

## 2.4 Source Code: -

```
package busbooking;

import javax.swing.*.*;
import java.awt.*.*;
import java.sql.*.*;
import java.util.Vector;
import javax.swing.table.DefaultTableModel;

public class availablebus extends JPanel{
    String usersql="root";
    String passwordsql="aditya123";
    String link="jdbc:mysql://localhost:3306/";
    DefaultTableModel model=new DefaultTableModel(new String[]{"BUS
ID","Source","Destination","Type","Departure Time","Arrival At Destination","Fare"},0);

    public static JTable list=new JTable();
    JButton select=new JButton();
    String data;

    JLabel notselect =new JLabel("PLEASE SELECT A BUS");
    JButton back=new JButton("BACK");
    public static Vector<String> i=new Vector<String>();
    JButton b1=new JButton("NEXT");
    public availablebus(String source,String destination ,String date) {
        try {

            Connection conn=DriverManager.getConnection(link,usersql,passwordsql);
            Statement smt=(Statement)conn.createStatement();

            ResultSet rs=smt.executeQuery("select * from `bus`.`availablebus` where
`Source`='"+source+"`"and `Destination`='"+destination+"`"and `Date`='"+date+"`" ");
            if(rs.next())
            {
                model.addRow(new Object[]{"BUS ID","Source","Destination","Type","Departure
Time","Arrival time","Fare"});
                rs.previous();
                while(rs.next())
                {
                    String id=rs.getString("ID");
                    String sou=rs.getString("Source");
                    String des=rs.getString("Destination");
                    String typ=rs.getString("Type");
                    String dep=rs.getString("Departure Time");
                    String arr=rs.getString("Arrival time");
                    String far=rs.getString("Fare");
```

```

        i.add(id);
        model.addRow(new Object[]{id,sou,des,typ,dep,arr,far});
    }

    list.setModel(model);
    list.setBounds(300, 320, 800,(list.getRowCount()*16));
    ImageIcon icon=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\bus.jpg");
    JLabel sor=new JLabel(icon);
    sor.setBounds(0,0,1400,300);

    this.add(sor);

    this.setBackground(Color.BLACK);
    b1.setBounds(700, 620, 100, 50);
    back.setBounds(500,620, 100, 50);
    Font font=new Font("Ariel",Font.BOLD,15);
    notselect.setFont(font);
    notselect.setForeground(Color.white);
    notselect.setBounds(500, 550, 200, 100);

    this.add(b1);
    this.add(list);
    //this.add(lab);
    this.add(back);
    }
    else
    {
        ImageIcon icon=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\sorry.jpg");
        Font font=new Font("Ariel",Font.BOLD,20);
        JLabel lab=new JLabel("SORRY NO BUSES FOUND FOR THE GIVEN LOCATION AT GIVEN DATE
PLEASE CHOOSE ANOTHER DATE.....");
        JLabel sor=new JLabel(icon);
        lab.setBounds(102, 300, 1400, 50);
        sor.setBounds(550, 400, 150, 150);
        this.add(sor);
        lab.setFont(font);
        this.add(lab);
        this.setBackground(Color.WHITE);
        back.setBackground(Color.WHITE);
        back.setBounds(580,600, 100, 50);
        this.add(back);
    }
    conn.close();
} catch (Exception e) {
    System.out.println(e);
}
}

```

```

    }

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package busbooking;

/**
 *
 * @author adity
 */
public class bookings extends javax.swing.JFrame {

    /**
     * Creates new form bookings
     */
    public bookings() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents
    private void initComponents() {

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGap(0, 400, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGap(0, 300, Short.MAX_VALUE)
        );

        pack();
    } // </editor-fold> //GEN-END: initComponents

```

```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
     */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(bookings.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(bookings.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(bookings.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(bookings.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        }
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new bookings().setVisible(true);
        }
    });
}

// Variables declaration - do not modify//GEN-BEGIN:variables
// End of variables declaration//GEN-END:variables
}

```

```
package busbooking;

import javax.swing.JFrame;

public class busbooking {

    public static void main(String[] args) {
        GUI f=new GUI();
        f.setTitle("BUS BOOKING SYSTEM");
        f.setVisible(true);
        f.setBounds(100,100,500,500);
        f.setExtendedState(JFrame.MAXIMIZED_BOTH);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```

```

package busbooking;

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;

public class cancel extends JFrame{
    JButton con=new JButton("CONFIRM");
    DefaultTableModel model=new DefaultTableModel(new String[]{"ID","First Name","Last
Name","Email","Seats","From","Destination","Journey Date","Fare"},0);
    String usersql="root";
    JButton back=new JButton("Book Another Ticket");
    String passwordsql="aditya123";
    String link="jdbc:mysql://localhost:3306/";
    String data="";

    public cancel(String email) {

        try {
            this.setLayout(null);
            Font font=new Font("Ariel",Font.BOLD,20);
            //String line="ID          "+"First Name          "+"Last Name          "+"Email
"+"Seats          "+"From          "+"Destination          "+"Journey Date";
            // JLabel lineJLabel= new JLabel(line);
            // lineJLabel.setForeground(Color.YELLOW);
            //lineJLabel.setBounds(200, 200, 1200, 50);
            Connection conn=DriverManager.getConnection(link,usersql,passwordsql);
            Statement smt=(Statement)conn.createStatement();
            model.addRow(new Object[]{"BUS ID","First Name","Last
Name","Email","Seats","From","Destination","Journey Date","Fare"});
            JLabel l=new JLabel("SELECT THE BOOKING ");
            l.setBounds(200, 200, 300, 50);

```

```

        l.setFont(font);
        this.add(l);

        ResultSet rs=smt.executeQuery("select * from `bus`.`reserve` where `Email`='"+email+"'");
        if(rs.next())
        {

            rs.previous();

            while(rs.next())
            {
                String id=rs.getString("ID");
                String fn=rs.getString("First Name");
                String ln=rs.getString("Last Name");
                String em=rs.getString("Email");
                String sea=rs.getString("Seats");
                String fr=rs.getString("From");
                String des=rs.getString("Destination");
                String jour=rs.getString("Journey Date");
                String fare=rs.getString("Total Amount");
                model.addRow(new Object[]{id,fn,ln,em,sea,fr,des,jour,fare});
            }
            JLabel lab=new JLabel("YOUR BOOKINGS ");

            lab.setFont(font);
            lab.setBounds(200, 150, 600, 50);
            JTable T=new JTable();
            T.setModel(model);
            T.setBounds(200,250,1000,(T.getRowCount()*16));
            this.add(lab);
            this.add(T);
            JButton cancel =new JButton("CANCEL BOOKING");
            cancel.setBounds(500, 550, 300, 50);
            this.add(cancel);

            cancel.addActionListener(new ActionListener() {

                public void actionPerformed(ActionEvent e) {
                    try {
                        String f=(String)T.getModel().getValueAt(T.getSelectedRow(), 4);

