

Core Java Project

Project Title: **RAILWAY RESERVATION SYSTEM**

Aim: The main of “**Railway Reservation System**” is to make the reservation process convenient for user.

Languages:

- Java and SQL

Software Requirements:

- Eclipse for Java
- MySQL for SQL

Synopsis:

In this **Railway Reservation System** there are 5 main operations-

- **Registration** - Passenger can perform that operation.
- **Login** - Both admin and passenger can perform that operation.
- **Seat Reservation** - Passenger can perform that operation.
- **Seat Cancellation**- Passenger can perform that operation.
- **Display** - Both the admin and passenger perform the operation.

The control is given for both the admin and also user. Admin can perform two operations one is the fetching information about the passenger who all are reserved the seat and second one is admin can able the seat how many seats are reserved. In that Passenger side there is five main operations are given. The first one is registration operation it will make permanent account creation. Second one is login whenever you need the seat reservation for that security purpose this module will help. Third one is seat reservation this will take personal information's about that person, railway timing, trip type, seat type, ticket fare. Forth one is display process after the reservation if passenger need to be seen

the details about the reservation it will show the information about the seat reservation. Final one is seat cancellation this will also asking the security questions after that it will cancel and also it refunds the ticket fare.

Initial Setup:

In Java:

Mavan Project Name: Project

Package Name: com.project

Classes Name :1. AdminClass

2. CustomerClass

3. MainClass

4. DataBaseConnection

In Database:

Create database query:

```
mysql> create projectdb;
```

Using database:

```
mysql> use projectdb;  
Database changed
```

Used Tables:

```
mysql> show tables;  
+-----+  
| Tables_in_projectdb |  
+-----+  
| admin_account        |  
| customer_acc         |  
| new_user_register    |  
| seatbook             |  
| ticketdetails        |  
| tripdetails          |  
+-----+  
6 rows in set (0.09 sec)
```

MainClass Code:

/* Railway Reservation System

* CSR CAPGEMINI TRAINING PROJECT
* EDUBRIDGE INDIA PRIVATE LIMITED
* PROJECT TITLE: RAILWAY RESERVATION SYSTEM
* UNDER THE GUIDENCE OF TRAINER MRS.INDRAKA MALLI
* @DONE BY Anushree Joshi

* In RAILWAY RESERVATION SYSTEM:

* MAIN OPERATIONS:

- * 1.REGISTRATION -->passenger can perform that operation.
- * 2.LOGIN -->Both admin and passenger can perform that operation.
- * 3.SEAT RESERVATION -->Passenger can perform that operation.
- * 4.SEAT CANCELLATION -->passenger can perform that operation.
- * 5.DISPLAY -->Both the admin and passenger perform the operation.

*

* case 1: Admin login

* Admin screen

* 1. show user ticket details

* 2. show how many tickets are reserved

* case 2: New Customer login

* - It will register the new user

* case 3: Registered Customer

* - registered user can book ticket

*/

package com.project;

import java.sql.SQLException;

```
import java.util.Scanner;
```

```
public class MainClass {
```

```
    public static void main(String[] args) throws SQLException{
```

```
        // TODO Auto-generated method stub
```

```
        String un="admin";
```

```
        String userpass="admin123";
```

```
        while(true) {
```

```
            System.out.println("*****Railway  
Reservation System*****");
```

```
            int ch;
```

```
            Scanner sc=new Scanner(System.in);
```

```
            System.out.println("Login/Admin or User");
```

```
            System.out.println("1. Admin login");
```

```
            System.out.println("2. New Customer Login");
```

```
            System.out.println("3. Registered Customer");
```

```
            System.out.println("4. Exit");
```

```
            System.out.println("Enter your choice");
```

```
            ch=sc.nextInt();
```

```
            switch(ch) {
```

```
                case 1:
```

```
                    System.out.println("For admin login please enter  
credentials");
```

```
                    System.out.println("Enter name");
```

```

        String aname=sc.next();
        System.out.println("Enter password");
        String password=sc.next();
        if(aaname.equals("admin")&&
password.equals("admin123")) {
            System.out.println("-----Admin Screen-----
-----");

            System.out.println("Admin Options");
            System.out.println("1. Show User Ticket Details");
            System.out.println("2. Show Reserved Seats");


            System.out.println("Enter your choice");
            int adminch=sc.nextInt();
            switch(adminch) {
            case 1: System.out.println("Display all tickets details");
                AdminClass.displayCustomer();
                break;
            case 2: System.out.println("Display How many seats are
reserved");
                AdminClass.admin_account();
                break;


            default: System.out.println("Invalid choice");
            }
            break;
        }
        break;

        case 2:      System.out.println("-----New User
Registration-----");

```

```

        CustomerClass.newUserRegister();
        break;
    case 3: //User login
        System.out.println("-----Registered User Login-----");
        CustomerClass.userLogin();
        break;
    case 4: //exit
        break;
    }

    System.out.println("Type y if you want to continue or press
n to terminate");

    ch=sc.next().charAt(0);
    if(ch=='n'||ch=='N')
        break;

    }

    System.out.println("Program is terminated");

}

}

```

Database Creation:

Admin Table:

```
mysql> desc admin_account;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| totalamt   | float(10,2)   | YES  |     | NULL    |       |
| seatno     | int           | YES  | MUL | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.07 sec)
```

User Registration Table:

```
mysql> desc new_user_register;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| user_name  | varchar(20)   | YES  |     | NULL    |       |
| password   | varchar(10)   | YES  |     | NULL    |       |
| userId     | varchar(20)   | NO   | PRI | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

Database Connection Code:

