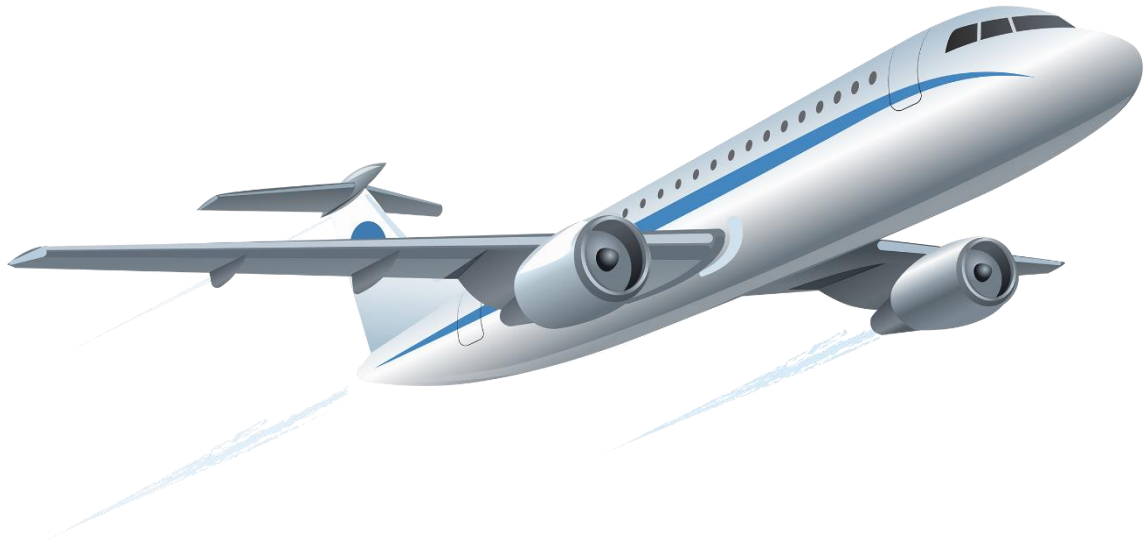


Airport Management System



Submitted To: Sir. Zulfiqar Ali

By

Irfan Danish

[FA15-BEL-012]

Ali Ferzain

[SP15-BEL-043]

Khizar Rattu

[SP13-BEL-022]

“Everybody in this country should learn to program a computer, because it teaches you how to think”
— Steve Jobs

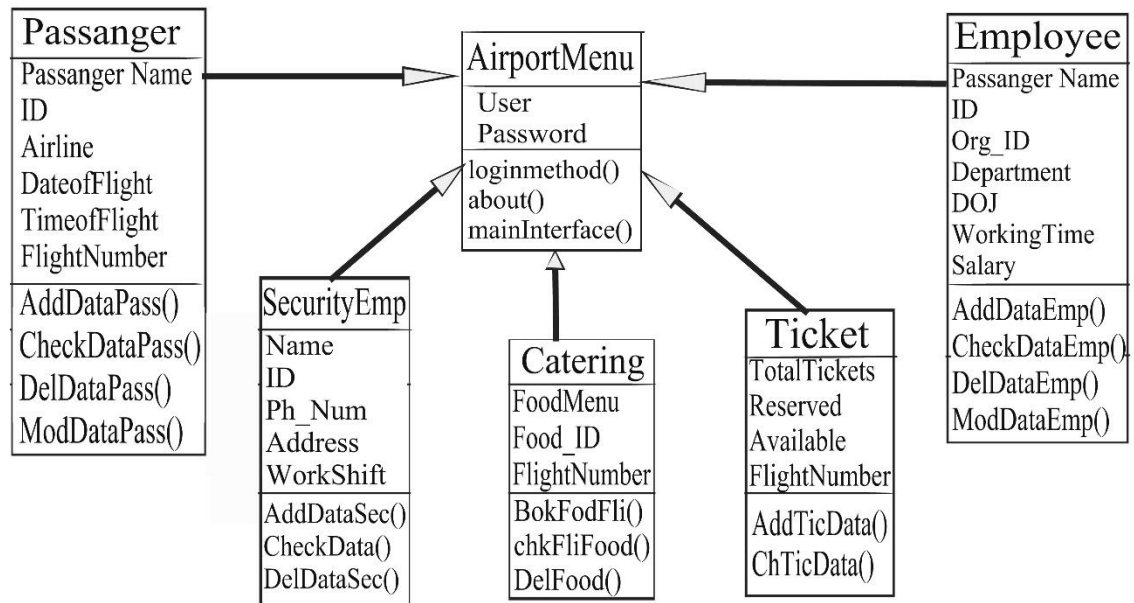
Existing System:

In the existing system Airport Database System there are many flaws. The system lack some features and also had some drawbacks. It was all console based, not user friendly. It lacks efficiency to remove, update and modify employee data, Flight data, seat reservations and their cancelation request, flight departure and arrival timings. Now keeping in view all the drawbacks and vulnerabilities we have developed this project.

Features of project:

- Developed on Graphical User Interface
- User Friendly
- Easy to add, update and modify records
- Shows
 1. Flight Timings
 2. Employee data
 3. Seat reservations
 4. Security guards timings and their names
- Seat reservation criteria
- Ticket purchasing system

Class Diagram:



Source Code:

1. airPortData class:

```

import java.awt.*;
import java.io.*;
import java.util.*;
import javax.swing.*;
import java.awt.Container;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.GridLayout;
import java.awt.Font;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JDialog;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;

```

```

import javax.swing.JTextField;
public class passangerData
{
    BufferedReader buR;
    public FileWriter pFW=null;
    public PrintWriter pPW=null;
    public void addPassangerData(JFrame mainframe)
    {
        JDialog pDialog = new JDialog();
        pDialog.setVisible(true);
        Container c = pDialog.getContentPane();
        pDialog.setSize(765, 730);
        pDialog.setTitle("Add the Data of
Passanger");
        pDialog.setLocationRelativeTo(mainframe);
        c.setLayout(null);

        JLabel laPaNam=new JLabel("Enter the Name of
Passanger:");
        laPaNam.setBounds(180, 210, 180, 25);
        c.add(laPaNam);
        JTextField tPaNam=new JTextField(30);
        tPaNam.setBounds(370, 210, 180, 25);
        c.add(tPaNam);

        JLabel laPaId=new JLabel("Enter the ID of
Passanger:");
        laPaId.setBounds(180, 250, 160, 25);
        c.add(laPaId);
        JTextField tPaId=new JTextField(30);
        tPaId.setBounds(370, 250, 180, 25);
        c.add(tPaId);

        JLabel laPaFl=new JLabel("Enter the Flight
Number:");
        laPaFl.setBounds(180, 290, 160, 25);
        c.add(laPaFl);
        JTextField tPaFl=new JTextField(30);
        tPaFl.setBounds(370, 290, 180, 25);
        c.add(tPaFl);

        JLabel laPaAi=new JLabel("Enter the
Airline:");
        laPaAi.setBounds(180, 330, 160, 25);
    }
}

```

```

c.add(laPaAi);
JTextField tPaAi=new JTextField(30);
tPaAi.setBounds(370, 330, 180, 25);
c.add(tPaAi);

JLabel laPaDt=new JLabel("Enter the Date");
laPaDt.setBounds(180, 370, 160, 25);
c.add(laPaDt);
JTextField tPaDt=new JTextField(30);
tPaDt.setBounds(370, 370, 180, 25);
c.add(tPaDt);

JLabel laPaTm=new JLabel("Enter the time of
Flight");
laPaTm.setBounds(180, 410, 160, 25);
c.add(laPaTm);
JTextField tPaTm=new JTextField(30);
tPaTm.setBounds(370, 410, 180, 25);
c.add(tPaTm);

JButton bOK=new JButton("ok");
bOK.setBounds(370, 450, 50, 25);
c.add(bOK);
bOK.setToolTipText("Press OK to Save Data");
bOK.addActionListener(new ActionListener()
{
    public void
actionPerformed(ActionEvent ae)
    {

if(tPaNam.getText().isEmpty()||tPaAi.getText().isEmpty
y()||tPaDt.getText().isEmpty()||tPaFl.getText().isEmpty
ty()||tPaId.getText().isEmpty()||tPaTm.getText().isEm
pty())

        {
            pDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "Complete all
the Text Fields");
        }
        else
        {
            String nam=tPaNam.getText();