                        smt.executeUpdate("delete from `bus`.`reserve` where `Seats`='"+f+"'");

                        sendEmail("YOUR BOOKING OF BUS FROM
"+(String)T.getModel().getValueAt(T.getSelectedRow(), 5)+"

```

```
TO" +(String)T.getModel().getValueAt(T.getSelectedRow(), 6) + "ON  
" +(String)T.getModel().getValueAt(T.getSelectedRow(), 7) + " HAS BEEN CANCELED", email);
```

```
        } catch (Exception E) {  
            System.out.println(E);  
        }  
  
    }  
});
```

```
//this.getContentPane().add(lineJLabel);
```

```
this.repaint();  
this.revalidate();  
this.setVisible(true);
```

```
}  
else  
{
```

```
ImageIcon icon=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\sorry.jpg");
```

```
    JLabel lab=new JLabel("SORRY YOU DON'T HAVE ANY BOOKING .....");  
    JLabel sor=new JLabel(icon);  
    lab.setFont(font);  
    lab.setBounds(402, 300, 1400, 50);  
    sor.setBounds(550, 400, 150, 150);  
    this.setBackground(Color.WHITE);  
    this.add(sor);  
    this.add(lab);  
}
```

```
        } catch (Exception e) {  
            System.out.println(e);  
        }  
  
    }
```

```
public static void sendEmail(String message,String to)  
{
```



```
String[] t=new String[1];
t[0]=to;
if( Emailsender.sendMail("aditya.it.msit@gmail.com","aditya/.",",message,t))
{
    System.out.println("Email sent");
}
else
{
    System.out.println("Email not sent");
}
}
```

```
package busbooking;
```

```
public class email {
```

```
}
```

```
apackage busbooking;
```

```
import java.util.Properties;
```

```
import javax.mail.Authenticator;
```

```
import javax.mail.Message;
```

```
import javax.mail.Session;
```

```
import javax.mail.Transport;
```

```
import javax.mail.internet.InternetAddress;
```

```
import javax.mail.internet.MimeMessage;
```

```
public class Emailsender {
```

```
    public static boolean sendMail(String from,String Password,String message,String to[])
```

```
    {
```

```
        String host="smtp.gmail.com";
```

```
        Properties props=System.getProperties();
```

```
        props.put("mail.smtp.starttls.enable", "true");
```

```
        props.put("mail.smtp.host", host);
```

```
        props.put("mail.smtp.user", from);
```

```
        props.put("mail.smtp.password", Password);
```

```
        props.put("mail.smtp.port",587);
```

```
        props.put("mail.smtp.auth","true");
```

```
        Session session=Session.getDefaultInstance(props,null);
```

```
        MimeMessage mimeMessage=new MimeMessage(session);
```

```
        try {
```

```
            mimeMessage.setFrom(new InternetAddress(from));
```

```
            InternetAddress[] toAddresses=new InternetAddress[to.length];
```

```
            for(int i=0;i<to.length;i++)
```

```
            {
```

```
                toAddresses[i]=new InternetAddress(to[i]);
```

```
            }
```

```
        }
```

```
        for(int i=0;i<toAddresses.length;i++)
```

```
        {
```

```
        mimeMessage.addRecipient(Message.RecipientType.TO,toAddresses[i]);

    }

    mimeMessage.setSubject("Your Bus Booking Ticket");
    mimeMessage.setText(message);
    Transport transPort=session.getTransport("smtp");
    transPort.connect(host,from>Password);
    transPort.sendMessage(mimeMessage, mimeMessage.getAllRecipients());
    transPort.close();
    return true;

} catch (Exception e) {
    System.out.println(e);
}

return false;
}

}
```

```

package busbooking;

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.sql.*;

public class enterlocation extends JPanel implements FocusListener
{
    JComboBox d;      //ComboBox
    JComboBox m;
    JComboBox y;
    String day[]=new String[32];
    JButton view=new JButton("VIEW YOUR OTHER BOOKINGS");
    String
month[]={"MONTH","january","february","march","april","may","june","july","august","september",
"october","november","december"};
    String year[]={"YEAR","2016","2017","2018","2019","2020"};
    Font f;
    JTextField from_tf =new JTextField();
    JTextField to_tf =new JTextField();
    JButton go=new JButton();
    public enterlocation()
    {
        ImagemIcon icon=new ImagemIcon("C:\\Users\\adity\\Desktop\\images\\icon.png");
        ImagemIcon icon2=new ImagemIcon("C:\\Users\\adity\\Desktop\\images\\cover.jpg");
        ImagemIcon icon3=new ImagemIcon("C:\\Users\\adity\\Desktop\\images\\go.png");
        for(int i=1;i<=31;i++)
            day[i]=Integer.toString(i);
        day[0]="DAY";

        d=new JComboBox(day);      //ComboBox
        m=new JComboBox(month);
        y=new JComboBox(year);
        f=new Font("Arial",Font.BOLD,20);
        JLabel from=new JLabel("From :-");
        JLabel to=new JLabel("To :-");
        from.setBounds(100, 350, 100, 50);
        from.setFont(f);
        to.setBounds(700, 350, 100, 50);
        to.setFont(f);

        JLabel frompic=new JLabel(icon);

```

```

JLabel topic=new JLabel(icon);
frompic.setBounds(260, 372, icon.getIconWidth(), icon.getIconHeight());
topic.setBounds(850, 372, icon.getIconWidth(), icon.getIconHeight());

from_tf.setBounds(280, 360, 200, 30);
// from_tf.setFont(f);

to_tf.setBounds(870, 360, 200, 30);
// to_tf.setFont(f);

JLabel pic=new JLabel("WELCOME");
pic.setIcon(icon2);
pic.setBounds(0, 0,1400,300);

