
93     contentPane.add(add);
94
95     JButton button = new JButton("前往商城");
96     button.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
97     button.setBackground(new Color(30, 144, 255));
98     button.setBounds(560, 408, 171, 41);
```

```
100     button.setOpaque(true);
101     button.setBorderPainted(false);
102     button.addActionListener(new ActionListener() {
103         public void actionPerformed(ActionEvent e) {
104             mainWindow frame_1 = new mainWindow();
105             frame_1.setVisible(true);
106             myPage.this.dispose();
107
108         }
109     });
110     contentPane.add(button);
111
112     textField_2 = new JTextField();
113     textField_2.setText(db_myPage.balance(log.account));
114     textField_2.setColumns(10);
115     textField_2.setBounds(608, 346, 180, 32);
116     contentPane.add(textField_2);
117
118     JLabel label_3 = new JLabel("我的主页");
119     label_3.setFont(new Font("Xingkai TC", Font.PLAIN, 30));
120     label_3.setBounds(576, 81, 155, 54);
121     contentPane.add(label_3);
122
123     JButton button_2 = new JButton("我的在售");
124     button_2.setOpaque(true);
125     button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
126     button_2.setBorderPainted(false);
127     button_2.setBackground(new Color(30, 144, 255));
128     button_2.setBounds(560, 474, 171, 41);
129     button_2.addActionListener(new ActionListener() {
130         public void actionPerformed(ActionEvent e) {
131             mySale frame_1 = new mySale();
132             frame_1.setVisible(true);
133             myPage.this.dispose();
134
135         }
136     });
137     contentPane.add(button_2);
138
139     JButton button_3 = new JButton("交易记录");
140     button_3.setOpaque(true);
141     button_3.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
142     button_3.setBorderPainted(false);
143     button_3.setBackground(new Color(30, 144, 255));
144     button_3.setBounds(560, 536, 171, 41);
145     button_3.addActionListener(new ActionListener() {
146         public void actionPerformed(ActionEvent e) {
147             record frame_1 = new record();
```

```

148         frame_1.setVisible(true);
149         myPage.this.dispose();
150
151     }
152 });
153 contentPane.add(button_3);
154
155 JButton button_1 = new JButton("注销");
156 button_1.addActionListener(new ActionListener() {
157     public void actionPerformed(ActionEvent e) {
158         log frame = new log();
159         frame.setVisible(true);
160         myPage.this.dispose();
161     }
162 });
163 button_1.setForeground(new Color(65, 105, 225));
164 button_1.setFont(new Font("Xingkai TC", Font.PLAIN, 16));
165 button_1.setBounds(800, 216, 104, 32);
166 contentPane.add(button_1);
167
168
169 lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg")
170                                     );
171 lblNewLabel_2.setBounds(413, 0, 683, 622);
172 contentPane.add(lblNewLabel_2);
173
174 }

```

#### 4.1.9 db\_mysale(我的在售界面所需数据)

```

1
2
3
4
5 import java.awt.List;
6 import java.sql.*;
7 import java.util.ArrayList;
8 import java.util.InputMismatchException;
9 import java.util.Scanner;
10
11 public class db_mySale {
12     public static String[] sale(String account) {
13         Connection conn = db.getConn();
14         ArrayList<String> list = new ArrayList<String>();
15         String sql = "select goods.name from goods,user_goods where goods.id =
16                                     user_goods.Gid and user_goods.

```

```
account = ?";

16     PreparedStatement pstmt;
17     try {
18         pstmt = (PreparedStatement) conn.prepareStatement(sql);
19         pstmt.setString(1, account);
20
21         ResultSet rs = pstmt.executeQuery();
22         while (rs.next()) {
23             list.add(rs.getString(1));
24
25         }
26         pstmt.close();
27         conn.close();
28     } catch (SQLException e) {
29         e.printStackTrace();
30     }
31
32     String[] arr = new String[list.size()]; // 创建一个和list长度一样的数组
33     for (int i = 0; i < list.size(); i++) {
34         arr[i] = list.get(i); // 数组赋值了。
35     }
36
37     return arr;
38 }
39 }
```

#### 4.1.10 mysale(我的在售界面)

```
1
2
3
4
5 import java.awt.BorderLayout;
6 import java.awt.EventQueue;
7
8 import javax.swing.JFrame;
9 import javax.swing.JPanel;
10 import javax.swing.border.EmptyBorder;
11 import java.sql.*;
12 import java.util.InputMismatchException;
13 import java.util.Scanner;
14 import java.awt.*;
15 import java.awt.event.ActionEvent;
16 import java.awt.event.ActionListener;
17
18 import javax.swing.*;
19 import javax.swing.border.EmptyBorder;
```

```
20
21 import javax.swing.border.BevelBorder;
22 import javax.swing.border.LineBorder;
23
24 public class mySale extends JFrame {
25
26     private JPanel contentPane;
27     private final JLabel lblNewLabel_2 = new JLabel("");
28
29
30     public mySale() {
31         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
32         setBounds(100, 100, 548*2, 316*2);
33         contentPane = new JPanel();
34         contentPane.setBackground(Color.WHITE);
35         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
36         setContentPane(contentPane);
37         contentPane.setLayout(null);
38         JLabel lblNewLabel = new JLabel("");
39         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
40         image.setImage(image.getImage().getScaledInstance(159*2, 294*2, Image.
            SCALE_DEFAULT));
41         lblNewLabel.setIcon(image);
42         lblNewLabel.setBounds(0, 0, 159*2, 294*2);
43         contentPane.add(lblNewLabel);
44         JLabel lblNewLabel_1 = new JLabel("");
45         lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
46         lblNewLabel_1.setBounds(317, 0, 84, 588);
47         contentPane.add(lblNewLabel_1);
48
49         JButton button = new JButton("前往商城");
50         button.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
51         button.setBackground(new Color(30, 144, 255));
52         button.setBounds(560, 449, 171, 41);
53         button.setOpaque(true);
54         button.setBorderPainted(false);
55         button.addActionListener(new ActionListener() {
56             public void actionPerformed(ActionEvent e) {
57                 mainWindow frame_1 = new mainWindow();
58                 frame_1.setVisible(true);
59                 mySale.this.dispose();
60
61             }
62         });
63         contentPane.add(button);
64
65         JButton button_2 = new JButton("返回主页");
66         button_2.setOpaque(true);
```