```

```

        String id=tPaId.getText();
        String ai=tPaAi.getText();
        String fl=tPaFl.getText();
        String dt=tPaDt.getText();
        String tm=tPaTm.getText();

        try
        {
            // pFW=new FileWriter(new
File("passangerData.txt"));
            pPW=new PrintWriter(new
FileOutputStream("passangerData.txt", true));
            String paData=nam+"-
"+id+"-"+ai+"-"+fl+"-"+dt+"-"+tm;
            pPW.println(paData);
            pPW.flush();
            pPW.close();

JOptionPane.showMessageDialog(null, "The Record of
the Passanger has been added successfully");

pDialog.setVisible(false);
        }
        catch (FileNotFoundException
e)
        {

pDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "File not
found!");
        }

    }

}

);

JButton pCancel=new JButton("Cancel");
pCancel.setBounds(430, 450, 80, 25);
c.add(pCancel);
pCancel.setToolTipText("Cancel The Dialog
Box");

```

```

        pCancel.addActionListener(new
ActionListener()
        {
            public void actionPerformed(ActionEvent
ae)
                {
                    pDialog.setVisible(false);
                }
        }
    );

} //end of method addPassangerData()

public void modPass(JFrame window)
{
    JDialog pdelDialog = new JDialog();
    pdelDialog.setVisible(true);
    Container c = pdelDialog.getContentPane();
    pdelDialog.setSize(630,250);
    pdelDialog.setTitle("Delete the Data of
Passanger");
    pdelDialog.setLocationRelativeTo(window);
    c.setLayout(null);

    JLabel dPass=new JLabel("Enter the ID Number
of Passanger:");
    dPass.setBounds(70, 70, 200, 25);
    c.add(dPass);
    JTextField dTPass=new JTextField(30);
    dTPass.setBounds(290, 70, 200, 25);
    c.add(dTPass);
    JButton dButton=new JButton("ok");
    dButton.setBounds(290, 100,50, 25);
    c.add(dButton);
    dButton.setToolTipText("Press ok to delete
the Record!");
    dButton.addActionListener(new
ActionListener()
    {
        public void actionPerformed(ActionEvent
ae)
            {
                if(dTPass.getText().isEmpty())
                {

```

```

        pdelDialog.setVisible(true);

        JOptionPane.showMessageDialog(null, "Empty
Fields, Please Enter Valid ID");
    }
    else
    {
        String id=dTPass.getText();

        try
        {
            BufferedReader dBr=new
BufferedReader(new FileReader("passangerData.txt"));
            StringBuffer sBu=new
StringBuffer();

            String line,nline;

            while((line=dBr.readLine())!=null)
            {
                if(line.contains(id))
                {

nline=line.replace(line,"");
                    line=nline;
                }
                if(line.equals(""))
                {

                }
                else
                {
                    sBu.append(line+"\n");

pdelDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "Data Relative to
ID is not Found");

                    break;
                }
            }

            dBr.close();
            FileWriter f=new FileWriter(new
File("passangerData.txt"),false);

```



```

        PrintWriter p=new PrintWriter(f);
        p.print(sBu.toString());
        f.flush();
        f.close();
        pdelDialog.setVisible(false);

        /*Writing new or modified data to
file */

        JDialog pDialog = new JDialog();
        pDialog.setVisible(true);
        Container c = pDialog.getContentPane();
        pDialog.setSize(765, 730);
        pDialog.setTitle("Add the Data of
Passanger");
        pDialog.setLocationRelativeTo(window);
        c.setLayout(null);

        JLabel laPaNam=new JLabel("Enter the Name of
Passanger:");
        laPaNam.setBounds(180, 210, 180, 25);
        c.add(laPaNam);
        JTextField tPaNam=new JTextField(30);
        tPaNam.setBounds(370, 210, 180, 25);
        c.add(tPaNam);

        JLabel laPaId=new JLabel("Enter the ID of
Passanger:");
        laPaId.setBounds(180, 250, 160, 25);
        c.add(laPaId);
        JTextField tPaId=new JTextField(30);
        tPaId.setBounds(370, 250, 180, 25);
        c.add(tPaId);

        JLabel laPaFl=new JLabel("Enter the Flight
Number:");
        laPaFl.setBounds(180, 290, 160, 25);
        c.add(laPaFl);
        JTextField tPaFl=new JTextField(30);
        tPaFl.setBounds(370, 290, 180, 25);
        c.add(tPaFl);

```

```

        JLabel laPaAi=new JLabel("Enter the
Airline:");
        laPaAi.setBounds(180, 330, 160, 25);
        c.add(laPaAi);
        JTextField tPaAi=new JTextField(30);
        tPaAi.setBounds(370, 330, 180, 25);
        c.add(tPaAi);

        JLabel laPaDt=new JLabel("Enter the Date");
        laPaDt.setBounds(180, 370, 160, 25);
        c.add(laPaDt);
        JTextField tPaDt=new JTextField(30);
        tPaDt.setBounds(370, 370, 180, 25);
        c.add(tPaDt);

        JLabel laPaTm=new JLabel("Enter the time of
Flight");
        laPaTm.setBounds(180, 410, 160, 25);
        c.add(laPaTm);
        JTextField tPaTm=new JTextField(30);
        tPaTm.setBounds(370, 410, 180, 25);
        c.add(tPaTm);

        JButton bOK=new JButton("ok");
        bOK.setBounds(370, 450, 50, 25);
        c.add(bOK);
        bOK.setToolTipText("Press OK to Save Data");
        bOK.addActionListener(new ActionListener()
        {
            public void
actionPerformed(ActionEvent ae)
            {

if(tPaNam.getText().isEmpty()||tPaAi.getText().isEmpty
y()||tPaDt.getText().isEmpty()||tPaFl.getText().isEmp
ty()||tPaId.getText().isEmpty()||tPaTm.getText().isEm
pty())

                {
                    pDialog.setVisible(true);

                    JOptionPane.showMessageDialog(null, "Complete all
the Text Fields");
                }
            }
        }
    }

```

```

        else
        {
            String nam=tPaNam.getText();
            String id=tPaId.getText();
            String ai=tPaAi.getText();
            String fl=tPaFl.getText();
            String dt=tPaDt.getText();
            String tm=tPaTm.getText();

            try
            {
                // pFW=new FileWriter(new
File("passangerData.txt"));
                pPW=new PrintWriter(new
FileOutputStream("passangerData.txt", true));
                String paData=nam+"-
"+id+"-"+ai+"-"+fl+"-"+dt+"-"+tm;
                pPW.println(paData);
                pPW.flush();
                pPW.close();

JOptionPane.showMessageDialog(null, "The Record of
the Passanger has been added successfully");

pDialog.setVisible(false);
            }
            catch (FileNotFoundException
e)
            {

pDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "File not
found!");
            }
        }
    }
};

JButton pCancel=new JButton("Cancel");
pCancel.setBounds(430, 450, 80, 25);

```

```

        c.add(pCancel);
        pCancel.setToolTipText("Cancel The Dialog
Box");
        pCancel.addActionListener(new
ActionListener()
        {
            public void actionPerformed(ActionEvent
ae)
            {
                pDialog.setVisible(false);
            }
        });

    }

    catch(IOException e)
    {
        pdelDialog.setVisible(true);
    }

JOptionPane.showMessageDialog(null, "Input Output
error during file Reading");
    }

    }
}

);

JButton pCancel=new JButton("Cancel");
pCancel.setBounds(350, 100, 80, 25);
c.add(pCancel);
pCancel.setToolTipText("Cancel The Dialog
Box");
pCancel.addActionListener(new
ActionListener()
{
    public void actionPerformed(ActionEvent
ae)
    {
        pdelDialog.setVisible(false);
    }
}

```

```

        }
    );
} //end of modPass()

public void chkPass(JFrame window)
{
    JDialog pCDialog = new JDialog();
    pCDialog.setVisible(true);
    Container pc = pCDialog.getContentPane();
    pCDialog.setSize(630,250);
    pCDialog.setTitle("Check the Data of
Passanger");
    pCDialog.setLocationRelativeTo(window);
    pc.setLayout(null);
    JLabel pcEmp=new JLabel("Enter the ID Number
of Passanger:");
    pcEmp.setBounds(70, 70, 200, 25);
    pc.add(pcEmp);
    JTextField pcTPas=new JTextField(30);
    pcTPas.setBounds(290, 70, 200, 25);
    pc.add(pcTPas);
    JButton ok=new JButton("ok");
    ok.setBounds(290, 100,50, 25);
    pc.add(ok);
    ok.setToolTipText("Press ok to Check the
record of Passanger");
    ok.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent
ae)
        {
            String id=pcTPas.getText();
            try
            {
                buR=new BufferedReader(new
FileReader("PassangerData.txt"));
                String l="";
                while ((l=buR.readLine())!=null)
                {
                    String pArr[]=l.split("-");

                    if(pArr[1].equals(id))

