|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **成 绩** |  | **批阅人** | 杨军 | **日 期** | 2021.12.8 |

.

**计算机科学与技术**

**课 程 设 计**

**（2021/2022 学年 第一学期）**

|  |  |
| --- | --- |
| 课程名称 | Java面向对象程序设计技术 |
| 设计名称 | 旅游管理系统 |
| 设计时间 | 2021年12月4日至2021年12月16日 |
| 指导教师 | 杨军 |

|  |  |  |  |
| --- | --- | --- | --- |
| 学生姓名 | 王飞飞 | 学号 | 20213330438 |
| 班级 | 计科专升本21级2班 | 专业 | 计算机科学与技术 |

目录

[一、绪论 3](#_Toc90565575)

[1.1、研究背景 3](#_Toc90565576)

[1.2、开发环境 4](#_Toc90565577)

[二、系统分析 5](#_Toc90565578)

[2.1、需求分析 5](#_Toc90565579)

[2.2、主要内容 5](#_Toc90565580)

[2.3、系统功能 5](#_Toc90565581)

[2.4、可行性分析 5](#_Toc90565582)

[2.5使用说明 6](#_Toc90565583)

[三、系统总体设计 7](#_Toc90565584)

[3.1、旅游系统实例图 7](#_Toc90565585)

[3.2总体结构 8](#_Toc90565586)

[3.3类图 10](#_Toc90565587)

[3.4数据库图 14](#_Toc90565588)

[3.5jdbc的连接代码 14](#_Toc90565589)

[四、程序运行结果 17](#_Toc90565590)

[4.1登录界面 17](#_Toc90565591)

[4.2注册页面 19](#_Toc90565592)

[4.3主界面 21](#_Toc90565593)

[4.4 报名旅游 28](#_Toc90565594)

[4.5订单管理界面 35](#_Toc90565595)

[五、系统测试 44](#_Toc90565596)

[5.1注册测试 44](#_Toc90565597)

[5.2登录测试 44](#_Toc90565598)

[5.3报名旅游 45](#_Toc90565599)

[5.5订单管理 45](#_Toc90565600)

[5.6退出系统 46](#_Toc90565601)

[六、结论与心得 47](#_Toc90565602)

一、绪论

## 1.1、研究背景

目前旅游业的信息量非常的大，旅游在正常运营中需要处理大量的信息。因此需要对旅游景点信息、旅游路线信息,旅游服务信息进行管理,及时了解各种信息的变更，针对变更及时做出调整，这样有利于提高管理效率和资源的利用率。通过旅游线路管理系统，来实现旅游的信息管理工作的规范化、系统化、自动化,旅游线路管理系统是为了提高管理信息的效率、资源的充分利用、利润的最大化。

## 1.2、开发环境

java是一门[面向对象](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1" \t "_blank)编程语言，Java具有简单性、面向对象、[分布式](https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F/19276232" \t "_blank)、[健壮性](https://baike.baidu.com/item/%E5%81%A5%E5%A3%AE%E6%80%A7/4430133)、[安全性](https://baike.baidu.com/item/%E5%AE%89%E5%85%A8%E6%80%A7/7664678)、平台独立与可移植性、[多线程](https://baike.baidu.com/item/%E5%A4%9A%E7%BA%BF%E7%A8%8B/1190404)、动态性等特点 。Java可以编写[桌面应用程序](https://baike.baidu.com/item/%E6%A1%8C%E9%9D%A2%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F/2331979" \t "_blank)、Web应用程序、[分布式系统](https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E7%B3%BB%E7%BB%9F/4905336" \t "_blank)和[嵌入式系统](https://baike.baidu.com/item/%E5%B5%8C%E5%85%A5%E5%BC%8F%E7%B3%BB%E7%BB%9F/186978)应用程序等

JDK 是Java开发工具包 (Java Development Kit ) 的缩写。从字面意思翻译过来就是java开发的工具。它是一种用于构建在 Java 平台上发布的应用程序、applet 和组件的开发环境。其中包括了JVM、Java编译器、大量的Java工具以及Java基础API里面是Java类库和Java的语言规范。

IntelliJ IDEA的每个方面都专门设计用于最大限度地提高开发人员的工强大的静态代码分析和符合人体工程学的设计使开发不仅具有高效性，而且还具有令人愉悦的体验。

在IntelliJ IDEA为您的源代码编制索引之后，它通过在每个上下文中提供相关建议提供了快速而智能的体验：即时和巧妙的代码完成，动态代码分析和可靠的重构工具。

MySQL 是一款安全、跨平台、高效的，并与 PHP、Java 等主流编程语言紧密结合的数据库系统。该数据库系统是由瑞典的 MySQL AB 公司开发、发布并支持，由 MySQL 的初始开发人员 David Axmark 和 Michael Monty Widenius 于 1995 年建立的。

二、系统分析

## 2.1、需求分析

了解用户要求。此用户即为系统的使用人员（超级管理员，普通管理员和用户）。了解他们在系统功能、性能等方面的要求及用户在硬件配置、开发周期处理方式等方面的意向与打算。把用户要求成文，完成系统分析报告。

## 2.2、主要内容

打开客户端，如果有账号、密码，输入账号、密码登录，根据账号的不同登录不同的用户，除了管理员之外，别的用户无法查看订单管理。

管理员拥有所有的权限。

普通用户可以查看热门线路。

## 2.3、系统功能

登录之后可点击查看热门线路、报名旅游、订单管理（只有管理员可操作）、用户切换、添加用户、退出系统、系统帮助

## 2.4、可行性分析

**（1）经济可行性分析**

城市旅游资源丰富，且各自的前期宣传也做得较到位，已有很多传统宣传工作在进行，所以其对宣传费用的投入是较好的所以从投入方面可以不用太担心。而经本网站的统一的高效益的宣传，对旅游业的综合系统的宣传和形象的塑造，对旅游业有很好的经济效益。

**（2）技术可行性分析**

本系统采用java作为客户端的书写语言，用Mysql存放后端数据。

java面向对象基于对象的编程更符合人的思维模式，使人们更容易编写程序,与平台无关是 Java 语言最大的优势。其它语言编写的程序面临的一个主要问题是操作系统的变化，处理器升级以及核心系统资源的变化，都可能导致程序出现错误或无法运行,java可以多线程操作，不用担心网络阻塞，java的基本就是类，可以动态的搭建自己想要的所有。

Mysql体积小、速度快、总体拥有成本低，开源；支持多种操作系统；是开源数据库，提供的接口支持多种语言连接操作 ；MySQL的核心程序采用完全的多线程编程。线程是轻量级的进程，它可以灵活地为用户提供服务，而不过多的系统资源。