```
package com.project;
```

```
import java.sql.Connection;
```

```
import java.sql.DriverManager;
```

```
public class DataBaseConnection {
```

```
    static String driver ="com.mysql.cj.jdbc.Driver";
```

```
    static String url="jdbc:mysql://localhost:3306/projectdb";
```

```
    static String un="root";
```

```
    static String up="root";
```

```
    static Connection conn;
```

```
    public static Connection getConnection() {
```

```

        try {
            Class.forName(driver);
            conn=DriverManager.getConnection(url,un,up);
            if(conn==null) {
                System.out.println("Connection Error");
            }
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        return conn;
    }
}

```

Registration Code:

For Admin:

1. The admin registration is done by back-end process. Reason is only one admin for the whole application that why the admin registration is created by backend process.
2. There is no option is given to create the admin registration process at front end.

User Registration Code:

/* new user registration

- * It accepts user id, name, password then stores in database
- * After successful registration it will ask for bank account details it is like digital wallet for
- * Railway reservation system for booking ticket
- */


```
//new user registration
```

```
    public static void newUserRegister() throws SQLException {  
        // TODO Auto-generated method stub  
        System.out.println("Welcome To Railway Reservatiion System");  
  
        myconn=DataBaseConnection.getConnection();  
        String new_register="select * from new_user_register where  
userId=?";  
        pst=myconn.prepareStatement(new_register);  
        System.out.println("Enter User ID");  
        userId=sc.nextInt();  
        pst.setInt(1, userId);  
        rs=pst.executeQuery();  
        if(!rs.next()) {  
            System.out.println("Enter user name to register");  
            username=sc.next();  
            System.out.println("Enter password");  
            password=sc.next();  
            while(true) {  
                String userinsert="insert into new_user_register  
values(?,?,?)";  
                pst=myconn.prepareStatement(userinsert);  
                pst.setString(1,username);  
                pst.setString(2, password);  
                pst.setInt(3, userId);  
                int i=pst.executeUpdate();  
                if(i>0) {
```

```

Registered!!!!");

Railway System");

System.out.println("Successfully

System.out.println("Create Bank Account In

System.out.println("Enter Your Account No");
String cus_acc=sc.next();
System.out.println("Enter Your Pin");
int cus_acc_pin=sc.nextInt();
System.out.println("Enter how much money
you want to deposit");

double bal=sc.nextDouble();
String seldeposit="select * from customer_acc
where userId=?";

pst=myconn.prepareStatement(seldeposit);
pst.setInt(1, userId);
rs=pst.executeQuery();
if(!rs.next()) {
    String deposit="insert into customer_acc
values(?,?,?,?)";

    pst=myconn.prepareStatement(deposit);
    pst.setString(1, cus_acc);
    pst.setInt(2, cus_acc_pin);
    pst.setDouble(3, bal);
    pst.setInt(4, userId);
    int depo=pst.executeUpdate();
    if(depo>0) {
        System.out.println("Bank Account
Created Successfully");
    }
}