```

```

{

/*Shows the output
dialog*/
pCDialog.setVisible(false);
JDialog pchkDialog=new
JDialog();

pchkDialog.setVisible(true);
Container
c=pchkDialog.getContentPane();

pchkDialog.setSize(400,500);

pchkDialog.setTitle("Check The Data of Passanger");

pchkDialog.setLocationRelativeTo(window);
c.setLayout(null);

JLabel pNam=new
JLabel("Name of Passanger:");
pNam.setBounds(80, 80,
180, 25);

c.add(pNam);
JLabel pONam=new
JLabel(pArr[0]);
pONam.setBounds(200, 80,
180, 25);

c.add(pONam);

JLabel pID=new JLabel("ID
of Passanger:");
pID.setBounds(80, 120,
180, 25);

c.add(pID);
JLabel pOID=new
JLabel(pArr[1]);
pOID.setBounds(200, 120,
180, 25);

c.add(pOID);

```

```

JLabel("Department:");
180, 25);

JLabel(pArr[2]);
180, 25);

JLabel("Date of Joining:");
180, 25);

JLabel(pArr[3]);
180, 25);

JLabel("Hours of Working:");
180, 25);

JLabel(pArr[4]);
180, 25);

JLabel("Salary:");
180, 25);

JLabel(pArr[5]);
180, 25);

JLabel pDep=new
pDep.setBounds(80, 160,
c.add(pDep);
JLabel pODep=new
pODep.setBounds(200, 160,
c.add(pODep);

JLabel pDOJ=new
pDOJ.setBounds(80, 200,
c.add(pDOJ);
JLabel pODOJ=new
pODOJ.setBounds(200, 200,
c.add(pODOJ);

JLabel pHoW=new
pHoW.setBounds(80, 240,
c.add(pHoW);
JLabel pOHoW=new
pOHoW.setBounds(200, 240,
c.add(pOHoW);

JLabel pSal=new
pSal.setBounds(80, 280,
c.add(pSal);
JLabel pOSal=new
pOSal.setBounds(200, 280,
c.add(pOSal);

```

```

        JButton pOK=new
JButton("ok");
        pOK.setBounds(200, 320,
50, 25);
        c.add(pOK);
        pOK.setToolTipText("Exit
the Output Dialog");
        pOK.addActionListener(new
ActionListener()
        {
            public void
actionPerformed(ActionEvent ae)
            {
                pchkDialog.setVisible(false);
            }
        });
        }
        else
        {
            System.out.println("The
Data of the ID is not Found!");
//JOptionPane.showMessageDialog(null,"The Data of the
ID is not Found!");
        }
    }
    buR.close();
}
catch(IOException io)
{
    JOptionPane.showMessageDialog(null,"Input Output
Error during File Reading");
}

}
);

JButton pCancel=new JButton("Cancel");

```



```

        pCancel.setBounds(350, 100, 80, 25);
        pc.add(pCancel);
        pCancel.setToolTipText("Cancel The Dialog
Box");
        pCancel.addActionListener(new
ActionListener()
        {
            public void actionPerformed(ActionEvent
ae)
            {
                pCDialog.setVisible(false);
            }
        }
    );

} //end of method chkPass()

public void delPass(JFrame window)
{
    JDialog pdelDialog = new JDialog();
    pdelDialog.setVisible(true);
    Container c = pdelDialog.getContentPane();
    pdelDialog.setSize(630,250);
    pdelDialog.setTitle("Delete the Data of
Passanger");
    pdelDialog.setLocationRelativeTo(window);
    c.setLayout(null);

    JLabel dPass=new JLabel("Enter the ID Number
of Passanger:");
    dPass.setBounds(70, 70, 200, 25);
    c.add(dPass);
    JTextField dTPass=new JTextField(30);
    dTPass.setBounds(290, 70, 200, 25);
    c.add(dTPass);
    JButton dButton=new JButton("ok");
    dButton.setBounds(290, 100,50, 25);
    c.add(dButton);
    dButton.setToolTipText("Press ok to delete
the Record!");
    dButton.addActionListener(new
ActionListener()
    {

```

```

        public void actionPerformed(ActionEvent
ae)
        {
            if(dTPass.getText().isEmpty())
            {
                pdelDialog.setVisible(true);

                JOptionPane.showMessageDialog(null, "Empty
Fields, Please Enter Valid ID");
            }
            else
            {
                String id=dTPass.getText();

                try
                {
                    BufferedReader dBr=new
BufferedReader(new FileReader("passangerData.txt"));
                    StringBuffer sBu=new
StringBuffer();

                    String line,nline;

                    while((line=dBr.readLine())!=null)
                    {
                        if(line.contains(id))
                        {

nline=line.replace(line,"");
                            line=nline;
                        }
                        if(line.equals(""))
                        {

                        }
                        else
                        {
                            sBu.append(line+"\n");

pdelDialog.setVisible(true);

//JOptionPane.showMessageDialog(null, "Data Relative
to ID is not Found");

                            break;

```

```

        }
    }
    dBr.close();
    FileWriter f=new FileWriter(new
File("passangerData.txt"),false);
    PrintWriter p=new PrintWriter(f);
    p.print(sBu.toString());
    f.flush();
    f.close();

JOptionPane.showMessageDialog(null, "The Data has
been deleted successfully");
    pdelDialog.setVisible(false);
}

    catch(IOException e)
    {
        pdelDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "Input Output
error during file Reading");
    }

    }
}
);

JButton pCancel=new JButton("Cancel");
pCancel.setBounds(350, 100, 80, 25);
c.add(pCancel);
pCancel.setToolTipText("Cancel The Dialog
Box");
    pCancel.addActionListener(new
ActionListener()
    {
        public void actionPerformed(ActionEvent
ae)
        {
            pdelDialog.setVisible(false);

        }
    }
);

```

```

    } // end of method delPass().
}

```

2. passanger class:

```

3. import java.awt.*;
4. import java.io.*;
5. import java.util.*;
6. import javax.swing.*;
7. import java.awt.Container;
8. import java.awt.event.ActionEvent;
9. import java.awt.event.ActionListener;
10.     import java.awt.GridLayout;
11.     import java.awt.Font;
12.     import javax.swing.ImageIcon;
13.     import javax.swing.JButton;
14.     import javax.swing.JDialog;
15.     import javax.swing.JFrame;
16.     import javax.swing.JLabel;
17.     import javax.swing.JMenu;
18.     import javax.swing.JMenuBar;
19.     import javax.swing.JMenuItem;
20.     import javax.swing.JOptionPane;
21.     import javax.swing.JPanel;
22.     import javax.swing.JPasswordField;
23.     import javax.swing.JTextField;
24.     public class passangerData
25.     {
26.         BufferedReader buR;
27.         public FileWriter pFW=null;
28.         public PrintWriter pPW=null;
29.         public void addPassangerData(JFrame
mainframe)
30.         {
31.             JDialog pDialog = new JDialog();
32.             pDialog.setVisible(true);
33.             Container c =
pDialog.getContentPane();
34.             pDialog.setSize(765, 730);
35.             pDialog.setTitle("Add the Data
of Passanger");

```

```

36.         pDialog.setLocationRelativeTo(mainframe)
           ;
37.         c.setLayout(null);
38.
39.         JLabel laPaNam=new
           JLabel("Enter the Name of Passanger:");
40.         laPaNam.setBounds(180, 210,
           180, 25);
41.         c.add(laPaNam);
42.         JTextField tPaNam=new
           JTextField(30);
43.         tPaNam.setBounds(370, 210, 180,
           25);
44.         c.add(tPaNam);
45.
46.         JLabel laPaId=new JLabel("Enter
           the ID of Passanger:");
47.         laPaId.setBounds(180, 250, 160,
           25);
48.         c.add(laPaId);
49.         JTextField tPaId=new
           JTextField(30);
50.         tPaId.setBounds(370, 250, 180,
           25);
51.         c.add(tPaId);
52.
53.         JLabel laPaFl=new JLabel("Enter
           the Flight Number:");
54.         laPaFl.setBounds(180, 290, 160,
           25);
55.         c.add(laPaFl);
56.         JTextField tPaFl=new
           JTextField(30);
57.         tPaFl.setBounds(370, 290, 180,
           25);
58.         c.add(tPaFl);
59.
60.         JLabel laPaAi=new JLabel("Enter
           the Airline:");
61.         laPaAi.setBounds(180, 330, 160,
           25);
62.         c.add(laPaAi);