JLabel Journey=new JLabel("Journey Date :-");
Journey.setFont(f);
Journey.setBounds(380, 450, 200, 50);
from_tf.setText("Enter City");
to_tf.setText("Enter City");

d.setBounds(580,460,90,30);
m.setBounds(680,460,90,30);
y.setBounds(780,460,90,30);
// go.setIcon(icon3);
go.setBounds(580, 600, 100, 50);
go.setText("NEXT");
this.add(topic);
this.add(Journey);
this.add(frompic);
this.add(to);
this.add(pic);
this.add(from);
this.add(to_tf);
this.add(from_tf);
this.add(d);
this.add(m);
this.add(y);
this.add(go);

from_tf.addFocusListener(this);
to_tf.addFocusListener(this);
view.setBounds(380, 670, 500, 30);
this.add(view);
}
public void focusLost(FocusEvent e)
{
    if(e.getSource()==from_tf)
    {

```

```

        if(from_tf.getText().isEmpty())
        {
            from_tf.setText("Enter City");
        }
    }
    if(e.getSource()==to_tf)
    {
        if(to_tf.getText().isEmpty())
        {
            to_tf.setText("Enter City");
        }
    }
}

```

```

public void focusGained(FocusEvent e)
{
    if(e.getSource()==from_tf)
    {
        if(from_tf.getText().equals("Enter City"))
        {
            from_tf.setText("");
        }
    }
    if(e.getSource()==to_tf)
    {
        if(to_tf.getText().equals("Enter City"))
        {
            to_tf.setText("");
        }
    }
}

```

```

}

```

```

package busbooking;

import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import static java.lang.Thread.sleep;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.*;
import jdk.nashorn.internal.objects.Global;
import busbooking.seats;
import java.awt.event.FocusEvent;
import java.awt.event.FocusListener;
import javax.swing.text.View;

public class GUI extends JFrame{

    JComboBox d;      //ComboBox
    JComboBox m;
    JComboBox y;
    String day[]=new String[31];
    String
month[]={ "january", "february", "march", "april", "may", "june", "july", "august", "september", "october",
"november", "december"};
    String year[]={ "2016", "2017", "2018", "2019", "2020"};
    JPanel content =new JPanel();
    JPanel signin =new JPanel();
    JPanel register =new JPanel();
    JPanel main =new JPanel();
    JButton sign=new JButton("signin");
    JButton reg=new JButton("register");
    JLabel l=new JLabel();
    thanks tk;
    public static String seat="";
    Container c;
    Font f;
    public mainpanel mp=new mainpanel();
    public enterlocation el=new enterlocation();
    public availablebus ab;
    public seats seats=new seats();

    static String source="";
    static String destination="";

```

```

static String date="";
static String IDno;
JScrollPane sp=new JScrollPane();
String set ="";
public GUI()
{
    c=this.getContentPane();

    mp.setLayout(null);
    c.add(mp);

    mp.login.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            if(mp.user1.getText().isEmpty())
            {
                mp.msg.setText("Plzz Enter Username");
                mp.bar.repaint();
                mp.bar.revalidate();
            }
            else if(mp.pass1.getText().isEmpty())
            {
                mp.msg.setText("Plzz Enter Password");
                mp.bar.repaint();
                mp.bar.revalidate();
            }

            else
            {

                try {
                    Connection
conn=DriverManager.getConnection(mp.link,mp.usersql,mp.passwordsql);
                    Statement smt=(Statement)conn.createStatement();
                    ResultSet rs=smt.executeQuery("select * from bus.user");

                    while(rs.next() || rs.isAfterLast())
                    {
                        if(rs.isAfterLast())
                        {

if(rs.getString("Email").equals(mp.user1.getText()) || rs.getString("Password").equals(mp.pass1.getTe
xt()))
                        {
                            c.remove(mp);

```



```

        c.revalidate();
        c.repaint();

        /* wait w=new wait();
        w.setLayout(null);
        c.add(w);
        c.revalidate();
        c.repaint();
        c.remove(w);*/
        mp.bar.remove(mp.msg);
        mp.bar.repaint();
        mp.bar.revalidate();

        el.setLayout(null);

        c.add(el);
        c.revalidate();
        c.repaint();
    }
    else
    {
        mp.msg.setText("Email id Dosen't exist");
        System.out.println("Hello");
        mp.bar.repaint();
        mp.bar.revalidate();
    }
}
else
if(rs.getString("Email").equals(mp.user1.getText()) || rs.getString("Password").equals(mp.pass1.getTe
xt()))
{
    c.remove(mp);
    c.revalidate();
    c.repaint();

    /* wait w=new wait();
    w.setLayout(null);
    c.add(w);
    c.revalidate();
    c.repaint();
    c.remove(w);*/
    mp.bar.remove(mp.msg);
    mp.bar.repaint();
    mp.bar.revalidate();
    el.setLayout(null);

    c.add(el);

```

```

        c.revalidate();
        c.repaint();
    }

}

conn.close();
}
catch (Exception E) {
    System.out.println(E);
}

}

}
});

```

```

mp.Siup.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e) {
        if(!mp.fn.getText().isEmpty() && !mp.ln.getText().isEmpty() &&
            !mp.email.getText().isEmpty() && !mp.reenter.getText().isEmpty()
            && !mp.pass.getPassword().toString().isEmpty()
            && !mp.d.getSelectedItem().toString().equals("DAY")
            && !mp.m.getSelectedItem().toString().equals("MONTH") &&
            !mp.y.getSelectedItem().toString().equals("YEAR"))
        {
            System.out.println("hello");
            if(mp.email.getText().equals(mp.reenter.getText()))
            {

```

```

String fn="",ln="",email="",pass="",gender="";
fn=mp.fn.getText();
ln=mp.ln.getText();
email=mp.email.getText();
pass=mp.pass.getText();
if(mp.male.isSelected())
    gender="male";
else
    gender="Female";

```