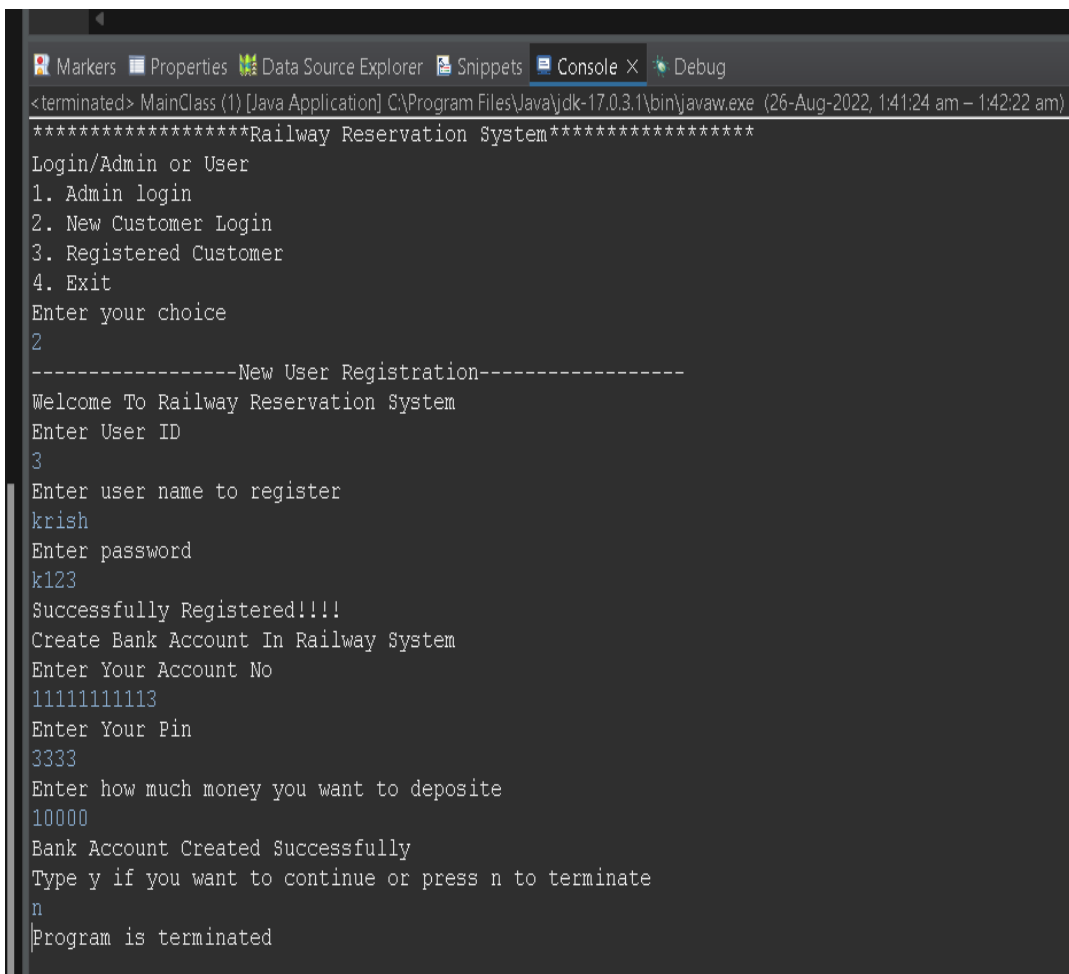
```

```

        }
        break;
    }
}
}
}else {
    System.out.println("User ID is already Registered");
}
}
}

```

User Registration Output:



```

<terminated> MainClass (1) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (26-Aug-2022, 1:41:24 am – 1:42:22 am)
*****Railway Reservation System*****
Login/Admin or User
1. Admin login
2. New Customer Login
3. Registered Customer
4. Exit
Enter your choice
2
-----New User Registration-----
Welcome To Railway Reservation System
Enter User ID
3
Enter user name to register
krish
Enter password
k123
Successfully Registered!!!!
Create Bank Account In Railway System
Enter Your Account No
1111111113
Enter Your Pin
3333
Enter how much money you want to deposit
10000
Bank Account Created Successfully
Type y if you want to continue or press n to terminate
n
Program is terminated

```

User Registration record stored in database:

```
mysql> select * from new_user_register;
+-----+-----+-----+
| user_name | password | userId |
+-----+-----+-----+
| anushree  | anu123   | 1      |
| ram       | ram123   | 2      |
| krish     | k123     | 3      |
+-----+-----+-----+
3 rows in set (0.01 sec)

mysql> select * from customer_acc;
+-----+-----+-----+-----+
| accno      | pin  | balance | userId |
+-----+-----+-----+-----+
| 1111111111 | 1111 | 8850.00 | 1      |
| 1111111112 | 2222 | 10000.00 | 2      |
| 1111111113 | 3333 | 10000.00 | 3      |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

Admin Login:

/*

* In this admin login process the back end we created one user name and password for the admin

*

* That user name and password can be used to login admin login module

*

* If the login process get successful means admin can perform display operations

*

* 1. display details of all seats - it gives information all the seat number

*

* 2. display seat reserved- it gives result about how many seats are reserved

*

*/

System.out.println("For admin login please enter credentials");

System.out.println("Enter name");

```

        String aname=sc.next();
        System.out.println("Enter password");
        String password=sc.next();
        if(aname.equals("admin")&&
password.equals("admin123")) {
            System.out.println("-----Admin Screen-----
-----");

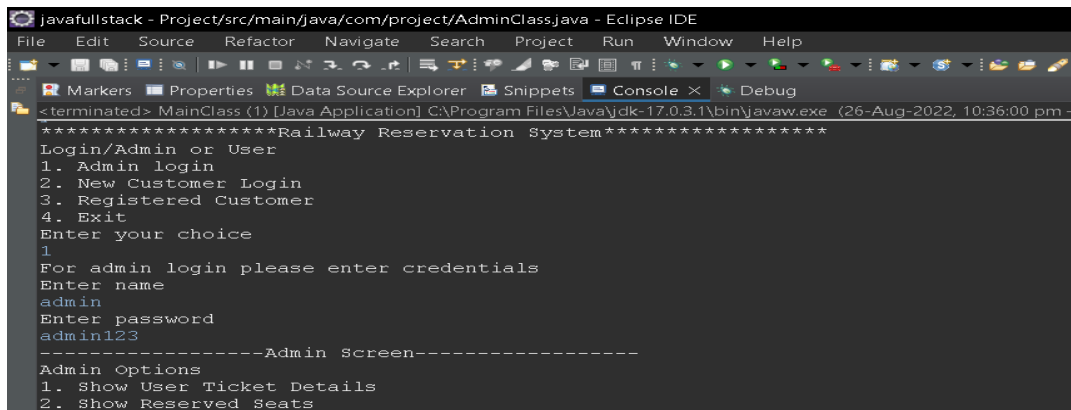
            System.out.println("Admin Options");
            System.out.println("1. Show User Ticket Details");
            System.out.println("2. Show Reserved Seats");


            System.out.println("Enter your choice");
            int adminch=sc.nextInt();
            switch(adminch) {
            case 1: System.out.println("Display all tickets details");
                AdminClass.displayCustomer();
                break;
            case 2: System.out.println("Display How many seats are
reserved");
                AdminClass.admin_account();
                break;


            default: System.out.println("Invalid choice");
            }
            break;
        }
        break;

```

Admin Login Output:



```
javafullstack - Project/src/main/java/com/project/AdminClass.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Markers Properties Data Source Explorer Snippets Console X Debug
<terminated> MainClass (1) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (26-Aug-2022, 10:36:00 pm --
*****Railway Reservation System*****
Login/Admin or User
1. Admin login
2. New Customer Login
3. Registered Customer
4. Exit
Enter your choice
1
For admin login please enter credentials
Enter name
admin
Enter password
admin123
-----Admin Screen-----
Admin Options
1. Show User Ticket Details
2. Show Reserved Seats
```