```

```

63.         JTextField tPaAi=new
        JTextField(30);
64.         tPaAi.setBounds(370, 330, 180,
        25);
65.         c.add(tPaAi);
66.
67.         JLabel laPaDt=new
        JLabel("Enter the Date");
68.         laPaDt.setBounds(180, 370, 160,
        25);
69.         c.add(laPaDt);
70.         JTextField tPaDt=new
        JTextField(30);
71.         tPaDt.setBounds(370, 370, 180,
        25);
72.         c.add(tPaDt);
73.
74.         JLabel laPaTm=new JLabel("Enter
        the time of Flight");
75.         laPaTm.setBounds(180, 410, 160,
        25);
76.         c.add(laPaTm);
77.         JTextField tPaTm=new
        JTextField(30);
78.         tPaTm.setBounds(370, 410, 180,
        25);
79.         c.add(tPaTm);
80.
81.         JButton bOK=new JButton("ok");
82.         bOK.setBounds(370, 450, 50,
        25);
83.         c.add(bOK);
84.         bOK.setToolTipText("Press OK to
        Save Data");
85.         bOK.addActionListener(new
        ActionListener()
86.         {
87.             public void
            actionPerformed(ActionEvent ae)
88.             {
89.
90.                 if(tPaNam.getText().isEmpty() || tPaAi.getText(
                    ) .isEmpty() || tPaDt.getText().isEmpty() || tPaFl

```

```

        .getText().isEmpty() || tPaId.getText().isEmpty()
        ) || tPaTm.getText().isEmpty())
91.                {
92.
93.            pDialog.setVisible(true);
94.            JOptionPane.showMessageDialog(null,
95.                "Complete all the Text Fields");
96.            }
97.            else
98.            {
99.                String
100.                nam=tPaNam.getText();
101.                String
102.                id=tPaId.getText();
103.                String
104.                ai=tPaAi.getText();
105.                String
106.                fl=tPaFl.getText();
107.                String
108.                dt=tPaDt.getText();
109.                String
110.                tm=tPaTm.getText();
111.            try
112.            {
113.                // pFW=new
114.                FileWriter(new File("passangerData.txt"));
115.                pPW=new
116.                PrintWriter(new
117.                FileOutputStream("passangerData.txt", true));
118.                String
119.                paData=nam+"-"+id+"-"+ai+"-"+fl+"-"+dt+"-
120.                "+tm;
121.
122.                pPW.println(paData);
123.                pPW.flush();
124.                pPW.close();
125.            }
126.            JOptionPane.showMessageDialog(null, "The
127.            Record of the Passanger has been added
128.            successfully");

```

```

113.        pDialog.setVisible(false);
114.    }
115.    catch (FileNotFoundException e)
116.    {
117.        pDialog.setVisible(true);
118.        JOptionPane.showMessageDialog(null, "File not
found!");
119.    }
120.
121.    }
122.
123.    }
124.    }
125.    );
126.
127.        JButton pCancel=new
        JButton("Cancel");
128.        pCancel.setBounds(430, 450, 80,
        25);
129.        c.add(pCancel);
130.        pCancel.setToolTipText("Cancel
The Dialog Box");
131.        pCancel.addActionListener(new
        ActionListener()
132.        {
133.            public void
            actionPerformed(ActionEvent ae)
134.            {
135.                pDialog.setVisible(false);
136.            }
137.        }
138.    );
139.
140.    }//end of method addPassangerData()
141.
142.    public void modPass(JFrame window)
143.    {
144.        JDialog pdelDialog = new
        JDialog();

```



```

145.         pdelDialog.setVisible(true);
146.         Container c =
            pdelDialog.getContentPane();
147.         pdelDialog.setSize(630,250);
148.         pdelDialog.setTitle("Delete the
            Data of Passanger");
149.         pdelDialog.setLocationRelativeTo(window)
            ;
150.         c.setLayout(null);
151.
152.         JLabel dPass=new JLabel("Enter
            the ID Number of Passanger:");
153.         dPass.setBounds(70, 70, 200,
            25);
154.         c.add(dPass);
155.         JTextField dTPass=new
            JTextField(30);
156.         dTPass.setBounds(290, 70, 200,
            25);
157.         c.add(dTPass);
158.         JButton dButton=new
            JButton("ok");
159.         dButton.setBounds(290, 100,50,
            25);
160.         c.add(dButton);
161.         dButton.setToolTipText("Press ok
            to delete the Record!");
162.         dButton.addActionListener(new
            ActionListener()
163.         {
164.             public void
            actionPerformed(ActionEvent ae)
165.             {
166.                 if(dTPass.getText().isEmpty())
167.                 {
168.                     pdelDialog.setVisible(true);
169.                     JOptionPane.showMessageDialog(null,
                        "Empty Fields,Please Enter Valid ID");
170.                 }
171.                 else

```

```

172.                {
173.                    String
        id=dTPass.getText();
174.
175.                try
176.                {
177.                    BufferedReader
        dBr=new BufferedReader(new
        FileReader("passangerData.txt"));
178.                    StringBuffer sBu=new
        StringBuffer();
179.
180.                    String line,nline;
181.
        while((line=dBr.readLine())!=null)
182.                {
183.
        if(line.contains(id))
184.                {
185.
        nline=line.replace(line,"");
186.                    line=nline;
187.                }
188.
        if(line.equals(""))
189.                {
190.
191.                }
192.                else
193.                {
194.
        sBu.append(line+"\n");
195.
        pdelDialog.setVisible(true);
196.
        JOptionPane.showMessageDialog(null, "Data
        Relative to ID is not Found");
197.                break;
198.                }
199.            }
200.            dBr.close();
201.            FileWriter f=new FileWriter(new
        File("passangerData.txt"),false);

```

```

202.        PrintWriter p=new
        PrintWriter(f);
203.        p.print(sBu.toString());
204.        f.flush();
205.        f.close();
206.        pdelDialog.setVisible(false);
207.
208.
209.        /*Writing new or
        modified data to file */
210.
211.
212.        JDialog pDialog = new JDialog();
213.        pDialog.setVisible(true);
214.        Container c =
        pDialog.getContentPane();
215.        pDialog.setSize(765, 730);
216.        pDialog.setTitle("Add the Data
        of Passanger");
217.
        pDialog.setLocationRelativeTo(window);
218.        c.setLayout(null);
219.
220.        JLabel laPaNam=new
        JLabel("Enter the Name of Passanger:");
221.        laPaNam.setBounds(180, 210,
        180, 25);
222.        c.add(laPaNam);
223.        JTextField tPaNam=new
        JTextField(30);
224.        tPaNam.setBounds(370, 210, 180,
        25);
225.        c.add(tPaNam);
226.
227.        JLabel laPaId=new JLabel("Enter
        the ID of Passanger:");
228.        laPaId.setBounds(180, 250, 160,
        25);
229.        c.add(laPaId);
230.        JTextField tPaId=new
        JTextField(30);
231.        tPaId.setBounds(370, 250, 180,
        25);
232.        c.add(tPaId);

```

```
233.
234.         JLabel laPaFl=new JLabel("Enter
the Flight Number:");
235.         laPaFl.setBounds(180, 290, 160,
25);
236.         c.add(laPaFl);
237.         JTextField tPaFl=new
JTextField(30);
238.         tPaFl.setBounds(370, 290, 180,
25);
239.         c.add(tPaFl);
240.
241.         JLabel laPaAi=new JLabel("Enter
the Airline:");
242.         laPaAi.setBounds(180, 330, 160,
25);
243.         c.add(laPaAi);
244.         JTextField tPaAi=new
JTextField(30);
245.         tPaAi.setBounds(370, 330, 180,
25);
246.         c.add(tPaAi);
247.
248.         JLabel laPaDt=new
JLabel("Enter the Date");
249.         laPaDt.setBounds(180, 370, 160,
25);
250.         c.add(laPaDt);
251.         JTextField tPaDt=new
JTextField(30);
252.         tPaDt.setBounds(370, 370, 180,
25);
253.         c.add(tPaDt);
254.
255.         JLabel laPaTm=new JLabel("Enter
the time of Flight");
256.         laPaTm.setBounds(180, 410, 160,
25);
257.         c.add(laPaTm);
258.         JTextField tPaTm=new
JTextField(30);
259.         tPaTm.setBounds(370, 410, 180,
25);
260.         c.add(tPaTm);
```



```

287.                                     // pFW=new
    FileWriter(new File("passangerData.txt"));
288.                                     pPW=new
    PrintWriter(new
    FileOutputStream("passangerData.txt", true));
289.                                     String
    paData=nam+"-"+id+"-"+ai+"-"+fl+"-"+dt+"-
    "+tm;
290.
    pPW.println(paData);
291.
    pPW.flush();
292.
    pPW.close();
293.
    JOptionPane.showMessageDialog(null, "The
    Record of the Passanger has been added
    successfully");
294.
    pDialog.setVisible(false);
295.                                     }
296.
    catch (FileNotFoundException e)
297.                                     {
298.
    pDialog.setVisible(true);
299.
    JOptionPane.showMessageDialog(null, "File not
    found!");
300.                                     }
301.
302.                                     }
303.
304.                                     }
305.                                     }
306.                                     );
307.
308.                                     JButton pCancel=new
    JButton("Cancel");
309.                                     pCancel.setBounds(430, 450, 80,
    25);
310.                                     c.add(pCancel);
311.                                     pCancel.setToolTipText("Cancel
    The Dialog Box");

