编号：

桂林电子科技大学

课程设计报告

|  |  |
| --- | --- |
| 课程名称： | 数据库系统原理 |
| 题 目： | 车票售票管理子系统 |
| 学 院： | 计算机工程学院 |
| 专 业： | 网络工程 |
| 学 号： | 191603010107 |
| 姓 名： | 刘健成 |
| 指导教师： |  |

2021年5月

摘要

中国高铁里程居世界之首，占全球高铁总里程的60%以上，具不完全统计，中国的在役高铁列车已经超过了2500列，乘车人数逐年增加。当今社会的生活节奏越来越快，人们对生活效率的要求也越来越高，在车站内排队购票不仅浪费了大量的时间，而且乘客无法实时了解所需车票是否还有剩余，容易耽误乘客的旅行。不仅如此，用人工来操作海量数据的工作效率不高，且成本较大。随着信息化时代的到来，利用计算机的高速运算来帮助人们处理数据已成必然，通过利用计算机来管理车站内的数据，便于乘客了解所需信息，提高乘客购票效率，同时也利于车站内部人员的管理。该系统主要包括用户端子系统、管理端子系统，充分利用网络快速地处理信息，满足了乘客与车站管理人员之间的信息交与需求。本系统的开发为车站管理和乘客购票提供了较大的便捷，使乘客足不出户就可以购买车票，也为车站提供了一个良好的售票窗口，减轻了工作负担。

**关键词：海量数据 计算机 高速运算 处理数据**

Abstract

The mileage of China's high-speed railway ranks first in the world, accounting for more than 60% of the total mileage of the world's high-speed railway. According to incomplete statistics, there are more than 2500 high-speed trains in service in China, and the number of passengers is increasing year by year. In today's society, the pace of life is faster and faster, people's demand for life efficiency is also higher and higher. Queuing in the station to buy tickets not only wastes a lot of time, but also passengers can't know whether there is any remaining tickets in real time, which is easy to delay passengers' travel. Moreover, manual operation of massive data is inefficient and costly. With the advent of the information age, it is inevitable to use the high-speed operation of the computer to help people deal with the data. By using the computer to manage the data in the station, it is convenient for passengers to understand the required information, improve the efficiency of passenger ticket purchase, and also conducive to the management of the internal staff of the station. The system mainly includes user terminal system and management terminal system. It makes full use of the network to process information quickly and meets the information exchange needs between passengers and station management personnel. The development of the system provides great convenience for the station management and passengers to buy tickets, so that passengers can buy tickets at home, and also provides a good ticket window for the station, reducing the workload.

**Key words: massive data computer high speed operation data processing**

# 目 录

[一 绪论 1](#_Toc73567822)

[1.1系统开发的背景描述 1](#_Toc73567823)

[1.2系统的开发环境 1](#_Toc73567824)

[1.2.1 Java编程语言介绍 1](#_Toc73567825)

[1.2.2 Eclipse 开发平台介绍 1](#_Toc73567826)

[1.2.3 Microsoft SQL Server 平台介绍 2](#_Toc73567827)

[1.2.4 JDBC接口介绍 2](#_Toc73567828)

[二 系统需求分析 3](#_Toc73567829)

[2.1系统可行性分析 3](#_Toc73567830)

[2.2系统功能需求分析 3](#_Toc73567831)

[三 系统设计 4](#_Toc73567832)

[3.1系统功能总体框架图 4](#_Toc73567833)

[3.2各功能模块设计 5](#_Toc73567834)

[3.2.1 用户注册模块 5](#_Toc73567835)

[3.2.2 用户登录模块 6](#_Toc73567836)

[3.2.3 用户找回密码模块 7](#_Toc73567837)

[3.2.4 查询列车信息模块 8](#_Toc73567838)

[3.2.5 菜单模块 9](#_Toc73567839)

[3.2.6 管理员登录模块 10](#_Toc73567840)

[3.2.7 管理员界面模块 10](#_Toc73567841)

[3.3数据库设计 11](#_Toc73567842)

[3.3.1 E-R图 11](#_Toc73567843)

[3.3.2 关系模式 13](#_Toc73567844)

[四 系统实现 14](#_Toc73567845)

[4.1 数据库和数据表实现 14](#_Toc73567846)

[4.1.1 数据库的建立和用户数据表的建立 14](#_Toc73567847)

[4.1.2 管理员数据表的建立 15](#_Toc73567848)

[4.1.3 列车数据表的建立 15](#_Toc73567849)

[4.1.4 车票数据表的建立 16](#_Toc73567850)

[4.1.5 购票人数据表的建立 17](#_Toc73567851)

[4.2通过视图，JAVA，GUI,JDBC等实现每个模块的功能 17](#_Toc73567852)

[4.2.1 显示用户查询车站列车信息的视图 17](#_Toc73567853)

[4.2.2 显示管理员查询车站列车信息的视图 17](#_Toc73567854)

[4.2.3 用户注册与登录模块，管理员登录模块 18](#_Toc73567855)

[4.2.3 用户找回密码模块 23](#_Toc73567856)

[4.2.4 查询列车信息与菜单模块 25](#_Toc73567857)

[4.2.5 管理员界面模块 48](#_Toc73567858)

[五 系统测试 58](#_Toc73567859)

[5.1 用户注册功能测试 58](#_Toc73567860)

[5.2 用户登录功能测试 59](#_Toc73567861)

[5.3 用户找回密码功能测试 59](#_Toc73567862)

[5.4 车站查询与日期查询 60](#_Toc73567863)

[5.5 购票功能测试 61](#_Toc73567864)

[5.6 退票功能测试 63](#_Toc73567865)

[5.7 管理员登录功能测试 64](#_Toc73567866)

[5.8 管理员增加列车信息功能测试 65](#_Toc73567867)

[5.9 管理员删除列车信息功能测试 67](#_Toc73567868)

[6.0 管理员修改列车信息功能测试 67](#_Toc73567869)

[6.1 管理员查询列车信息功能测试 69](#_Toc73567870)

[六 结论 70](#_Toc73567871)

[参考文献 70](#_Toc73567872)

[七 附录 71](#_Toc73567873)

# 绪论

## 1.1系统开发的背景描述

铁路票务管理是一件不允许一点失误的事情，随着经济水平的不断提高，交通道路环境的不断改善，人口流动频繁，车站信息量迅猛增长，仅仅依靠线下购票无法为每一位乘客办理业务。为了加快业务办理速度，满足乘客快捷出行的需求，减轻车站工作人员负担，设计开发一种高效的车票管理系统就显得颇为重要。

计算机是20世纪最先进的科学技术发明之一，对人类的生产活动和社会活动产生了极其重要的影响，并以强大的生命力飞速发展。它的应用领域从最初的军事科研应用扩展到社会的各个领域，已形成了规模巨大的计算机产业，带动了全球范围的技术进步，由此引发了深刻的社会变革，计算机已遍及一般学校、企事业单位，进入寻常百姓家，成为信息社会中必不可少的工具。

随着信息化时代的到来，计算机应用于火车站售票的日常管理为铁路票务管理领域的飞速发展提供了无限的潜力，采用计算机管理信息系统已成为铁路票务管理科学化和现代化的重要标志。

## 1.2系统的开发环境

### 1.2.1 Java编程语言介绍

java是一门面向对象编程语言，不仅吸收了C++语言的各种优点，还摒弃了C++里难以理解的多继承、指针等概念，因此Java语言具有功能强大和简单易用两个特征。Java语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。

Java具有简单性、面向对象、分布式、健壮性、安全性、平台独立与可移植性、多线程、动态性等特点 。Java可以编写桌面应用程序、Web应用程序、分布式系统和嵌入式系统应用程序等 。

### 1.2.2 Eclipse 开发平台介绍

Eclipse是著名的跨平台的自由集成开发环境（IDE）。最初主要用来Java语言开发，通过安装不同的插件Eclipse可以支持不同的计算机语言，比如C++和Python等开发工具。

Eclipse 是一个开放[源代码](https://baike.baidu.com/item/%E6%BA%90%E4%BB%A3%E7%A0%81/3969)的、基于[Java](https://baike.baidu.com/item/Java/85979)的可扩展开发平台。就其本身而言，它只是一个框架和一组服务，用于通过插件组件构建开发环境。幸运的是，Eclipse 附带了一个标准的插件集，包括Java[开发工具](https://baike.baidu.com/item/%E5%BC%80%E5%8F%91%E5%B7%A5%E5%85%B7)（Java Development Kit，[JDK](https://baike.baidu.com/item/JDK/1011)）。

### 1.2.3 Microsoft SQL Server 平台介绍

SQL Server 是Microsoft 公司推出的关系型数据库管理系统。具有使用方便可伸缩性好与相关软件集成程度高等优点，可跨越从运行Microsoft Windows 98 的膝上型电脑到运行Microsoft Windows 2012 的大型多处理器的服务器等多种平台使用。

Microsoft SQL Server 是一个全面的数据库平台，使用集成的商业智能 (BI)工具提供了企业级的数据管理。Microsoft SQL Server 数据库引擎为关系型数据和结构化数据提供了更安全可靠的存储功能，使用户可以构建和管理用于业务的高可用和高性能的数据应用程序。

### 1.2.4 JDBC接口介绍

Java数据库连接，（Java Database Connectivity，简称JDBC）是Java语言中用来规范客户端程序如何来访问数据库的应用程序接口，是一种标准Java应用编程接口（ JAVA API），用来连接 Java 编程语言和广泛的数据库，提供了诸如查询和更新数据库中数据的方法。JDBC 的 API 支持两层和三层处理模式进行数据库访问，但一般的 JDBC 架构由两层处理模式组成：

* JDBC API: 提供了应用程序对 JDBC 管理器的连接。
* JDBC Driver API: 提供了 JDBC 管理器对驱动程序连接。

JDBC API 使用驱动程序管理器和数据库特定的驱动程序来提供异构（heterogeneous）数据库的透明连接。

JDBC 驱动程序管理器可确保正确的驱动程序来访问每个数据源。该驱动程序管理器能够支持连接到多个异构数据库的多个并发的驱动程序。

以下是结构图，其中显示了驱动程序管理器相对于在 JDBC 驱动程序和 Java 应用程序所处的位置。

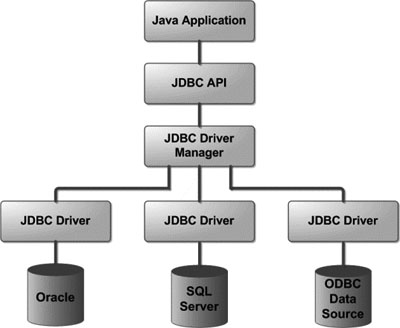


图1 ‑1 JDBC结构图

常见的 JDBC 组件

JDBC 的 API 提供了以下接口和类：

**DriverManager** ：这个类管理一系列数据库驱动程序。匹配连接使用通信子协议从 JAVA 应用程序中请求合适的数据库驱动程序。识别 JDBC 下某个子协议的第一驱动程序将被用于建立数据库连接。

**Driver** : 这个接口处理与数据库服务器的通信。你将很少直接与驱动程序互动。相反，你使用 DriverManager 中的对象，它管理此类型的对象。它也抽象与驱动程序对象工作相关的详细信息。

**Connection** : 此接口具有接触数据库的所有方法。该连接对象表示通信上下文，即，所有与数据库的通信仅通过这个连接对象进行。

**Statement** : 使用创建于这个接口的对象将 SQL 语句提交到数据库。除了执行存储过程以外，一些派生的接口也接受参数。

**ResultSet** : 在你使用语句对象执行 SQL 查询后，这些对象保存从数据获得的数据。它作为一个迭代器，让您可以通过它的数据来移动。

**SQLException** : 这个类处理发生在数据库应用程序的任何错误。

# 系统需求分析

## 2.1系统可行性分析

该开发系统需要的应用软件、硬件系统易于获得。开发成本较低。而引进使用本系统后，与传统方式相比，具有高效率、低成本、高质量的特点，可以节省不少资源。所以，从经济的总体角度来看，该系统可行。