```

        String
d=mp.d.getSelectedItem().toString()+"/"+mp.m.getSelectedItem().toString()+"/"+mp.y.getSelectedItem().toString();
        try {
            Connection
conn=DriverManager.getConnection(mp.link,mp.usersql,mp.passwordsql);
            Statement smt=(Statement)conn.createStatement();
            smt.executeUpdate("insert into `bus`.`user` (`First Name`,`Last
Name`,`Email`,`password`,`Date of
Birth`,`Gender`)values(\""+fn+"\", \""+ln+"\", \""+email+"\", \""+pass+"\", \""+d+"\", \""+gender+"\"))
;

            mp.below.add(mp.msg2);
            mp.msg2.setText("USER REGISTERED");
            c.revalidate();
            c.repaint();
            mp.below.revalidate();
            mp.below.repaint();
            conn.close();
        }
        catch(Exception E)
        {
            System.out.println(E);
        }

    }
    else{

        mp.msg2.setText("Email does not match ");
        mp.below.add(mp.msg2);
        c.revalidate();
        c.repaint();
        mp.below.revalidate();
        mp.below.repaint();

    }
}

        else if(mp.fn.getText().equals("First Name") || mp.ln.getText().equals("Last
Name") || mp.email.getText().equals("Email") || mp.reenter.getText().equals("Reenter
Email") || mp.pass.getText().equals("Password"))
        {
            // ADD FOR EMPTY FIELDS

            mp.msg2.setText("Please fill all details");
            mp.below.add(mp.msg2);

            mp.below.revalidate();
            mp.below.repaint();

```

```

        c.revalidate();
        c.repaint();
    }

    }

});

mp.email.addFocusListener(new FocusListener(){

    public void focusGained(FocusEvent f)
    {
        mp.below.remove(mp.msg2);

        mp.msg2.setText("");
        c.revalidate();
        c.repaint();
        mp.below.revalidate();
        mp.below.repaint();

    }

    public void focusLost(FocusEvent f)
    {
        try {
            Connection connect=DriverManager.getConnection(mp.link,mp.usersql,mp.passwordsql);
            Statement smt=connect.createStatement();
            ResultSet rs=smt.executeQuery("select * from `bus`.`user`");
            while(rs.next())
            {
                if(mp.email.getText().equals(rs.getString("Email")))
                {

                    mp.below.add(mp.msg2);
                    mp.msg2.setText("Username not available");

                    mp.below.revalidate();
                    mp.below.repaint();
                    c.revalidate();
                    c.repaint();
                }
            }

        }
        catch(Exception e)
        {

```

```

    }
}
});

```

```

el.view.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        view v=new view(mp.user1.getText());
        v.setTitle("BUS BOOKING SYSTEM");
        v.setVisible(true);
        v.setBounds(100,100,500,500);
        v.setExtendedState(JFrame.MAXIMIZED_BOTH);
        v.cancel.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {
                cancel c =new cancel(mp.user1.getText());
                c.setVisible(true);
                //c.setBounds(100,100,500,500);
                c.setExtendedState(JFrame.MAXIMIZED_BOTH);
                v.setVisible(false);
                /*System.out.println("hello");*/

            }
        });
    }
});

```

```

el.go.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {

        source=el.from_tf.getText() ;
        destination=el.to_tf.getText() ;

```

```

date=el.d.getSelectedItemAt().toString()+"/"+el.m.getSelectedIndex()+"/"+el.y.getSelectedItemAt().toString();

```

```

        ab=new availablebus(source,destination,date);
        c.remove(el);
        c.add(ab);
        ab.setLayout(null);
        c.revalidate();
        c.repaint();
        ab.back.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {

                el.setLayout(null);
                c.remove(ab);

```

```

        c.add(e1);
        c.revalidate();
        c.repaint();
    }
});

ab.b1.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        try {

            Connection
conn=DriverManager.getConnection(mp.link,mp.usersql,mp.passwordsql);
            Statement smt=(Statement)conn.createStatement();
            if(ab.list.getSelectedRow()-1)
            {
                IDno=ab.i.get(ab.list.getSelectedRow());
                String st =availablebus.i.get(availablebus.list.getSelectedRow());
                //FOR DISPLAYING BOOKED SEATS
                ResultSet rs=smt.executeQuery("select * from `bus`.`reserve` where
`ID`=\""+st+"\"");

                while(rs.next())
                {
                    set=set+rs.getString("Seats");
                }

                char [] s=set.toCharArray();

                for(int i=0;i<s.length;i=i+3)
                {
                    String sa="" +s[i]+s[i+1];

                    for(int j=0;j<seats.buttonlist.size();j++)
                    {
                        if(seats.buttonlist.get(j).getText().equals(sa))
                        {
                            seats.buttonlist.get(j).setEnabled(false);
                        }
                    }

                }

            }
            c.remove(ab);
            c.add(seats);
            seats.setLayout(null);
            c.revalidate();

```

```

        c.repaint();

    }
    else
    {
        ab.add(ab.notselect);
        c.revalidate();
        c.repaint();
    }

    }
    catch (Exception E)
    {
        System.out.println(E);
    }
    }
    });

    }
    });
seats.back.addActionListener(new ActionListener() {

    public void actionPerformed(ActionEvent e) {
        c.remove(seats);
        c.add(ab);
        ab.setLayout(null);
        c.revalidate();
        c.repaint();
    }
    });

seats.next.addActionListener(new ActionListener() {

    public void actionPerformed(ActionEvent e) {
        if(seats.al.isEmpty())
            seats.msg.setText("Please Select minimum one seat");
        else
        {
            try {
                Connection conn=DriverManager.getConnection(mp.link,mp.usersql,mp.passwordsql);
                Statement smt=(Statement)conn.createStatement();
                ResultSet rs;
                String id=IDno;
                String firn="";
                String lasn="";
                String email="";
            }
            catch (Exception E) {
                seats.msg.setText(E.getMessage());
            }
        }
    }
});

```

```
String fare="";
String from="";
String destination2="";
String jdate="";
```

```
rs=smt.executeQuery("select * from `bus`.`user` where  
`Email`='"+mp.user1.getText()+"'");
```

```
//if(!rs.next())  
// rs=smt.executeQuery("select * from `bus`.`user` where  
`Email`='"+mp.email.getText()+"'");
```

```
if(rs.next())  
{  
    firn=rs.getString("First name");  
    lasn=rs.getString("Last name");  
    email=rs.getString("Email");  
}
```

```
rs=smt.executeQuery("select * from `bus`.`availablebus` where `ID`='"+id+"'");
```

```
if(rs.next())  
{  
    Integer a=Integer.parseInt(rs.getString("Fare"))*(seats.al.size());  
    fare=a.toString();  
}
```