```
67 button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
68 button_2.setBorderPainted(false);
69 button_2.setBackground(new Color(30, 144, 255));
70 button_2.setBounds(560, 518, 171, 41);
71 button_2.addActionListener(new ActionListener() {
72     public void actionPerformed(ActionEvent e) {
73         myPage frame_1 = new myPage();
74         frame_1.setVisible(true);
75         mySale.this.dispose();
76
77     }
78 });
79 contentPane.add(button_2);
80
81 JButton add = new JButton("发布");
82 add.setBounds(723, 33, 155, 29);
83 add.setForeground(new Color(65, 105, 225));
84 add.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
85 add.addActionListener(new ActionListener() {
86     public void actionPerformed(ActionEvent e) {
87         addGoods frame_1 = new addGoods();
88         frame_1.setVisible(true);
89         mySale.this.dispose();
90
91     }
92 });
93 contentPane.add(add);
94
95 JList list = new JList();
96 list.setModel(new AbstractListModel() {
97     String[] values = db_mySale.sale(log.account);
98     public int getSize() {
99         return values.length;
100     }
101     public Object getElementAt(int index) {
102         return values[index];
103     }
104 });
105 list.setBounds(479, 74, 348, 342);
106 list.setBorder(new LineBorder(Color.LIGHT_GRAY, 1, true));
107 list.setFont(new Font("Yuanti TC", Font.PLAIN, 16));
108 list.setFixedCellWidth(200);
109 list.setFixedCellHeight(50);
110 contentPane.add(list);
111
112 lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
113 lblNewLabel_2.setBounds(413, 0, 683, 622);
114 contentPane.add(lblNewLabel_2);
```

```
115
116     }
117 }
```

#### 4.1.11 db\_addGoods.java(发布商品界面)

```
1
2
3
4 import java.sql.*;
5 import java.util.InputMismatchException;
6 import java.util.Scanner;
7
8 public class db_addGoods {
9     public static void upload(String name,int price,String describe,String pic,
10                               String category){
11         Connection conn = db.getConn();
12         String sql1 = "insert into goods values(?,?,?,?,?)";
13         String sql2 = "select count(*) from goods";
14         String sql3 = "insert into user_goods values(?,?)";
15         String sql4 = "insert into goods_category values(?,?)";
16         PreparedStatement pstmt1;
17         int i = 0; //已有商品数目
18         int cid = 0;
19         try {
20             pstmt1 = (PreparedStatement) conn.prepareStatement(sql2);
21             ResultSet rs = pstmt1.executeQuery();
22             while(rs.next()){
23                 String n = rs.getString(1);
24                 i=Integer.parseInt(n);
25             }
26             pstmt1.close();
27         } catch (SQLException e) {
28             e.printStackTrace();
29         }
30
31         String id = i+1+"";
32         PreparedStatement pstmt;
33         try {
34             pstmt = (PreparedStatement) conn.prepareStatement(sql1);
35             pstmt.setString(1, id);
36             pstmt.setString(2, name);
37             pstmt.setInt(3, price);
38             pstmt.setString(4, describe);
39             pstmt.setString(5, pic);
40             pstmt.executeUpdate();
```

```
41         pstmt.close();
42
43     } catch (SQLException e) {
44         e.printStackTrace();
45     }
46
47
48     PreparedStatement pstmt2;
49     try {
50         pstmt2 = (PreparedStatement) conn.prepareStatement(sql3);
51         pstmt2.setString(1, log.account);
52         pstmt2.setString(2, id);
53         pstmt2.executeUpdate();
54         pstmt2.close();
55
56     } catch (SQLException e) {
57         e.printStackTrace();
58     }
59
60     switch (category) {
61         case "图书课本":
62             cid = 1;
63             break;
64         case "美妆用品":
65             cid = 2;
66             break;
67         case "手机数码":
68             cid = 3;
69             break;
70         case "衣物服饰":
71             cid = 4;
72             break;
73         case "美食生鲜":
74             cid = 5;
75             break;
76         case "虚拟产品":
77             cid = 6;
78             break;
79         case "生活用品":
80             cid = 7;
81             break;
82     }
83
84     PreparedStatement pstmt3;
85     try {
86         pstmt3 = (PreparedStatement) conn.prepareStatement(sql4);
87         pstmt3.setString(1, id);
88         pstmt3.setInt(2, cid);
```



```
89         pstmt3.executeUpdate();
90         pstmt3.close();
91         conn.close();
92     } catch (SQLException e) {
93         e.printStackTrace();
94     }
95 }
96 }
```

#### 4.1.12 addGoods.java(发布商品界面)

```
1
2
3
4
5 import java.sql.*;
6 import java.util.InputMismatchException;
7 import java.util.Scanner;
8 import java.awt.*;
9 import java.awt.event.ActionEvent;
10 import java.awt.event.ActionListener;
11 import java.io.File;
12
13 import javax.swing.*;
14 import javax.swing.border.EmptyBorder;
15
16 public class addGoods extends JFrame {
17
18     private JPanel contentPane;
19     private final JLabel lblNewLabel_2 = new JLabel("");
20     private JTextField textField;
21     private JTextField textField_1;
22     private JTextField textField_2;
23     private JTextField textField_3;
24
25     String[] cate = new String[] { "图书课本", "美妆用品", "手机数码", "衣物服饰",
26                                     "美食生鲜", "虚拟产品", "生活用品" };
27
28     private File openFile;
29
30     public addGoods() {
31         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
32         setBounds(100, 100, 548*2, 316*2);
33         contentPane = new JPanel();
34         contentPane.setBackground(Color.WHITE);
35         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
```

```
36     setContentPane(contentPane);
37     contentPane.setLayout(null);
38     JLabel lblNewLabel = new JLabel("");
39     ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
40     image.setImage(image.getImage().getScaledInstance(159*2, 294*2, Image.
        SCALE_DEFAULT));
41     lblNewLabel.setIcon(image);
42     lblNewLabel.setBounds(0, 0, 159*2, 294*2);
43     contentPane.add(lblNewLabel);
44     JLabel lblNewLabel_1 = new JLabel("");
45     lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
46     lblNewLabel_1.setBounds(317, 0, 84, 588);
47     contentPane.add(lblNewLabel_1);
48
49
50
51     JLabel label = new JLabel("商品名");
52     label.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
53     label.setBounds(506, 156, 95, 41);
54     contentPane.add(label);
55
56     JLabel label_1 = new JLabel("价格");
57     label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
58     label_1.setBounds(506, 223, 95, 41);
59     contentPane.add(label_1);
60
61     JLabel label_2 = new JLabel("商品说明");
62     label_2.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
63     label_2.setBounds(506, 291, 107, 41);
64     contentPane.add(label_2);
65
66     JLabel label_3 = new JLabel("图片");
67     label_3.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
68     label_3.setBounds(506, 357, 95, 41);
69     contentPane.add(label_3);
70
71     textField = new JTextField();
72     textField.setBounds(632, 156, 196, 36);
73     contentPane.add(textField);
74     textField.setColumns(10);
75
76     textField_1 = new JTextField();
77     textField_1.setColumns(10);
78     textField_1.setBounds(632, 228, 196, 36);
79     contentPane.add(textField_1);
80
81     textField_2 = new JTextField();
82     textField_2.setColumns(10);
```