软件方面包括Java，GUI，JDBC,以及SQL Server数据库，硬件方面为Microsoft Windows10开发系统环境，所以从软硬件方法看，该系统可行。

## 2.2系统功能需求分析

**本系统主要实现用户注册功能，用户登录功能，用户找回密码功能，管理员登录功能，列车信息查询功能，日期查询功能，用户购票功能和用户退票功能，管理员对列车信息的增删改查功能。**

**用户注册功能**：首先查询注册的用户名是否在数据库中已存在，如果存在，则提示用户“用户名已经存在，请重新输入。”；如果用户未输入用户密码或密保或密保答案，则提示用户“请输入注册的各项内容。”，并将用户的各个信息导入到数据库对应的数据表中；若注册成功，则提示用户“注册成功”，并跳转到登录界面。

**用户登录功能**：查询用户所输入的用户名及密码是否与数据库存储的信息对应，若不对应，则提示用户“用户名或密码错误，请重新输入。”；若对应，则关闭登录界面并跳转到菜单模块。

**用户找回密码功能：**查询用户所输入的用户名，密保以及密保答案是否对应，若不对应，则提示用户“输入错误”；若对应，则输出该用户的密码。

**管理员登录功能：**此功能与用户登录功能一致，登录成功后会进入管理员界面，对列车信息进行增删改查的操作。

**列车信息查询功能**：根据用户选择的出发站和终点站，查询对应列车，并且输出列车相关信息，如列车号，发车日期，发车时间，到达时间，起始站，终点站，各类票种的剩余票数；若两站之间并无列车运营，则提示用户“很抱歉！未找到您所查询的车站间的列车信息”。

**日期查询功能**：根据用户输入的日期，查询在该日期里发车的所有列车的相关信息。

**用户购票功能**：用户购买所需列车的相关车票，并将用户信息和车票信息记录在数据库里的客户信息表中，相关列车的相关票数-1。

**用户退票功能：**用户退还车票，数据库根据车票信息删除用户持票信息，相关列车的相关票数+1。

**管理员对列车信息的增删改查功能：**遇到特殊情况时。管理员能及时对列车的信息进行增删改查，以免耽误乘客的旅行。

# 系统设计

## 3.1系统功能总体框架图

车票管理子系统功能总体框架图如下：



图 3‑1车票管理子系统总体功能框架图

## 3.2各功能模块设计

### 3.2.1 用户注册模块

用户注册模块的结构图如下：



图 3‑2 用户注册模块结构图

用户注册模块的逻辑流程图如下：



图 3‑3 用户注册模块的逻辑流程图

### 3.2.2 用户登录模块

用户登录模块的结构图如下：



图 3‑4 用户登录模块结构图

用户登录模块的逻辑流程图如下：



图 3‑5 用户登录模块的逻辑流程图

### 3.2.3 用户找回密码模块

用户找回密码模块的结构图如下：



图 3‑6 用户找回密码模块结构图

用户找回密码模块的逻辑流程图如下：



图 3‑7 用户找回密码模块逻辑流程图

### 3.2.4 查询列车信息模块

查询列车信息模块的结构图如下：



图 3‑8 查询列车信息模块结构图

查询列车信息模块的逻辑流程图如下：



图 3‑9 查询列车信息模块逻辑流程图

### 3.2.5 菜单模块

菜单模块的结构图如下：



图 3‑10 菜单模块结构图

菜单模块的逻辑流程图如下：



图 3‑11 菜单模块逻辑流程图

### 3.2.6 管理员登录模块

管理员登录模块的结构依靠于上图3-4用户登录模块结构图。

管理员登录模块的逻辑流程图如下：



图 3‑12 管理员登录模块的逻辑流程图

### 3.2.7 管理员界面模块

管理界面模块的结构图如下：



图 3‑13 管理员界面结构图

管理界面模块的逻辑流程图如下：



图 3‑14 管理界面模块逻辑流程图

## 3.3数据库设计

### 3.3.1 E-R图

**（1）用户实体：**

保存注册系统的用户的信息，主键为用户名，用来作为每一个用户的唯一标记，用户实体的每一个属性都不能为空。用户实体E-R图如下：



图 3‑15 用户实体E-R图

**（2）管理员实体：**

主键为管理员编号，用来作为每一个管理员的唯一标记，管理员实体的每一个属性都不能为空。管理员实体E-R图如下：



图 3‑16 管理员实体E-R图

**（3）列车实体：**

存储列车相关信息，主键为列车号，列车实体E-R图如下：



图 3‑17 列车实体E-R图

**（4）车票实体：**

存储车票相关信息，主键为列车号，车票实体E-R图如下：



图 3‑18 车票实体E-R如下

**（5）购票人实体：**

存储购票人相关信息，主键为用户名，购票人实体E-R图如下：



图 3‑19 购票人实体E-R图

**（6）整体E-R图**

车票管理子系统整体E-R图如下：



图 3‑20 车票管理子系统整体E-R图

### 3.3.2 关系模式

将车票管理子系统整体E-R图转化为关系模式。

用户（用户名,身份证,电话号码,用户密码,密保,密保答案）

管理员（管理员编号,密码,等级,权限）

列车（列车号,列车长,起始站,出发时间,终点站,到站时间,日期,联系电话）

车票（列车号,商务票数,一等票数,二等票数,无座票数,商务票价,一等票价,二等票价,无座票价）

购票人（存储用户名,列车号,商务票数,一等票数,二等票数,无座票数）

将各个E-R图转化为数据表

图 3‑21 用户数据表 图 3‑22管理员数据表 图 3‑23 列车数据表

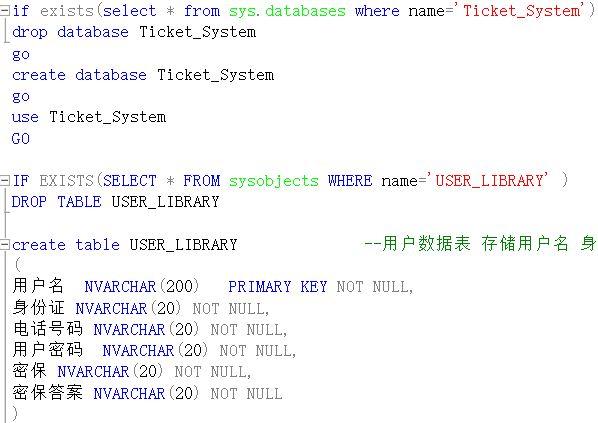
 

图 3‑24 车票数据表 图 3‑25 购买人数据表

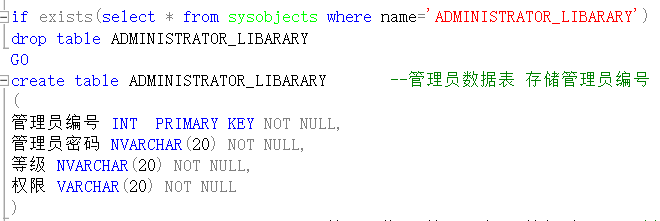
# 系统实现

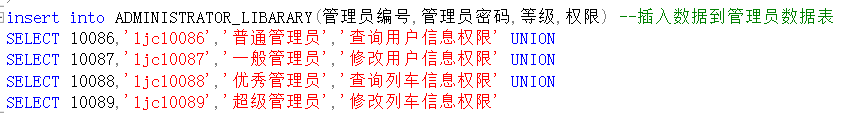
## 4.1 数据库和数据表实现

### 4.1.1 数据库的建立和用户数据表的建立

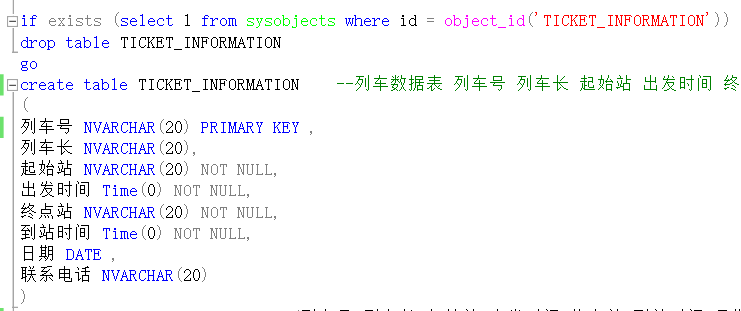


### 4.1.2 管理员数据表的建立



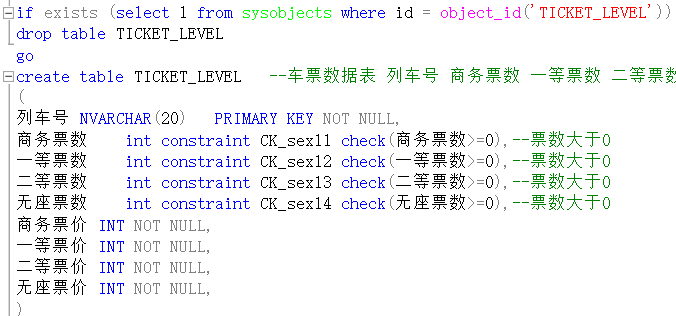


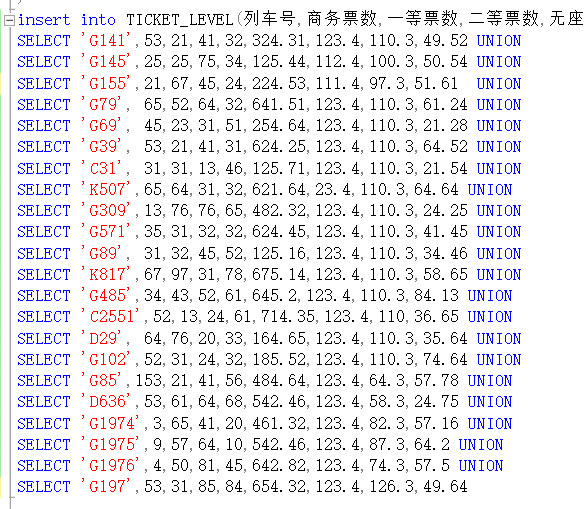
### 4.1.3 列车数据表的建立



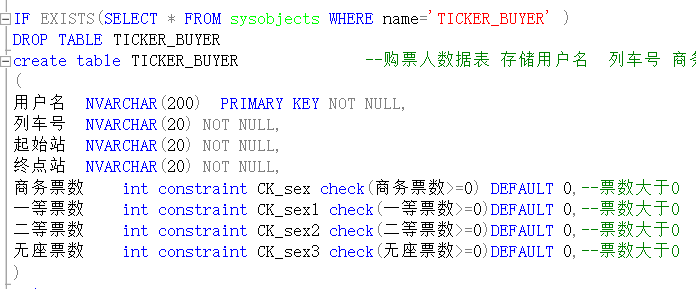


### 4.1.4 车票数据表的建立



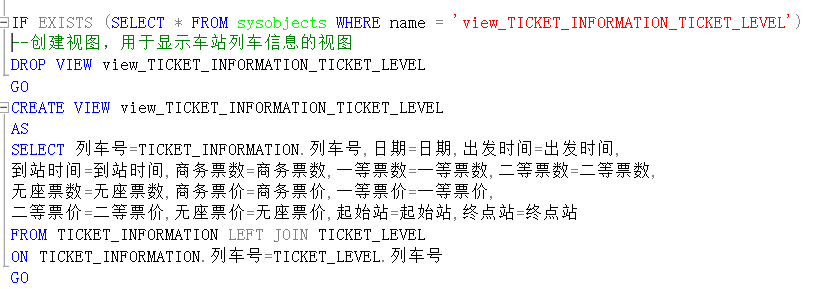


### 4.1.5 购票人数据表的建立

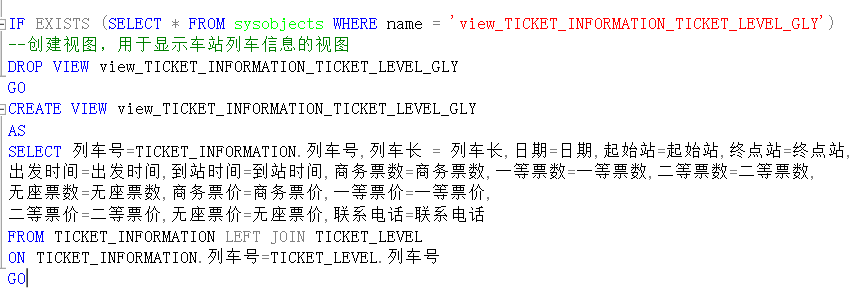


## 4.2通过视图，JAVA，GUI,JDBC等实现每个模块的功能

### 4.2.1 显示用户查询车站列车信息的视图



### 4.2.2 显示管理员查询车站列车信息的视图



### 4.2.3 用户注册与登录模块，管理员登录模块

结构如图所示：

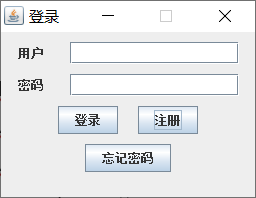
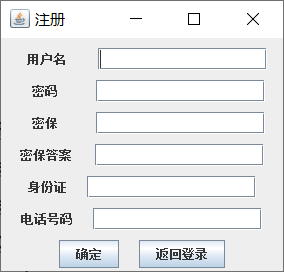
 