```
for(int i=0;i<seats.al.size();i++)  
{  
    seat=seat+seats.al.get(i).getText()+",";
```

```
}  
from=el.from_tf.getText();  
destination2=el.to_tf.getText();
```

```
jdate=el.d.getSelectedItem().toString()+"/"+el.m.getSelectedIndex()+"/"+el.y.getSelectedItem().toString();
```

```
smt.executeUpdate("insert into `bus`.`reserve`(`ID`,`First Name`,`Last  
Name`,`Email`,`Seats`,`From`,`Destination`,`Journey Date`,`Total Amount`) values  
('"+IDno+"','"+firn+"','"+lasn+"','"+email+"','"+seat+"','"+from+"','"+destination2+"','"+jdate+"','"+fare  
+"')");
```

```
conn.close();  
c.remove(seats);  
tk=new thanks();
```



```

        c.add(tk);
        tk.setLayout(null);
        c.revalidate();
        c.repaint();
        tk.back.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {
                el=new enterlocation();
                c.remove(tk);
                el.setLayout(null);
                c.add(el);
                c.revalidate();
                c.repaint();

                }

            });

    }
    catch (Exception E) {
        System.out.println(E);
    }

}

}
});

this.pack();

}
}

```

```

package busbooking;

```

```

import java.awt.*;
import java.awt.event.FocusListener;

```

```

import javax.swing.*.*;
import javax.swing.border.Border;
import java.awt.event.*;
import java.sql.*;

public class mainpanel extends JPanel implements FocusListener{

    String day[]=new String[32];
    String
    month[]={ "MONTH","january","february","march","april","may","june","july","august","september",
    "october","november","december"};
    String year[]=new String[51];
    JComboBox d;          //ComboBox
    JComboBox m;
    JComboBox y;
    JRadioButton male =new JRadioButton("Male");    //RadioButton
    JRadioButton female =new JRadioButton("Female") ;
    ButtonGroup b1=new ButtonGroup();
    JPanel bar=new JPanel();
    JPanel below =new JPanel();
    JTextField fn=new JTextField();
    JTextField ln=new JTextField();
    JTextField email=new JTextField();
    JTextField reenter=new JTextField();
    JPasswordField pass=new JPasswordField();
    JLabel msg=new JLabel();
    String usersql="root";
    String passwordsql="aditya123";
    String link="jdbc:mysql://localhost:3306/";
    JTextField user1 =new JTextField();
    JPasswordField pass1=new JPasswordField();
    JButton login=new JButton("LOG IN");
    JButton Siup =new JButton("Signup");
    JLabel msg2=new JLabel("USER REGISTERED PLEASE SIGNIN WITH YOUR ACCOUNT");
    public mainpanel()
    {

    {          //BAR PANEL
    //FOR SIGN IN
        Border border = BorderFactory.createLineBorder(Color.BLUE);

        bar.setLayout(null);
        bar.setBounds(0, 0, 1400, 150);
        bar.setBackground(Color.BLACK);
        this.add(bar);
        Font f=new Font("Bradley Hand ITC",Font.ITALIC,50);
        Font f1=new Font("Bradley Hand ITC",Font.BOLD,25);
    }
    }
}

```

```
Font f2=new Font("Bradley Hand ITC",Font.BOLD,15);
Font f3=new Font("Ariel",Font.BOLD,15);
```

```
JLabel mybus=new JLabel("PRIME BUS");
JLabel username=new JLabel("Email or Phone");
JLabel password=new JLabel("Password");
```

```
mybus.setBounds(50, 50, 500, 60);
username.setBounds(700, 30, 200, 50);
msg.setBounds(700, 110, 200, 30);
password.setBounds(950, 30, 200, 50);
```

```
login.setBounds(1160, 50, 150, 50);
```

```
mybus.setForeground(Color.WHITE);
username.setForeground(Color.WHITE);
password.setForeground(Color.WHITE);
login.setForeground(Color.WHITE);
```

```
login.setBackground(Color.BLACK);
```

```
login.setBorder(border);
msg.setForeground(Color.WHITE);
msg.setFont(f3);
mybus.setFont(f);
username.setFont(f3);
password.setFont(f3);
```

```
login .setFont(f3);
```

```
user1.setBounds(700, 70, 200, 30);
pass1.setBounds(950, 70, 200, 30);
```

```
Cursor cur=new Cursor(Cursor.HAND_CURSOR);
login.setCursor(cur);
```

```
bar.add(mybus);
bar.add(username);
bar.add(password);
```

```
bar.add(user1);
bar.add(pass1);
bar.add(login);
bar.add(msg);
```

```
}
```

```
//BELOW JPANEL  
//FOR SIGN UP
```

```
for(int i=1;i<51;i++)  
year[i]=Integer.toString(i+1970);  
for(int i=1;i<32;i++)  
day[i]=Integer.toString(i);
```

```
year[0]="YEAR";  
day[0]="DAY";
```

```
d=new JComboBox(day);    //ComboBox  
m=new JComboBox(month);  
y=new JComboBox(year);
```

```
ImageIcon icon =new ImageIcon("C:\\Users\\adity\\Desktop\\images\\thamks.png");
```

```
Font f=new Font("Britannic",Font.ITALIC,70);  
Font f1=new Font("Britannic",Font.BOLD,45);  
Font f2=new Font("Britannic",Font.BOLD,18);  
below.setLayout(null);  
below.setBounds(0, 150, 1400, 700);  
below.setBackground(Color.ORANGE);  
this.add(below);  
JLabel Signup =new JLabel("Signup");  
JLabel free =new JLabel("Its Free And Always Will Be");  
JLabel Birthday =new JLabel("Birthday");
```

```
JLabel pic=new JLabel(icon);
```

```
fn.addFocusListener(this);  
ln.addFocusListener(this);  
email.addFocusListener(this);  
reenter.addFocusListener(this);  
pass.addFocusListener(this);
```

```
Signup.setBounds(700, 10, 200, 100);  
free.setBounds(705, 70, 300, 100);  
fn.setBounds(705, 170, 200, 30);  
ln.setBounds(925, 170, 200, 30);
```