```

```

312.            pCancel.addActionListener(new
                ActionListener()
313.            {
314.                public void
                actionPerformed(ActionEvent ae)
315.                {
316.                    pDialog.setVisible(false);
317.                }
318.            }
319.        );
320.
321.
322.                }
323.
324.                catch (IOException e)
325.                {
326.                    pdelDialog.setVisible(true);
327.                    JOptionPane.showMessageDialog(null, "Input
                Output error during file Reading");
328.                }
329.
330.            }
331.        }
332.    }
333.    );
334.
335.        JButton pCancel=new
        JButton("Cancel");
336.        pCancel.setBounds(350, 100, 80,
        25);
337.        c.add(pCancel);
338.        pCancel.setToolTipText("Cancel
        The Dialog Box");
339.        pCancel.addActionListener(new
        ActionListener()
340.        {
341.            public void
            actionPerformed(ActionEvent ae)
342.            {
343.                pdelDialog.setVisible(false);

```

```

344.
345.         }
346.     }
347.     );
348. } //end of modPass()
349.
350.     public void chkPass(JFrame window)
351.     {
352.
353.         JDialog pCDialog = new JDialog();
354.         pCDialog.setVisible(true);
355.         Container pc =
            pCDialog.getContentPane();
356.         pCDialog.setSize(630,250);
357.         pCDialog.setTitle("Check the
            Data of Passanger");
358.
            pCDialog.setLocationRelativeTo(window);
359.         pc.setLayout(null);
360.         JLabel pcEmp=new JLabel("Enter
            the ID Number of Passanger:");
361.         pcEmp.setBounds(70, 70, 200,
            25);
362.         pc.add(pcEmp);
363.         JTextField pcTPas=new
            JTextField(30);
364.         pcTPas.setBounds(290, 70, 200,
            25);
365.         pc.add(pcTPas);
366.         JButton ok=new JButton("ok");
367.         ok.setBounds(290, 100,50, 25);
368.         pc.add(ok);
369.         ok.setToolTipText("Press ok to
            Check the record of Passanger");
370.         ok.addActionListener(new
            ActionListener()
371.         {
372.             public void
            actionPerformed(ActionEvent ae)
373.             {
374.                 String
            id=pcTPas.getText();
375.                 try
376.                 {

```



```

377.                                     buR=new
    BufferedReader(new
    FileReader("PassangerData.txt"));
378.                                     String l="";
379.
    while((l=buR.readLine())!=null)
380.     {
381.         String
        pArr[]=l.split("-");
382.
383.
384.
        if(pArr[1].equals(id))
385.     {
386.
387.
388.
389.                                     /*Shows the
        output dialog*/
390.
        pCDialog.setVisible(false);
391.                                     JDialog
        pchkDialog=new JDialog();
392.
        pchkDialog.setVisible(true);
393.                                     Container
        c=pchkDialog.getContentPane();
394.
        pchkDialog.setSize(400,500);
395.
        pchkDialog.setTitle("Check The Data of
        Passanger");
396.
        pchkDialog.setLocationRelativeTo(window);
397.
        c.setLayout(null);
398.
399.                                     JLabel
        pNam=new JLabel("Name of Passanger");
400.
        pNam.setBounds(80, 80, 180, 25);
401.                                     c.add(pNam);
402.                                     JLabel
        pONam=new JLabel(pArr[0]);

```

```

403.
    pONam.setBounds(200, 80, 180, 25);
404.
    c.add(pONam);
405.
406.                                JLabel
    pID=new JLabel("ID of Passanger:");
407.
    pID.setBounds(80, 120, 180, 25);
408.                                c.add(pID);
409.                                JLabel
    pOID=new JLabel(pArr[1]);
410.
    pOID.setBounds(200, 120, 180, 25);
411.                                c.add(pOID);
412.
413.                                JLabel
    pDep=new JLabel("Department:");
414.
    pDep.setBounds(80, 160, 180, 25);
415.                                c.add(pDep);
416.                                JLabel
    pODep=new JLabel(pArr[2]);
417.
    pODep.setBounds(200, 160, 180, 25);
418.
    c.add(pODep);
419.
420.                                JLabel
    pDOJ=new JLabel("Date of Joining:");
421.
    pDOJ.setBounds(80, 200, 180, 25);
422.                                c.add(pDOJ);
423.                                JLabel
    pODOJ=new JLabel(pArr[3]);
424.
    pODOJ.setBounds(200, 200, 180, 25);
425.
    c.add(pODOJ);
426.
427.                                JLabel
    pHoW=new JLabel("Hours of Working:");
428.
    pHoW.setBounds(80, 240, 180, 25);

```

```

429.                                c.add(pHoW);
430.                                JLabel
    pOHoW=new JLabel(pArr[4]);
431.
    pOHoW.setBounds(200, 240, 180, 25);
432.
    c.add(pOHoW);
433.
434.                                JLabel
    pSal=new JLabel("Salary:");
435.
    pSal.setBounds(80, 280, 180, 25);
436.                                c.add(pSal);
437.                                JLabel
    pOSal=new JLabel(pArr[5]);
438.
    pOSal.setBounds(200, 280, 180, 25);
439.
    c.add(pOSal);
440.
441.                                JButton
    pOK=new JButton("ok");
442.
    pOK.setBounds(200, 320, 50, 25);
443.                                c.add(pOK);
444.
    pOK.setToolTipText("Exit the Output Dialog");
445.
    pOK.addActionListener(new ActionListener()
446.                                {
447.                                public
    void actionPerformed(ActionEvent ae)
448.                                {
449.
    pchkDialog.setVisible(false);
450.                                }
451.                                }
452.                                );
453.
454.                                }
455.                                else
456.                                {

```

```

457.         System.out.println("The Data of the ID is not
Found!");
458.         //JOptionPane.showMessageDialog(null,"The
Data of the ID is not Found!");
459.             }
460.         }
461.         buR.close();
462.     }
463.     catch(IOException io)
464.     {
465.         JOptionPane.showMessageDialog(null,"Input
Output Error during File Reading");
466.     }
467.
468.
469.     }
470. }
471. );
472.
473.         JButton pCancel=new
JButton("Cancel");
474.         pCancel.setBounds(350, 100, 80,
25);
475.         pc.add(pCancel);
476.         pCancel.setToolTipText("Cancel
The Dialog Box");
477.         pCancel.addActionListener(new
ActionListener()
478.         {
479.             public void
actionPerformed(ActionEvent ae)
480.             {
481.                 pCDialog.setVisible(false);
482.             }
483.         }
484.     );
485.
486.     }//end of method chkPass()
487.
488.     public void delPass(JFrame window)

```

```

489.      {
490.          JDialog pdelDialog = new
      JDialog();
491.          pdelDialog.setVisible(true);
492.          Container c =
      pdelDialog.getContentPane();
493.          pdelDialog.setSize(630,250);
494.          pdelDialog.setTitle("Delete the
      Data of Passanger");
495.          pdelDialog.setLocationRelativeTo(window)
      ;
496.          c.setLayout(null);
497.
498.          JLabel dPass=new JLabel("Enter
      the ID Number of Passanger:");
499.          dPass.setBounds(70, 70, 200,
      25);
500.          c.add(dPass);
501.          JTextField dTPass=new
      JTextField(30);
502.          dTPass.setBounds(290, 70, 200,
      25);
503.          c.add(dTPass);
504.          JButton dButton=new
      JButton("ok");
505.          dButton.setBounds(290, 100,50,
      25);
506.          c.add(dButton);
507.          dButton.setToolTipText("Press ok
      to delete the Record!");
508.          dButton.addActionListener(new
      ActionListener()
509.          {
510.              public void
      actionPerformed(ActionEvent ae)
511.              {
512.                  if(dTPass.getText().isEmpty())
513.                  {
514.                      pdelDialog.setVisible(true);