图4‑1 登录界面 图4‑2 注册界面

**public** **class** LoginUI

{

**static** String *a*;

**public** **void** loginUI() //登录模块

{

JFrame login\_frame = **new** JFrame("登录");

login\_frame.setSize(260,200);

login\_frame.setResizable(**false**); //不允许修改窗口

login\_frame.setLocationRelativeTo(**null**);//窗口位置居中

login\_frame.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,20,10));///流式布局

login\_frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

JLabel usrname = **new** JLabel("用户 ");

JLabel psw = **new** JLabel("密码 ");

JTextField f1 = **new** JTextField(15);

JPasswordField f2 = **new** JPasswordField(15);

JButton login = **new** JButton("登录");

JButton reset = **new** JButton("注册");

JButton forget = **new** JButton("忘记密码");

login\_frame.add(usrname);

login\_frame.add(f1);

login\_frame.add(psw);

login\_frame.add(f2);

login\_frame.add(login);

login\_frame.add(reset);

login\_frame.add(forget);

login\_frame.setVisible(**true**);

forget.addActionListener(**new** FindPassword());

ActionListener listener = **new** ActionListener() //登录监听

{

**public** **void** actionPerformed(ActionEvent event)

{

String textName = f1.getText();

String textPassword = **new** String(f2.getPassword());

login\_frame.setVisible(**false**);

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Statement stmt1;

Connection dbConn = **null**;

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();stmt1=dbConn.createStatement();

String s="(Select \* from USER\_LIBRARY where 用户名='"+textName+"'And 用户密码='"+textPassword+"' )";

String s1="(Select \* from ADMINISTRATOR\_LIBARARY where 管理员编号='"+textName+"'And 管理员密码='"+textPassword+"' )";

ResultSet r=stmt.executeQuery(s);

ResultSet r1=stmt1.executeQuery(s1);

**if**(r.next())

{

login\_frame.dispose();

*a*=f1.getText();

JOptionPane.*showMessageDialog*(**null**, "尊敬的用户，欢迎使用本系统！","温馨提示",JOptionPane.***PLAIN\_MESSAGE***);//弹窗

**new** MainUI().mainUI(); //菜单模块

}

**else** **if**(r1.next())

{

login\_frame.dispose();

*a*=f1.getText();

JOptionPane.*showMessageDialog*(**null**, "管理员"+textName+"\n您好，欢迎您的使用！","温馨提示",JOptionPane.***PLAIN\_MESSAGE***);//弹窗

**new** GLYUI().glyUI();

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "用户名或密码输入错误，请重新输入！！！","温馨提示",JOptionPane.***ERROR\_MESSAGE***);//弹窗

f1.setText("");f2.setText("");

loginUI();

}

dbConn.close();

}

**catch** (Exception e)

{

e.printStackTrace();

System.***out***.print("连接失败");

}

}

};

ActionListener listener1 = **new** ActionListener() //注册监听

{

**public** **void** actionPerformed(ActionEvent event)

{

login\_frame.setVisible(**false**);

JFrame Reset\_frame = **new** JFrame("注册");

Reset\_frame.setSize(300,280);

//Reset\_frame.setResizable(false); //不允许修改窗口

Reset\_frame.setLocationRelativeTo(**null**);//窗口位置居中

Reset\_frame.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,20,10));///流式布局

Reset\_frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

JLabel usrname = **new** JLabel(" 用户名 ");

JLabel psw = **new** JLabel(" 密码 ");

JLabel psw1 = **new** JLabel(" 密保 ");

JLabel psw2 = **new** JLabel("密保答案 ");

JLabel sfz = **new** JLabel("身份证");

JLabel dh = **new** JLabel("电话号码");

JButton back = **new** JButton("返回登录");

JButton sureButton1 = **new** JButton("确定");

JTextField f1 = **new** JTextField(15);

JTextField f3 = **new** JTextField(15);

JTextField f4 = **new** JTextField(15);

JTextField f5= **new** JTextField(15);

JTextField f6 = **new** JTextField(15);

JPasswordField f2 = **new** JPasswordField(15);

Reset\_frame.add(usrname);

Reset\_frame.add(f1);

Reset\_frame.add(psw);

Reset\_frame.add(f2);

Reset\_frame.add(psw1);

Reset\_frame.add(f3);

Reset\_frame.add(psw2);

Reset\_frame.add(f4);

Reset\_frame.add(sfz);

Reset\_frame.add(f5);

Reset\_frame.add(dh);

Reset\_frame.add(f6);

Reset\_frame.add(sureButton1);

Reset\_frame.add(back);

Reset\_frame.setVisible(**true**);

ActionListener listener1 = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent e) {

Reset\_frame.setVisible(**false**);

login\_frame.setVisible(**true**);

}

};

back.addActionListener(listener1);

ActionListener errorListener = **new** ActionListener()

{

**public** **void** actionPerformed(ActionEvent event)

{

String textName = f1.getText();

String Secret = f3.getText();

String SecretKey = f4.getText();

String sfz = f5.getText();

String dh = f6.getText();

String textPassword = **new** String(f2.getPassword());

login\_frame.setVisible(**false**);

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Connection dbConn = **null**;

**if**(textName.isEmpty()||Secret.isEmpty()||SecretKey.isEmpty()||textPassword.isEmpty()||sfz.isEmpty()||dh.isEmpty()) {

JOptionPane.*showMessageDialog*(**null**, "请输入注册的各项内容","温馨提示",JOptionPane.***ERROR\_MESSAGE***);//消息对话框弹窗

f1.setText("");f2.setText("");f3.setText("");f4.setText("");f5.setText("");f6.setText("");

Reset\_frame.setVisible(**true**);

}

**else**

{

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="(Select \* from USER\_LIBRARY where 用户名='"+textName+"')";

ResultSet r=stmt.executeQuery(s); //游标

**if**(r.next())

{

Reset\_frame.setVisible(**false**);

JOptionPane.*showMessageDialog*(**null**, "用户名已经存在，请重新输入。","温馨提示",JOptionPane.***ERROR\_MESSAGE***);//消息对话框弹窗

f1.setText("");f2.setText("");f3.setText("");f4.setText("");f5.setText("");f6.setText("");

Reset\_frame.setVisible(**true**);

}

**else** {

String sql = "insert into USER\_LIBRARY values('"+textName+"','"+textPassword+"','"+Secret+"','"+SecretKey+"','"+sfz+"','"+dh+"')";

PreparedStatement pst = dbConn.prepareStatement(sql,Statement.***RETURN\_GENERATED\_KEYS***);

pst.executeUpdate();

pst.close();

JOptionPane.*showMessageDialog*(**null**,"注册成功"); //消息对话框

Reset\_frame.setVisible(**false**);

loginUI();

}

}

**catch** (Exception e)

{

e.printStackTrace();

System.***out***.print("连接失败");

}

}

}

};

sureButton1.addActionListener(errorListener);

}

};

login.addActionListener(listener);

reset.addActionListener(listener1);

}

**public** **static** **void** main(String[] agrs)

{

LoginUI qq = **new** LoginUI();

qq.loginUI();

}

}

### 4.2.3 用户找回密码模块

结构如图所示：

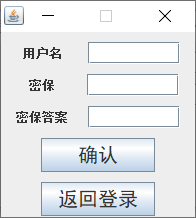


图4‑3 找回密码界面

**public** **class** FindPassword **implements** ActionListener{

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Connection dbConn = **null**;

**public** **void** actionPerformed(ActionEvent e) {

JFrame f = **new** JFrame();

JLabel lc = **new** JLabel(" 用户名 ");

JLabel mb = **new** JLabel(" 密保 ");

JLabel mbdaan = **new** JLabel("密保答案");

JButton chax = **new** JButton(" 确认 ");chax.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JButton back = **new** JButton("返回登录");back.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JTextField f1 = **new** JTextField(8);

JTextField f2 = **new** JTextField(8);

JTextField f3 = **new** JTextField(8);

f.add(lc);

f.add(f1);

f.add(mb);

f.add(f2);

f.add(mbdaan);

f.add(f3);

f.add(chax);

f.add(back);

f.setResizable(**false**); //不允许修改窗口

f.setSize(200,220);

f.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,20,10));///流式布局

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**true**);

ActionListener listener = **new** ActionListener() //确认退票监听

{

**public** **void** actionPerformed(ActionEvent e) {

String yhm = f1.getText();

String mb = f2.getText();

String mbdaan = f3.getText();

**try**

{ Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="(Select 用户密码 from USER\_LIBRARY where 用户名='"+yhm+"' AND 密保='"+mb+"'and 密保答案 = '"+mbdaan+"')";

ResultSet r=stmt.executeQuery(s); //游标 查询表内内容

**if**(r.next()) {

f1.setText("");f2.setText("");f3.setText("");

JOptionPane.*showMessageDialog*(**null**, "您的密码为："+r.getString(1),"温馨提示",JOptionPane.***PLAIN\_MESSAGE***);//弹窗

}**else** {

f1.setText("");f2.setText("");f3.setText("");

JOptionPane.*showMessageDialog*(**null**, "输入错误！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

}

}**catch** (Exception e1) {

}

}

};

chax.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent e) {

f.setVisible(**false**);

}

};

back.addActionListener(listener1);

}

}

### 4.2.4 查询列车信息与菜单模块

结构如所示：

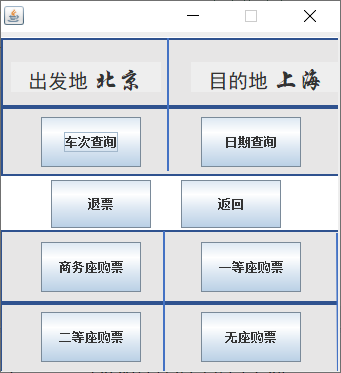
 

图4‑4 查询列车信息 图4‑5 菜单模块图

**public** **class** MainUI {

//菜单模块

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Connection dbConn = **null**;

**public** **void** mainUI()

{

JFrame f = **new** JFrame("查询车票");

JButton sureButton = **new** JButton("查询列车");

JButton fh = **new** JButton("返回登录");

f.setResizable(**false**); //不允许修改窗口

f.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,20,10));

f.setSize(200,200);

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

JLabel starting = **new** JLabel("起始站");

JLabel end = **new** JLabel("终点站");

JComboBox<String>comboBox = **new** JComboBox<>();

JComboBox<String>comboBox1 = **new** JComboBox<>();

comboBox.addItem("北京");comboBox1.addItem("北京");

comboBox.addItem("上海");comboBox1.addItem("上海");

comboBox.addItem("广州");comboBox1.addItem("广州");

comboBox.addItem("重庆");comboBox1.addItem("重庆");

comboBox.addItem("成都");comboBox1.addItem("成都");

comboBox.addItem("长沙");comboBox1.addItem("长沙");

comboBox.addItem("天津");comboBox1.addItem("天津");

comboBox.addItem("哈尔滨");comboBox1.addItem("哈尔滨");

f.add(starting);

f.add(end);

f.add(comboBox);

f.add(comboBox1);

f.add(sureButton);

f.add(fh);