```
83     textField_2.setBounds(632, 296, 196, 36);
84     contentPane.add(textField_2);
85
86     textField_3 = new JTextField();
87     textField_3.setColumns(10);
88     textField_3.setBounds(632, 367, 196, 36);
89     contentPane.add(textField_3);
90     lblNewLabel_2.setFont(new Font("Lucida Grande", Font.PLAIN, 20));
91
92
93     JLabel label_4 = new JLabel("商品分类");
94     label_4.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
95     label_4.setBounds(506, 98, 95, 41);
96     contentPane.add(label_4);
97
98
99     JComboBox comboBox = new JComboBox(cate);
100    comboBox.setBounds(632, 109, 196, 27);
101    contentPane.add(comboBox);
102    lblNewLabel_2.setFont(new Font("Lucida Grande", Font.PLAIN, 20));
103
104
105
106    JButton btnNewButton = new JButton("浏览");
107    btnNewButton.setFont(new Font("Lucida Grande", Font.PLAIN, 12));
108    btnNewButton.setBounds(830, 369, 36, 29);
109    btnNewButton.addActionListener(new ActionListener() {
110        public void actionPerformed(ActionEvent e) {
111
112            JFileChooser chooser = new JFileChooser(); // 文件选择
113            chooser.showOpenDialog(chooser);          // 打开文件选择窗
114            openFile = chooser.getSelectedFile();      // 获取选择的文件
115            textField_3.setText(openFile.getPath());  // 获取选择文件的路径
116
117        }
118    });
119    contentPane.add(btnNewButton);
120
121
122    JButton button_2 = new JButton("上传");
123    button_2.setOpaque(true);
124    button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
125    button_2.setBorderPainted(false);
126    button_2.setBackground(new Color(30, 144, 255));
127    button_2.setBounds(565, 456, 171, 41);
128    button_2.addActionListener(new ActionListener() {
129        public void actionPerformed(ActionEvent e) {
```

```
130         String name = textField.getText();
131         int price = Integer.parseInt(textField_1.getText());
132         String describe = textField_2.getText();
133         String pic = textField_3.getText();
134         String category = (String)comboBox.getSelectedItemAt();
135         db_addGoods.upload(name, price, describe, pic, category);
136         mySale frame_1 = new mySale();
137         frame_1.setVisible(true);
138         addGoods.this.dispose();
139     }
140 }));
141 contentPane.add(button_2);
142
143 JButton button = new JButton("返回");
144 button.setBounds(723, 33, 155, 29);
145 button.setForeground(new Color(65, 105, 225));
146 button.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
147 button.addActionListener(new ActionListener() {
148     public void actionPerformed(ActionEvent e) {
149         mySale frame_1 = new mySale();
150         frame_1.setVisible(true);
151         addGoods.this.dispose();
152     }
153 });
154 contentPane.add(button);
155
156
157
158
159
160 lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
161 lblNewLabel_2.setBounds(413, 0, 683, 622);
162 contentPane.add(lblNewLabel_2);
163
164 }
165 }
```

#### 4.1.13 db\_charge.java(余额充值界面所需数据)

```
1
2
3
4
5 import java.awt.BorderLayout;
6 import java.awt.EventQueue;
7
8 import javax.swing.JFrame;
```

```
9 import javax.swing.JPanel;
10 import javax.swing.border.EmptyBorder;
11 import java.sql.*;
12 import java.util.InputMismatchException;
13 import java.util.Scanner;
14 import java.awt.*;
15 import java.awt.event.ActionEvent;
16 import java.awt.event.ActionListener;
17
18 import javax.swing.*;
19 import javax.swing.border.EmptyBorder;
20
21 import javax.swing.border.BevelBorder;
22 import javax.swing.border.LineBorder;
23
24 public class charge extends JFrame {
25
26     private JPanel contentPane;
27     private final JLabel lblNewLabel_2 = new JLabel("");
28     private JTextField textField_1;
29
30
31
32     public charge(){
33         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
34         setBounds(100, 100, 548*2, 316*2);
35         contentPane = new JPanel();
36         contentPane.setBackground(Color.WHITE);
37         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
38         setContentPane(contentPane);
39         contentPane.setLayout(null);
40         JLabel lblNewLabel = new JLabel("");
41         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
42         image.setImage(image.getImage().getScaledInstance(159*2, 294*2, Image.
            SCALE_DEFAULT));
43         lblNewLabel.setIcon(image);
44         lblNewLabel.setBounds(0, 0, 159*2, 294*2);
45         contentPane.add(lblNewLabel);
46         JLabel lblNewLabel_1 = new JLabel("");
47         lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
48         lblNewLabel_1.setBounds(317, 0, 84, 588);
49         contentPane.add(lblNewLabel_1);
50
51
52
53         JLabel label_1 = new JLabel("充值金额");
54         label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
55         label_1.setBounds(506, 223, 95, 41);
```

```
56     contentPane.add(label_1);
57
58     textField_1 = new JTextField();
59     textField_1.setColumns(10);
60     textField_1.setBounds(632, 228, 196, 36);
61     contentPane.add(textField_1);
62     lblNewLabel_2.setFont(new Font("Lucida Grande", Font.PLAIN, 20));
63
64     JButton button_2 = new JButton("完成");
65     button_2.setOpaque(true);
66     button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
67     button_2.setBorderPainted(false);
68     button_2.setBackground(new Color(30, 144, 255));
69     button_2.setBounds(560, 327, 171, 41);
70     button_2.addActionListener(new ActionListener() {
71     public void actionPerformed(ActionEvent e) {
72         db_charge.alter(Integer.parseInt(textField_1.getText()));
73         myPage frame = new myPage();
74         frame.setVisible(true);
75         charge.this.dispose();
76     }
77 });
78     contentPane.add(button_2);
79
80     lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
81     lblNewLabel_2.setBounds(413, 0, 683, 622);
82     contentPane.add(lblNewLabel_2);
83
84 }
85 }
```

#### 4.1.14 charge.java (余额充值界面)

```
1
2
3
4
5 import java.awt.BorderLayout;
6 import java.awt.EventQueue;
7
8 import javax.swing.JFrame;
9 import javax.swing.JPanel;
10 import javax.swing.border.EmptyBorder;
11 import java.sql.*;
12 import java.util.InputMismatchException;
13 import java.util.Scanner;
14 import java.awt.*;
```