```

```

515.         JOptionPane.showMessageDialog(null,
        "Empty Fields, Please Enter Valid ID");
516.     }
517.     else
518.     {
519.         String
        id=dTPass.getText();
520.
521.         try
522.         {
523.             BufferedReader
        dBr=new BufferedReader(new
        FileReader("passangerData.txt"));
524.             StringBuffer sBu=new
        StringBuffer();
525.
526.             String line,nline;
527.             while((line=dBr.readLine())!=null)
528.             {
529.                 if(line.contains(id))
530.                 {
531.                     nline=line.replace(line, "");
532.                     line=nline;
533.                 }
534.                 if(line.equals(""))
535.                 {
536.
537.                 }
538.                 else
539.                 {
540.
541.                     sBu.append(line+"\n");
542.                     pdelDialog.setVisible(true);
543.                     //JOptionPane.showMessageDialog(null, "Data
        Relative to ID is not Found");
544.                     break;
        }
    }

```

```

545.                                }
546.                                dBr.close();
547.                                FileWriter f=new
FileWriter(new
File("passangerData.txt"),false);
548.                                PrintWriter p=new
PrintWriter(f);
549.
p.print(sBu.toString());
550.                                f.flush();
551.                                f.close();
552.
JOptionPane.showMessageDialog(null, "The Data
has been deleted successfully");
553.
pdelDialog.setVisible(false);
554.                                }
555.
556.                                catch(IOException e)
557.                                {
558.
pdelDialog.setVisible(true);
559.
JOptionPane.showMessageDialog(null, "Input
Output error during file Reading");
560.                                }
561.
562.                                }
563.                                }
564.                                }
565.                                );
566.
567.                                JButton pCancel=new
JButton("Cancel");
568.                                pCancel.setBounds(350, 100, 80,
25);
569.                                c.add(pCancel);
570.                                pCancel.setToolTipText("Cancel
The Dialog Box");
571.                                pCancel.addActionListener(new
ActionListener()
572.                                {
573.                                public void
actionPerformed(ActionEvent ae)

```

```

574.         {
575.
576.             pdelDialog.setVisible(false);
577.         }
578.     }
579. );
580.
581. } // end of method delPass().
582. }

```

583. employee class:

```

584. import java.awt.*;
585. import java.io.*;
586. import java.util.*;
587. import javax.swing.*;
588. import java.awt.Container;
589. import java.awt.event.ActionEvent;
590. import java.awt.event.ActionListener;
591. import java.awt.GridLayout;
592. import java.awt.Font;
593. import javax.swing.ImageIcon;
594. import javax.swing.JButton;
595. import javax.swing.JDialog;
596. import javax.swing.JFrame;
597. import javax.swing.JLabel;
598. import javax.swing.JMenu;
599. import javax.swing.JMenuBar;
600. import javax.swing.JMenuItem;
601. import javax.swing.JOptionPane;
602. import javax.swing.JPanel;
603. import javax.swing.JPasswordField;
604. import javax.swing.JTextField;
605. import javax.swing.*;
606. public class employeeData
607. {
608.     BufferedReader buR;
609.     public PrintWriter ePW=null;
610.     public void addEmp(JFrame window)
611.     {
612.
613.
614.         JDialog eDialog = new JDialog();
615.         eDialog.setVisible(true);

```



```
616.         Container c =
           eDialog.getContentPane();
617.         eDialog.setSize(705, 630);
618.         eDialog.setTitle("Add the Data of
           Employees");
619.         eDialog.setLocationRelativeTo(window);
620.         c.setLayout(null);
621.
622.         JLabel laEmNam=new JLabel("Enter
           the Name of Employee:");
623.         laEmNam.setBounds(180, 210, 180,
           25);
624.         c.add(laEmNam);
625.         JTextField tEmNam=new
           JTextField(30);
626.         tEmNam.setBounds(370, 210, 180,
           25);
627.         c.add(tEmNam);
628.
629.         JLabel laEmId=new JLabel("Enter the
           ID of Employee:");
630.         laEmId.setBounds(180, 250, 160,
           25);
631.         c.add(laEmId);
632.         JTextField tEmId=new
           JTextField(30);
633.         tEmId.setBounds(370, 250, 180, 25);
634.         c.add(tEmId);
635.
636.         JLabel laEmDe=new JLabel("Enter the
           Department:");
637.         laEmDe.setBounds(180, 290, 160,
           25);
638.         c.add(laEmDe);
639.         JTextField tEmDe=new
           JTextField(30);
640.         tEmDe.setBounds(370, 290, 180, 25);
641.         c.add(tEmDe);
642.
643.         JLabel laEmCId=new JLabel("Enter
           the Organizational ID:");
644.         laEmCId.setBounds(180, 330, 160,
           25);
645.         c.add(laEmCId);
646.         JTextField tEmCId=new
           JTextField(30);
```

```

647.            tEmCId.setBounds(370, 330, 180,
        25);
648.            c.add(tEmCId);
649.
650.            JLabel laEmDt=new JLabel("Enter
        the Date of Joining:");
651.            laEmDt.setBounds(180, 370, 160,
        25);
652.            c.add(laEmDt);
653.            JTextField tEmDt=new
        JTextField(30);
654.            tEmDt.setBounds(370, 370, 180, 25);
655.            c.add(tEmDt);
656.
657.            JLabel laEmTm=new JLabel("Enter the
        Hours of Working:");
658.            laEmTm.setBounds(180, 410, 160,
        25);
659.            c.add(laEmTm);
660.            JTextField tEmTm=new
        JTextField(30);
661.            tEmTm.setBounds(370, 410, 180, 25);
662.            c.add(tEmTm);
663.
664.            JLabel laEmSa=new JLabel("Enter the
        Salary:");
665.            laEmSa.setBounds(180, 450, 160,
        25);
666.            c.add(laEmSa);
667.            JTextField tEmSa=new
        JTextField(30);
668.            tEmSa.setBounds(370, 450, 180, 25);
669.            c.add(tEmSa);
670.
671.            JButton bOK=new JButton("ok");
672.            bOK.setBounds(370, 490, 50, 25);
673.            c.add(bOK);
674.            bOK.setToolTipText("Press OK to
        Save Data");
675.            bOK.addActionListener(new
        ActionListener()
676.            {
677.                public void
        actionPerformed(ActionEvent ae)
678.                {
679.

```

```

680.
    if (tEmNam.getText().isEmpty() || tEmCId.getText().
        isEmpty() || tEmDe.getText().isEmpty() || tEmDt.getT
        ext().isEmpty() || tEmId.getText().isEmpty() || tEmS
        a.getText().isEmpty() || tEmTm.getText().isEmpty()
    )
681.                {
682.
        eDialog.setVisible(true);
683.
        JOptionPane.showMessageDialog(null,
            "Complete all the Text Fields");
684.                }
685.                else
686.                {
687.                    String
        nam=tEmNam.getText();
688.                    String
        id=tEmId.getText();
689.                    String
        de=tEmDe.getText();
690.                    String
        cid=tEmCId.getText();
691.                    String
        dt=tEmDt.getText();
692.                    String
        tm=tEmTm.getText();
693.                    String
        sa=tEmSa.getText();
694.
695.                try
696.                {
697.
698.                    ePW=new
        PrintWriter(new
        FileOutputStream("employeeData.txt", true));
699.                    String
        emData=nam+"-"+id+"-"+de+"-"+cid+"-"+dt+"-
        "+tm+"-"+sa;
700.
        ePW.println(emData);
701.                    ePW.flush();
702.                    ePW.close();
703.
        JOptionPane.showMessageDialog(null, "The Record
        of the Employee has been added successfully");

```

```

704.         eDialog.setVisible(false);
705.     }
706.     catch (FileNotFoundException e)
707.     {
708.         JOptionPane.showMessageDialog(null, "File not
found!");
709.     }
710.
711.     }
712.
713.     }
714.     }
715. );
716.
717.         JButton eCancel=new
JButton("Cancel");
718.         eCancel.setBounds(430, 490, 80,
25);
719.         c.add(eCancel);
720.         eCancel.setToolTipText("Cancel The
Dialog Box");
721.         eCancel.addActionListener(new
ActionListener()
722.         {
723.             public void
actionPerformed(ActionEvent ae)
724.             {
725.                 eDialog.setVisible(false);
726.             }
727.         }
728.     );
729.
730.     }//end of addEmp() method.
731.
732.
733.
734.     public void chkEmp(JFrame window)
735.     {
736.
737.
738.         JDialog eCDialog = new JDialog();
739.         eCDialog.setVisible(true);
740.         Container ec =
eCDialog.getContentPane();