ActionListener listener = **new** ActionListener() //返回登录的监听

{

**public** **void** actionPerformed(ActionEvent event)

{

f.setVisible(**false**);

**new** LoginUI().loginUI();

}

};

fh.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //查询列车监听

{

**public** **void** actionPerformed(ActionEvent event)

{

String city = comboBox.getSelectedItem().toString(); //获取下拉框中的值

String city1 = comboBox1.getSelectedItem().toString(); //获取下拉框中的值

f.setVisible(**false**);

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="(Select \* from TICKET\_INFORMATION where 起始站='"+city+"' AND 终点站='"+city1+"')";

ResultSet r=stmt.executeQuery(s); //游标 查询表内内容

**if**(r.next())

{ //查询窗口

JLayeredPane layeredPane; //创建一个JLayeredPane用于分层的。

JPanel jp;

JLabel jl; //创建一个Panel和一个Label用于存放图片，作为背景。

JButton cc =**new** JButton("车次查询");

JButton rq =**new** JButton("日期查询");

JButton sw =**new** JButton("商务座购票");

JButton yd =**new** JButton("一等座购票");

JButton ed =**new** JButton("二等座购票");

JButton wz =**new** JButton("无座购票");

JButton fh =**new** JButton("返回");

JButton tp =**new** JButton("退票");

ImageIcon image;

tp.addActionListener(**new** BackTicket(city, city1));//退票的监听

rq.addActionListener(**new** DateMonitoring(city, city1)); //日期查询的监听

sw.addActionListener(**new** BusinessMonitoring(city, city1)); //商务购票的监听

yd.addActionListener(**new** FirstClassTicket(city, city1)); //一等购票的监听

ed.addActionListener(**new** SecondClassTicket(city, city1)); //二等购票的监听

wz.addActionListener(**new** NoSeatTicket(city, city1)); //无座购票的监听

JFrame jf= **new** JFrame();

layeredPane=**new** JLayeredPane();

image=**new** ImageIcon("C:\\Users\\DEATH\\Desktop\\1.png");

jp=**new** JPanel();

jp.setBounds(0,0,image.getIconWidth(),image.getIconHeight()); //面板的位置和宽高

jl=**new** JLabel(image);

jp.add(jl);

JPanel train=**new** JPanel(); //存储列车信息的面板

JPanel panel\_north=**new** JPanel();

JPanel panel\_north1=**new** JPanel();

JLabel starting\_city = **new** JLabel("出发地");starting\_city.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JLabel destination = **new** JLabel("目的地");destination.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JLabel c = **new** JLabel(city);c.setFont(**new** Font("华文行楷",Font.***PLAIN***,25));//设置字体的字体，大小

JLabel c1 = **new** JLabel(city1);c1.setFont(**new** Font("华文行楷",Font.***PLAIN***,25));//设置字体的字体，大小

panel\_north.add(starting\_city);

panel\_north.add(c);

panel\_north1.add(destination);

panel\_north1.add(c1);

panel\_north.setBounds(10,30, 150, 30);

panel\_north1.setBounds(190,30, 150, 30);

tp.setBounds(50, 148, 100, 48);

fh.setBounds(180,148, 100, 48);

cc.setBounds(40,85, 100, 50);

rq.setBounds(200,85, 100, 50);

sw.setBounds(40, 210, 100, 50);

yd.setBounds(200, 210, 100, 50);

ed.setBounds(40, 280, 100, 50);

wz.setBounds(200, 280, 100, 50);

layeredPane.add(jp,JLayeredPane.***DEFAULT\_LAYER***);

layeredPane.add(panel\_north,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(panel\_north1,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(tp,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(fh,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(cc,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(rq,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(sw,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(yd,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(ed,JLayeredPane.***MODAL\_LAYER***);

layeredPane.add(wz,JLayeredPane.***MODAL\_LAYER***);

jf.setLayeredPane(layeredPane);

jf.setLayout(**null**);

jf.setResizable(**false**); //不允许修改窗口

jf.setSize(345,375);

jf.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

jf.setLocationRelativeTo(**null**);//窗口位置居中

jf.setVisible(**true**);

f.setVisible(**false**);

ActionListener listener = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent event)

{

jf.setVisible(**false**);

f.setVisible(**true**);

}

};

fh.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //车次查询监听

{

**public** **void** actionPerformed(ActionEvent event)

{

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="Select \* from view\_TICKET\_INFORMATION\_TICKET\_LEVEL where 起始站='"+city+"' AND 终点站='"+city1+"'order by 列车号";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

Vector columnNames=**new** Vector();//设置列名

columnNames.add("列车号");

columnNames.add("日期");

columnNames.add("发车时间");

columnNames.add("到站时间");

columnNames.add("商务座票数");

columnNames.add("一等座票数");

columnNames.add("二等座票数");

columnNames.add("无座票数");

Vector rowData = **new** Vector();

**while**(rs.next()) {

//rowData可以存放多行

Vector hang=**new** Vector();

hang.add(rs.getString(1));

hang.add(rs.gettring(2));

hang.add(rs.getString(3)) hang.add(rs.getString(4)); hang.add(rs.getString(5)); hang.add(rs.getString(6)); hang.add(rs.getString(7)); hang.add(rs.getString(8));

rowData.add(hang);

}

JTable jt = **new** JTable(rowData,columnNames);

DefaultTableCellRenderer tcr = **new** DefaultTableCellRenderer();// 设置table内容居中

tcr.setHorizontalAlignment(SwingConstants.***CENTER***);// 这句和上句作用一样

jt.setDefaultRenderer(Object.**class**, tcr);

jt.setEnabled(**false**);//不允许修改表格内容

jt.setFont(**new** Font("楷体", Font.***PLAIN***, 18));// 设置表格字体

JScrollPane jsp = **new** JScrollPane(jt);

JPanel jp1 =**new** JPanel();

jp1.add(jsp);

JFrame jf = **new** JFrame();

jf.add(jsp);

jf.setSize(985,900);

jf.setLocationRelativeTo(**null**);//窗口位置居中

jf.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

};

cc.addActionListener(listener1);

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "很抱歉！未找到您所查询的车站间的列车信息","提示",JOptionPane.***INFORMATION\_MESSAGE***);

**new** MainUI().mainUI();

}

//dbConn.close(); //关闭数据库

}

**catch** (Exception e)

{

e.printStackTrace();

System.***out***.print("连接失败");

}

}

};

sureButton.addActionListener(listener1);

}

**public** **static** **void** main(String[] agrs)

{

LoginUI qq = **new** LoginUI();

qq.loginUI();

}

}

**日期查询功能**

**public** **class** DateMonitoring **implements** ActionListener {

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

String c;

String c1;

Connection dbConn = **null**;

JComboBox<String>comboBox = **new** JComboBox<>();

**public** **static** java.sql.Date StringToDate(String sDate) {

String str = sDate;

SimpleDateFormat format = **new** SimpleDateFormat("yyyyMMdd");

java.util.Date d = **null**;

**try** {

d = format.parse(str);

} **catch** (Exception e) {

JOptionPane.*showMessageDialog*(**null**, "日期格式输入错误！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

}

java.sql.Date date = **new** java.sql.Date(d.getTime());

**return** date;

}

**public** **void** actionPerformed(ActionEvent e) {

JFrame f = **new** JFrame("列车查询");

JTextField f1 = **new** JTextField(15);

JButton chax = **new** JButton("查询");chax.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JButton back = **new** JButton("返回");back.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JLabel rq =**new** JLabel("出发日期");rq.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

f.setResizable(**false**); //不允许修改窗口

f.setSize(200,200);

f.add(rq);

f.add(f1);

f.add(chax);

f.add(back);

f.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,20,10));///流式布局

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**true**);

ActionListener listener = **new** ActionListener() //查询列车监听

{

**public** **void** actionPerformed(ActionEvent e) {

**try**

{ String cfrq= f1.getText();

java.sql.Date dt=*StringToDate*(cfrq);

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="select \* from view\_TICKET\_INFORMATION\_TICKET\_LEVEL where 日期='"+dt+"'and 起始站 = '"+c+"'and 终点站 = '"+c1+"'order by 列车号";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**if**(!rs.next()) {

JOptionPane.*showMessageDialog*(**null**, "很抱歉！未找到您所查询的车站间的列车信息","提示",JOptionPane.***INFORMATION\_MESSAGE***);

f1.setText("");

}

**else** {

Vector columnNames=**new** Vector();//设置列名

columnNames.add("列车号");

columnNames.add("日期");

columnNames.add("发车时间");

columnNames.add("到站时间");

columnNames.add("商务座票数");

columnNames.add("一等座票数");

columnNames.add("二等座票数");

columnNames.add("无座票数");

Vector rowData = **new** Vector();

**while**(rs.next()) {

//rowData可以存放多行

Vector hang=**new** Vector();

hang.add(rs.getString(1));

hang.add(rs.getString(2));

hang.add(rs.getString(3));

hang.add(rs.getString(4));

hang.add(rs.getString(5));

hang.add(rs.getString(6));

hang.add(rs.getString(7));

hang.add(rs.getString(8));

rowData.add(hang);

}

JTable jt = **new** JTable(rowData,columnNames);

DefaultTableCellRenderer tcr = **new** DefaultTableCellRenderer();// 设置table内容居中

tcr.setHorizontalAlignment(SwingConstants.***CENTER***);// 这句和上句作用一样

jt.setDefaultRenderer(Object.**class**, tcr);

jt.setEnabled(**false**);//不允许修改表格内容

jt.setFont(**new** Font("楷体", Font.***PLAIN***, 18));// 设置表格字体

JScrollPane jsp = **new** JScrollPane(jt);

JPanel jp1 =**new** JPanel();

jp1.add(jsp);

JFrame jf = **new** JFrame();

jf.add(jsp);

jf.setSize(985,900);

jf.setLocationRelativeTo(**null**);//窗口位置居中

jf.setVisible(**true**);

}

} **catch** (Exception e1) {

f1.setText("");f.setVisible(**true**);

}

}

};

chax.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent e) {

f.setVisible(**false**);

}

};

back.addActionListener(listener1);

}

**public** DateMonitoring(String city,String city1){

c = city;c1 = city1;

}

}

**购票功能**

**public** **class** BusinessMonitoring **implements** ActionListener {

String c;

String c1;

JComboBox<String>comboBox = **new** JComboBox<>();

**public** **void** actionPerformed(ActionEvent e) {

JFrame f = **new** JFrame("购买车票");

JButton sureButton = **new** JButton("商务购票");sureButton.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JTextField f1 = **new** JTextField(5);

JButton back = **new** JButton("返回");back.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JLabel lc =**new** JLabel("列车号");

JLabel gps =**new** JLabel("购票数");

f.setResizable(**false**); //不允许修改窗口

f.setLayout(**null**);

f.setSize(200,240);

f.add(lc);

f.add(sureButton);

f.add(gps);

f.add(f1);

f.add(back);

lc.setBounds(20,5, 65, 40);

gps.setBounds(10, 50, 65, 40);

f1.setBounds(80,50,40,30);

sureButton.setBounds(40,100, 120, 45);

back.setBounds(65,150, 75, 40);

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**false**);

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String url="jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String user="sa";//sa超级管理员

String password="sa";//密码

Connection conn=DriverManager.*getConnection*(url,user,password);

Statement stmt=conn.createStatement();

String s="(SELECT 列车号 FROM TICKET\_INFORMATION WHERE 起始站='"+c+"' AND 终点站='"+c1+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**while**(rs.next()){

comboBox.addItem(rs.getString(1));

}

}**catch** (Exception e1)

{

e1.printStackTrace();

System.***out***.print("连接失败");

}

comboBox.setBounds(90,10, 50, 30);

f.add(comboBox);