```
15 import java.awt.event.ActionEvent;
16 import java.awt.event.ActionListener;
17
18 import javax.swing.*;
19 import javax.swing.border.EmptyBorder;
20
21 import javax.swing.border.BevelBorder;
22 import javax.swing.border.LineBorder;
23
24 public class charge extends JFrame {
25
26     private JPanel contentPane;
27     private final JLabel lblNewLabel_2 = new JLabel("");
28     private JTextField textField_1;
29
30
31
32     public charge(){
33         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
34         setBounds(100, 100, 548*2, 316*2);
35         contentPane = new JPanel();
36         contentPane.setBackground(Color.WHITE);
37         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
38         setContentPane(contentPane);
39         contentPane.setLayout(null);
40         JLabel lblNewLabel = new JLabel("");
41         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
42         image.setImage(image.getImage().getScaledInstance(159*2, 294*2, Image.
            SCALE_DEFAULT));
43         lblNewLabel.setIcon(image);
44         lblNewLabel.setBounds(0, 0, 159*2, 294*2);
45         contentPane.add(lblNewLabel);
46         JLabel lblNewLabel_1 = new JLabel("");
47         lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
48         lblNewLabel_1.setBounds(317, 0, 84, 588);
49         contentPane.add(lblNewLabel_1);
50
51
52
53         JLabel label_1 = new JLabel("充值金额");
54         label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
55         label_1.setBounds(506, 223, 95, 41);
56         contentPane.add(label_1);
57
58         textField_1 = new JTextField();
59         textField_1.setColumns(10);
60         textField_1.setBounds(632, 228, 196, 36);
61         contentPane.add(textField_1);
```

```

62    lblNewLabel_2.setFont(new Font("Lucida Grande", Font.PLAIN, 20));
63
64    JButton button_2 = new JButton("完成");
65    button_2.setOpaque(true);
66    button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
67    button_2.setBorderPainted(false);
68    button_2.setBackground(new Color(30, 144, 255));
69    button_2.setBounds(560, 327, 171, 41);
70    button_2.addActionListener(new ActionListener() {
71    public void actionPerformed(ActionEvent e) {
72        db_charge.alter(Integer.parseInt(textField_1.getText()));
73        myPage frame = new myPage();
74        frame.setVisible(true);
75        charge.this.dispose();
76    }
77    });
78    contentPane.add(button_2);
79
80    lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
81    lblNewLabel_2.setBounds(413, 0, 683, 622);
82    contentPane.add(lblNewLabel_2);
83
84    }
85    }

```

#### 4.1.15 db\_buy.java(购买界面所需数据)

```

1
2
3
4  import java.sql.*;
5  import java.util.InputMismatchException;
6  import java.util.Scanner;
7
8  public class db_buy{
9      public static String des(String name){
10          Connection conn = db.getConn();
11          String sql1 = "select goods.describe from goods where goods.name=?";
12          PreparedStatement pstmt;
13          String describe = "";
14          try {
15              pstmt = (PreparedStatement) conn.prepareStatement(sql1);
16              pstmt.setString(1, name);
17              ResultSet rs = pstmt.executeQuery();
18              while(rs.next()){
19                  describe = rs.getString(1);
20              }

```



```
21         pstmt.close();
22         conn.close();
23     } catch (SQLException e) {
24         e.printStackTrace();
25     }
26
27     return describe;
28 }
29
30 public static float price(){
31     Connection conn = db.getConn();
32     String sql1 = "select goods.price from goods where goods.name=?";
33     PreparedStatement pstmt;
34     float price = 0;
35     try {
36         pstmt = (PreparedStatement) conn.prepareStatement(sql1);
37         pstmt.setString(1, mainWindow.goodsName);
38         ResultSet rs = pstmt.executeQuery();
39         while(rs.next()){
40             price = rs.getFloat(1);
41         }
42         pstmt.close();
43         conn.close();
44     } catch (SQLException e) {
45         e.printStackTrace();
46     }
47     return price*buy.num;
48 }
49
50 public static String seller(){
51     Connection conn = db.getConn();
52     String sql1 = "select user.account from user,user_goods,goods where user.
                    account = user_goods.account and
                    goods.id = user_goods.Gid and
                    goods.name = ?";
53
54     PreparedStatement pstmt;
55     String seller = "";
56     try {
57         pstmt = (PreparedStatement) conn.prepareStatement(sql1);
58         pstmt.setString(1, mainWindow.goodsName);
59         ResultSet rs = pstmt.executeQuery();
60         while(rs.next()){
61             seller = rs.getString(1);
62         }
63         pstmt.close();
64         conn.close();
65     } catch (SQLException e) {
66         e.printStackTrace();
67     }
```

```
66     }
67     return seller;
68 }
69
70 public static boolean testBal(){
71     Connection conn = db.getConn();
72     String sql = "select balance from user where user.account = ?";
73     PreparedStatement pstmt;
74     float bal = 0;
75     try {
76         pstmt = (PreparedStatement) conn.prepareStatement(sql);
77         pstmt.setString(1, log.account);
78         ResultSet rs = pstmt.executeQuery();
79         while(rs.next()){
80             bal = rs.getFloat(1);
81         }
82         pstmt.close();
83         conn.close();
84     } catch (SQLException e) {
85         e.printStackTrace();
86     }
87     return(bal>=price());
88 }
89
90
91 public static void updateRecord(){
92     String sql ="insert record values(?,?,?)";
93     PreparedStatement pstmt;
94     Connection conn = db.getConn();
95     System.out.println(seller());
96     try {
97         pstmt = (PreparedStatement) conn.prepareStatement(sql);
98
99         pstmt.setString(1, seller());
100        pstmt.setString(2, log.account);
101        pstmt.setString(3, mainWindow.goodsName);
102
103        pstmt.executeUpdate();
104        pstmt.close();
105        conn.close();
106    } catch (SQLException e) {
107        e.printStackTrace();
108    }
109 }
110
111 public static void balChange(){
112     Connection conn = db.getConn();
113     String sql1 = "update user set balance = balance - ? where user.account = ?";
```

```
114         String sql2 = "update user set balance = balance + ? where user.account = ?";
115         PreparedStatement pstmt;
116         try {
117             pstmt = (PreparedStatement) conn.prepareStatement(sql1);
118             pstmt.setFloat(1, price());
119             pstmt.setString(2, log.account);
120
121             pstmt.executeUpdate();
122             pstmt.close();
123
124         } catch (SQLException e) {
125             e.printStackTrace();
126         }
127
128         PreparedStatement pstmt1;
129         try {
130             pstmt1 = (PreparedStatement) conn.prepareStatement(sql2);
131             pstmt1.setFloat(1, price());
132             pstmt1.setString(2, seller());
133             pstmt1.executeUpdate();
134             pstmt1.close();
135             conn.close();
136         } catch (SQLException e) {
137             e.printStackTrace();
138         }
139     }
140
141     public static void pulloff(){
142         String sql ="delete from goods where name = ?";
143         PreparedStatement pstmt;
144         Connection conn = db.getConn();
145         try {
146             pstmt = (PreparedStatement) conn.prepareStatement(sql);
147
148             pstmt.setString(1, mainWindow.goodsName);
149
150             pstmt.executeUpdate();
151             pstmt.close();
152             conn.close();
153         } catch (SQLException e) {
154             e.printStackTrace();
155         }
156     }
157 }
```

## 4.1.16 buy.java (购买界面)