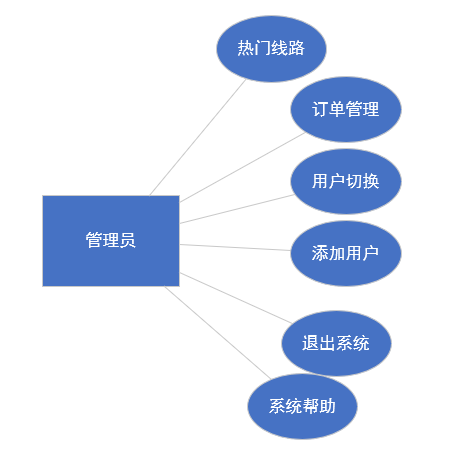
## 2.5使用说明

（1）打开登录界面，用管理员账号成功登录可以查看订单的详细情况包括对订单的查找，修改，删除，添加。

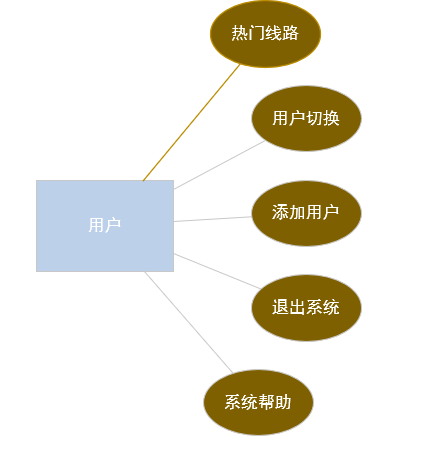
（2）普通用户登录之后，可以查找热门的了旅游线路，可以报名旅游

三、系统总体设计

## 3.1、旅游系统实例图



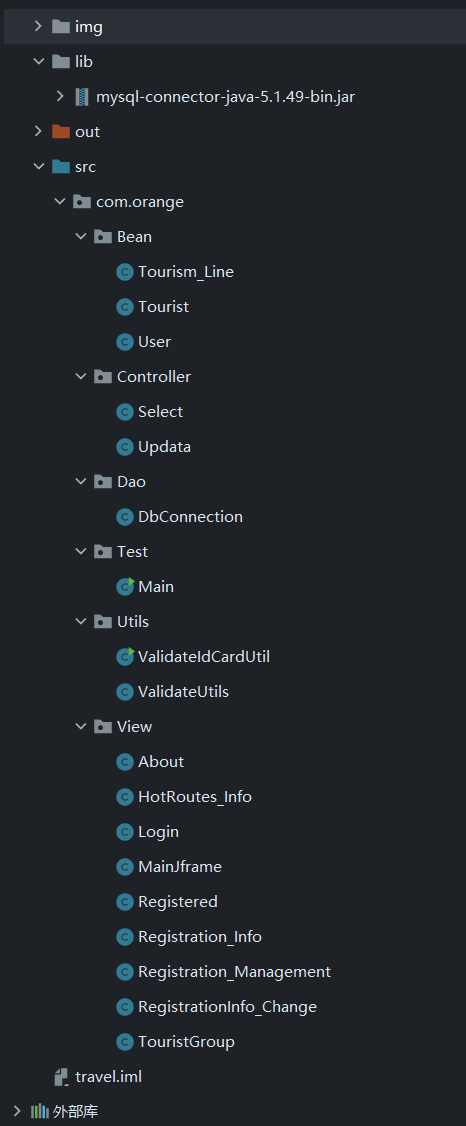
管理员实例图



用户实例图

## 3.2总体结构

该系统总体项目结构如下图所示，该项目名称为travel,在src目录下有6个包。



基于maven的逻辑设计。

lib目录存放链接数据数据库的jar包

Bean包存放用到的变量和get、set方法

Controller包存放数据的命令

Dao包存放连接数据库的代码（jdbc）

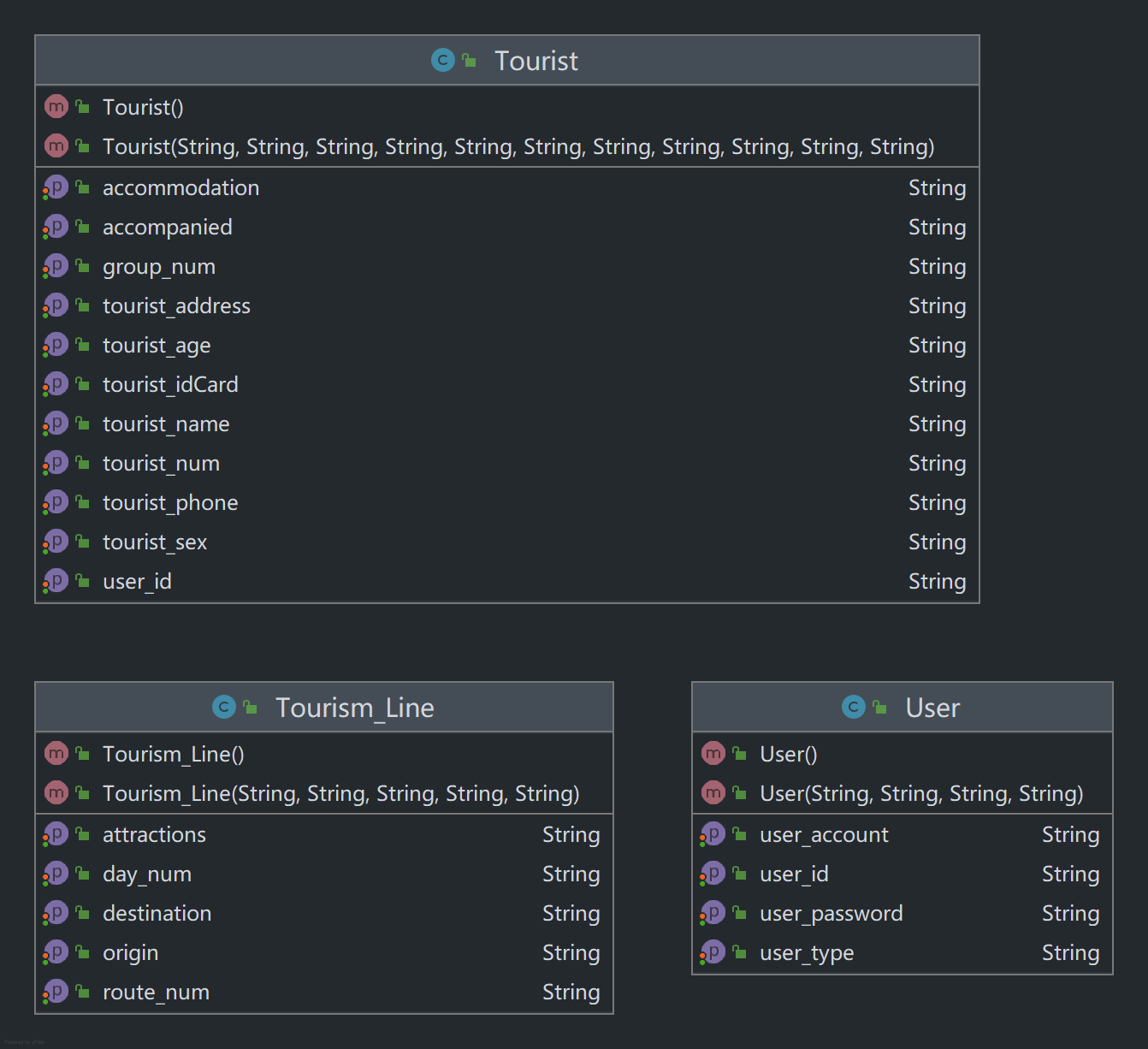
Test包存放主程序

Utils包存放检查验证输入数据的代码

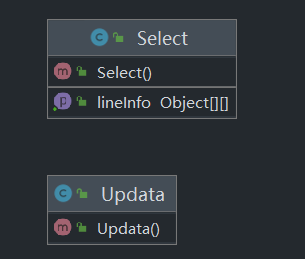
View包存放客户端的页面设计

## 3.3类图

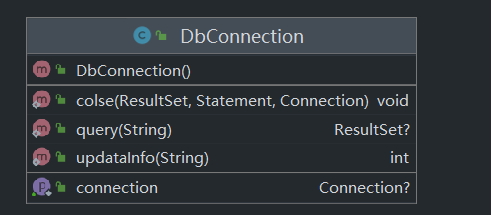
bean包



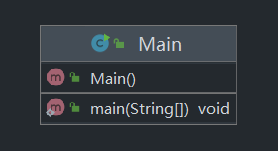
Controller



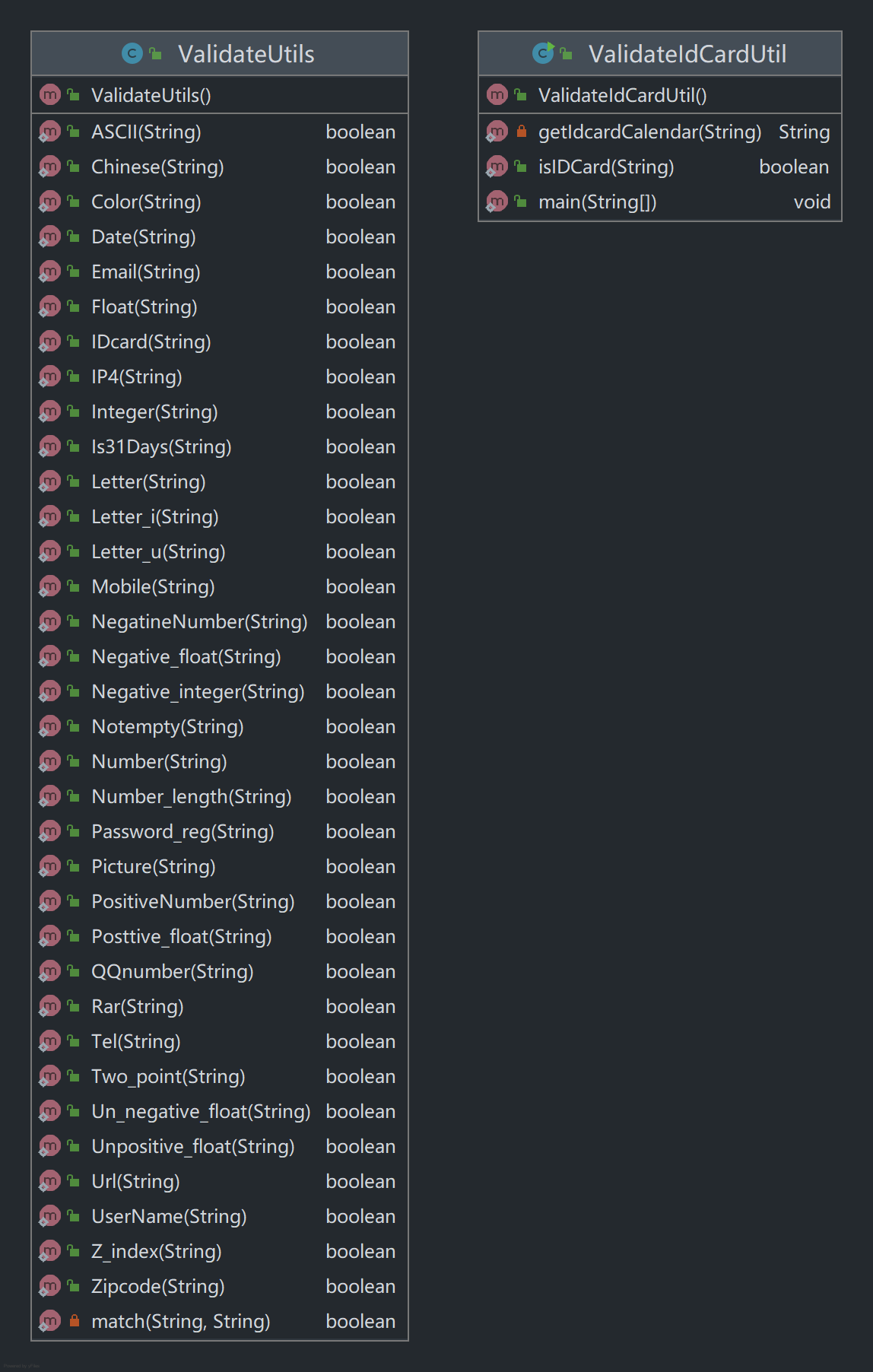
Dao包



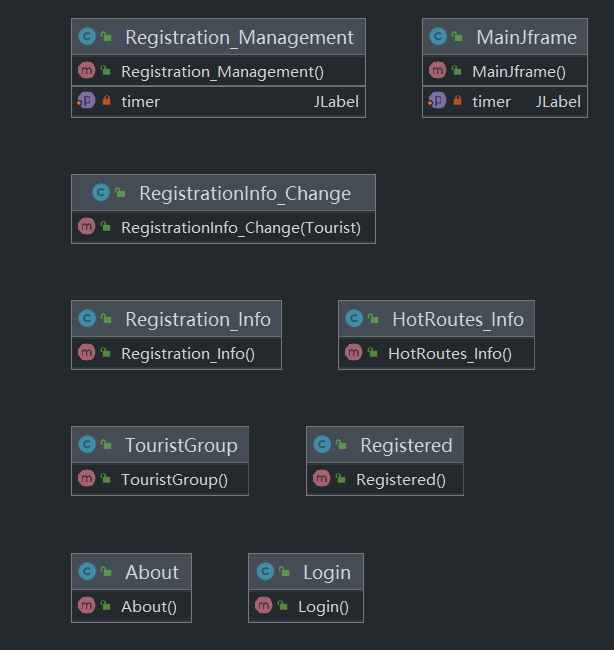
Test包



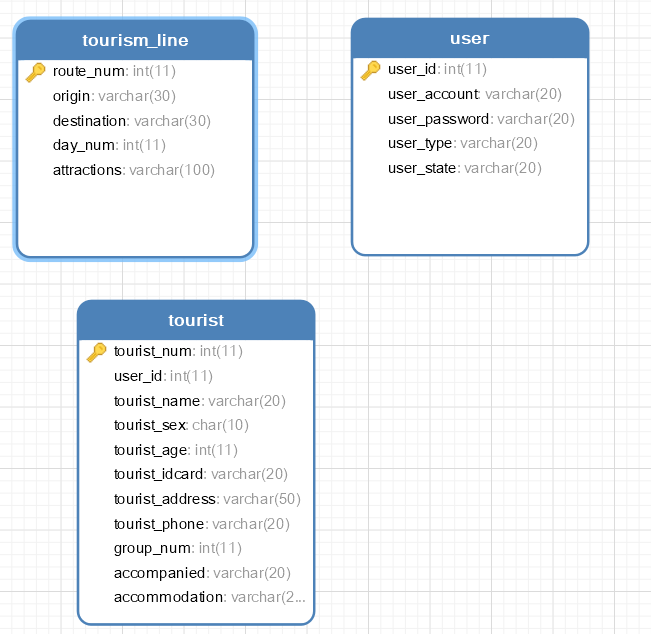
Utils包



View包



## 3.4数据库图

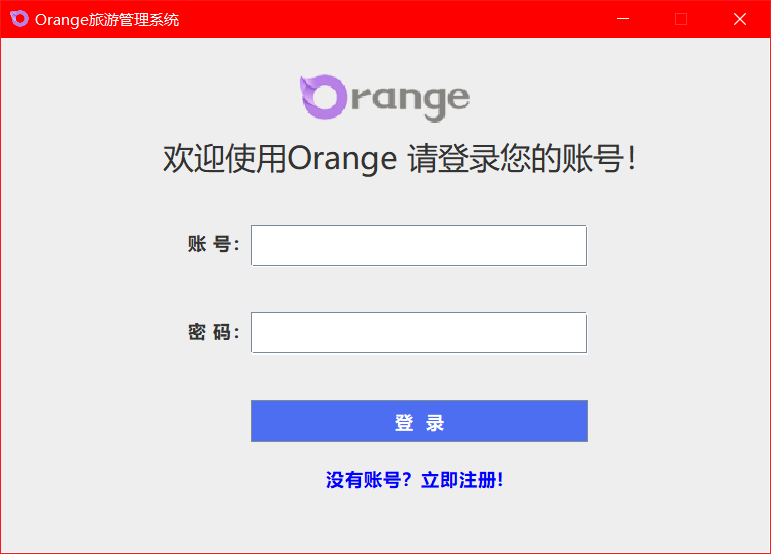


## 3.5jdbc的连接代码

public class DbConnection {  
 //驱动类的类名  
 private static final String *DRIVERNAME*="com.mysql.jdbc.Driver";  
 //连接数据的URL路径  
 private static final String *URL*="jdbc:mysql://127.0.0.1:3306/orange\_travel\_manage";  
 //数据库登录账号  
 private static final String *USER*="root";  
 //数据库登录密码  
 private static final String *PASSWORD*="root";  
 //加载驱动  
 static{  
 try {  
 Class.*forName*(*DRIVERNAME*);  
 } catch (ClassNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