```
email.setBounds(705, 220, 420, 30);
reenter.setBounds(705, 270, 420, 30);
pass.setBounds(705, 320, 420, 30);
Birthday.setBounds(705, 370, 420, 30);
d.setBounds(705,420,90,30);
m.setBounds(805,420,90,30);
y.setBounds(905,420,90,30);
pic.setBounds(100, 100, 400, 400);
male.setBounds(705,470,100,25); //RadioButton for gender
female.setBounds(905,470,100,25);
```

```
male.setBackground(Color.orange);
female.setBackground(Color.orange);
b1.add(male);
b1.add(female);
```

```
Siup.setBounds(790,500,100,30);
msg2.setBounds(900,500,500,30);
```

```
Signup.setFont(f1);
free.setFont(f2);
Birthday.setFont(f2);
male.setFont(f2);
female.setFont(f2);
```

```
fn.setText("First Name");
ln.setText("Last Name");
email.setText("Email");
pass.setEchoChar((char)0);
pass.setText("Password");
reenter.setText("Reenter Email");
```

```
below.add(Signup);
below.add(free);
below.add(Birthday);
below.add(Siup);
below.add(fn);
below.add(ln);
below.add(email);
below.add(reenter);
below.add(pass);
below.add(d);
below.add(y);
below.add(m);
below.add(male);
```

```
below.add(female);
below.add(pic);
below.repaint();
below.revalidate();
```

```
}
```

```
public void focusLost(FocusEvent e)
{
    if(e.getSource()==fn)
    {
        if(fn.getText().isEmpty())
        {
            fn.setText("First Name");
            below.repaint();
            below.revalidate();
        }
    }
    if(e.getSource()==ln)
    {
        if(ln.getText().isEmpty())
        {
            ln.setText("Last Name");
            below.repaint();
            below.revalidate();
        }
    }
    if(e.getSource()==email)
    {
        if(email.getText().isEmpty())
        {
            email.setText("Email");
            below.repaint();
            below.revalidate();
        }
    }
    if(e.getSource()==pass)
    {
```

```

        if(pass.getText().isEmpty())
        {
            pass.setEchoChar((char)0);
            pass.setText("Password");
            below.repaint();
            below.revalidate();
        }
    }
    if(e.getSource()==reenter)
    {
        if(reenter.getText().isEmpty())
        {
            reenter.setText("Reenter Email");
            below.repaint();
            below.revalidate();
        }
    }
}

public void focusGained(FocusEvent e)
{
    if(e.getSource()==fn)
    {
        if(fn.getText().equals("First Name"))
        {
            fn.setText("");
            below.repaint();
            below.revalidate();
        }
    }
    if(e.getSource()==ln)
    {
        if(ln.getText().equals("Last Name"))
        {
            ln.setText("");
            below.repaint();
            below.revalidate();
        }
    }
    if(e.getSource()==email)
    {
        if(email.getText().equals("Email"))
        {
            email.setText("");
            below.repaint();
            below.revalidate();
        }
    }
}

```

```
}
if(e.getSource()==pass)
{
    pass.setEchoChar('*');
    if(pass.getText().equals("Password"))
    {
        pass.setText("");
        below.repaint();
        below.revalidate();
    }
}
if(e.getSource()==reenter)
{
    if(reenter.getText().equals("Reenter Email"))
    {
        reenter.setText("");
        below.repaint();
        below.revalidate();
    }
}
}
}
```



```

package busbooking;

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
import javax.swing.*;
import javax.swing.border.Border;
import busbooking.avaliablebus;

public class seats extends JPanel implements ActionListener{
    public static String seet="";
    JPanel bus=new JPanel();
    ImageIcon icon=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\seat.png");
    ImageIcon icon1=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\steering.png");
    ImageIcon icon2=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\seatgrey.png");
    ImageIcon icon3=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\busimage.png");
    Font font=new Font("Britannic", Font.ITALIC, 30);
    JLabel discription=new JLabel(icon3);
    JButton back=new JButton("BACK");
    JLabel plz=new JLabel("PLEASE SELECT YOUR SEATS");
    JButton a1=new JButton("a1",icon);
    JButton a2=new JButton("a2",icon);

    JButton b1=new JButton("b1",icon);
    JButton b2=new JButton("b2",icon);

    JButton c1=new JButton("c1",icon);
    JButton c2=new JButton("c2",icon);

    JButton d1=new JButton("d1",icon);
    JButton d2=new JButton("d2",icon);

    JButton e1=new JButton("e1",icon);
    JButton e2=new JButton("e2",icon);

    JButton f1=new JButton("f1",icon);
    JButton f2=new JButton("f2",icon);

    JButton g1=new JButton("g1",icon);
    JButton g2=new JButton("g2",icon);

    JButton h1=new JButton("h1",icon);
    JButton h2=new JButton("h2",icon);

    JButton i1=new JButton("i1",icon);

```

```

JButton i2=new JButton("i2",icon);

JButton j1=new JButton("j1",icon);
JButton j2=new JButton("j2",icon);

JButton k1=new JButton("k1",icon);
JButton k2=new JButton("k2",icon);

JButton l1=new JButton("l1",icon);
JButton l2=new JButton("l2",icon);

JButton next =new JButton("NEXT");

JLabel steer=new JLabel(icon1);

ArrayList<JButton> buttonlist=new ArrayList<JButton>();

JLabel msg=new JLabel();
ArrayList<JButton> al=new ArrayList<JButton>();

public seats() {
    Border border = BorderFactory.createLineBorder(Color.WHITE);

    {

    }

    plz.setFont(font);
    plz.setBounds(100, 50, 500, 100);
    bus.setBounds(100, 200, 500, 200);
    bus.setBackground(Color.white);
    bus.setLayout(null);
    steer.setBounds(10,10, 30, 30);
    a1.setBounds(150, 10, 30, 30);
    a1.setBorder(border);
    a2.setBounds(150, 50, 30, 30);
    a2.setBorder(border);

    b1.setBounds(210, 10, 30, 30);
    b1.setBorder(border);
    b2.setBounds(210, 50, 30, 30);
    b2.setBorder(border);

    c1.setBounds(270, 10, 30, 30);

```

```
c1.setBorder(border);  
c2.setBounds(270, 50, 30, 30);  
c2.setBorder(border);
```

```
d1.setBounds(330, 10, 30, 30);  
d1.setBorder(border);  
d2.setBounds(330, 50, 30, 30);  
d2.setBorder(border);
```