User Login Code:

```
/*
* In this passenger login process the back end we created one user name and
* password for the user
*
* That user name and pass word can be used to login user login module
*
* If the login process get successful means user can perform operations
* Operation is seat reservation, display and cancellation
*
* 1. seat reservation-> this module is used to reserve the seat and
* display ticket info
* 2. seat cancellation-> this module is used to cancelling the reserved seat
*
*/
```

//Registered User Login Process

```
public static void userLogin() throws SQLException {
    // TODO Auto-generated method stub
    myconn=DataBaseConnection.getConnection();
```

```
while(true) {  
    System.out.println("Enter User Name");  
    username=sc.next();  
    System.out.println("Enter Password");  
    password=sc.next();  
    String checkusername="select * from new_user_register  
where user_name=?";  
    pst=myconn.prepareStatement(checkusername);  
    pst.setString(1, username);  
    rs=pst.executeQuery();  
    if(rs.next()) {  
        String checkpassword="select * from  
new_user_register where password=?";  
        pst=myconn.prepareStatement(checkpassword);  
        pst.setString(1, password);  
        rs=pst.executeQuery();  
        if(rs.next()) {  
            System.out.println("Login Successfull");  
  
            while(true) {  
                System.out.println("1. Seat Reservation");  
                System.out.println("2. Seat Cancellation");  
                System.out.println("3. Exit");  
  
                System.out.println("Enter your choice..");  
                int ch1=sc.nextInt();  
                switch(ch1) {
```

```

Reservation");
        case 1: System.out.println("You choosed Seat
CustomerClass.bookTicket();
        break;
        case 2: System.out.println("You choosed Seat
Cancellation");
        CustomerClass.cancelTicket();
        break;
        case 3://exit
        break;
        default: System.out.println("Invalid Choice");
        break;
    }

    System.out.println("Do you want to continue the seat
reservation or seat cancellation yes/no");
    String in=sc.next();
    if(in.equalsIgnoreCase("no"))
        break;

    }

    }

    }else {
        System.out.println("Invalid user name");
        System.out.println("If you are already registred then
again try to login");
        System.out.println("Otherwise go for Registration");
    }
}

```



```

        System.out.println("Do you want to go for one more
chance yes/no");

        String input=sc.next();

        if(input.equalsIgnoreCase("no")) {

            break;

        }

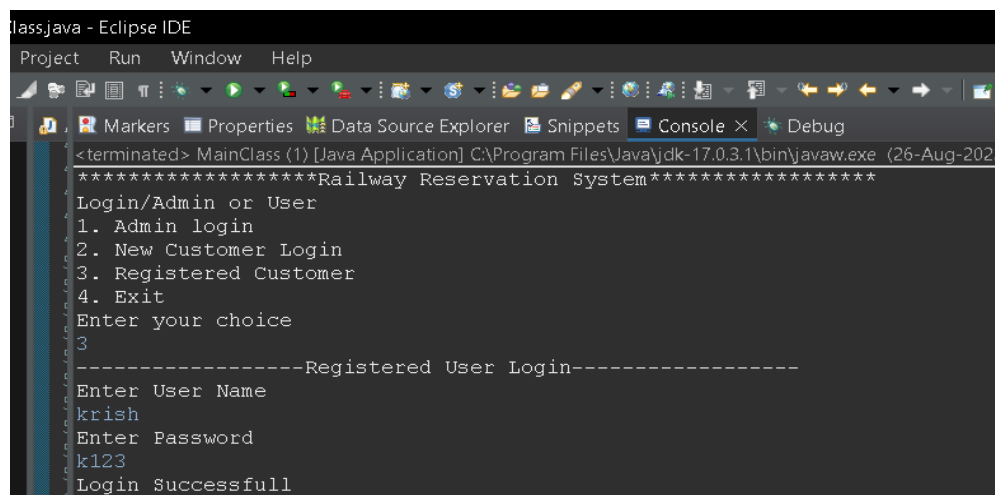
    } break;

}

}

```

User Login Output:



```

class.java - Eclipse IDE
Project Run Window Help
Markers Properties Data Source Explorer Snippets Console X Debug
<terminated> MainClass (1) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (26-Aug-202
*****Railway Reservation System*****
Login/Admin or User
1. Admin login
2. New Customer Login
3. Registered Customer
4. Exit
Enter your choice
3
-----Registered User Login-----
Enter User Name
krish
Enter Password
k123
Login Successfull

```

Reservation Process:

//Seat Reservation:

/**

* In this seat reservation process there are trip choosing option, timing option, ticket type, ticket class, payment

* These are the key modules are working in side this method.

*

* While reservation process it also takes the personal information about the passenger.

*

* Main thing in this process choosing the seat no it shows how many seats are reserved and the seat numbers also.