 //获取数据库连接  
 public static *Connection* getConnection() {  
 try {  
 return DriverManager.*getConnection*(*URL*,*USER*,*PASSWORD*);  
 } catch (SQLException e) {   
 e.printStackTrace();  
 }  
 return null;  
 }  
 //查询  
 public static *ResultSet* query(String sql) {  
 System.*out*.println(sql);  
 //获取连接  
 *Connection* connection=*getConnection*();  
 *PreparedStatement* psd;  
 try {  
 psd = connection.prepareStatement(sql);  
 return psd.executeQuery();  
 } catch (SQLException e) {  
 JOptionPane.*showMessageDialog*(null,"执行语句出错\n"+e.toString());  
 e.printStackTrace();  
 }  
 return null;  
 }  
 //增、删、改、查  
 public static int updataInfo(String sql) {  
 System.*out*.println(sql);  
 //获取连接  
 *Connection* connection=*getConnection*();  
 try {  
 *PreparedStatement* psd=connection.prepareStatement(sql);  
 return psd.executeUpdate();  
 } catch (SQLException e) {  
 JOptionPane.*showMessageDialog*(null,"执行语句出错\n"+e.toString());  
 e.printStackTrace();  
 }  
 return 0;  
 }  
 //关闭连接  
 public static void colse(*ResultSet* rs,*Statement* stmt,*Connection* conn) throws Exception{  
 try { if (rs != null){ rs.close(); }  
 if (stmt != null) { stmt.cancel(); }  
 if (conn != null) { conn.close(); }  
 } catch (Exception e) {  
 e.printStackTrace(); throw new Exception();  
 }  
 }  
}