```
e1.setBounds(390, 10, 30, 30);  
e1.setBorder(border);  
e2.setBounds(390, 50, 30, 30);  
e2.setBorder(border);
```

```
f1.setBounds(450, 10, 30, 30);  
f1.setBorder(border);  
f2.setBounds(450, 50, 30, 30);  
f2.setBorder(border);
```

```
g1.setBounds(150, 120, 30, 30);  
g1.setBorder(border);  
g2.setBounds(150, 160, 30, 30);  
g2.setBorder(border);
```

```
h1.setBounds(210, 120, 30, 30);  
h1.setBorder(border);  
h2.setBounds(210, 160, 30, 30);  
h2.setBorder(border);
```

```
i1.setBounds(270, 120, 30, 30);  
i1.setBorder(border);  
i2.setBounds(270, 160, 30, 30);  
i2.setBorder(border);
```

```
j1.setBounds(330, 120, 30, 30);  
j1.setBorder(border);  
j2.setBounds(330, 160, 30, 30);  
j2.setBorder(border);
```

```
k1.setBounds(390, 120, 30, 30);  
k1.setBorder(border);  
k2.setBounds(390, 160, 30, 30);  
k2.setBorder(border);
```

```
l1.setBounds(450, 120, 30, 30);  
l1.setBorder(border);
```

```
l2.setBounds(450, 160, 30, 30);  
l2.setBorder(border);
```

```
discription.setBounds(750, 200, 308, 97);  
next.setBounds(700, 500, 100, 50);  
back.setBounds(400,500, 100, 50);  
msg.setBounds(500,600,300,50);  
this.add(back);  
this.add(bus);  
this.add(discription);  
this.add(next);  
this.add(plz);  
this.add(msg);  
    //a1.setText("a1");  
    // System.out.println(a1.getText());
```

```
bus.add(steer);  
bus.add(a1);  
bus.add(a2);  
bus.add(b1);  
bus.add(b2);  
bus.add(c1);  
bus.add(c2);  
bus.add(d1);  
bus.add(d2);  
bus.add(e1);  
bus.add(e2);  
bus.add(f1);  
bus.add(f2);  
bus.add(g1);  
bus.add(g2);  
bus.add(h1);  
bus.add(h2);  
bus.add(i1);  
bus.add(i2);  
bus.add(j1);  
bus.add(j2);  
bus.add(k1);  
bus.add(k2);  
bus.add(l1);  
bus.add(l2);  
a1.addActionListener(this);  
a2.addActionListener(this);  
b1.addActionListener(this);  
b2.addActionListener(this);  
c1.addActionListener(this);  
c2.addActionListener(this);
```

```
d1.addActionListener(this);
d2.addActionListener(this);
e1.addActionListener(this);
e2.addActionListener(this);
f1.addActionListener(this);
f2.addActionListener(this);
g1.addActionListener(this);
g2.addActionListener(this);
h1.addActionListener(this);
h2.addActionListener(this);
i1.addActionListener(this);
i2.addActionListener(this);
j1.addActionListener(this);
j2.addActionListener(this);
k1.addActionListener(this);
k2.addActionListener(this);
l1.addActionListener(this);
l2.addActionListener(this);
```

```
buttonlist.add(a1);
buttonlist.add(a2);
buttonlist.add(b1);
buttonlist.add(b2);
buttonlist.add(c1);
buttonlist.add(c2);
buttonlist.add(d1);
buttonlist.add(d2);
buttonlist.add(e1);
buttonlist.add(e2);
buttonlist.add(f1);
buttonlist.add(f2);
buttonlist.add(g1);
buttonlist.add(g2);
buttonlist.add(h1);
buttonlist.add(h2);
buttonlist.add(i1);
buttonlist.add(i2);
buttonlist.add(j1);
buttonlist.add(j2);
buttonlist.add(k1);
buttonlist.add(k2);
buttonlist.add(l1);
buttonlist.add(l2);
```

```

    }

    public void actionPerformed(ActionEvent e)
    {
        JButton b=(JButton)e.getSource();
        if(b.getIcon()==icon)
        {
            b.setIcon(icon2);

            al.add(b);

        }
        else
        {
            b.setIcon(icon);
            for(int i=0;i<al.size();i++)
            {
                if(al.get(i).getText().equals(b.getText()))
                {
                    al.remove(i);
                }
            }
        }
    }
}

```

```

package busbooking;

import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.*;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import busbooking.GUI;
import javax.swing.table.DefaultTableModel;

public class thanks extends JPanel{
    JButton con=new JButton("CONFIRM");
    DefaultTableModel model=new DefaultTableModel(new String[]{"ID","First Name","Last
Name","Email","Seats","From","Destination","Journey Date","Fare"},0);

    String usersql="root";
    JButton back=new JButton("Book Another Ticket");
    String passwordsql="aditya123";
    String link="jdbc:mysql://localhost:3306/";
    String data="";
    JButton ticket=new JButton("Send Ticket To email");
    String Message;
    public thanks() {
        try {

            Font font=new Font("Arial",Font.BOLD,20);
            JLabel msg= new JLabel("Thanks For Traveling With PRIME BUS");
            JLabel msg2= new JLabel("PLEASE CHECH YOUR DETAILS");
            JLabel msg3= new JLabel("PAYMENT WILL BE DONE AT THE BOARDING POINT");
            JLabel msg4= new JLabel("FOR ANY QUERIES OR CANCELATION PLEASE CONTACT CUSTOMER
SUPPORT :-1800633121");
            model.addRow(new Object[]{"BUS ID","First Name","Last
Name","Email","Seats","From","Destination","Journey Date","Fare"});
            msg.setBounds(200, 100, 700, 100);
            msg.setForeground(Color.YELLOW);
            msg2.setForeground(Color.YELLOW);
            msg3.setForeground(Color.YELLOW);
            msg4.setForeground(Color.YELLOW);
            msg4.setBounds(200, 150, 700, 100);

            this.setBackground(Color.blue);
            // String line="ID          "+"First Name          "+"Last Name          "+"Email
"+"Seats          "+"From          "+"Destination          "+"Journey Date";
            // JLabel lineJLabel= new JLabel(line);
            // lineJLabel.setForeground(Color.YELLOW);

```