```
1
2
3
4
5 import java.awt.BorderLayout;
6 import java.awt.EventQueue;
7
8 import javax.swing.JFrame;
9 import javax.swing.JPanel;
10 import javax.swing.border.EmptyBorder;
11 import java.sql.*;
12 import java.util.InputMismatchException;
13 import java.util.Scanner;
14 import java.awt.*;
15 import java.awt.event.ActionEvent;
16 import java.awt.event.ActionListener;
17
18 import javax.swing.*;
19 import javax.swing.border.EmptyBorder;
20
21 import javax.swing.border.BevelBorder;
22 import javax.swing.border.LineBorder;
23 import javax.swing.event.DocumentEvent;
24
25 public class buy extends JFrame {
26
27     private JPanel contentPane;
28     private final JLabel lblNewLabel_2 = new JLabel("");
29     private JTextField textField;
30     private JTextField textField_1;
31     public static int num;
32
33     public buy() {
34         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
35         setBounds(100, 100, 548 * 2, 316 * 2);
36         contentPane = new JPanel();
37         contentPane.setBackground(Color.WHITE);
38         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
39         setContentPane(contentPane);
40         contentPane.setLayout(null);
41         JLabel lblNewLabel = new JLabel("");
42         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
43         image.setImage(image.getImage().getScaledInstance(159 * 2, 294 * 2, Image.
44                                     SCALE_DEFAULT));
45         lblNewLabel.setIcon(image);
46         lblNewLabel.setBounds(0, 0, 159 * 2, 294 * 2);
```

```
46 contentPane.add(lblNewLabel);
47 JLabel lblNewLabel_1 = new JLabel("");
48 lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
49 lblNewLabel_1.setBounds(317, 0, 84, 588);
50 contentPane.add(lblNewLabel_1);
51
52 JLabel label = new JLabel("购买数量");
53 label.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
54 label.setBounds(472, 318, 95, 41);
55 contentPane.add(label);
56
57 JLabel label_2 = new JLabel("总价");
58 label_2.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
59 label_2.setBounds(741, 318, 95, 41);
60 contentPane.add(label_2);
61
62 JTextField_1 = new JTextField();
63 textField_1.setColumns(10);
64 textField_1.setBounds(799, 325, 73, 33);
65
66 contentPane.add(textField_1);
67
68 textField = new JTextField();
69 textField.setBounds(579, 325, 73, 33);
70 textField.setText("0");
71 textField.getDocument().addDocumentListener(new javax.swing.event.
    DocumentListener() {
72
73     public void changedUpdate(DocumentEvent e) {
74
75     }
76
77     @Override
78     public void insertUpdate(DocumentEvent e) {
79         String number = textField.getText();
80         buy.num = Integer.parseInt(number);
81         textField_1.setText(db_buy.price() + "");
82         // System.out.println(num);
83         // System.out.println(db_buy.price());
84
85     }
86
87     @Override
88     public void removeUpdate(DocumentEvent e) {
89
90     }
91
92 });
```

```
93     contentPane.add(textField);
94     textField.setColumns(10);
95
96     JButton button_2 = new JButton("确认购买");
97     button_2.setOpaque(true);
98     button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
99     button_2.setBorderPainted(false);
100    button_2.setBackground(new Color(30, 144, 255));
101    button_2.setBounds(518, 390, 171, 41);
102    button_2.addActionListener(new ActionListener() {
103        public void actionPerformed(ActionEvent e) {
104            if (db_buy.testBal()) {
105                db_buy.updateRecord();
106                db_buy.balChange();
107                db_buy.pullOff();
108                JOptionPane.showMessageDialog(null, "", "购买成功", JOptionPane.
                    INFORMATION_MESSAGE);
109                mainWindow frame = new mainWindow();
110                frame.setVisible(true);
111                buy.this.dispose();
112            } else {
113                JOptionPane.showMessageDialog(null, "ERROR", "余额不足, 请充值",
                    JOptionPane.ERROR_MESSAGE);
114            }
115        }
116    });
117    contentPane.add(button_2);
118
119
120    JLabel label_1 = new JLabel("商品描述");
121    label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
122    label_1.setBounds(438, 78, 95, 41);
123    contentPane.add(label_1);
124
125    JTextArea textArea = new JTextArea();
126    textArea.setBounds(562, 93, 318, 200);
127    textArea.setBorder(new LineBorder(new Color(211, 211, 211), 1, true));
128    textArea.setText(db_buy.des(mainWindow.goodsName));
129    contentPane.add(textArea);
130
131    JButton button = new JButton("返回");
132    button.addActionListener(new ActionListener() {
133        public void actionPerformed(ActionEvent e) {
134            mainWindow frame = new mainWindow();
135            frame.setVisible(true);
136            buy.this.dispose();
137        }
138    });
```

```
139 button.setOpaque(true);
140 button.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
141 button.setBorderPainted(false);
142 button.setBackground(new Color(30, 144, 255));
143 button.setBounds(518, 454, 171, 41);
144 contentPane.add(button);
145
146 lblNewLabel_2.setFont(new Font("Lucida Grande", Font.PLAIN, 20));
147
148 lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
149 lblNewLabel_2.setBounds(413, 0, 683, 622);
150 contentPane.add(lblNewLabel_2);
151
152 }
153 }
```

#### 4.1.17 db\_alterPsd.java(修改密码界面)

```
1
2
3
4 import java.sql.*;
5 import java.util.InputMismatchException;
6 import java.util.Scanner;
7
8 public class db_addGoods {
9     public static void upload(String name,int price,String describe,String pic,
10                               String category){
11         Connection conn = db.getConn();
12         String sql1 = "insert into goods values(?,?,?,?,?)";
13         String sql2 = "select count(*) from goods";
14         String sql3 = "insert into user_goods values(?,?)";
15         String sql4 = "insert into goods_category values(?,?)";
16         PreparedStatement pstmt1;
17         int i = 0; //已有商品数目
18         int cid = 0;
19         try {
20             pstmt1 = (PreparedStatement) conn.prepareStatement(sql2);
21             ResultSet rs = pstmt1.executeQuery();
22             while(rs.next()){
23                 String n = rs.getString(1);
24                 i=Integer.parseInt(n);
25             }
26             pstmt1.close();
27         } catch (SQLException e) {
28             e.printStackTrace();
29         }
```