```

```

741.         eCDialog.setSize(630,250);
742.         eCDialog.setTitle("Check the Data of
Employee");
743.
        eCDialog.setLocationRelativeTo(window);
744.         ec.setLayout(null);
745.         JLabel ecEmp=new JLabel("Enter the
ID Number of Employee:");
746.         ecEmp.setBounds(70, 70, 200, 25);
747.         ec.add(ecEmp);
748.         JTextField ecTEmp=new
JTextField(30);
749.         ecTEmp.setBounds(290, 70, 200, 25);
750.         ec.add(ecTEmp);
751.         JButton ok=new JButton("ok");
752.         ok.setBounds(290, 100,50, 25);
753.         ec.add(ok);
754.         ok.setToolTipText("Press ok to Check
the record of Employee");
755.         ok.addActionListener(new
ActionListener()
756.         {
757.             public void
actionPerformed(ActionEvent ae)
758.             {
759.                 String id=ecTEmp.getText();
760.                 try
761.                 {
762.                     buR=new
BufferedReader(new
FileReader("employeeData.txt"));
763.                     String l="";
764.
while((l=buR.readLine())!=null)
765.                     {
766.                         String
eArr[]=l.split("-");
767.
768.
769.                         if(eArr[1].equals(id))
770.                             {
771.
772.
773.
System.out.println("equal");

```

```

774.                                     /*Shows the
    output dialog*/
775.    eCDialog.setVisible(false);
776.                                     JDialog
    echkDialog=new JDialog();
777.
    echkDialog.setVisible(true);
778.                                     Container
    c=echkDialog.getContentPane();
779.
    echkDialog.setSize(400,500);
780.
    echkDialog.setTitle("Check The Data of
    Employee");
781.
    echkDialog.setLocationRelativeTo(window);
782.
    c.setLayout(null);
783.
784.                                     JLabel eNam=new
    JLabel("Name of Employee:");
785.
    eNam.setBounds(80, 80, 180, 25);
786.                                     c.add(eNam);
787.                                     JLabel eONam=new
    JLabel(eArr[0]);
788.
    eONam.setBounds(200, 80, 180, 25);
789.                                     c.add(eONam);
790.
791.                                     JLabel eID=new
    JLabel("ID of Employee:");
792.
    eID.setBounds(80, 120, 180, 25);
793.                                     c.add(eID);
794.                                     JLabel eOID=new
    JLabel(eArr[1]);
795.
    eOID.setBounds(200, 120, 180, 25);
796.                                     c.add(eOID);
797.
798.                                     JLabel eDep=new
    JLabel("Department:");
799.
    eDep.setBounds(80, 160, 180, 25);
800.                                     c.add(eDep);

```

```

801.                                JLabel eODep=new
      JLabel(eArr[2]);
802.      eODep.setBounds(200, 160, 180, 25);
803.                                c.add(eODep);
804.
805.                                JLabel eEOID=new
      JLabel("Organizational ID:");
806.      eEOID.setBounds(80, 200, 180, 25);
807.                                c.add(eEOID);
808.                                JLabel eOOID=new
      JLabel(eArr[3]);
809.      eOOID.setBounds(200, 200, 180, 25);
810.                                c.add(eOOID);
811.
812.                                JLabel eDOJ=new
      JLabel("Date of Joining:");
813.      eDOJ.setBounds(80, 280, 180, 25);
814.                                c.add(eDOJ);
815.                                JLabel eODOJ=new
      JLabel(eArr[4]);
816.      eODOJ.setBounds(200, 280, 180, 25);
817.                                c.add(eODOJ);
818.
819.                                JLabel eHoW=new
      JLabel("Hours of Working:");
820.      eHoW.setBounds(80, 240, 180, 25);
821.                                c.add(eHoW);
822.                                JLabel eOHoW=new
      JLabel(eArr[5]);
823.      eOHoW.setBounds(200, 240, 180, 25);
824.                                c.add(eOHoW);
825.
826.                                JLabel eSal=new
      JLabel("Salary:");
827.      eSal.setBounds(80, 320, 180, 25);
828.                                c.add(eSal);
829.                                JLabel eOSal=new
      JLabel(eArr[6]);

```

```

830.        eOSal.setBounds(200, 320, 180, 25);
831.                c.add(eOSal);
832.
833.                JButton eOK=new
        JButton("ok");
834.
        eOK.setBounds(200, 360, 50, 25);
835.                c.add(eOK);
836.
        eOK.setToolTipText("Exit the Output Dialog");
837.
        eOK.addActionListener(new ActionListener()
838.                {
839.                        public void
        actionPerformed(ActionEvent ae)
840.                                {
841.
        echkDialog.setVisible(false);
842.                                }
843.                                }
844.                                );
845.
846.                                }
847.                                else
848.                                {
849.
        JOptionPane.showMessageDialog(null,"The Data of
        the ID is not Found!");
850.                                }
851.                                }
852.                                buR.close();
853.                                }
854.                                catch(IOException io)
855.                                {
856.
        JOptionPane.showMessageDialog(null,"Input Output
        Error during File Reading");
857.                                }
858.
859.
860.                                }
861.                                }
862.                                );
863.
864.                JButton eCancel=new
        JButton("Cancel");

```



```

865.            eCancel.setBounds(350, 100, 80,
25);
866.            ec.add(eCancel);
867.            eCancel.setToolTipText("Cancel The
Dialog Box");
868.            eCancel.addActionListener(new
ActionListener()
869.            {
870.                public void
actionPerformed(ActionEvent ae)
871.                {
872.                    eCDialog.setVisible(false);
873.                }
874.            }
875.            );
876.
877.
878.        }
879.        public void delEmp(JFrame window)
880.        {
881.            JDialog edelDialog = new JDialog();
882.            edelDialog.setVisible(true);
883.            Container c =
edelDialog.getContentPane();
884.            edelDialog.setSize(630,250);
885.            edelDialog.setTitle("Delete the Data
of Employee");
886.
            edelDialog.setLocationRelativeTo(window);
887.            c.setLayout(null);
888.
889.            JLabel dPass=new JLabel("Enter the
ID Number of Employee:");
890.            dPass.setBounds(70, 70, 200, 25);
891.            c.add(dPass);
892.            JTextField dTPass=new
JTextField(30);
893.            dTPass.setBounds(290, 70, 200, 25);
894.            c.add(dTPass);
895.            JButton dButton=new JButton("ok");
896.            dButton.setBounds(290, 100,50, 25);
897.            c.add(dButton);
898.            dButton.setToolTipText("Press ok to
delete the Record!");
899.            dButton.addActionListener(new
ActionListener()
900.            {

```

```

901.         public void
          actionPerformed(ActionEvent ae)
902.         {
903.             if (dTPass.getText().isEmpty())
904.             {
905.                 edelDialog.setVisible(true);
906.                 JOptionPane.showMessageDialog(null, "Empty
Fields, Please Enter Valid ID");
907.             }
908.             else
909.             {
910.                 String
          id=dTPass.getText();
911.
912.                 try
913.                 {
914.                     BufferedReader
          dBr=new BufferedReader(new
          FileReader("employeeData.txt"));
915.                     StringBuffer sBu=new
          StringBuffer();
916.
917.                     String line,nline;
918.                     while((line=dBr.readLine())!=null)
919.                     {
920.                         if(line.contains(id))
921.                         {
922.                             nline=line.replace(line,"");
923.                             line=nline;
924.                         }
925.                         if(line.equals(""))
926.                         {
927.
928.                         }
929.                         else
930.                         {
931.
          sBu.append(line+"\n");
932.
          edelDialog.setVisible(true);