四、程序运行结果

## 4.1登录界面



输入账号密码和数据库中的账号密码进行比对，若账号密码比对成功，则进入系统，如失败则弹出对话框提升重新输入，点击注册跳转到注册界面

*详细代码：*

JButton button = new JButton("登 录");  
button.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
button.setBounds(200, 290, 270, 34);  
button.setFocusPainted(false);//去掉按钮周围的焦点框  
button.setBackground(new Color(78, 110, 242));  
button.setForeground(Color.*WHITE*);  
getContentPane().add(button);  
  
button.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 String account=userField.getText();  
 @SuppressWarnings("deprecation")  
 String password=passwordField.getText();  
 if (account.equals("")&&password.equals("")) {  
 JOptionPane.*showMessageDialog*(null, "账户名或密码未填写！");  
 }   
 else {  
 String sql = "select COUNT(\*) from user where user\_password='"+password+"' and user\_account='"+account+"'";  
 int reselt = select.getCount(sql);  
 int i = updata.addData("UPDATE user SET user\_state='已登录' WHERE user\_account='"+account+"';");  
 if (reselt>0 && i>0) {  
 //打开主界面窗口  
 MainJframe m = new MainJframe();  
 m.setVisible(true);  
 dispose();//关闭旧窗口  
 } else {  
 JOptionPane.*showMessageDialog*(null, "账户名或密码不正确！请重新输入！");  
 }  
 }  
 }  
});  
  
//注册跳转  
JLabel registerLabel = new JLabel("没有账号？立即注册!");  
registerLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
registerLabel.setForeground(Color.*blue*);  
registerLabel.setBounds(260, 340, 168, 27);  
getContentPane().add(registerLabel);  
registerLabel.addMouseListener(new MouseListener(){  
 public void mouseClicked(MouseEvent e) {  
 //打开注册界面  
 Registered m = new Registered();  
 m.setVisible(true);  
 }

## 4.2注册页面



输入账号名和密码，点击注册之后在数据库的user表中增加用户注册成功之后跳转到登录页面

*详细代码：*

//账户名  
 JLabel usernameLabel = new JLabel("账 号 名：");  
 usernameLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 usernameLabel.setBounds(165, 138, 72, 18);  
 getContentPane().add(usernameLabel);  
   
 userField = new JTextField();  
 userField.setBounds(248, 130, 255, 34);  
 getContentPane().add(userField);  
 userField.setColumns(10);  
   
 //密码  
 JLabel passwordLabel = new JLabel("输入密码：");  
 passwordLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 passwordLabel.setBounds(165, 208, 83, 18);  
 getContentPane().add(passwordLabel);  
   
 passwordField = new JPasswordField();  
 passwordField.setBounds(248, 267, 255, 34);  
 getContentPane().add(passwordField);  
 passwordField.setColumns(10);  
   
 JLabel passwordAgainLabel = new JLabel("确认密码：");  
 passwordAgainLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 passwordAgainLabel.setBounds(165, 275, 92, 18);  
 getContentPane().add(passwordAgainLabel);  
   
 passwordAgainField = new JPasswordField();  
 passwordAgainField.setBounds(248, 200, 255, 34);  
 getContentPane().add(passwordAgainField);  
 passwordAgainField.setColumns(10);  
   
 //确认注册按钮  
 JButton button = new JButton("确定注册");  
 button.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 button.setBounds(250, 330, 250, 34);  
 button.setFocusPainted(false);//去掉按钮周围的焦点框  
 button.setBackground(new Color(78, 110, 242));  
 button.setForeground(Color.*WHITE*);  
 getContentPane().add(button);  
   
 button.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 String username = userField.getText();  
 @SuppressWarnings("deprecation")  
 String password = passwordField.getText();  
 @SuppressWarnings("deprecation")  
 String passwordAgain = passwordAgainField.getText();  
   
 //确认输入的信息是否为空  
 if (username.equals("")&&password.equals("")&&passwordAgain.equals("")) {  
 JOptionPane.*showMessageDialog*(null, "请完整输入信息！");  
 }  
 else {  
 //判断两次密码是否一致  
 if (password.equals(passwordAgain)) {  
 String sql = "INSERT INTO `user` VALUES (null, '" + username + "', '"+password + "', '游客', '未登录');";  
 int reselt = updata.addData(sql);  
 if (reselt>0) {  
 JOptionPane.*showMessageDialog*(null, "注册成功，即将跳转到登录页面！");  
 //打开登录界面  
 Login l = new Login();  
 l.setVisible(true);  
 dispose();  
 }  
 else {  
 JOptionPane.*showMessageDialog*(null, "注册失败，请联系管理员！");  
 }  
 }  
 else {  
 JOptionPane.*showMessageDialog*(null, "两次输入的密码不一致，请重新输入！");  
 }  
 }  
 }  
 });  
}

## 4.3主界面



显示用户可以做的操作（只有管理员能操作订单管理）

*详细代码：*

public MainJframe() {  
   
 this.setTitle("Orange旅游管理系统");  
 this.setBounds(0, 0, 1130, 700);  
 this.setLocationRelativeTo(null);//让窗口在屏幕中间显示  
 this.setResizable(false);//让窗口大小不可改变  
 getContentPane().setLayout(null);  
 this.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);//用户单击窗口的关闭按钮时程序执行的操作  
   
 //设置窗口图标  
 Toolkit tk = Toolkit.*getDefaultToolkit*();  
 Image frameImage=tk.createImage("img/logo.png");   
 this.setIconImage(frameImage);  
   
 //logo添加  
 JLabel jl\_logo = new JLabel();  
 jl\_logo.setIcon(new ImageIcon("img/logo2.png"));//文件路径  
 jl\_logo.setBounds(950, 6, 200, 50);  
 this.add(jl\_logo);  
   
 //热门线路按钮  
 ImageIcon icon1 = new ImageIcon("img/Icon1.png");  
 JButton searchButton = new JButton("热门线路查询",icon1);  
 searchButton.setBounds(20, 11, 145, 35);  
 searchButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 searchButton.setContentAreaFilled(false);//设置按钮透明背景  
 searchButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
// searchButton.setBackground(new Color(255, 153, 41));  
 searchButton.setForeground(Color.*black*);  
 getContentPane().add(searchButton);  
   
 searchButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 //显示热门线路界面  
 HotRoutes\_Info hr = new HotRoutes\_Info();  
 hr.setVisible(true);  
// dispose();关闭旧窗口  
 }  
 });  
   
 //报名旅游按钮  
 ImageIcon icon2 = new ImageIcon("img/Icon2.png");  
 JButton signButton = new JButton("报名旅游",icon2);  
 signButton.setBounds(173, 11, 120, 35);  
 signButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 signButton.setContentAreaFilled(false);//设置按钮透明背景  
 signButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
// signButton.setBackground(new Color(255, 153, 41));  
 signButton.setForeground(Color.*black*);  
 getContentPane().add(signButton);  
   
 signButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 //判断当前是否有用户登录  
 String sql = "SELECT COUNT(\*) FROM `user` WHERE user\_state='已登录'";  
 int reselt = select.getCount(sql);  
 if (reselt>0) {  
 //打开报名界面  
 Registration\_Info r = new Registration\_Info();  
 r.setVisible(true);  
 }   
 else {  
 JOptionPane.*showMessageDialog*(null, "请先登录！");  
 //打开登录界面  
 Login l = new Login();  
 l.setVisible(true);  
// dispose();  
 }  
 }  
 });  
   
 //订单管理按钮  
 ImageIcon icon3 = new ImageIcon("img/Icon3.png");  
 JButton orderButton = new JButton("订单管理",icon3);  
 orderButton.setBounds(300, 11, 125, 35);  
 orderButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 orderButton.setContentAreaFilled(false);//设置按钮透明背景  
 orderButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
// yhdl.setBackground(new Color(255, 153, 41));  
 orderButton.setForeground(Color.*black*);  
 getContentPane().add(orderButton);  
   
 orderButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 //判断当前是否有用户登录  
 String sql = "SELECT COUNT(\*) FROM `user` WHERE user\_state='已登录'";  
 int reselt = select.getCount(sql);  
 if (reselt>0) {  
 //判断当前登录的用户身份  
 String user\_type = select.getString("SELECT user\_type FROM `user` WHERE user\_state='已登录'");  
 if (user\_type.equals("管理员")) {  
 //打开订单管理界面  
 Registration\_Management r = new Registration\_Management();  
 r.setVisible(true);  
 dispose();  
 }else{  
 JOptionPane.*showMessageDialog*(null, "当前用户无权限！请登录管理员账号！");  
 }  
 } else {  
 JOptionPane.*showMessageDialog*(null, "请先登录！");  
 //打开登录界面  
 Login l = new Login();  
 l.setVisible(true);  
// dispose();//关闭旧窗口  
 }  
 }  
 });  
   