```
29     }
30
31     String id = i+1+"";
32     PreparedStatement pstmt;
33     try {
34         pstmt = (PreparedStatement) conn.prepareStatement(sql1);
35         pstmt.setString(1, id);
36         pstmt.setString(2, name);
37         pstmt.setInt(3, price);
38         pstmt.setString(4, describe);
39         pstmt.setString(5, pic);
40         pstmt.executeUpdate();
41         pstmt.close();
42
43     } catch (SQLException e) {
44         e.printStackTrace();
45     }
46
47
48     PreparedStatement pstmt2;
49     try {
50         pstmt2 = (PreparedStatement) conn.prepareStatement(sql3);
51         pstmt2.setString(1, log.account);
52         pstmt2.setString(2, id);
53         pstmt2.executeUpdate();
54         pstmt2.close();
55
56     } catch (SQLException e) {
57         e.printStackTrace();
58     }
59
60     switch (category) {
61         case "图书课本":
62             cid = 1;
63             break;
64         case "美妆用品":
65             cid = 2;
66             break;
67         case "手机数码":
68             cid = 3;
69             break;
70         case "衣物服饰":
71             cid = 4;
72             break;
73         case "美食生鲜":
74             cid = 5;
75             break;
76         case "虚拟产品":
```



```
77         cid = 6;
78         break;
79     case "生活用品":
80         cid = 7;
81         break;
82     }
83
84     PreparedStatement pstmt3;
85     try {
86         pstmt3 = (PreparedStatement) conn.prepareStatement(sql4);
87         pstmt3.setString(1, id);
88         pstmt3.setInt(2, cid);
89         pstmt3.executeUpdate();
90         pstmt3.close();
91         conn.close();
92     } catch (SQLException e) {
93         e.printStackTrace();
94     }
95 }
96 }
```

#### 4.1.18 alterPsd(修改密码界面)

```
1
2
3
4
5 import java.awt.BorderLayout;
6 import java.awt.EventQueue;
7
8 import javax.swing.JFrame;
9 import javax.swing.JPanel;
10 import javax.swing.border.EmptyBorder;
11 import java.sql.*;
12 import java.util.InputMismatchException;
13 import java.util.Scanner;
14 import java.awt.*;
15 import java.awt.event.ActionEvent;
16 import java.awt.event.ActionListener;
17
18 import javax.swing.*;
19 import javax.swing.border.EmptyBorder;
20
21 import javax.swing.border.BevelBorder;
22 import javax.swing.border.LineBorder;
23
24 public class alterPsd extends JFrame {
```

```
25
26 private JPanel contentPane;
27     private final JLabel lblNewLabel_2 = new JLabel("");
28     private JTextField textField_1;
29
30
31
32 public alterPsd(){
33     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
34     setBounds(100, 100, 548*2, 316*2);
35     contentPane = new JPanel();
36     contentPane.setBackground(Color.WHITE);
37     contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
38     setContentPane(contentPane);
39     contentPane.setLayout(null);
40     JLabel lblNewLabel = new JLabel("");
41     ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
42     image.setImage(image.getImage().getScaledInstance(159*2, 294*2,Image.
        SCALE_DEFAULT));
43     lblNewLabel.setIcon(image);
44     lblNewLabel.setBounds(0, 0, 159*2, 294*2);
45     contentPane.add(lblNewLabel);
46     JLabel lblNewLabel_1 = new JLabel("");
47     lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
48     lblNewLabel_1.setBounds(317, 0, 84, 588);
49     contentPane.add(lblNewLabel_1);
50
51
52
53     JLabel label_1 = new JLabel("新密码");
54     label_1.setFont(new Font("Xingkai TC", Font.PLAIN, 22));
55     label_1.setBounds(506, 223, 95, 41);
56     contentPane.add(label_1);
57
58     textField_1 = new JTextField();
59     textField_1.setColumns(10);
60     textField_1.setBounds(632, 228, 196, 36);
61     contentPane.add(textField_1);
62     lblNewLabel_2.setFont(new Font("Lucida Grande", Font.PLAIN, 20));
63
64     JButton button_2 = new JButton("完成");
65     button_2.setOpaque(true);
66     button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
67     button_2.setBorderPainted(false);
68     button_2.setBackground(new Color(30, 144, 255));
69     button_2.setBounds(560, 327, 171, 41);
70     button_2.addActionListener(new ActionListener() {
71     public void actionPerformed(ActionEvent e) {
```

```
72         db_alterPsd.alter(textField_1.getText());
73         log frame = new log();
74         frame.setVisible(true);
75         alterPsd.this.dispose();
76     }
77 });
78     contentPane.add(button_2);
79
80     lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
81     lblNewLabel_2.setBounds(413, 0, 683, 622);
82     contentPane.add(lblNewLabel_2);
83
84 }
85 }
```

#### 4.1.19 db\_record (交易记录所需数据)

```
1
2
3
4
5 import java.awt.List;
6 import java.sql.*;
7 import java.util.ArrayList;
8 import java.util.InputMismatchException;
9 import java.util.Scanner;
10
11 public class db_record {
12
13     public static String[] record(String account) {
14         Connection conn = db.getConn();
15         ArrayList<String> list = new ArrayList<String>();
16
17         String sql = "select record.goods from record where record.buyer = ?";
18         PreparedStatement pstmt;
19         try {
20             pstmt = (PreparedStatement) conn.prepareStatement(sql);
21             pstmt.setString(1, account);
22
23             ResultSet rs = pstmt.executeQuery();
24             while (rs.next()) {
25                 list.add(rs.getString(1));
26             }
27             pstmt.close();
28             conn.close();
29         } catch (SQLException e) {
30             e.printStackTrace();
31         }
32     }
33 }
```

```

31     }
32
33     String[] buyed = new String[list.size()];
34     for (int i = 0; i < list.size(); i++) {
35         buyed[i] = list.get(i) + " (购入) ";
36     }
37
38     Connection conn1 = db.getConn();
39     ArrayList<String> list1 = new ArrayList<String>();
40
41     String sql1 = "select record.goods from record where record.seller = ?";
42     PreparedStatement pstmt1;
43     try {
44         pstmt1 = (PreparedStatement) conn1.prepareStatement(sql1);
45         pstmt1.setString(1, account);
46
47         ResultSet rs1 = pstmt1.executeQuery();
48         while (rs1.next()) {
49             list1.add(rs1.getString(1));
50         }
51         pstmt1.close();
52         conn1.close();
53     } catch (SQLException e) {
54         e.printStackTrace();
55     }
56
57     String[] solded = new String[list1.size()];
58     for (int i = 0; i < list1.size(); i++) {
59         solded[i] = list1.get(i) + " (售出) ";
60     }
61
62     return (concat(selled, buyed));
63
64 }
65
66 static String[] concat(String[] a, String[] b) {
67     String[] c = new String[a.length + b.length];
68     System.arraycopy(a, 0, c, 0, a.length);
69     System.arraycopy(b, 0, c, a.length, b.length);
70     return c;
71 }
72 }

```

#### 4.1.20 record (交易记录界面)