*/

```
static Connection myconn = null;
```

```
    static ResultSet rs = null;
```

```
    static PreparedStatement pst = null;
```

```
    //newUserRegister() method variables
```

```
    static String username;
```

```
    static String password;
```

```
    static int userId;
```

```
    //bookSeat() tripdetails
```

```
    static String tripname;
```

```
    static String departure;
```

```
    static String arrival;
```

```
    static String name;
```

```
    //bookSeat() ticketdetails
```

```
    static float ticketprice;
```

```
    static String coach;
```

```
    static int tid;
```

```
    static float total;
```

```
    static int seatno;
```

```
    static long accno;
```

```
    static Scanner sc = new Scanner(System.in);
```

```
//Seat Reservation
```

```
    public static void bookTicket() throws SQLException {
```

```
        int totalNoOfSeats = 50;
```

```

int avalseat=50;

myconn = DataBaseConnection.getConnection();
String seat = "select seatno from seatbook order by seatno";
pst = myconn.prepareStatement(seat);
rs = pst.executeQuery();
while (rs.next()) {
    System.out.print(rs.getInt(1) + " ");
}
System.out.println("Already These seats are reserved");
String count = "select count(*) from seatbook";
pst = myconn.prepareStatement(count);
rs = pst.executeQuery();
while (rs.next()) {
    int reservedseat = rs.getInt(1);

    // selecting seat no
    while (true) {
//            System.out.println(reservedseat+" "+avalseat);
        System.out.println("Total number of seats " +
totalNoOfSeats);
        System.out.println("available seat count " + (avalseat -
reservedseat));
        System.out.println("Enter Seat No:");
        seatno = sc.nextInt();
        String sel = "select * from seatbook where seatno =?";
        pst = myconn.prepareStatement(sel);

```

```

        pst.setInt(1, seatno);
        rs = pst.executeQuery();
        // entering customer details
        if (!rs.next()) {
            System.out.println("Enter Customer Name:");
            name = sc.next();
            System.out.println("Enter phone no:");
            String phone = sc.next();

            // select coach

            while(true) {
                System.out.println("Which Trip you want to choose");
                System.out.println("1. Banglore To Pune");
                System.out.println("2. Banglore To Mumbai");
                System.out.println("3. Banglore To Hyderabad");
                int ch=sc.nextInt();
                if(ch<4) {
                    switch(ch) {
                        case 1: System.out.println("You selected
Banglore to Pune Trip");
                            tripname="Banglore To Pune";
                            break;
                        case 2: System.out.println("You selected
Banglore to Mumbai Trip");
                            tripname="Banglore To Mumbai";
                            break;

```

```

        case 3: System.out.println("You selected
Banglore to Hyderabad Trip");
        tripname="Banglore to Hyderabad";
        break;
    }
    break;
}
else {
    System.out.println("Invalid Option");
}
}
while(true) {
    System.out.println("Select which timing you want");
    System.out.println("1. DEPARTURE TIME is
6:00AM ARRIVAL TIME is 2:00PM");
    System.out.println("2. DEPARTURE TIME is
7:00AM ARRIVAL TIME is 5:00PM");
    System.out.println("3. DEPARTURE TIME is
3:00PM ARRIVAL TIME is 10:00PM");
    int ch=sc.nextInt();
    if(ch<4) {
        switch(ch){
            case 1: System.out.println("Your DEPARTURE
TIME is 6:00AM");
            departure="6:00:00";
            System.out.println("Your ARRIVAL TIME is
2:00PM");
            arrival="2:00:00";

            break;

```

```

TIME is 7:00AM");

5:00PM");

TIME is 3:00PM");

10:00PM");

case 2: System.out.println("Your DEPARTURE
departure="7:00:00";
System.out.println("Your ARRIVAL TIME is
arrival="5:00:00";
break;

case 3: System.out.println("Your DEPARTURE
departure="3:00:00";
System.out.println("Your ARRIVAL TIME is
arrival="10:00:00";
break;
}
break;
}else {
    System.out.println("Invalid option..");
}
}
while (true) {
    System.out.println("Enter which ticket you want to
book");

    System.out.println("1. First_AC Price 1940 Rs");
    System.out.println("2. Second_AC Price 1150 Rs");
    System.out.println("3. Third_AC Price 950 Rs");
    tid = sc.nextInt();
    if (tid < 4) {

```

```

switch (tid) {
case 1:
    System.out.println("you selected first_AC
ticket");

    coach = "First_AC";
    ticketprice = 1940;
    break;
case 2:
    System.out.println("you selected second_AC
ticket");

    coach = "Second_AC";
    ticketprice = 1150;
    break;
case 3:
    System.out.println("you selected third_AC
ticket");

    coach = "Third_AC";
    ticketprice = 950;
    break;
}
break;
} else {
    System.out.println("invalid choice");
}
}

// payment process

//PAYMENT PROCESS

```

/**

* The payment process also very simple and also securely handled.

*

* It also having checking processes taking informations from the passengers account number

* and pin it two data's are correct then it do the payment process

*

* Here the account number or pin number is invalid means it gives the alert and gives one more chances.

*

* May be both are correct but the balance is insufficient means it gives that insufficient message also.

*

* If the payment is done then only it reserve that seat otherwise it not takes all the information before the payment option.