 //用户切换按钮  
 ImageIcon icon4 = new ImageIcon("img/Icon4.png");  
 JButton switchButton = new JButton("用户切换",icon4);  
 switchButton.setBounds(433, 11, 120, 35);  
 switchButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 switchButton.setContentAreaFilled(false);//设置按钮透明背景  
 switchButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
// switchButton.setBackground(new Color(255, 153, 41));  
 switchButton.setForeground(Color.*black*);  
 getContentPane().add(switchButton);  
   
 switchButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 //判断当前是否有用户登录  
 String sql = "SELECT COUNT(\*) FROM `user` WHERE user\_state='已登录'";  
 int reselt = select.getCount(sql);  
 if (reselt>0) {  
 String i = select.getString("SELECT user\_account FROM `user` WHERE user\_state='已登录'");  
 int a = JOptionPane.*showConfirmDialog*(null,"当前已有用户" + " ”"+ i +"” " + "登录！是否注销？","注销提示",0,1);  
 if(a == JOptionPane.*OK\_OPTION*){  
 JOptionPane.*showMessageDialog*(null, "已注销前账户！");  
 updata.addData("UPDATE user SET user\_state='未登录' WHERE user\_account='"+ i +"';");  
 //打开登录界面  
 Login l = new Login();  
 l.setVisible(true);  
 dispose();//关闭旧窗口  
 }  
 }else {  
 //打开登录界面  
 Login l = new Login();  
 l.setVisible(true);  
 dispose();//关闭旧窗口  
 }  
 }  
 });  
   
 //添加用户按钮  
 ImageIcon icon5 = new ImageIcon("img/Icon5.png");  
 JButton appendButton = new JButton("添加用户",icon5);  
 appendButton.setBounds(560, 11, 120, 35);  
 appendButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 appendButton.setContentAreaFilled(false);//设置按钮透明背景  
 appendButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
// appendButton.setBackground(new Color(255, 153, 41));  
 appendButton.setForeground(Color.*black*);  
 getContentPane().add(appendButton);  
   
 appendButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 //判断当前是否有用户登录  
 String sql = "SELECT COUNT(\*) FROM `user` WHERE user\_state='已登录'";  
 int reselt = select.getCount(sql);  
 if (reselt>0) {  
 String i = select.getString("SELECT user\_account FROM `user` WHERE user\_state='已登录'");  
// JOptionPane.showMessageDialog(null, "当前已有用户"+" ”"+i+"” "+"登录！是否注销？");  
 int a = JOptionPane.*showConfirmDialog*(null,"当前已有用户" + " ”" + i + "” " + "登录！是否注销？","注销提示",0,1);  
 if(a == JOptionPane.*OK\_OPTION*){  
 JOptionPane.*showMessageDialog*(null, "已注销前账户！");  
 updata.addData("UPDATE user SET user\_state='未登录' WHERE user\_account='" + i + "';");  
 //打开注册界面  
 Registered r = new Registered();  
 r.setVisible(true);  
 //dispose();//打开新窗口的时候关闭旧窗口  
 }  
 }else {  
 //打开注册界面  
 Registered r = new Registered();  
 r.setVisible(true);  
 //dispose();  
 }  
 }  
 });  
   
 //退出系统按钮  
 ImageIcon icon6 = new ImageIcon("img/Icon6.png");  
 JButton exitButton = new JButton("退出系统",icon6);  
 exitButton.setBounds(687, 11, 120, 35);  
 exitButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 exitButton.setContentAreaFilled(false);//设置按钮透明背景  
 exitButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
// exitButton.setBackground(new Color(255, 153, 41));  
 exitButton.setForeground(Color.*black*);  
 getContentPane().add(exitButton);  
   
 exitButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 int result = JOptionPane.*showConfirmDialog*(null,"您现在要关闭系统吗?关闭后同时注销账号!","退出提示",0,1);  
 if(result == JOptionPane.*OK\_OPTION*){  
 JOptionPane.*showMessageDialog*(null, "已退出系统，欢迎下次使用！");  
 updata.addData("UPDATE user SET user\_state='未登录';");  
 System.*exit*(0);  
 }  
 }  
 });  
   
 //系统帮助按钮  
 ImageIcon icon7 = new ImageIcon("img/Icon7.png");  
 JButton helpButton = new JButton("系统帮助",icon7);  
 helpButton.setBounds(815, 11, 120, 35);  
 helpButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 helpButton.setContentAreaFilled(false);//设置按钮透明背景  
 helpButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
// helpButton.setBackground(new Color(101, 113, 175));  
 helpButton.setForeground(Color.*black*);  
 getContentPane().add(helpButton);  
   
 helpButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 //打开系统帮助界面  
 About a = new About();  
 a.setVisible(true);  
 }  
 });  
   
 //当前时间  
 JLabel dqsj = new JLabel("当前时间 ：");  
 dqsj.setBounds(825, 70, 85, 35);  
 dqsj.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 dqsj.setForeground(Color.*white*);  
 getContentPane().add(dqsj);  
   
 JLabel time1 = new JLabel();  
 time1.setBounds(910, 71, 236, 35);  
 time1.setFont(new Font("微软雅黑", Font.*CENTER\_BASELINE*, 16));  
 time1.setForeground(Color.*white*);  
 getContentPane().add(time1);  
 this.setTimer(time1);  
   
 //背景图片添加  
 JLabel jl = new JLabel();  
 jl.setIcon(new ImageIcon("img/timebg.png"));//文件路径  
 jl.setBounds(810, 62, 750, 40);  
 this.add(jl);

## 4.4 报名旅游



在里面填写的信息会被写入到数据库中，方便保存信息

*代码详细*：

super("修改报名信息");  
 this.setBounds(0, 0, 800, 600);  
 this.setLocationRelativeTo(null);//让窗口在屏幕中间显示  
 this.setResizable(false);//让窗口大小不可改变  
 getContentPane().setLayout(null);  
   
 //设置窗口图标  
 Toolkit tk = Toolkit.*getDefaultToolkit*();  
 Image frameImage=tk.createImage("img/logo.png");   
 this.setIconImage(frameImage);  
   
 //图片添加  
 JLabel jl = new JLabel();  
 jl.setIcon(new ImageIcon("img/brand.png"));//文件路径  
 jl.setBounds(453, 10, 800, 300);  
 this.add(jl);  
   
 //姓名  
 JLabel nameLabel= new JLabel("姓 名：");  
 nameLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 nameLabel.setBounds(130, 39, 72, 18);  
 getContentPane().add(nameLabel);  
   
 nameField = new JTextField();  
 nameField.setBounds(191, 36, 240, 28);  
 getContentPane().add(nameField);  
 nameField.setColumns(10);  
   
 //性别  
 JLabel sexLabel= new JLabel("性 别：");  
 sexLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 sexLabel.setBounds(130, 90, 72, 18);  
 getContentPane().add(sexLabel);  
   
 JComboBox sexcomboBox = new JComboBox();  
 sexcomboBox.setModel(new DefaultComboBoxModel(new String[] {"男", "女"}));  
 sexcomboBox.setBounds(191, 87, 240, 28);  
 getContentPane().add(sexcomboBox);  
   
 //年龄  
 JLabel ageLabel= new JLabel("年 龄：");  
 ageLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 ageLabel.setBounds(130, 138, 72, 18);  
 getContentPane().add(ageLabel);  
   
 ageField = new JTextField();  
 ageField.setBounds(191, 135, 240, 28);  
 getContentPane().add(ageField);  
 ageField.setColumns(10);  
   
 //身份证号码  
 JLabel IDcardLabel= new JLabel("身份证号码：");  
 IDcardLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 IDcardLabel.setBounds(98, 188, 117, 18);  
 getContentPane().add(IDcardLabel);  
   
 IDcardField = new JTextField();  
 IDcardField.setBounds(191, 185, 240, 28);  
 getContentPane().add(IDcardField);  
 IDcardField.setColumns(10);  
   
 //家庭住址  
 JLabel addressLabel= new JLabel("家庭住址：");  
 addressLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 addressLabel.setBounds(113, 234, 100, 18);  
 getContentPane().add(addressLabel);  
   
 addressField = new JTextField();  
 addressField.setBounds(191, 231, 240, 28);  
 getContentPane().add(addressField);  
 addressField.setColumns(10);  
   
 //联系电话  
 JLabel phoneLabel= new JLabel("联系电话：");  
 phoneLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 phoneLabel.setBounds(113, 280, 100, 18);  
 getContentPane().add(phoneLabel);  
   
 phoneField = new JTextField();  
 phoneField.setBounds(191, 277, 240, 28);  
 getContentPane().add(phoneField);  
 phoneField.setColumns(10);  
   