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

ActionListener listener1 = **new** ActionListener() //购票监听

{

**public** **void** actionPerformed(ActionEvent event)

{ String shu = f1.getText();

String piao = comboBox.getSelectedItem().toString(); //获取下拉框中的值

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Connection dbConn=DriverManager.*getConnection*(dbURL,userName,userPwd);

Statement stmt=dbConn.createStatement();

Statement stmt1=dbConn.createStatement();

String s1="UPDATE TICKET\_LEVEL SET 商务票数 =商务票数-'"+shu+"' WHERE 列车号 = '"+piao+"'";

**int** r1=stmt.executeUpdate(s1);

**if**(r1==1)

{

String s="(SELECT \* FROM TICKER\_BUYER WHERE 用户名='"+LoginUI.*a*+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**if**(!rs.next()) {

String s2= "insert into TICKER\_BUYER(用户名,列车号,起始站,终点站,商务票数) values('"+LoginUI.*a*+"','"+piao+"','"+c+"','"+c1+"','"+shu+"')";

**int** r2=stmt1.executeUpdate(s2);

**if**(r2==1) {

JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}**else** {

String s2="UPDATE TICKER\_BUYER SET 商务票数 = 商务票数+'"+shu+"' WHERE 列车号 = '"+piao+"'and 用户名 ='"+LoginUI.*a*+"'and 起始站 ='"+c+"'and 终点站 ='"+c1+"'";

stmt1.executeUpdate(s2);JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "购票失败","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

**catch** (Exception e)

{

JOptionPane.*showMessageDialog*(**null**, "购票失败,余票不足！！！","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

};

sureButton.addActionListener(listener1);

ActionListener listener = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent event)

{

f.setVisible(**false**);

}

};

back.addActionListener(listener);

}

**public** BusinessMonitoring(String city,String city1){

c=city;c1=city1;

}

}

**public** **class** FirstClassTicket **implements** ActionListener {

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

String c;

String c1;

Connection dbConn = **null**;

JComboBox<String>comboBox = **new** JComboBox<>();

**public** **void** actionPerformed(ActionEvent e) {

JFrame f = **new** JFrame("购买车票");

JButton sureButton = **new** JButton("一等购票");sureButton.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JTextField f1 = **new** JTextField(5);

JButton back = **new** JButton("返回");back.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JLabel lc =**new** JLabel("列车号");

JLabel gps =**new** JLabel("购票数");

f.setResizable(**false**); //不允许修改窗口

f.setLayout(**null**);

f.setSize(200,240);

f.add(lc);

f.add(sureButton);

f.add(gps);

f.add(f1);

f.add(back);

lc.setBounds(20,5, 65, 40);

gps.setBounds(10, 50, 65, 40);

f1.setBounds(80,50,40,30);

sureButton.setBounds(40,100, 120, 45);

back.setBounds(65,150, 75, 40);

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**false**);

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String url="jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String user="sa";//sa超级管理员

String password="sa";//密码

Connection conn=DriverManager.*getConnection*(url,user,password);

Statement stmt=conn.createStatement();

String s="(SELECT 列车号 FROM TICKET\_INFORMATION WHERE 起始站='"+c+"' AND 终点站='"+c1+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**while**(rs.next()){

comboBox.addItem(rs.getString(1));

}

}**catch** (Exception e1)

{

e1.printStackTrace();

System.***out***.print("连接失败");

}

comboBox.setBounds(90,10, 50, 30);

f.add(comboBox);

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

ActionListener listener1 = **new** ActionListener() //购票监听

{

**public** **void** actionPerformed(ActionEvent event)

{ String shu = f1.getText();

String piao = comboBox.getSelectedItem().toString(); //获取下拉框中的值

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Connection dbConn=DriverManager.*getConnection*(dbURL,userName,userPwd);

Statement stmt=dbConn.createStatement();

Statement stmt1=dbConn.createStatement();

String s1="UPDATE TICKET\_LEVEL SET 一等票数 =一等票数-'"+shu+"' WHERE 列车号 = '"+piao+"'";

**int** r1=stmt.executeUpdate(s1);

**if**(r1==1)

{

String s="(SELECT \* FROM TICKER\_BUYER WHERE 用户名='"+LoginUI.*a*+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**if**(!rs.next()) {

String s2= "insert into TICKER\_BUYER(用户名,列车号,起始站,终点站,一等票数) values('"+LoginUI.*a*+"','"+piao+"','"+c+"','"+c1+"','"+shu+"')";

**int** r2=stmt1.executeUpdate(s2);

**if**(r2==1) {

System.***out***.println(r2);

JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}**else** {

String s2="UPDATE TICKER\_BUYER SET 一等票数 = 一等票数+'"+shu+"' WHERE 列车号 = '"+piao+"'and 用户名 ='"+LoginUI.*a*+"'and 起始站 ='"+c+"'and 终点站 ='"+c1+"'";

stmt1.executeUpdate(s2);JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "购票失败","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

**catch** (Exception e)

{

JOptionPane.*showMessageDialog*(**null**, "购票失败,余票不足！！！","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

};

sureButton.addActionListener(listener1);

ActionListener listener = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent event)

{

f.setVisible(**false**);

}

};

back.addActionListener(listener);

}

**public** FirstClassTicket(String city,String city1){

c=city;c1=city1;

}

}

**public** **class** SecondClassTicket **implements** ActionListener {

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

String c;

String c1;

Connection dbConn = **null**;

JComboBox<String>comboBox = **new** JComboBox<>();

**public** **void** actionPerformed(ActionEvent e) {

JFrame f = **new** JFrame("购买车票");

JButton sureButton = **new** JButton("二等购票");sureButton.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JTextField f1 = **new** JTextField(5);

JButton back = **new** JButton("返回");back.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JLabel lc =**new** JLabel("列车号");

JLabel gps =**new** JLabel("购票数");

f.setResizable(**false**); //不允许修改窗口

f.setLayout(**null**);

f.setSize(200,240);

f.add(lc);

f.add(sureButton);

f.add(gps);

f.add(f1);

f.add(back);

lc.setBounds(20,5, 65, 40);

gps.setBounds(10, 50, 65, 40);

f1.setBounds(80,50,40,30);

sureButton.setBounds(40,100, 120, 45);

back.setBounds(65,150, 75, 40);

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**false**);

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String url="jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String user="sa";//sa超级管理员

String password="sa";//密码

Connection conn=DriverManager.*getConnection*(url,user,password);

Statement stmt=conn.createStatement();

String s="(SELECT 列车号 FROM TICKET\_INFORMATION WHERE 起始站='"+c+"' AND 终点站='"+c1+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**while**(rs.next()){

comboBox.addItem(rs.getString(1));

}

}**catch** (Exception e1)

{

e1.printStackTrace();

System.***out***.print("连接失败");

}

comboBox.setBounds(90,10, 50, 30);

f.add(comboBox);

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

ActionListener listener1 = **new** ActionListener() //购票监听

{

**public** **void** actionPerformed(ActionEvent event)

{ String shu = f1.getText();

String piao = comboBox.getSelectedItem().toString(); //获取下拉框中的值

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Connection dbConn=DriverManager.*getConnection*(dbURL,userName,userPwd);

Statement stmt=dbConn.createStatement();

Statement stmt1=dbConn.createStatement();

String s1="UPDATE TICKET\_LEVEL SET 二等票数 =二等票数-'"+shu+"' WHERE 列车号 = '"+piao+"'";

**int** r1=stmt.executeUpdate(s1);

**if**(r1==1)

{

String s="(SELECT \* FROM TICKER\_BUYER WHERE 用户名='"+LoginUI.*a*+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**if**(!rs.next()) {

String s2= "insert into TICKER\_BUYER(用户名,列车号,起始站,终点站,二等票数) values('"+LoginUI.*a*+"','"+piao+"','"+c+"','"+c1+"','"+shu+"')";

**int** r2=stmt1.executeUpdate(s2);

**if**(r2==1) {

System.***out***.println(r2);

JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}**else** {

String s2="UPDATE TICKER\_BUYER SET 二等票数 = 二等票数+'"+shu+"' WHERE 列车号 = '"+piao+"'and 用户名 ='"+LoginUI.*a*+"'and 起始站 ='"+c+"'and 终点站 ='"+c1+"'";

stmt1.executeUpdate(s2);JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "购票失败","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

**catch** (Exception e)

{

JOptionPane.*showMessageDialog*(**null**, "购票失败,余票不足！！！","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

};

sureButton.addActionListener(listener1);

ActionListener listener = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent event)

{

f.setVisible(**false**);

}

};

back.addActionListener(listener);

}

**public** SecondClassTicket(String city,String city1){

c=city;c1=city1;

}

}

**public** **class** NoSeatTicket **implements** ActionListener {

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

String c;

String c1;

Connection dbConn = **null**;

JComboBox<String>comboBox = **new** JComboBox<>();

**public** **void** actionPerformed(ActionEvent e) {

JFrame f = **new** JFrame("购买车票");

JButton sureButton = **new** JButton("无座购票");sureButton.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JTextField f1 = **new** JTextField(5);

JButton back = **new** JButton("返回");back.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JLabel lc =**new** JLabel("列车号");

JLabel gps =**new** JLabel("购票数");

f.setResizable(**false**); //不允许修改窗口

f.setLayout(**null**);

f.setSize(200,240);

f.add(lc);

f.add(sureButton);

f.add(gps);

f.add(f1);

f.add(back);

lc.setBounds(20,5, 65, 40);

gps.setBounds(10, 50, 65, 40);

f1.setBounds(80,50,40,30);

sureButton.setBounds(40,100, 120, 45);

back.setBounds(65,150, 75, 40);

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**false**);

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String url="jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String user="sa";//sa超级管理员

String password="sa";//密码

Connection conn=DriverManager.*getConnection*(url,user,password);

Statement stmt=conn.createStatement();

String s="(SELECT 列车号 FROM TICKET\_INFORMATION WHERE 起始站='"+c+"' AND 终点站='"+c1+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**while**(rs.next()){

comboBox.addItem(rs.getString(1));

}

}**catch** (Exception e1)

{

e1.printStackTrace();

System.***out***.print("连接失败");

}

comboBox.setBounds(90,10, 50, 30);

f.add(comboBox);

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

ActionListener listener1 = **new** ActionListener() //购票监听

{

**public** **void** actionPerformed(ActionEvent event)

{ String shu = f1.getText();

String piao = comboBox.getSelectedItem().toString(); //获取下拉框中的值

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Connection dbConn=DriverManager.*getConnection*(dbURL,userName,userPwd);

Statement stmt=dbConn.createStatement();

Statement stmt1=dbConn.createStatement();

String s1="UPDATE TICKET\_LEVEL SET 无座票数 =无座票数-'"+shu+"' WHERE 列车号 = '"+piao+"'";

**int** r1=stmt.executeUpdate(s1);

**if**(r1==1)

{

String s="(SELECT \* FROM TICKER\_BUYER WHERE 用户名='"+LoginUI.*a*+"')";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

**if**(!rs.next()) {

String s2= "insert into TICKER\_BUYER(用户名,列车号,起始站,终点站,无座票数) values('"+LoginUI.*a*+"','"+piao+"','"+c+"','"+c1+"','"+shu+"')";

**int** r2=stmt1.executeUpdate(s2);

**if**(r2==1) {

System.***out***.println(r2);

JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}**else** {

String s2="UPDATE TICKER\_BUYER SET 无座票数 = 无座票数+'"+shu+"' WHERE 列车号 = '"+piao+"'and 用户名 ='"+LoginUI.*a*+"'and 起始站 ='"+c+"'and 终点站 ='"+c1+"'";

stmt1.executeUpdate(s2);JOptionPane.*showMessageDialog*(**null**, "购票成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

}

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "购票失败","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

**catch** (Exception e)

{

JOptionPane.*showMessageDialog*(**null**, "购票失败,余票不足！！！","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

};

sureButton.addActionListener(listener1);

ActionListener listener = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent event)

{

f.setVisible(**false**);

}

};

back.addActionListener(listener);

}

**public** NoSeatTicket(String city,String city1){

c=city;c1=city1;