*/

```
                while (true) {
                    total = ticketprice;
                    //
System.out.println("total"+ticketprice+"\t"+total);
                    System.out.println("Payment Section");
                    System.out.println("Enter UserId:");
                    userId=sc.nextInt();
                    String checkUserid="select accno from
customer_acc where userId=?";

                    pst=myconn.prepareStatement(checkUserid);
                    pst.setInt(1, userId);
                    rs=pst.executeQuery();
                    if(rs.next()) {
```



```

        System.out.println("Enter account no");
        accno = sc.nextLong();
        System.out.println("Enter pin");
        int pin = sc.nextInt();
        String paymentacc = "select accno from
customer_acc where accno=?";

        pst =
myconn.prepareStatement(paymentacc);

        pst.setLong(1, accno);
        rs = pst.executeQuery();
        if (rs.next()) {
            String paymentpin = "select pin
from customer_acc where pin=?";

            pst =
myconn.prepareStatement(paymentpin);

            pst.setInt(1, pin);
            rs = pst.executeQuery();

            if (rs.next()) {
                String in = "insert into seatbook
values(?,?,?)";

                pst =
myconn.prepareStatement(in);

                pst.setInt(1, seatno);
                pst.setString(2, name);
                pst.setString(3, phone);
                int j = pst.executeUpdate();

                if(j>0) {

```

```
values(?,?,?,?);
```

```
pst=myconn.prepareStatement(ins1);
```

```
String ins1="insert into tripdetails
```

```
pst.setString(1,tripname);
```

```
pst.setString(2, departure);
```

```
pst.setString(3, arrival);
```

```
pst.setInt(4, seatno);
```

```
int k=pst.executeUpdate();
```

```
ticketdetails values(?,?,?,?);
```

```
myconn.prepareStatement(ins);
```

```
if (k > 0) {
```

```
String ins = "insert into
```

```
pst =
```

```
pst.setString(1, coach);
```

```
pst.setFloat(2, ticketprice);
```

```
pst.setInt(3, seatno);
```

```
pst.setFloat(4, total);
```

```
int i = pst.executeUpdate();
```

```
if (i > 0) {
```

```
String bal = "select balance from  
customer_acc where(select balance from customer_acc where accno=?)>0";
```

```
pst =
```

```
myconn.prepareStatement(bal);
```

```
pst.setLong(1, accno);
```

```
rs = pst.executeQuery();
```

[illegible]

```

                                break;

                    }else {

                                break;

                    }

                    }

                    break;
                }}
            else {
                System.out.println("insufficient
balance");

                String del = "delete from seatbook
where seatno =?";

                pst =

                myconn.prepareStatement(del);

                pst.setInt(1, seatno);
                int df = pst.executeUpdate();
                if (df > 0) {
                    String del1 = "delete from
ticketdetails where seatno =?";

                    pst =

                    myconn.prepareStatement(del1);

                    pst.setInt(1, seatno);
                    int ds = pst.executeUpdate();
                    if (ds > 0) {
                        System.out.println("check your
balance");

                        break;
                    }
                }
            }
        }
    }
}

```

```
}  
  
}  
  
}  
  
}  
  
}  
  
}  
  
}else {  
    System.out.println("invalid account no or pin");  
}  
  
break;  
}else{  
System.out.println("Invalid User Id");  
  
}  
  
        break;  
}  
}  
}else {  
    System.out.println("this seat no is already reserved");  
}  
avalseat = totalNoOfSeats - 1;  
  
break;
```

Seat Reservation Output:

```
a/com/project/CustomerClass.java - Eclipse IDE
Navigate Search Project Run Window Help
Markers Properties Data Source Explorer Snippets Console X Debug
<terminated> MainClass (1) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (26-Aug-2022, 10:53:48 pm – 10:57:26 pm)
*****Railway Reservation System*****
Login/Admin or User
1. Admin login
2. New Customer Login
3. Registered Customer
4. Exit
Enter your choice
3
-----Registered User Login-----
Enter User Name
krish
Enter Password
kl23
Login Successfull
1. Seat Reservation
2. Seat Cancellation
3. Exit
Enter your choice..
1
You choosed Seat Reservation
1 2 Already These seats are reserved
Total number of seats 50
available seat count 48
Enter Seat No:
3
Enter Customer Name:
krish
Enter phone no:
9087890767
Which Trip you want to choose
1. Banglore To Pune
2. Banglore To Mumbai
3. Banglore To Hyderabad
3
You selected Banglore to Hyderabad Trip
Select which timing you want
1. DEPARTURE TIME is 6:00AM ARRIVAL TIME is 2:00PM
2. DEPARTURE TIME is 7:00AM ARRIVAL TIME is 5:00PM
3. DEPARTURE TIME is 3:00PM ARRIVAL TIME is 10:00PM
1
```

```
a/com/project/CustomerClass.java - Eclipse IDE
Navigate Search Project Run Window Help
Markers Properties Data Source Explorer Snippets Console X Debug
<terminated> MainClass (1) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (26-Aug-2022, 10:53:48 pm – 10:57:26 pm)
You selected Banglore to Hyderabad Trip
Select which timing you want
1. DEPARTURE TIME is 6:00AM ARRIVAL TIME is 2:00PM
2. DEPARTURE TIME is 7:00AM ARRIVAL TIME is 5:00PM
3. DEPARTURE TIME is 3:00PM ARRIVAL TIME is 10:00PM
1
Your DEPARTURE TIME is 6:00AM
Your ARRIVAL TIME is 2:00PM
Enter which ticket you want to book
1. First_AC Price 1940 Rs
2. Second_AC Price 1150 Rs
3. Third_AC Price 950 Rs
1
you selected first_AC ticket
Payment Section
Enter UserId:
3
Enter account no
1111111113
Enter pin
3333
Ticket Booked
```

Seat Cancellation Process:

/**

* This cancellation process is given to the passengers convenient.