 //团号  
 JLabel touristNumberLabel= new JLabel("团 号：");  
 touristNumberLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 touristNumberLabel.setBounds(130, 325, 72, 18);  
 getContentPane().add(touristNumberLabel);  
   
 touristNumberField = new JTextField();  
 touristNumberField.setBounds(191, 322, 240, 28);  
 getContentPane().add(touristNumberField);  
 touristNumberField.setColumns(10);  
   
 //陪同  
 JLabel accompanyLabel= new JLabel("是否陪同：");  
 accompanyLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 accompanyLabel.setBounds(113, 377, 100, 18);  
 getContentPane().add(accompanyLabel);  
   
 accompanyField = new JTextField();  
 accompanyField.setText("是否选择导游陪同？");  
 accompanyField.setToolTipText("");  
 accompanyField.setBounds(191, 374, 240, 28);  
 getContentPane().add(accompanyField);  
 accompanyField.setColumns(10);  
   
 JLabel foodsLabel= new JLabel("是否食宿：");  
 foodsLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 foodsLabel.setBounds(113, 427, 100, 18);  
 getContentPane().add(foodsLabel);  
   
 foodsField = new JTextField();  
 foodsField.setText("是否选择宾馆住宿？");  
 foodsField.setBounds(191, 424, 240, 28);  
 getContentPane().add(foodsField);  
 foodsField.setColumns(10);  
   
 JButton button\_1 = new JButton("查看旅游团信息");  
 button\_1.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 TouristGroup tg = new TouristGroup();  
 tg.setVisible(true);  
 }  
 });  
 button\_1.setBounds(453, 324, 158, 27);  
 button\_1.setFocusPainted(false);//去掉按钮周围的焦点框  
 button\_1.setFont(new Font("微软雅黑", Font.*BOLD*, 13));  
 button\_1.setBackground(new Color(243, 152, 0));  
 button\_1.setForeground(Color.*WHITE*);  
 getContentPane().add(button\_1);  
   
   
 JButton button\_2 = new JButton("是");  
 button\_2.setBounds(453, 376, 72, 27);  
 button\_2.setFocusPainted(false);//去掉按钮周围的焦点框  
 button\_2.setFont(new Font("微软雅黑", Font.*BOLD*, 13));  
 button\_2.setBackground(new Color(0, 189, 93));  
 button\_2.setForeground(Color.*WHITE*);  
 getContentPane().add(button\_2);  
 button\_2.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 accompanyField.setText("是");  
 }  
 });  
   
 JButton button\_3 = new JButton("否");  
 button\_3.setBounds(539, 376, 72, 27);  
 button\_3.setFocusPainted(false);//去掉按钮周围的焦点框  
 button\_3.setFont(new Font("微软雅黑", Font.*BOLD*, 13));  
 button\_3.setBackground(new Color(238, 10, 36));  
 button\_3.setForeground(Color.*WHITE*);  
 getContentPane().add(button\_3);  
 button\_3.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 accompanyField.setText("无");  
 }  
 });  
   
 JButton button\_4 = new JButton("是");  
 button\_4.setBounds(453, 424, 72, 27);  
 button\_4.setFocusPainted(false);//去掉按钮周围的焦点框  
 button\_4.setFont(new Font("微软雅黑", Font.*BOLD*, 13));  
 button\_4.setBackground(new Color(0, 189, 93));  
 button\_4.setForeground(Color.*WHITE*);  
 getContentPane().add(button\_4);  
 button\_4.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 foodsField.setText("是");  
 }  
 });  
   
 JButton button\_5 = new JButton("否");  
 button\_5.setBounds(539, 424, 72, 27);  
 button\_5.setFocusPainted(false);//去掉按钮周围的焦点框  
 button\_5.setFont(new Font("微软雅黑", Font.*BOLD*, 13));  
 button\_5.setBackground(new Color(238, 10, 36));  
 button\_5.setForeground(Color.*WHITE*);  
 getContentPane().add(button\_5);  
 button\_5.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 foodsField.setText("无");  
 }  
 });  
   
 //设置文本框初始值  
 nameField.setText(tourist.getTourist\_name());  
 String t = tourist.getTourist\_sex();  
 sexcomboBox.setSelectedItem(t);  
 ageField.setText(tourist.getTourist\_age());  
 IDcardField.setText(tourist.getTourist\_idCard());  
 addressField.setText(tourist.getTourist\_address());  
 phoneField.setText(tourist.getTourist\_phone());  
 touristNumberField.setText(tourist.getGroup\_num());  
 accompanyField.setText(tourist.getAccompanied());  
 foodsField.setText(tourist.getAccommodation());  
   
 //确认修改按钮  
 JButton submitButton = new JButton("确认修改");  
 submitButton.setFont(new Font("微软雅黑", Font.*PLAIN*, 18));  
 submitButton.setBounds(320, 495, 150, 33);  
 submitButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 submitButton.setBackground(new Color(78, 110, 242));  
 submitButton.setForeground(Color.*WHITE*);  
 getContentPane().add(submitButton);  
   
 submitButton.addActionListener(new ActionListener() {  
   
 @Override  
 public void actionPerformed(ActionEvent e) {  
 String id = tourist.getTourist\_num();  
 name = nameField.getText();  
 sex = sexcomboBox.getSelectedItem().toString();  
 age = ageField.getText();  
 IDcard = IDcardField.getText();  
 address = addressField.getText();  
 phone = phoneField.getText();  
 touristNumber = touristNumberField.getText();  
 accompany = accompanyField.getText();  
 if (accompany.equals("是否选择导游陪同？")) {  
 accompany="无";  
 }  
 foods = foodsField.getText();  
 if (foods.equals("是否选择宾馆住宿？")) {  
 foods="无";  
 }  
 //判断输入的信息是否为空，是否完整  
 if (name.equals("")||sex.equals("")||age.equals("")||IDcard.equals("")||address.equals("")||phone.equals("")||touristNumber.equals("")||accompany.equals("")||foods.equals("")) {  
 JOptionPane.*showMessageDialog*(null, "请输入完整信息！");  
 } else {  
 //判断身份证号码  
 if (!ValidateUtils.*IDcard*(IDcard)) {  
 JOptionPane.*showMessageDialog*(null, "身份证号码错误！请检查！");  
 } else {  
 String i = select.getString("SELECT user\_id FROM `user` WHERE user\_state='已登录'");  
 String sql = "UPDATE tourist SET tourist\_name='"+name+"',tourist\_sex='"+sex+"',tourist\_age='"+age+"',tourist\_idcard='"+IDcard+"',tourist\_address='"+address+"',tourist\_phone='"+phone+"',group\_num='"+touristNumber+"',accompanied='"+accompany+"',accommodation='"+foods+"' WHERE tourist\_num='"+id+"';";  
 int result = updata.addData(sql);  
 //判断手机号  
 String regex = "^((13[0-9])|(14[5|7])|(15([0-3]|[5-9]))|(17[013678])|(18[0,5-9]))\\d{8}$";  
 if(phone.length() != 11){  
 JOptionPane.*showMessageDialog*(null, "手机号应为11位数！");  
 }else{  
 Pattern p = Pattern.*compile*(regex);  
 Matcher m = p.matcher(phone);  
 boolean isMatch = m.matches();  
 if(!isMatch){  
 JOptionPane.*showMessageDialog*(null, "您的手机号" + phone + "是错误格式！");  
 }else {  
 //判断插入结果  
 if (result>0) {  
 JOptionPane.*showMessageDialog*(null, "修改成功！");  
 dispose();  
// Registration\_Management r = new Registration\_Management();  
// r.dispose();  
// r.setVisible(true);  
 } else {  
 JOptionPane.*showMessageDialog*(null, "修改失败，请与管理员联系！");  
 }  
 }  
 }

## 4.5订单管理界面

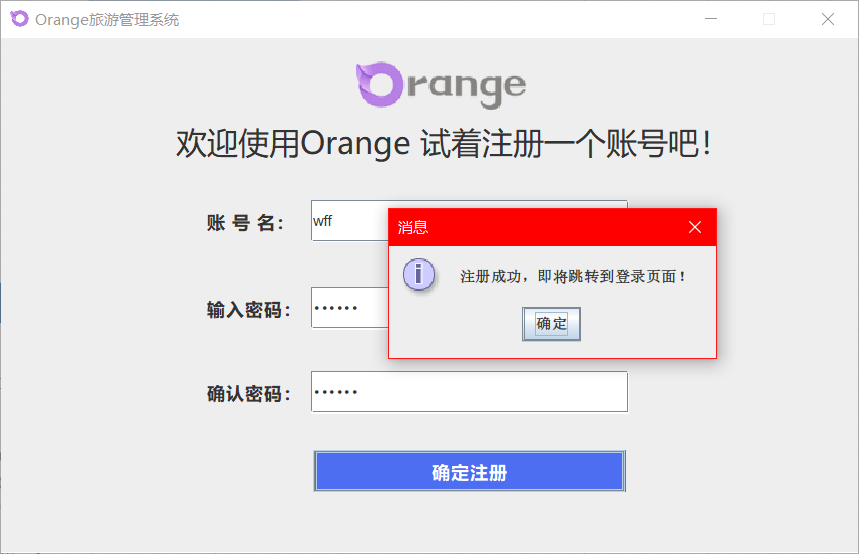


只有管理员可以进入订单管理界面，在订单管理界面操作，直接修改数据库的信息

private static final long *serialVersionUID* = 1L;  
  