```

1
2

```

```
3
4
5 import java.awt.BorderLayout;
6 import java.awt.EventQueue;
7
8 import javax.swing.JFrame;
9 import javax.swing.JPanel;
10 import javax.swing.border.EmptyBorder;
11 import java.sql.*;
12 import java.util.InputMismatchException;
13 import java.util.Scanner;
14 import java.awt.*;
15 import java.awt.event.ActionEvent;
16 import java.awt.event.ActionListener;
17
18 import javax.swing.*;
19 import javax.swing.border.EmptyBorder;
20
21 import javax.swing.border.BevelBorder;
22 import javax.swing.border.LineBorder;
23
24 public class record extends JFrame {
25     private JPanel contentPane;
26     private final JLabel lblNewLabel_2 = new JLabel("");
27
28     public record() {
29         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
30         setBounds(100, 100, 548*2, 316*2);
31         contentPane = new JPanel();
32         contentPane.setBackground(Color.WHITE);
33         contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
34         setContentPane(contentPane);
35         contentPane.setLayout(null);
36         JLabel lblNewLabel = new JLabel("");
37         ImageIcon image = new ImageIcon("/Users/wangchenhui/Desktop/icon1.jpg");
38         image.setImage(image.getImage().getScaledInstance(159*2, 294*2, Image.
39                                     SCALE_DEFAULT));
40         lblNewLabel.setIcon(image);
41         lblNewLabel.setBounds(0, 0, 159*2, 294*2);
42         contentPane.add(lblNewLabel);
43         JLabel lblNewLabel_1 = new JLabel("");
44         lblNewLabel_1.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon2.png"));
45         lblNewLabel_1.setBounds(317, 0, 84, 588);
46         contentPane.add(lblNewLabel_1);
47
48         JButton button = new JButton("前往商城");
49         button.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
50         button.setBackground(new Color(30, 144, 255));
```

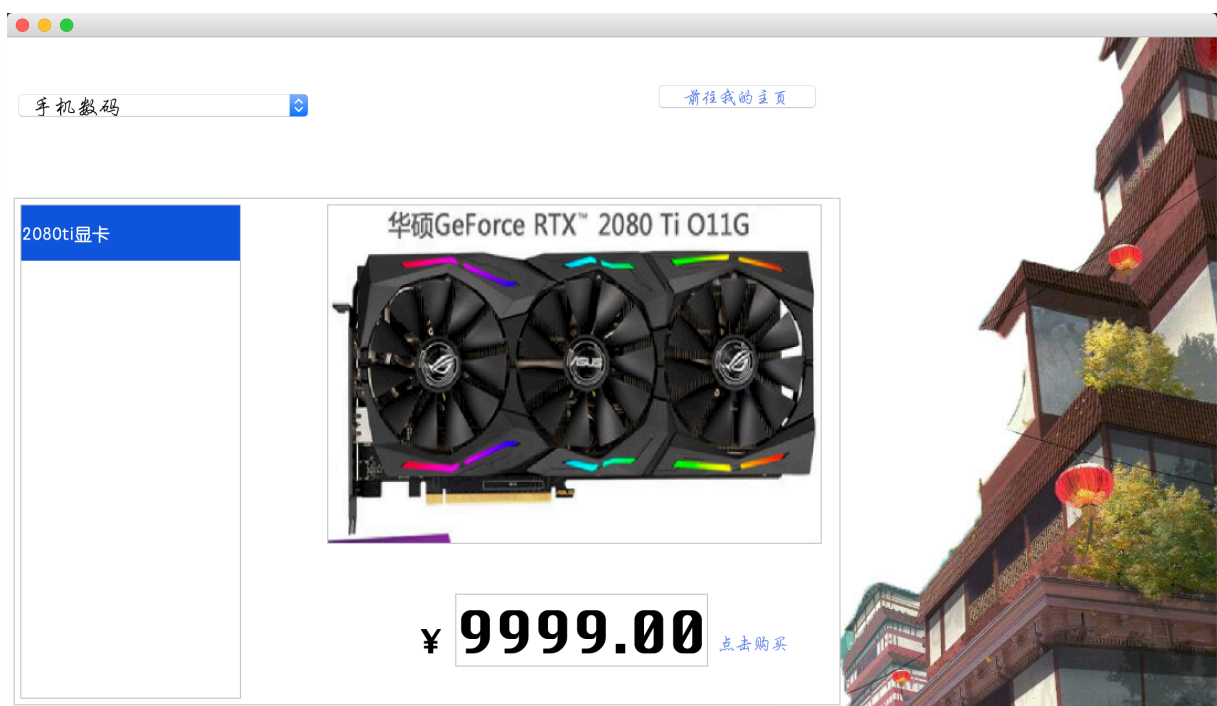
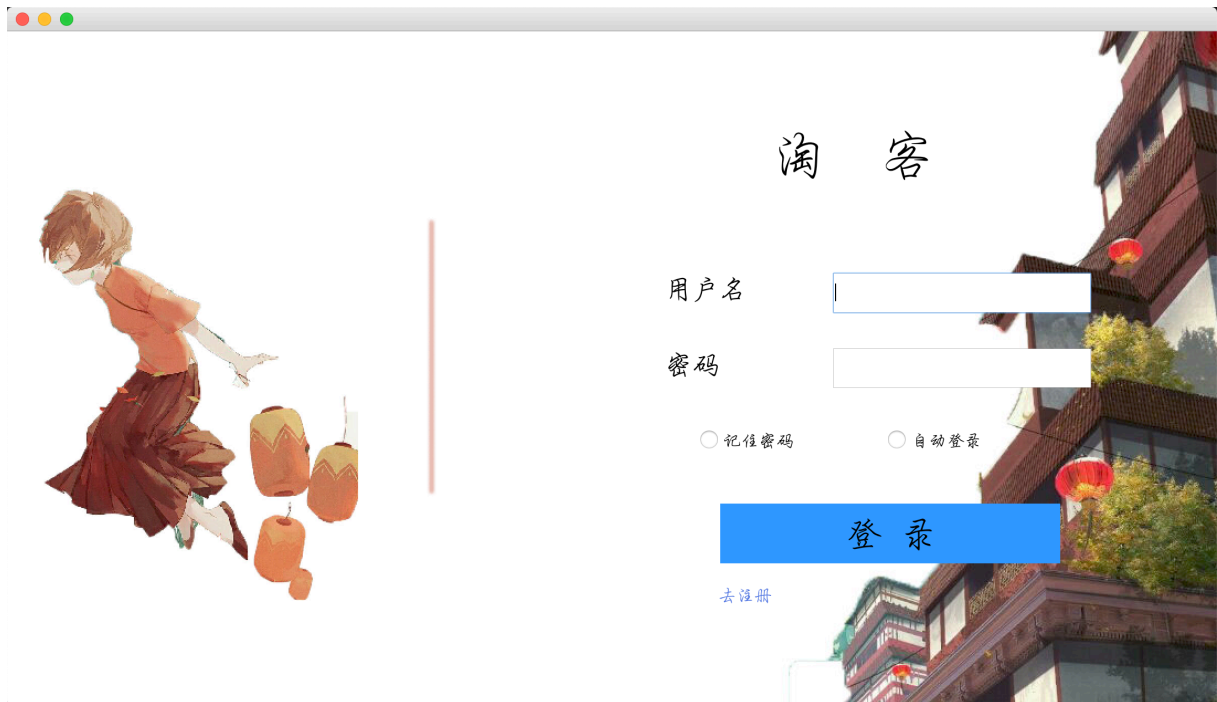
```
50 button.setBounds(560, 449, 171, 41);
51 button.setOpaque(true);
52 button.setBorderPainted(false);
53 button.addActionListener(new ActionListener() {
54     public void actionPerformed(ActionEvent e) {
55         mainWindow frame_1 = new mainWindow();
56         frame_1.setVisible(true);
57         record.this.dispose();
58     }
59 });
60 contentPane.add(button);
61
62 JButton button_2 = new JButton("返回主页");
63 button_2.setOpaque(true);
64 button_2.setFont(new Font("Xingkai TC", Font.PLAIN, 20));
65 button_2.setBorderPainted(false);
66 button_2.setBackground(new Color(30, 144, 255));
67 button_2.setBounds(560, 518, 171, 41);
68 button_2.addActionListener(new ActionListener() {
69     public void actionPerformed(ActionEvent e) {
70         myPage frame_1 = new myPage();
71         frame_1.setVisible(true);
72         record.this.dispose();
73     }
74 });
75 contentPane.add(button_2);
76
77 JList list = new JList();
78 db_record.record(log.account);
79 list.setModel(new AbstractListModel() {
80
81     String[] values = db_record.record(log.account);
82     public int getSize() {
83         return values.length;
84     }
85     public Object getElementAt(int index) {
86         return values[index];
87     }
88 });
89 list.setBounds(479, 74, 348, 342);
90 list.setBorder(new LineBorder(Color.LIGHT_GRAY, 1, true));
91 list.setFont(new Font("Yuanti TC", Font.PLAIN, 16));
92 list.setFixedCellWidth(200);
93 list.setFixedCellHeight(50);
94 contentPane.add(list);
```