*

- * If may be situation passenger not able to travel means it will gives the chance
- * to cancelling the seat and conditions applied amount is refund.
- *
- * It asking the major information regarding that passengers email id,
- * username, password and seat number every information are correct
- * then only it will cancel the seat otherwise it will not cancel.
- */

//cancel ticket

```

public static void cancelTicket() throws SQLException {
    myconn = DataBaseConnection.getConnection();
    while (true) {
        System.out.println("Enter Your User name");
        username=sc.next();
        System.out.println("Enter Your Password");
        password=sc.next();
        System.out.println("Enter User ID");
        userId=sc.nextInt();
        System.out.println("Enter Your Seat No");
        seatno=sc.nextInt();
        String sel = "select * from new_user_register where
userId=?";

        pst = myconn.prepareStatement(sel);
        pst.setInt(1, userId);
        rs = pst.executeQuery();
        if(rs.next()) {
            String sel1="select * from new_user_register where
user_name=?";

            pst = myconn.prepareStatement(sel1);

```

```

        pst.setString(1, username);
        rs = pst.executeQuery();

        if(rs.next()) {
            String refund="select * from ticketdetails where seatno=?";
            pst=myconn.prepareStatement(refund);
            pst.setInt(1, seatno);
            rs = pst.executeQuery();
            while(rs.next()) {
                float moneyreturn=rs.getFloat(4);

                String sel2="select * from new_user_register where
password=?";

                pst = myconn.prepareStatement(sel2);
                pst.setString(1, password);
                rs = pst.executeQuery();
                if(rs.next()) {
                    String sel3="select seatno=? from seatbook where
cname=?";

                    pst=myconn.prepareStatement(sel3);
                    pst.setInt(1, seatno);
                    pst.setString(2, username);
                    rs=pst.executeQuery();
                    if (rs.next()) {
                        String del = "delete from seatbook where seatno=?";
                        pst = myconn.prepareStatement(del);
                        pst.setInt(1, seatno);
                        int d = pst.executeUpdate();

```



```

        if (d > 0) {
            String del1 = "delete from ticketdetails where
seatno=?";

            pst = myconn.prepareStatement(del1);
            pst.setInt(1, seatno);
            int d1 = pst.executeUpdate();

            if(d1>0) {
                String del2="delete from admin_account where
seatno=?";

                pst = myconn.prepareStatement(del2);
                pst.setInt(1, seatno);
                int d2 = pst.executeUpdate();

                if(d2>0) {
                    String del3="delete from tripdetails where
seatno=?";

                    pst=myconn.prepareStatement(del3);
                    pst.setInt(1, seatno);
                    int d3=pst.executeUpdate();

                    if (d3 > 0) {
                        System.out.println("Your Reservation is
cancelled");
                        //
                        System.out.println(moneyreturn+"
"+userId);

                        String refund2="update customer_acc set
balance=balance+? where userId=?";

                        pst=myconn.prepareStatement(refund2);

```

```

        pst.setFloat(1, moneyreturn);
        pst.setInt(2, userId);
        int rfd=pst.executeUpdate();
        if(rfd>0) {
            System.out.println("Money
refunded successfully into your account");
        }
        break;
    }
}
}
}} else {
    System.out.println("you not reserved any seat");

}
}}

break;
}else {
    System.out.println("Invalid UserID");
}

}}}
```

Seat Cancellation Output:

```
Java - Eclipse IDE
Run Window Help
Markers Properties Data Source Explorer Snippets Console X Debug
<terminated> MainClass (1) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (26-Aug-2022, 11:55:06 pm - 11:55:49 pm)
*****Railway Reservation System*****
Login/Admin or User
1. Admin login
2. New Customer Login
3. Registered Customer
4. Exit
Enter your choice
3
-----Registered User Login-----
Enter User Name
krish
Enter Password
k123
Login Successfull
1. Seat Reservation
2. Seat Cancellation
3. Exit
Enter your choice..
2
You choosed Seat Cancellation
Enter Your User name
krish
Enter Your Password
k123
Enter User ID
3
Enter Your Seat No
3
Your Reservation is cancelled
Money refunded successfullly into your account
Do you want to continue the seat reservation or seat cancellation yes/no
no
Type y if you want to continue or press n to terminate
n
Program is terminated
```

Database Records:

```
MySQL 8.0 Command Line Client

mysql> select * from ticketdetails;
+-----+
| coach | ticketprice | seatno | totalamt |
+-----+
| Second_AC | 1150.00 | 1 | 1150.00 |
| Second_AC | 1150.00 | 2 | 1150.00 |
+-----+
2 rows in set (0.02 sec)

mysql> select * from tripdetails;
+-----+
| tripname | departuretime | arrivaltime | seatno |
+-----+
| Bangalore To Mumbai | 06:00:00 | 02:00:00 | 1 |
| Bangalore To Mumbai | 07:00:00 | 05:00:00 | 2 |
+-----+
2 rows in set (0.04 sec)

mysql> select * from seatbook;;
+-----+
| seatno | cname | phoneno |
+-----+
| 1 | anushree | 9870567854 |
| 2 | ram | 9087656789 |
+-----+
2 rows in set (0.00 sec)
```

Seat Count for Admin:

//admin account details

```
public static void admin_account() throws SQLException {
```

```

myconn=DataBaseConnection.getConnection();
String count_seat="select count(*) from admin_account";
pst=myconn.prepareStatement(count_seat);
rs=pst.executeQuery();
while(rs.next()) {
System.out.println("Total Seats Reserved="+rs.getInt(1));
}

String acc="select * from admin_account";
pst=myconn.prepareStatement(acc);
rs=pst.executeQuery();
System.out.println("Display Reserved Seats");
System.out.println("TotalAmount\tSeatNo");
while(rs.next()) {

    float total=rs.getFloat(1);
    int seatno=rs.getInt(2);

    System.out.println(total+"\t\t"+seatno);

}
}