Select select = new Select();  
Updata updata = new Updata();  
JTable jTable;  
DefaultTableModel dt;  
  
Object[] header = {"订单编号","客户姓名","性别","年龄","身份证号","家庭住址","联系电话","所属旅游团","陪同","食宿"};  
Object[][] data = select.getTourist("SELECT \* FROM tourist");  
private JTextField orderNumberField;//订单编号  
private JTextField touristNumberField;//团号  
private JTextField customernameField;//客户姓名  
  
public Registration\_Management() {  
 super("订单信息管理");  
 this.setBounds(0, 0, 1200, 700);  
 this.setLocationRelativeTo(null);//让窗口在屏幕中间显示  
 this.setResizable(false);//让窗口大小不可改变  
 getContentPane().setLayout(null);  
   
 //设置窗口图标  
 Toolkit tk = Toolkit.*getDefaultToolkit*();  
 Image frameImage=tk.createImage("img/logo.png");   
 this.setIconImage(frameImage);  
   
 //logo添加  
 JLabel jl\_logo = new JLabel();  
 jl\_logo.setIcon(new ImageIcon("img/logo2.png"));//文件路径  
 jl\_logo.setBounds(993, 15, 200, 50);  
 this.add(jl\_logo);  
   
 //当前时间  
 JLabel dqsj = new JLabel("当前时间 ：");  
 dqsj.setBounds(875, 70, 85, 35);  
 dqsj.setFont(new Font("微软雅黑", Font.*BOLD*, 15));  
 dqsj.setForeground(Color.*white*);  
 getContentPane().add(dqsj);  
   
 JLabel time1 = new JLabel();  
 time1.setBounds(960, 71, 236, 35);  
 time1.setFont(new Font("微软雅黑", Font.*CENTER\_BASELINE*, 16));  
 time1.setForeground(Color.*white*);  
 getContentPane().add(time1);  
 this.setTimer(time1);  
   
 //图片添加  
 JLabel jl = new JLabel();  
 jl.setIcon(new ImageIcon("img/timebg.png"));//文件路径  
 jl.setBounds(860, 62, 750, 40);  
 this.add(jl);   
   
 //窗口关闭时打开主界面窗口  
 this.addWindowListener(new WindowAdapter() {  
   
 public void windowClosing(WindowEvent e) {  
 super.windowClosing(e);  
 //加入动作  
 MainJframe m = new MainJframe();  
 m.setVisible(true);  
 }  
 });  
   
 //创建表模型  
 dt=new DefaultTableModel(data,header){  
 private static final long *serialVersionUID* = 1L;  
 //设置表格内容不可被编辑  
 public boolean isCellEditable(int row, int column) {  
 return false;//返回true表示能编辑，false表示不能编辑  
 }  
 };  
 jTable=new JTable(dt);//创建表格  
 jTable.getTableHeader().setFont(new Font("微软雅黑", Font.*BOLD*, 14));//设置表头名称字体样式  
 jTable.getTableHeader().setForeground(Color.*white*);//设置表头名称字体颜色  
 jTable.getTableHeader().setReorderingAllowed(false);//设置表头不允许拖动  
 jTable.getTableHeader().setBackground(new Color(68, 114, 196));//设置表头背景色  
   
 //设置表头高度  
 Dimension size = jTable.getTableHeader().getPreferredSize();  
 size.height = 40;//设置新的表头高度40  
 jTable.getTableHeader().setPreferredSize(size);  
   
 //设置行高  
 jTable.setRowHeight(30);//指定每一行的行高50  
   
 //设置表格列宽  
 int[] length = {100 , 100 , 50 , 50 , 200 , 300 , 150 , 100 , 50 , 50 };//表格的列宽  
 *TableColumnModel* model0 = jTable.getColumnModel();  
 for (int i = 0 ; i<model0.getColumnCount() ; i++)  
 {  
 TableColumn column = model0.getColumn(i);//获取列对象  
 column.setPreferredWidth(length[i]);//设置列宽  
 }  
   
 jTable.setFont(new Font("微软雅黑", Font.*PLAIN*, 14));//设置表格字体  
 int v=*ScrollPaneConstants*.*VERTICAL\_SCROLLBAR\_AS\_NEEDED*;//水平滚动条  
 int h=*ScrollPaneConstants*.*HORIZONTAL\_SCROLLBAR\_AS\_NEEDED*;//垂直滚动条  
 JScrollPane jsp=new JScrollPane(jTable,v,h);//创建滚动容器  
 jsp.setBounds(20, 120, 1150, 530);  
 getContentPane().add(jsp);  
   
 //订单编号  
 JLabel orderNumberLabel = new JLabel("订单编号");  
 orderNumberLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 orderNumberLabel.setBounds(20, 70, 89, 18);  
 getContentPane().add(orderNumberLabel);  
   
 orderNumberField = new JTextField();  
 orderNumberField.setBounds(90, 70, 120, 24);  
 getContentPane().add(orderNumberField);  
 orderNumberField.setColumns(10);//设置内容容量  
   
 //团号  
 JLabel touristNumberLabel = new JLabel("团号");  
 touristNumberLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 touristNumberLabel.setBounds(240, 70, 52, 18);  
 getContentPane().add(touristNumberLabel);  
   
 touristNumberField = new JTextField();  
 touristNumberField.setColumns(10);  
 touristNumberField.setBounds(280, 70, 114, 24);  
 getContentPane().add(touristNumberField);  
   
 //客户姓名  
 JLabel customernameLabel = new JLabel("客户姓名");  
 customernameLabel.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 customernameLabel.setBounds(420, 70, 80, 18);  
 getContentPane().add(customernameLabel);  
   
 customernameField = new JTextField();  
 customernameField.setColumns(10);  
 customernameField.setBounds(490, 70, 114, 24);  
 getContentPane().add(customernameField);  
   
 //查询按钮  
 JButton queryButton = new JButton("查询");  
 queryButton.setBounds(20, 20, 110, 30);  
 queryButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 queryButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 queryButton.setBackground(new Color(78, 110, 242));  
 queryButton.setForeground(Color.*WHITE*);  
 getContentPane().add(queryButton);  
   
 queryButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 String touristNumber,th,name,sql;  
 touristNumber = orderNumberField.getText();  
 th = touristNumberField.getText();  
 name = customernameField.getText();  
   
 if (touristNumber.equals("")&&th.equals("")&&name.equals("")) {  
 sql = "SELECT \* FROM tourist";  
 }  
 else if(touristNumber.equals("")&&th.equals("")){  
 sql = "SELECT \* FROM tourist WHERE tourist\_name='"+name+"';";  
 }  
 else if(name.equals("")&&th.equals("")){  
 sql = "SELECT \* FROM tourist WHERE tourist\_num='"+touristNumber+"';";  
 }  
 else if(touristNumber.equals("")&&name.equals("")){  
 sql = "SELECT \* FROM tourist WHERE group\_num='"+th+"';";  
 }  
 else if(touristNumber.equals("")){  
 sql = "SELECT \* FROM tourist WHERE group\_num='"+th+"' and tourist\_name='"+name+"';";  
 }  
 else if(th.equals("")){  
 sql = "SELECT \* FROM tourist WHERE tourist\_num='"+touristNumber+"' and tourist\_name='"+name+"';";  
 }  
 else if(name.equals("")){  
 sql = "SELECT \* FROM tourist WHERE tourist\_num='"+touristNumber+"' and group\_num='"+th+"';";  
 }  
 else {  
 sql = "SELECT \* FROM tourist WHERE tourist\_num='"+touristNumber+"' and group\_num='"+th+"' and tourist\_name='"+name+"';";  
 }  
 data = select.getTourist(sql);  
 dt.setDataVector(data,header);  
   