```
98
99     lblNewLabel_2.setIcon(new ImageIcon("/Users/wangchenhui/Desktop/icon3.jpg"));
100    lblNewLabel_2.setBounds(413, 0, 683, 622);
101    contentPane.add(lblNewLabel_2);
102
103    }
104
105    }
```

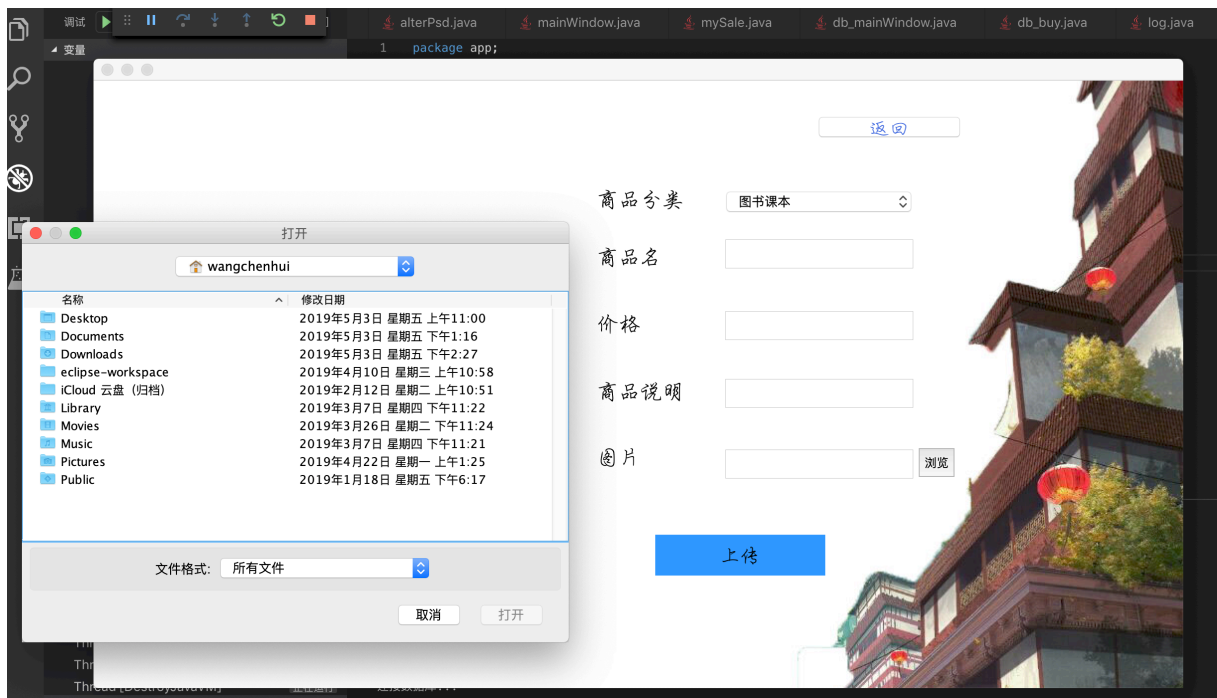
#### 4.1.21 app.java(主方法入口)

```
1
2
3
4
5  import java.sql.*;
6  import java.util.InputMismatchException;
7  import java.util.Scanner;
8  import java.awt.*;
9  import java.awt.event.ActionEvent;
10 import java.awt.event.ActionListener;
11
12 import javax.swing.*;
13 import javax.swing.border.EmptyBorder;
14
15 public class App {
16
17     public static void main(String[] args) {
18         log frame = new log();
19         frame.setVisible(true);
20     }
```

## 4.2 成品展示









## 第 5 章 总结与心得

### 5.1 事必躬亲

在这次程序设计中，最大的收获便是知道了事必躬亲的道理，书本上学到的终归是纸上谈兵，只有自己动手才能发现问题所在，通过实践，深入掌握 public 与 private、静态与非静态各自的用处；学会了怎么从一个页面利用事件内部类中调用另一个界面的构造方法跳转到另一个页面。

### 5.2 工欲善其事，必先利其器

高效的利用已有的工具可以很好的提高我们写代码的效率，例如 eclipse 中的插件——windowbuilder，可以通过可视化拖动实现界面的设计；又比如选择了这一款高效的编辑器——VScode，强大的代码高亮和自动补全功能，让我这个初学者能够在短短 80 多小时里写下 3000 行代码

### 5.3 自我学习

在程序设计过程中碰到了很多问题，都需要自己去解决，通过各种方式搜集信息，寻找结局问题的方法；另外在使用 L<sup>A</sup>T<sub>E</sub>X 排版的过程中，也查阅大量资料，这是一个自我不断扩充的过程。

## A 第一附录

该 java 项目可访问我的 Github 下载:<https://github.com/CanoeByGuitar/taoke>