```

Seat Count Output for Admin:

```

<terminated> MainClass (1) [Java Application] C:\Program Files\Java\jdk-17.0.3.1\bin\javaw.exe (27-Aug-2022, 3
*****Railway Reservation System*****
Login/Admin or User
1. Admin login
2. New Customer Login
3. Registered Customer
4. Exit
Enter your choice
1
For admin login please enter credentials
Enter name
admin
Enter password
admin123
-----Admin Screen-----
Admin Options
1. Show User Ticket Details
2. Show Reserved Seats
Enter your choice
2
Display How many Seats are reserved
Total Seats Reserved=2
Display Reserved Seats
TotalAmount      SeatNo
1150.0           1
1150.0           2
Type y if you want to continue or press n to terminate
n
Program is terminated

```

User Display Code:

```
/**
 * This module is used to display the information about the seat reservation.
 *
 * After the seat reservation it will ask you want to know about your seat
 reservation information
 * If the passenger given yes means it show all the details about the reservation.
 */
// display particular record

    public static void displayReservation() throws SQLException {
        myconn = DataBaseConnection.getConnection();
        String dis = "select * from seatbook where seatno=?";
        pst = myconn.prepareStatement(dis);
        pst.setInt(1, seatno);
        rs = pst.executeQuery();

        while (rs.next()) {
            System.out.println("*****Customer
Details*****");
            System.out.println("-----
-----");

            System.out.println("SeatNO\t\tCustomerName\t\tPhoneNo");
            int seatn = rs.getInt(1);
            String cname = rs.getString(2);
            String phoneno = rs.getString(3);
```

```

        System.out.println("-----
        -----");

        System.out.println(seatn + "\t\t" + cname + "\t\t" +
phoneno);

        System.out.println();
    }

    String sel1 = "select * from ticketdetails where seatno=?";
    pst = myconn.prepareStatement(sel1);
    pst.setInt(1, seatno);
    rs = pst.executeQuery();

    while (rs.next()) {
        System.out.println("*****Ticket
Details*****");
        System.out.println("-----
        -");

        System.out.println("CoachName\t\tPrice\t\tSeatNo\t\tTotalAmt");

        String coach = rs.getString(1);
        float price = rs.getFloat(2);
        int seatn = rs.getInt(3);
        float total = rs.getFloat(4);

        System.out.println("-----
        -----");

        System.out.println(coach + "\t\t" + price + "\t\t" + seatn +
"\t\t" + total);
    }
}

```

User Display Code Output:

```
Ticket Booked
Do you want to see the whole details about your reservation yes/no
yes
*****Customer Details*****
-----
SeatNo      CustomerName      PhoneNo
-----
3           krish           9087890767

*****Ticket Details*****
-----
CoachName      Price      SeatNo      TotalAmt
-----
First_AC       1940.0     3           1940.0
Do you want to continue the seat reservation or seat cancellation yes/no
no
Type y if you want to continue or press n to terminate
n
Program is terminated
```

Admin Display Code:

```
public class AdminClass {

    static Connection myconn=null;

    static ResultSet rs=null;

    static PreparedStatement pst=null;

    static Scanner sc=new Scanner(System.in);

    //display all customer details

    public static void displayCustomer() throws SQLException {

        myconn=DataBaseConnection.getConnection();

        String sel="select * from seatbook";

        pst=myconn.prepareStatement(sel);

        rs=pst.executeQuery();
```

```
System.out.println("Display User Details");
System.out.println("SeatNo\t\tCustomerName\tPhoneNo");
while(rs.next()) {

    int seatn=rs.getInt(1);
    String cname=rs.getString(2);
    String phoneno=rs.getString(3);

    System.out.println(seatn+"\t\t"+cname+"\t\t"+phoneno);
}
String sel1="select * from ticketdetails ";
pst=myconn.prepareStatement(sel1);
rs=pst.executeQuery();
System.out.println("Display Reserved Ticket Details");
System.out.println("CoachName\t\tPrice\t\tSeatNo\t\tTotalAmt");
while(rs.next()) {

    String coach=rs.getString(1);
    float price=rs.getFloat(2);
    int seatn=rs.getInt(3);
    float total=rs.getFloat(4);

    System.out.println(coach+"\t\t"+price+"\t\t"+seatn+"\t\t"+total);
}
}
```


Admin Display Code:

```
y
*****Railway Reservation System*****
Login/Admin or User
1. Admin login
2. New Customer Login
3. Registered Customer
4. Exit
Enter your choice
1
For admin login please enter credentials
Enter name
admin
Enter password
admin123
-----Admin Screen-----
Admin Options
1. Show User Ticket Details
2. Show Reserved Seats
Enter your choice
1
Display all tickets details
Display User Details
SeatNo      CustomerName      PhoneNo
1           anushree           9870567854
2           ram              9087656789
Display Reserved Ticket Details
CoachName      Price      SeatNo      TotalAmt
Second_AC      1150.0     1           1150.0
Second_AC      1150.0     2           1150.0
Type y if you want to continue or press n to terminate
```

Database Record for Admin:

```
no query specified
mysql> select * from admin_account;
+-----+-----+
| totalamt | seatno |
+-----+-----+
| 1150.00 | 1 |
| 1150.00 | 2 |
+-----+-----+
2 rows in set (0.00 sec)
```