 //设置表格列宽  
 int[] length = {100 , 100 , 50 , 50 , 200 , 300 , 150 , 100 , 50 , 50 };//表格的列宽  
 *TableColumnModel* model0 = jTable.getColumnModel();  
 for (int i = 0 ; i<model0.getColumnCount() ; i++)  
 {  
 TableColumn column = model0.getColumn(i);//获取列对象  
 column.setPreferredWidth(length[i]);//设置列宽  
 }  
 }  
 });  
   
 //修改按钮  
 JButton alterButton = new JButton("修改");  
 alterButton.setBounds(150, 20, 110, 30);  
 alterButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 alterButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 alterButton.setBackground(new Color(101, 113, 175));  
 alterButton.setForeground(Color.*WHITE*);  
 getContentPane().add(alterButton);  
 alterButton.addActionListener(new ActionListener() {  
   
 public void actionPerformed(ActionEvent arg0) {  
 if (jTable.getSelectedRow()<0) {  
 JOptionPane.*showMessageDialog*(null, "您未选中要修改的数据！");  
 }   
 else {  
 //获取用户选择的数据  
 String name,sex,age,IDcard,address,phone,touristNumber,accompany,foods;  
 String id = jTable.getValueAt(jTable.getSelectedRow(), 0).toString();  
 String user\_id = select.getString("SELECT user\_id FROM `user` WHERE user\_state='已登录'");  
   
 name = jTable.getValueAt(jTable.getSelectedRow(), 1).toString();  
 sex = jTable.getValueAt(jTable.getSelectedRow(), 2).toString();  
 age = jTable.getValueAt(jTable.getSelectedRow(), 3).toString();  
 IDcard = jTable.getValueAt(jTable.getSelectedRow(), 4).toString();  
 address = jTable.getValueAt(jTable.getSelectedRow(), 5).toString();  
 phone = jTable.getValueAt(jTable.getSelectedRow(), 6).toString();  
 touristNumber = jTable.getValueAt(jTable.getSelectedRow(), 7).toString();  
 accompany = jTable.getValueAt(jTable.getSelectedRow(), 8).toString();  
 foods = jTable.getValueAt(jTable.getSelectedRow(), 9).toString();  
   
 Tourist tourist=new Tourist(id, user\_id, name, sex, age, IDcard, address, phone, touristNumber, accompany, foods);  
 RegistrationInfo\_Change frame=new RegistrationInfo\_Change(tourist);  
   
 frame.setVisible(true);  
 }  
 }  
 });  
   
 //添加按钮  
 JButton addButton = new JButton("添加");  
 addButton.setBounds(280, 20, 110, 30);  
 addButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 addButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 addButton.setBackground(new Color(231, 121, 24));  
 addButton.setForeground(Color.*WHITE*);  
 getContentPane().add(addButton);  
   
 addButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 //打开添加界面  
 Registration\_Info r = new Registration\_Info();  
 r.setVisible(true);  
 }  
 });  
   
 //删除按钮  
 JButton deleteButton = new JButton("删除");  
 deleteButton.setBounds(410, 20, 110, 30);  
 deleteButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 deleteButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 deleteButton.setBackground(new Color(216, 30, 6));  
 deleteButton.setForeground(Color.*WHITE*);  
 getContentPane().add(deleteButton);  
 deleteButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 if (jTable.getSelectedRow()<0) {  
 JOptionPane.*showMessageDialog*(null, "您未选中要删除的数据！");  
 }   
 else {  
 //获取用户选择的数据  
 String id = jTable.getValueAt(jTable.getSelectedRow(), 0).toString();  
 String name = jTable.getValueAt(jTable.getSelectedRow(), 1).toString();  
 int result = JOptionPane.*showConfirmDialog*(null,"您确定要删除用户 “"+name+"” 的报名信息吗?","删除提示",0,1);  
 if(result == JOptionPane.*OK\_OPTION*){  
 int i = updata.addData("DELETE FROM tourist WHERE tourist\_num='"+id+"';");  
 if (i>0){  
 JOptionPane.*showMessageDialog*(null, "用户 “"+name+"” ，已被删除成功！");  
 }   
 else {  
 JOptionPane.*showMessageDialog*(null, "删除失败！");  
 }  
 data=select.getTourist("SELECT \* FROM tourist");  
 dt.setDataVector(data,header);  
   
 //设置表格列宽  
 int[] length = {100 , 100 , 50 , 50 , 200 , 300 , 150 , 100 , 50 , 50 };//表格的列宽  
 *TableColumnModel* model0 = jTable.getColumnModel();  
 for (int i1 = 0 ; i1<model0.getColumnCount() ; i1++)  
 {  
 TableColumn column = model0.getColumn(i1);//获取列对象  
 column.setPreferredWidth(length[i1]);//设置列宽  
 }  
 }  
 }  
 }  
 });  
   
 //提示按钮  
 JButton tipsButton = new JButton("提示");  
 tipsButton.setBounds(540, 20, 110, 30);  
 tipsButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 tipsButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 tipsButton.setBackground(new Color(78, 110, 242));  
 tipsButton.setForeground(Color.*WHITE*);  
 getContentPane().add(tipsButton);  
 tipsButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 JOptionPane.*showMessageDialog*(null, "<html>查询：直接点击将列出所有报名信息，也可填写游客编号、团号和性别查询。<br>修改：点击游客将会将游客编号绑定到文本框中，可以对该游客编号对应的游客进行团号和性别修改。<br>删除：点击要删除的信息，点击删除即可。<br>添加：对报名信息进行添加。</html>");  
 }  
 });  
   
 //刷新按钮  
 JButton flushButton = new JButton("刷新");  
 flushButton.setBounds(670, 20, 110, 30);  
 flushButton.setFocusPainted(false);//去掉按钮周围的焦点框  
 flushButton.setFont(new Font("微软雅黑", Font.*BOLD*, 14));  
 flushButton.setBackground(new Color(28, 175, 8));  
 flushButton.setForeground(Color.*WHITE*);  
 getContentPane().add(flushButton);  
 flushButton.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent arg0) {  
 data=select.getTourist("SELECT \* FROM tourist");  
 dt.setDataVector(data,header);  
   
 //设置表格列宽  
 int[] length = {100 , 100 , 50 , 50 , 200 , 300 , 150 , 100 , 50 , 50 };//表格的列宽  
 *TableColumnModel* model0 = jTable.getColumnModel();  
 for (int i = 0 ; i<model0.getColumnCount() ; i++)  
 {  
 TableColumn column = model0.getColumn(i);//获取列对象  
 column.setPreferredWidth(length[i]);//设置列宽  
 }  
 }  
 });  
   
 //设置单元格内容居中显示  
 DefaultTableCellRenderer r = new DefaultTableCellRenderer();   
 r.setHorizontalAlignment(JLabel.*CENTER*);   
 jTable.setDefaultRenderer(Object.class, r);  
}  
   
//设置Timer 1000ms实现一次动作 实际是一个线程  
private void setTimer(JLabel time) {  
 final JLabel varTime = time;  
 Timer timeAction = new Timer(100, new ActionListener() {  
 public void actionPerformed(ActionEvent e) {  
 long timemillis = System.*currentTimeMillis*();  
 // 转换日期显示格式  
 SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  
 varTime.setText(df.format(new Date(timemillis)));  
 }  
 });  
 timeAction.start();  
}

五、系统测试

## 5.1注册测试



## 5.2登录测试



## 5.3报名旅游



## 5.5订单管理



## 5.6退出系统



六、结论与心得

通过此次课程设计，使我更加扎实的掌握了有关java方面的知识，在设计过程中虽然遇到了一些问题，但经过一次又一次的思考，一遍又一遍的检查终于找出了原因所在，也暴露出了前期我在这方面的知识欠缺和经验不足。实践出真知，通过亲自动手制作，使我们掌握的知识不再是纸上谈兵。 过而能改，善莫大焉。在课程设计过程中，我们不断发现错误，不断改正，不断领悟，不断获取。

这次的课程设计使我对抽象的理论有了具体的认识。通过这次课程设计，我熟悉了java；了解了java方法；以及如何提高java的性能等等，掌握了java的方法和技术，通过查询资料，也了解了java原理。

在自己设计的这个系统中还存在很多的问题，在注册了新用户之后，登录用于是数据库中存放的第一个用户，在多次切换用户的操作中会出现系统崩溃的现象，在提交旅游订单的时候，自己加的规则好像没什么用，出现了身份证不够16位依然可以提交，对于普通用户和管理员页面的设计不合理等等等一系列bug。

通过这个课程设计，希望自己以后在设计系统的时候可以多考虑一些问题，防止某些不必要的bug出现，增加数据库的安全性，增加页面的美化。