}

}

**退票功能**

**import** javax.swing.table.DefaultTableCellRenderer;

**public** **class** BackTicket **implements** ActionListener {

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Statement stmt1;

Connection dbConn = **null**;

String c;

String c1;

**public** **void** actionPerformed(ActionEvent e) {

JFrame f = **new** JFrame("退票");

JTextField f1 = **new** JTextField(8);

JTextField f2 = **new** JTextField(5);

JLabel lc = **new** JLabel("列车");

JComboBox<String>comboBox = **new** JComboBox<>();

comboBox.addItem("商务座退票");

comboBox.addItem("一等座退票");

comboBox.addItem("二等座退票");

comboBox.addItem("无座退票");

JLabel gps =**new** JLabel("退票数");

f.add(lc);

f.add(f1);

f.add(comboBox);

JButton chax = **new** JButton("确认退票");chax.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

JButton back = **new** JButton("返回");back.setFont(**new** Font("黑体",Font.***PLAIN***,20));//设置字体的字体，大小

f.setResizable(**false**); //不允许修改窗口

f.setSize(189,250);

f.add(gps);

f.add(f2);

f.add(chax);

f.add(back);

f.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,28,10));///流式布局

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**true**);

ActionListener listener = **new** ActionListener() //确认退票监听

{

**public** **void** actionPerformed(ActionEvent e) {

String lch = f1.getText();

String shu = f2.getText();

**try**

{ Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

stmt1=dbConn.createStatement();

String s="(Select \* from TICKER\_BUYER where 起始站='"+c+"' AND 终点站='"+c1+"'and 列车号 = '"+lch+"')";

ResultSet r=stmt.executeQuery(s); //游标 查询表内内容

**if**(r.next()) {

**if**(comboBox.getSelectedIndex()==0) {

String q ="UPDATE TICKET\_LEVEL SET 商务票数 =商务票数+'"+shu+"' WHERE 列车号 = '"+lch+"'";

String tp="UPDATE TICKER\_BUYER SET 商务票数 = 商务票数-'"+shu+"' WHERE 列车号 = '"+lch+"'and 用户名 ='"+**new** LoginUI().*a*+"'";

stmt.executeUpdate(tp);**int** r1=stmt1.executeUpdate(q);

f1.setText("");JOptionPane.*showMessageDialog*(**null**, "退票成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

**else** **if**(comboBox.getSelectedIndex()==1) {

f1.setText(""); JOptionPane.*showMessageDialog*(**null**, "退票成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

String q ="UPDATE TICKET\_LEVEL SET 一等票数 =一等票数+'"+shu+"' WHERE 列车号 = '"+lch+"'";

String tp="UPDATE TICKER\_BUYER SET 一等票数 = 一等票数-'"+shu+"' WHERE 列车号 = '"+lch+"'and 用户名 ='"+**new** LoginUI().*a*+"'";

stmt.executeUpdate(tp);stmt1.executeUpdate(q);

}**else** **if**(comboBox.getSelectedIndex()==2) {

f1.setText(""); JOptionPane.*showMessageDialog*(**null**, "退票成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

String q ="UPDATE TICKET\_LEVEL SET 二等票数 =二等票数+'"+shu+"' WHERE 列车号 = '"+lch+"'";

String tp="UPDATE TICKER\_BUYER SET 二等票数 = 二等票数-'"+shu+"' WHERE 列车号 = '"+lch+"'and 用户名 ='"+**new** LoginUI().*a*+"'";

stmt.executeUpdate(tp);stmt1.executeUpdate(q);

}**else** {

f1.setText(""); JOptionPane.*showMessageDialog*(**null**, "退票成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

String q ="UPDATE TICKET\_LEVEL SET 无等票数 =无等票数+'"+shu+"' WHERE 列车号 = '"+lch+"'";

String tp="UPDATE TICKER\_BUYER SET 无座票数 = 无座票数-'"+shu+"' WHERE 列车号 = '"+lch+"'and 用户名 ='"+**new** LoginUI().*a*+"'";

stmt.executeUpdate(tp);stmt1.executeUpdate(q);

}

}

**else** {

f1.setText(""); f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "退票失败，请核对列车信息！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

}

} **catch** (Exception e1) {

JOptionPane.*showMessageDialog*(**null**, "您尚未购买该车票！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

f.setVisible(**true**);

}

}

};

chax.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent e) {

f.setVisible(**false**);

}

};

back.addActionListener(listener1);

}

**public** BackTicket(String city,String city1){

c = city;c1 = city1;

}

}

### 4.2.5 管理员界面模块



图4‑6 管理员界面图

**public** **class** GLYUI {

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Connection dbConn = **null**;

**public** **void** glyUI() {

JFrame f = **new** JFrame();

JButton j1 = **new** JButton("增加信息");

JButton j2 = **new** JButton("删除信息");

JButton j3 = **new** JButton("修改信息");

JButton j4 = **new** JButton("查询信息");

JButton fh = **new** JButton("返回登录");

f.add(j1);

f.add(j2);

f.add(j3);

f.add(j4);

f.add(fh);

f.setResizable(**false**); //不允许修改窗口

f.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,20,10));

f.setSize(250,190);

f.setLocationRelativeTo(**null**);//窗口位置居中

f.setVisible(**true**);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

j1.addActionListener(**new** AddTicket()); //增加信息监听

j2.addActionListener(**new** DeleteTICKET()); //删除信息监听

j3.addActionListener(**new** UpdateTicket()); //修改信息监听

ActionListener listener = **new** ActionListener() //车次查询监听

{

**public** **void** actionPerformed(ActionEvent event)

{

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="Select \* from view\_TICKET\_INFORMATION\_TICKET\_LEVEL\_GLY order by 列车号";

ResultSet rs=stmt.executeQuery(s); //游标 查询表内内容

Vector columnNames=**new** Vector();//设置列名

columnNames.add("列车号");

columnNames.add("列车长");

columnNames.add("日期");

columnNames.add("终点站");

columnNames.add("起始站");

columnNames.add("发车时间");

columnNames.add("到站时间");

columnNames.add("商务座票数");

columnNames.add("一等座票数");

columnNames.add("二等座票数");

columnNames.add("无座票数");

columnNames.add("商务座票价");

columnNames.add("一等座票价");

columnNames.add("二等座票价");

columnNames.add("无座票价");

columnNames.add("联系电话");

Vector rowData = **new** Vector();

**while**(rs.next()) {

//rowData可以存放多行

Vector hang=**new** Vector();

hang.add(rs.getString(1));

hang.add(rs.getString(2));

hang.add(rs.getString(3));

hang.add(rs.getString(4));

hang.add(rs.getString(5));

hang.add(rs.getString(6));

hang.add(rs.getString(7));

hang.add(rs.getString(8));

hang.add(rs.getString(9));

hang.add(rs.getString(10));

hang.add(rs.getString(11));

hang.add(rs.getString(12));

hang.add(rs.getString(13));

hang.add(rs.getString(14));

hang.add(rs.getString(15));

hang.add(rs.getString(16));

rowData.add(hang);

}

JTable jt = **new** JTable(rowData,columnNames);

DefaultTableCellRenderer tcr = **new** DefaultTableCellRenderer();// 设置table内容居中

tcr.setHorizontalAlignment(SwingConstants.***CENTER***);// 这句和上句作用一样

jt.setDefaultRenderer(Object.**class**, tcr);

jt.setEnabled(**false**);//不允许修改表格内容

jt.setFont(**new** Font("楷体", Font.***PLAIN***, 18));// 设置表格字体

JScrollPane jsp = **new** JScrollPane(jt);

JPanel jp1 =**new** JPanel();

jp1.add(jsp);

JFrame jf = **new** JFrame();

jf.add(jsp);

jf.setSize(985,900);

jf.setLocationRelativeTo(**null**);//窗口位置居中

jf.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

};

j4.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //返回登录监听

{

**public** **void** actionPerformed(ActionEvent e) {

f.setVisible(**false**);

**new** LoginUI().loginUI();

}

};

fh.addActionListener(listener1);

}

}

**增加信息**

**public** **class** AddTicket **implements** ActionListener{

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Statement stmt1;

Connection dbConn = **null**;

**public** **static** java.sql.Date StringToDate(String sDate) {

String str = sDate;

SimpleDateFormat format = **new** SimpleDateFormat("yyyyMMdd");

java.util.Date d = **null**;

**try** {

d = format.parse(str);

} **catch** (Exception e) {

JOptionPane.*showMessageDialog*(**null**, "请输入正确的日期格式！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

}

java.sql.Date date = **new** java.sql.Date(d.getTime());

**return** date;

}

**public** **void** actionPerformed(ActionEvent e) {

JFrame jf= **new** JFrame("增加列车信息");

JLabel lch = **new** JLabel(" 列车号 ");

JLabel lcz = **new** JLabel(" 列车长 ");

JLabel starting = **new** JLabel(" 起始站 ");

JLabel time = **new** JLabel("出发时间");

JLabel ending = **new** JLabel(" 终点站 ");

JLabel time2 = **new** JLabel("到站时间");

JLabel date = **new** JLabel(" 日期 ");

JLabel phone = **new** JLabel("联系电话");

JButton fh = **new** JButton(" 返回 ");

JButton zj = **new** JButton("确认增加");

JTextField f1 = **new** JTextField(10);

JTextField f2 = **new** JTextField(10);

JTextField f3 = **new** JTextField(10);

JTextField f4 = **new** JTextField(10);

JTextField f5 = **new** JTextField(10);

JTextField f6 = **new** JTextField(10);

JTextField f7 = **new** JTextField(10);

JTextField f8 = **new** JTextField(10);

jf.add(lch);

jf.add(f1);

jf.add(lcz);

jf.add(f2);

jf.add(starting);

jf.add(f3);

jf.add(time);

jf.add(f4);

jf.add(ending);

jf.add(f5);

jf.add(time2);

jf.add(f6);

jf.add(date);

jf.add(f7);

jf.add(phone);

jf.add(f8);

jf.add(zj);

jf.add(fh);

jf.setResizable(**false**); //不允许修改窗口

jf.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,35,10));

jf.setSize(300,350);

jf.setLocationRelativeTo(**null**);//窗口位置居中

jf.setVisible(**true**);

jf.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

ActionListener listener = **new** ActionListener() //确认增加监听

{

**public** **void** actionPerformed(ActionEvent e) {

String lc = f1.getText();

String lcz = f2.getText();

String starting = f3.getText();

String time = f4.getText();

String ending = f5.getText();

String time2 = f6.getText();

String date= f7.getText();

String phone = f8.getText();

java.sql.Date dt=*StringToDate*(date);

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="insert into TICKET\_INFORMATION values ('"+lc+"','"+lcz+"','"+starting+"','"+time+"','"+ending+"','"+time2+"','"+dt+"','"+phone+"')";

stmt.executeUpdate(s);

JOptionPane.*showMessageDialog*(**null**, "添加成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

f1.setText("");f2.setText("");f3.setText("");f4.setText("");

f5.setText("");f6.setText("");f7.setText("");f8.setText("");

}

**catch** (Exception e1)

{

JOptionPane.*showMessageDialog*(**null**, "请输入正确的时间信息格式！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

}

}

};

zj.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent e) {

jf.setVisible(**false**);

}

};

fh.addActionListener(listener1);

}

}

**删除信息**

**public** **class** DeleteTICKET **implements** ActionListener{

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Statement stmt1;

Connection dbConn = **null**;

**public** **void** actionPerformed(ActionEvent e) {

JFrame jf= **new** JFrame("删除列车信息");

JLabel lch = **new** JLabel("列车号 ");

JButton fh = **new** JButton("返回 ");

JButton zj = **new** JButton("确认删除");

JTextField f1 = **new** JTextField(10);

jf.add(lch);

jf.add(f1);

jf.add(zj);

jf.add(fh);

jf.setResizable(**false**); //不允许修改窗口

jf.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,35,10));

jf.setSize(300,250);

jf.setLocationRelativeTo(**null**);//窗口位置居中

jf.setVisible(**true**);

jf.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

ActionListener listener = **new** ActionListener() //确认删除监听

{

**public** **void** actionPerformed(ActionEvent e) {

String lc = f1.getText();