```

// lineJLabel.setBounds(200, 200, 1200, 50);
Connection conn=DriverManager.getConnection(link,usersql,passwordsql);
Statement smt=(Statement)conn.createStatement();
ResultSet rs=smt.executeQuery("select * from `bus`.`reserve` where
`Seats`='"+GUI.seat+"'");
Message="";
while(rs.next())
{
    String id=rs.getString("ID");
    String fn=rs.getString("First Name");
    String ln=rs.getString("Last Name");
    String em=rs.getString("Email");
    String sea=rs.getString("Seats");
    String fr=rs.getString("From");
    String des=rs.getString("Destination");
    String jour=rs.getString("Journey Date");
    String fare=rs.getString("Total Amount");
    model.addRow(new Object[]{id,fn,ln,em,sea,fr,des,jour,fare});
    Message="ID =" +id+"\n";
    Message=Message+"First Name =" +fn+"\n";
    Message=Message+"Last Name =" +ln+"\n";
    Message=Message+"Email =" +em+"\n";
    Message=Message+"Seats =" +sea+"\n";
    Message=Message+"FROM =" +fr+"\n";
    Message=Message+"Destination=" +des+"\n";
    Message=Message+"Journey Date=" +jour+"\n";
    Message=Message+"Fare =" +fare+"\n";

}
JTable T=new JTable();
T.setModel(model);
T.setBounds(200,250,1000,(T.getRowCount()*16));
msg3.setBounds(200,350,1200,100);
msg4.setBounds(200, 450, 700, 100);
back.setBounds(500, 550, 300, 50);
ticket.setBounds(500, 650, 300, 50);
this.add(ticket);
this.add(msg);
this.add(msg2);
this.add(msg3);
this.add(msg4);
//this.add(lineJLabel);
this.add(back);
this.add(T);

ticket.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

```



```

        try {
            if(rs.isAfterLast())
                rs.previous();

            sendEmail(Message,rs.getString("Email"));
        } catch (Exception E) {
            System.out.println(E);
        }

    }

});

    } catch (Exception e) {
    }
}

public static void sendEmail(String message,String to)
{
    String[] t=new String[1];
    t[0]=to;
    if( Emailsender.sendMail("aditya.it.msit@gmail.com","aditya/./",message,t))
    {
        System.out.println("Email sent");
    }
    else
    {
        System.out.println("Email not sent");
    }
}

}

```

```

package busbooking;

import static busbooking.thanks.sendEmail;
import java.awt.Color;
import java.awt.Container;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.Vector;
import javax.swing.DefaultListModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JList;
import javax.swing.JPanel;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;

public class view extends JFrame{

    JButton con=new JButton("CONFIRM");
    JButton cancel =new JButton("CANCEL YOUR BOOKING");
    DefaultTableModel model=new DefaultTableModel(new String[]{"ID","First Name","Last Name","Email","Seats","From","Destination","Journey Date","Fare"},0);
    String usersql="root";
    JButton back=new JButton("Book Another Ticket");
    String passwordsql="aditya123";
    String link="jdbc:mysql://localhost:3306/";
    String data="";
    Container contain;

    public view(String email) {

        try {
            this.getContentPane().setLayout(null);
            Font font=new Font("Ariel",Font.BOLD,20);
            //String line="ID          "+"First Name          "+"Last Name          "+"Email
"+"Seats          "+"From          "+"Destination          "+"Journey Date";

```

```

// JLabel lineJLabel= new JLabel(line);
// lineJLabel.setForeground(Color.YELLOW);
//lineJLabel.setBounds(200, 200, 1200, 50);
Connection conn=DriverManager.getConnection(link,usersql,passwordsql);
Statement smt=(Statement)conn.createStatement();
model.addRow(new Object[]{"BUS ID", "First Name", "Last
Name", "Email", "Seats", "From", "Destination", "Journey Date", "Fare"});
contain=this.getContentPane();

ResultSet rs=smt.executeQuery("select * from `bus`.`reserve` where `Email`=\""+email+"\"");
if(rs.next())
{

rs.previous();

while(rs.next())
{
String id=rs.getString("ID");
String fn=rs.getString("First Name");
String ln=rs.getString("Last Name");
String em=rs.getString("Email");
String sea=rs.getString("Seats");
String fr=rs.getString("From");
String des=rs.getString("Destination");
String jour=rs.getString("Journey Date");
String fare=rs.getString("Total Amount");
model.addRow(new Object[]{id,fn,ln,em,sea,fr,des,jour,fare});

}
JLabel lab=new JLabel("YOUR BOOKINGS ");

lab.setFont(font);
lab.setBounds(200, 150, 600, 50);
JTable T=new JTable();
T.setModel(model);
T.setBounds(200,250,1000,(T.getRowCount()*16));
this.getContentPane().add(lab);
this.getContentPane().add(T);

//this.getContentPane().add(lineJLabel);

cancel.setBounds(500, 550, 300, 50);
this.add(cancel);

```

```
}  
else  
{
```

```
ImageIcon icon=new ImageIcon("C:\\Users\\adity\\Desktop\\images\\sorry.jpg");
```

```
    JLabel lab=new JLabel("SORRY YOU DON'T HAVE ANY BOOKING .....");  
    JLabel sor=new JLabel(icon);  
    lab.setFont(font);  
    lab.setBounds(402, 300, 1400, 50);  
    sor.setBounds(550, 400, 150, 150);  
    this.setBackground(Color.WHITE);  
    this.add(sor);  
    this.add(lab);  
  
}
```

```
    } catch (Exception e) {  
        System.out.println(e);  
    }
```

```
}
```

```
}
```

```

package busbooking;

import java.awt.*;
import java.awt.event.*;
import static java.lang.Thread.sleep;
import javax.swing.*;
import java.sql.*;

class delay extends Thread
{
    public void run()
    {

    }
}

public class wait extends JPanel
{

    JLabel pleasewait=new JLabel("PLEASE WAIT LOGGING IN");
    JLabel dot=new JLabel(". . . ");
    String d=".";
    public wait()
    {
        this.setBackground(Color.LIGHT_GRAY);

        pleasewait.setBounds(100, 100, 500, 300);
        Font f=new Font("Ariel",Font.BOLD,30);
        pleasewait.setFont(f);
        dot.setBounds(600, 100, 500, 300);
        dot.setFont(f);
        this.add(pleasewait);
        this.add(dot);

        }

    // return;

}

```