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s1="(select \*from TICKET\_INFORMATION where 列车号='"+lc+"')";

ResultSet r=stmt.executeQuery(s1); //游标 查询表内内容

**if**(r.next()) {

String s="delete from TICKET\_INFORMATION where 列车号='"+lc+"'";

stmt.executeUpdate(s);

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的信息删除成功","提示消息",JOptionPane.***PLAIN\_MESSAGE***);

f1.setText("");

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "暂无该列车的相关信息，请重新输入","提示消息",JOptionPane.***ERROR\_MESSAGE***);

}

}

**catch** (Exception e1)

{

e1.printStackTrace();

}

}

};

zj.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent e) {

jf.setVisible(**false**);

}

};

fh.addActionListener(listener1);

}

}

**修改信息**

**public** **class** UpdateTicket **implements** ActionListener{

String driverName = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String dbURL = "jdbc:sqlserver://127.0.0.1:1433;DatabaseName=Ticket\_System";

String userName = "sa";

String userPwd = "sa";

Statement stmt;

Statement stmt1;

Connection dbConn = **null**;

**public** **void** actionPerformed(ActionEvent e) {

JFrame jf= **new** JFrame("修改列车信息");

JLabel lc = **new** JLabel(" 列车号 ");

JComboBox<String>comboBox = **new** JComboBox<>();

comboBox.addItem("列车长");

comboBox.addItem("起始站");

comboBox.addItem("出发时间");

comboBox.addItem("终点站");

comboBox.addItem("到站时间");

comboBox.addItem("日期");

comboBox.addItem("联系电话");

JButton fh = **new** JButton("返回");

JButton sb = **new** JButton("确认修改");

JTextField f1 = **new** JTextField(10);

JTextField f2 = **new** JTextField(10);

jf.add(lc);

jf.add(f1);

jf.add(comboBox);

jf.add(f2);

jf.add(sb);

jf.add(fh);

jf.setResizable(**false**); //不允许修改窗口

jf.setLayout(**new** FlowLayout(FlowLayout.***CENTER***,35,10));

jf.setSize(300,200);

jf.setLocationRelativeTo(**null**);//窗口位置居中

jf.setVisible(**true**);

jf.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***); //关闭窗口，释放内存

ActionListener listener = **new** ActionListener() //确认修改监听

{

**public** **void** actionPerformed(ActionEvent e) {

String lc = f1.getText();

String xg = f2.getText();

**int** PD= comboBox.getSelectedIndex();

**try**

{

Class.*forName*(driverName);

dbConn = DriverManager.*getConnection*(dbURL, userName, userPwd);

stmt=dbConn.createStatement();

String s="(Select \* from TICKET\_INFORMATION where 列车号 = '"+lc+"')";

ResultSet r=stmt.executeQuery(s); //游标 查询表内内容

**if**(r.next()) {

**if**(PD==0) {

String s1="update TICKET\_INFORMATION SET 列车长 = '"+xg+"' WHERE 列车号 = '"+lc+"'";

stmt.executeUpdate(s1);

f1.setText("");f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的列车长信息修改成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

**else** **if**(PD==1){

String s1="update TICKET\_INFORMATION SET 起始站 = '"+xg+"' WHERE 列车号 = '"+lc+"'";

stmt.executeUpdate(s1);

f1.setText("");f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的起始站信息修改成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

**else** **if**(PD==2){

String s1="update TICKET\_INFORMATION SET 出发时间 = '"+xg+"' WHERE 列车号 = '"+lc+"'";

stmt.executeUpdate(s1);

f1.setText("");f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的出发时间信息修改成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

**else** **if**(PD==3){

String s1="update TICKET\_INFORMATION SET 终点站 = '"+xg+"' WHERE 列车号 = '"+lc+"'";

stmt.executeUpdate(s1);

f1.setText("");f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的终点站信息修改成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

**else** **if**(PD==4){

String s1="update TICKET\_INFORMATION SET 到站时间 = '"+xg+"' WHERE 列车号 = '"+lc+"'";

stmt.executeUpdate(s1);

f1.setText("");f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的到站时间信息修改成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

**else** **if**(PD==5){

String s1="update TICKET\_INFORMATION SET 日期 = '"+xg+"' WHERE 列车号 = '"+lc+"'";

stmt.executeUpdate(s1);

f1.setText("");f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的日期信息修改成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

**else** **if**(PD==6){

String s1="update TICKET\_INFORMATION SET 联系电话= '"+xg+"' WHERE 列车号 = '"+lc+"'";

stmt.executeUpdate(s1);

f1.setText("");f2.setText("");

JOptionPane.*showMessageDialog*(**null**, "列车号为"+lc+"的联系电话信息修改成功！！！","提示",JOptionPane.***PLAIN\_MESSAGE***);

}

}

**else** {

JOptionPane.*showMessageDialog*(**null**, "请输入正确的列车号！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

f1.setText("");f2.setText("");

}

}

**catch** (Exception e1)

{

JOptionPane.*showMessageDialog*(**null**, "请输入正确的信息格式！！！","提示",JOptionPane.***ERROR\_MESSAGE***);

}

}

};

sb.addActionListener(listener);

ActionListener listener1 = **new** ActionListener() //返回监听

{

**public** **void** actionPerformed(ActionEvent e) {

jf.setVisible(**false**);

}

};

fh.addActionListener(listener1);

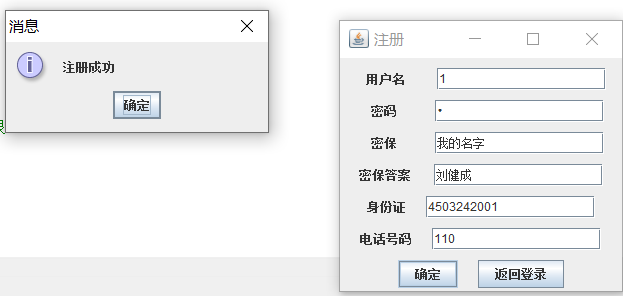
}

}

# 系统测试

## 5.1 用户注册功能测试

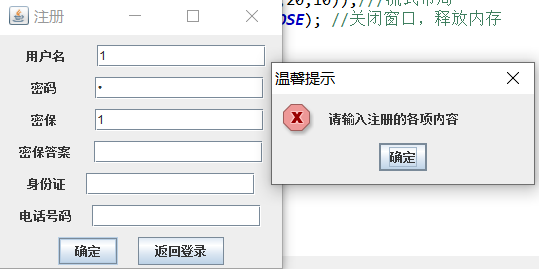
**当用户输入的用户名为被注册时，且其他数据皆符合规定时，注册成功，将用户数据导入用户信息表。**



**假如用户名已被注册，则提示“用户名已经存在，请重新输入”，清空文本框并返回注册界面。**



**假如用户并未将信息输入完全，则提示“请输入注册的各项内容”。**

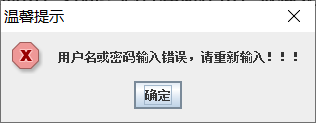


## 5.2 用户登录功能测试

**当输入的用户名和密码皆存储在用户数据表中，且一一对应，用户登录成功，进入查询列车界面。**

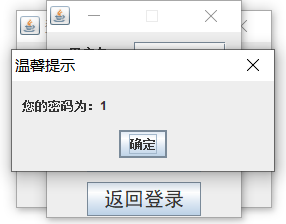
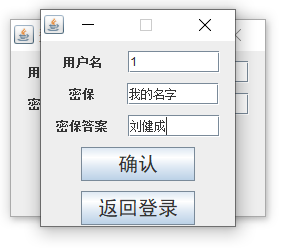
 

**当输入的用户名与密码不是一一对应，则提示“用户名或密码输入错误，请重新输入”，清空文本框，返回登录界面。**

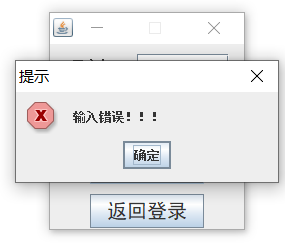
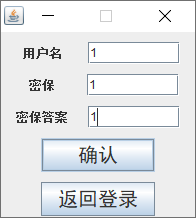


## 5.3 用户找回密码功能测试

**用户输入相关信息，若在用户表中查询到相关信息，输出用户密码。**



**用户输入的相关信息与用户表中的数据并不一致，提示“输入错误！！！”。**

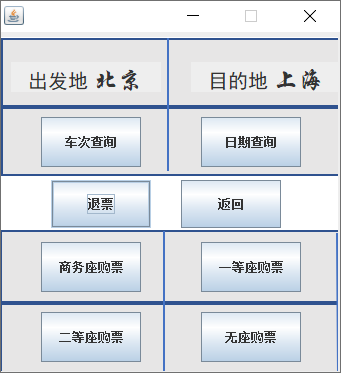


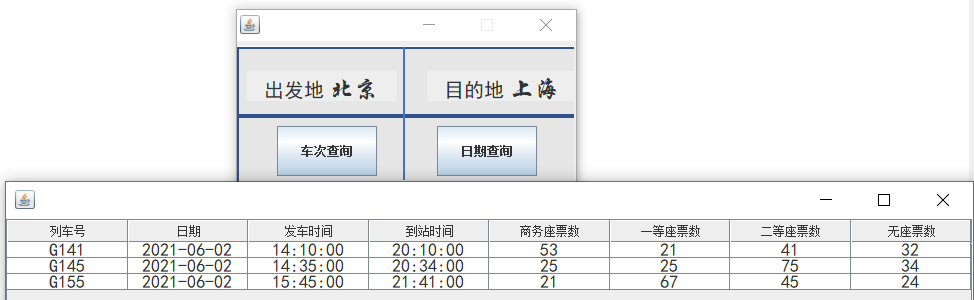
## 5.4 车站查询与日期查询

**当未找到用户选择的车站信息，则提示“很抱歉！为找到您所查询车站间的列车信息”，返回车站查询界面。**

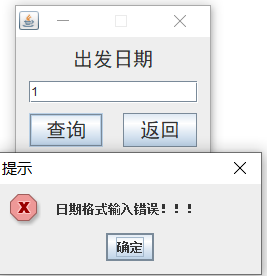


**若存放了用户选择的两站列车信息，则进入菜单模块，用户点击车次查询按键，则输出列车信息。**

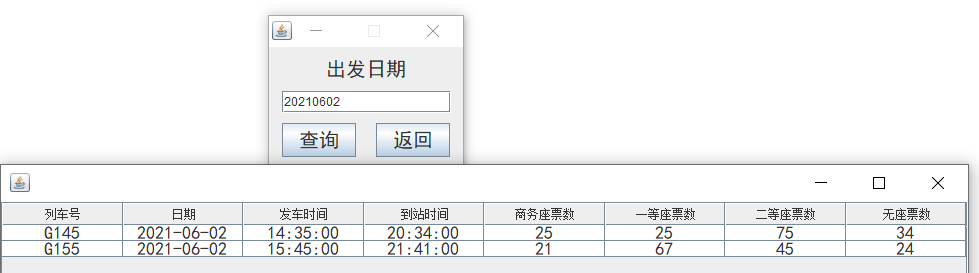
****



**当用户点击日期查询按键，输入的日期格式错误时，提示用户日期格式输入错误，清空文本框，返回日期查询界面。**

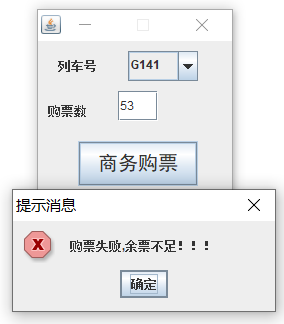


**当用户输入正确的日期格式，则输出该日经过起始站和终点站的列车相关信息。**

****

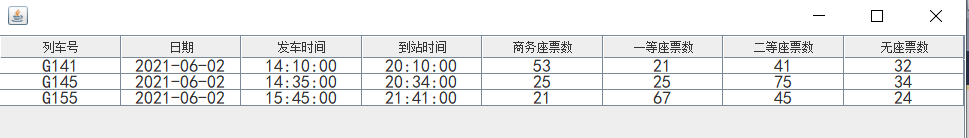
## 5.5 购票功能测试

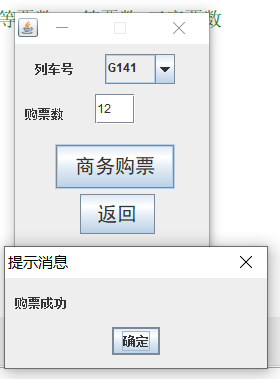
**用户购票车票，如果该类座位并无余票，则提示用户“购票失败，余票不足！！！”。**



**若有余票，则提示“购票成功”，相关票数减去购票数，将用户数据插入购票人表中。**

**购票前G141次列车的商务票为53。**





**购买G141次列车的商务座票12张后，信息变为41张。**

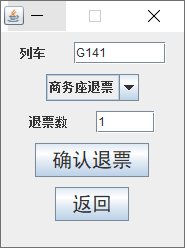


**同时将购票人信息插入购票人表中。**



## 5.6 退票功能测试

**当用户输入已购买的列车号以及票数时，购票人表中相应减少对应票数，车票数据表中相应增加对应票数。**

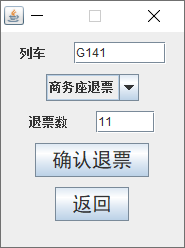
**G141次列车的商务座票数加1。**



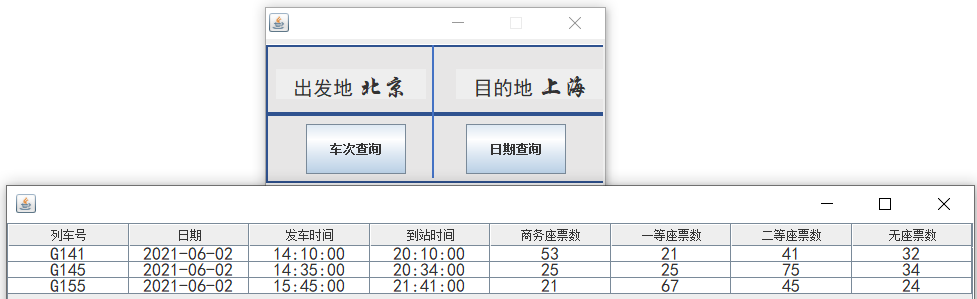
**购票人表中的商务座票数减1。**



**一次性退多张票。**

**列车表中G141次列车的商务座票数增加11。**



**购票人表中用户名为1的商务票数减去11。**

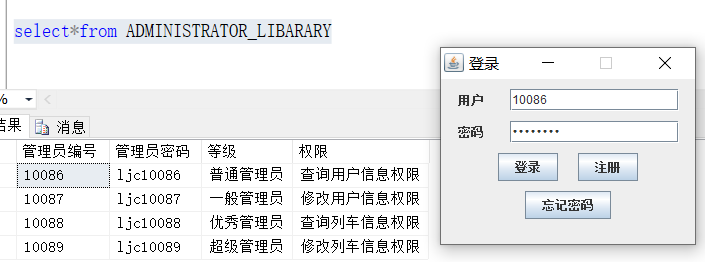


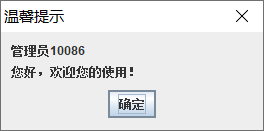
**退未购买的车票。**

****

## 5.7 管理员登录功能测试

**管理员输入管理员信息表中存在的数据。**



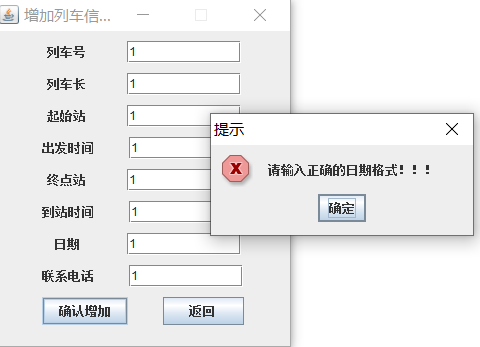


**进入管理员界面。**

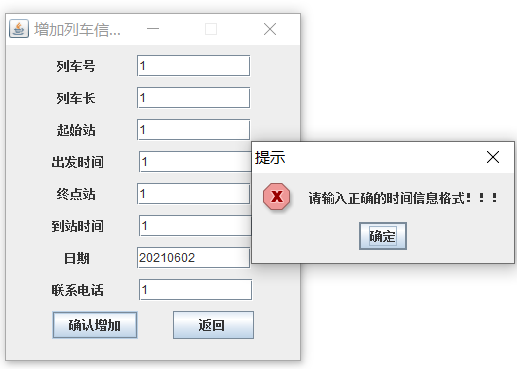
****

## 5.8 管理员增加列车信息功能测试

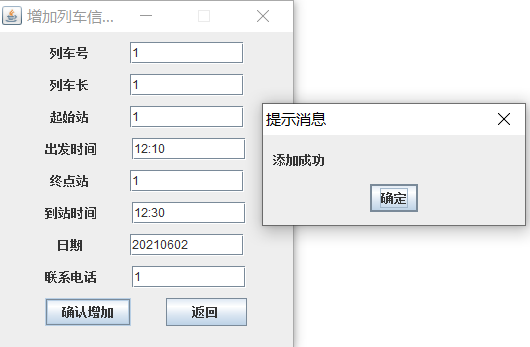
**当管理员输入的日期不合法时，提示“请输入正确的日期格式！！！”。**



**当管理员输入的时间格式不合法时，提示“请输入正确的时间信息格式！！！”**



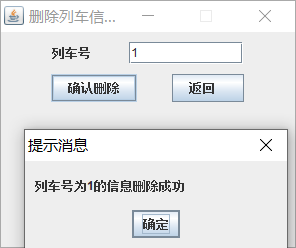
**当管理员正确输入数据**

****

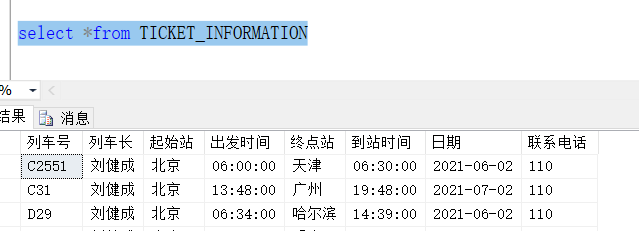
**将信息插入到列车信息表**

****

## 5.9 管理员删除列车信息功能测试

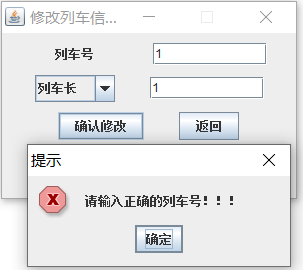


**列车信息表中删除对应列车信息**

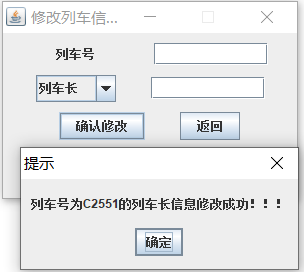
****

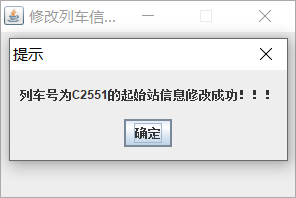
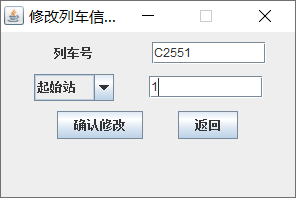
## 6.0 管理员修改列车信息功能测试

**当管理员企图修改未存在的列车号时，提示“请输入正确的列车！！！”,清空文本框，返回修改界面。**

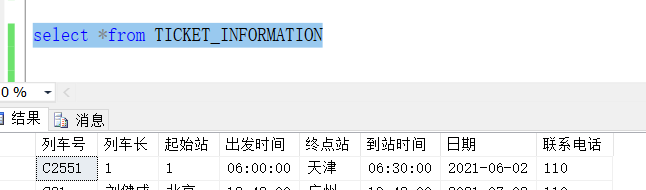
****

**管理员修改G551次列车信息**

****

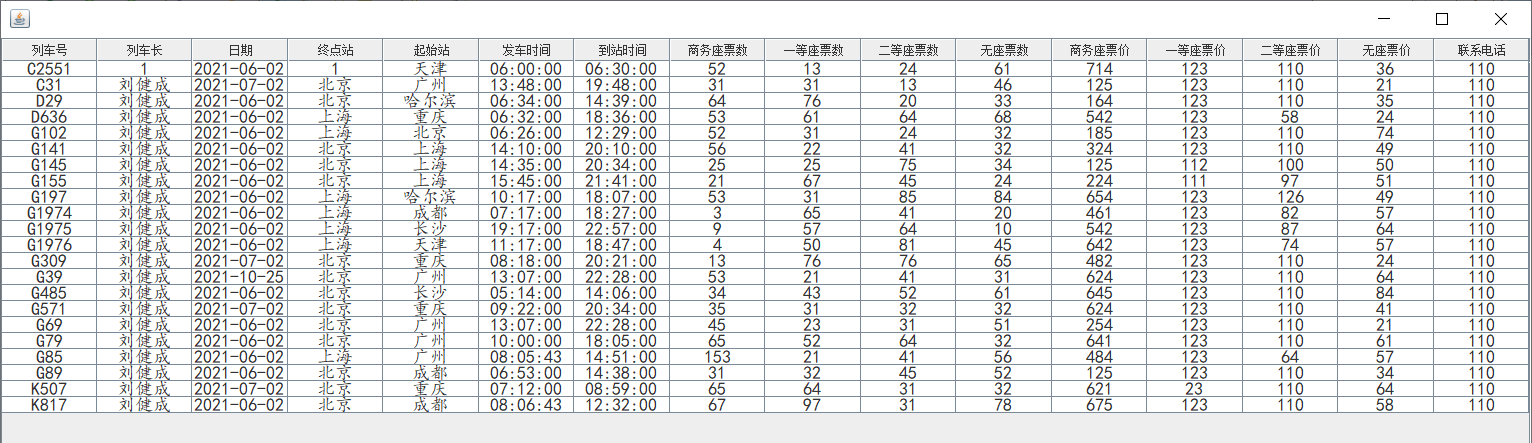
****

**列车信息表中C2551次列车信息修改为**

****

## 6.1 管理员查询列车信息功能测试

**管理员点击查询信息按钮，输出车站所有列车信息**



# 结论

本系统完成了用户的注册登录，查询列车信息，自定义购票退票等功能，以及管理员的登录功能和管理员对列车信息的增删改查功能。本系统能利用计算机快速地处理信息，满足了乘客与车站管理人员之间的信息交与需求。本系统的开发为车站管理和乘客购票提供了较大的便捷，使乘客足不出户就可以购买车票，也为车站提供了一个良好的售票窗口，减轻了工作负担。当然，本系统也有不足之处

（1）对用户等数据的保护并不安全，未进行封装。

（2）界面还不够美观，功能也并不完善。

（3）耗时耗力，事倍功半。

当今流行运用能以模板化的方式简单、高效地添加动态网页内容的JSP技术进行界面设计，期望能针对这些建议结合自己的技术能力对系统加以改进、完善。

## 参考文献

[1]刘金岭、冯万利、张有东主编，数据库原理及应用——SQL server 2012，清华大学出版社，2017.7.

[2]黑马程序员编著，Java基础入门（第二版），清华大学出版社，2018.12

[3] 赵满来编著，可视化Java SWT/JFace GUI程序设计教程——基于Eclipse Windo，清华大学出版社，2017.8

[4] 赵满来编著，可视化Java GUI程序设计，清华大学出版社，2010

[5] 姜志强编著，Java语言程序设计，电子工业出版社，2021

[6] 明日科技编著，Java从入门到精通（第5版），清华大学出版社，2019.3

# 附录