```

```

933.      JOptionPane.showMessageDialog(null, "Data
      Relative to ID is not Found");
934.                      break;
935.                      }
936.                  }
937.                  dBr.close();
938.                  FileWriter f=new
      FileWriter(new File("employeeData.txt"), false);
939.                  PrintWriter p=new
      PrintWriter(f);
940.                  p.print(sBu.toString());
941.                  f.flush();
942.                  f.close();
943.      JOptionPane.showMessageDialog(null, "The Data
      has been deleted successfully");
944.      edelDialog.setVisible(false);
945.                  }
946.
947.                  catch (IOException e)
948.                  {
949.
950.      edelDialog.setVisible(true);
951.      JOptionPane.showMessageDialog(null, "Input
      Output error during file Reading");
952.                  }
953.                  }
954.                  }
955.                  }
956.                  );
957.
958.      JButton eCancel=new
      JButton("Cancel");
959.      eCancel.setBounds(350, 100, 80, 25);
960.      c.add(eCancel);
961.      eCancel.setToolTipText("Cancel The
      Dialog Box");
962.      eCancel.addActionListener(new
      ActionListener()
963.      {
964.          public void
      actionPerformed(ActionEvent ae)
965.          {

```

```

966.         edelDialog.setVisible(false);
967.
968.             }
969.         }
970.     );
971.
972.     }
973.
974.     public void modEmployee(JFrame window)
975.     {
976.         JDialog edelDialog = new JDialog();
977.         edelDialog.setVisible(true);
978.         Container c =
979.             edelDialog.getContentPane();
980.         edelDialog.setSize(630,250);
981.         edelDialog.setTitle("Delete the Data
of Employee");
982.         edelDialog.setLocationRelativeTo(window);
983.         c.setLayout(null);
984.         JLabel dPass=new JLabel("Enter the
ID Number of Employee:");
985.         dPass.setBounds(70, 70, 200, 25);
986.         c.add(dPass);
987.         JTextField dTPass=new
JTextField(30);
988.         dTPass.setBounds(290, 70, 200, 25);
989.         c.add(dTPass);
990.         JButton dButton=new JButton("ok");
991.         dButton.setBounds(290, 100,50, 25);
992.         c.add(dButton);
993.         dButton.setToolTipText("Press ok to
delete the Record!");
994.         dButton.addActionListener(new
ActionListener()
995.         {
996.             public void
actionPerformed(ActionEvent ae)
997.             {
998.
999.                 if(dTPass.getText().isEmpty())
1000.                 {

```

```

1001.        JOptionPane.showMessageDialog(null, "Empty
        Fields, Please Enter Valid ID");
1002.    }
1003.    else
1004.    {
1005.        String
        id=dTPass.getText();
1006.
1007.        try
1008.        {
1009.            BufferedReader
            dBr=new BufferedReader(new
            FileReader("employeData.txt"));
1010.            StringBuffer sBu=new
            StringBuffer();
1011.
1012.            String line,nline;
1013.            while((line=dBr.readLine())!=null)
1014.            {
1015.                if(line.contains(id))
1016.                {
1017.                    nline=line.replace(line,"");
1018.                    line=nline;
1019.                }
1020.                if(line.equals(""))
1021.                {
1022.                }
1023.            }
1024.            else
1025.            {
1026.                sBu.append(line+"\n");
1027.                edelDialog.setVisible(true);
1028.                JOptionPane.showMessageDialog(null, "Data
                Relative to ID is not Found");
1029.                break;
1030.            }
1031.        }
1032.        dBr.close();
1033.        FileWriter f=new
        FileWriter(new File("employeData.txt"), false);

```

```

1034.                PrintWriter p=new
        PrintWriter(f);
1035.                p.print(sBu.toString());
1036.                f.flush();
1037.                f.close();
1038.
        //JOptionPane.showMessageDialog(null, "The Data
        has been deleted successfully");
1039.
        edelDialog.setVisible(false);
1040.
1041.                /* Witing Data*/
1042.                JDialog eDialog = new JDialog();
1043.                eDialog.setVisible(true);
1044.                eDialog.setSize(705, 630);
1045.                Container e =
        eDialog.getContentPane();
1046.                eDialog.setTitle("Add the Data of
        Employees");
1047.
        eDialog.setLocationRelativeTo(window);
1048.                e.setLayout(null);
1049.
1050.                JLabel laEmNam=new JLabel("Enter
        the Name of Employee:");
1051.                laEmNam.setBounds(180, 210, 180,
        25);
1052.                e.add(laEmNam);
1053.                JTextField tEmNam=new
        JTextField(30);
1054.                tEmNam.setBounds(370, 210, 180,
        25);
1055.                e.add(tEmNam);
1056.
1057.                JLabel laEmId=new JLabel("Enter the
        ID of Employee:");
1058.                laEmId.setBounds(180, 250, 160,
        25);
1059.                e.add(laEmId);
1060.                JTextField tEmId=new
        JTextField(30);
1061.                tEmId.setBounds(370, 250, 180, 25);
1062.                e.add(tEmId);
1063.
1064.                JLabel laEmDe=new JLabel("Enter the
        Department:");

```

```
1065.            laEmDe.setBounds(180, 290, 160,
25);
1066.            e.add(laEmDe);
1067.            JTextField tEmDe=new
JTextField(30);
1068.            tEmDe.setBounds(370, 290, 180, 25);
1069.            e.add(tEmDe);
1070.
1071.            JLabel laEmCId=new JLabel("Enter
the Organizational ID:");
1072.            laEmCId.setBounds(180, 330, 160,
25);
1073.            e.add(laEmCId);
1074.            JTextField tEmCId=new
JTextField(30);
1075.            tEmCId.setBounds(370, 330, 180,
25);
1076.            e.add(tEmCId);
1077.
1078.            JLabel laEmDt=new JLabel("Enter
the Date of Joining:");
1079.            laEmDt.setBounds(180, 370, 160,
25);
1080.            e.add(laEmDt);
1081.            JTextField tEmDt=new
JTextField(30);
1082.            tEmDt.setBounds(370, 370, 180, 25);
1083.            e.add(tEmDt);
1084.
1085.            JLabel laEmTm=new JLabel("Enter the
Hours of Working:");
1086.            laEmTm.setBounds(180, 410, 160,
25);
1087.            e.add(laEmTm);
1088.            JTextField tEmTm=new
JTextField(30);
1089.            tEmTm.setBounds(370, 410, 180, 25);
1090.            e.add(tEmTm);
1091.
1092.            JLabel laEmSa=new JLabel("Enter the
Salary:");
1093.            laEmSa.setBounds(180, 450, 160,
25);
1094.            e.add(laEmSa);
1095.            JTextField tEmSa=new
JTextField(30);
1096.            tEmSa.setBounds(370, 450, 180, 25);
```

```

1097.            e.add(tEmSa);
1098.
1099.            JButton bOK=new JButton("ok");
1100.            bOK.setBounds(370, 490, 50, 25);
1101.            e.add(bOK);
1102.            bOK.setToolTipText("Press OK to
Save Data");
1103.            bOK.addActionListener(new
ActionListener()
1104.            {
1105.                public void
actionPerformed(ActionEvent ae)
1106.                {
1107.
1108.                    if(tEmNam.getText().isEmpty()||tEmCId.getText().
isEmpty()||tEmDe.getText().isEmpty()||tEmDt.getT
ext().isEmpty()||tEmId.getText().isEmpty()||tEmS
a.getText().isEmpty()||tEmTm.getText().isEmpty()
)
1109.                    {
1110.                        eDialog.setVisible(true);
1111.                        JOptionPane.showMessageDialog(null,
"Complete all the Text Fields");
1112.                    }
1113.                    else
1114.                    {
1115.                        String
nam=tEmNam.getText();
1116.                        String
id=tEmId.getText();
1117.                        String
de=tEmDe.getText();
1118.                        String
cid=tEmCId.getText();
1119.                        String
dt=tEmDt.getText();
1120.                        String
tm=tEmTm.getText();
1121.                        String
sa=tEmSa.getText();
1122.
1123.                        try
1124.                        {
1125.

```



```

1126.                                     ePW=new
        PrintWriter(new
        FileOutputStream("employeeData.txt", true));
1127.                                     String
        emData=nam+"-"+id+"-"+de+"-"+cid+"-"+dt+"-
        "+tm+"-"+sa;
1128.
        ePW.println(emData);
1129.                                     ePW.flush();
1130.                                     ePW.close();
1131.
        JOptionPane.showMessageDialog(null, "The Record
        of the Employee has been added successfully");
1132.
        eDialog.setVisible(false);
1133.                                     }
1134.
        catch(FileNotFoundException e)
1135.                                     {
1136.
        JOptionPane.showMessageDialog(null, "File not
        found!");
1137.                                     }
1138.
1139.                                     }
1140.
1141.                                     }
1142.                                     }
1143.                                     );
1144.                                     }
1145.
1146.                                     catch(IOException e)
1147.                                     {
1148.
        edelDialog.setVisible(true);
1149.
        JOptionPane.showMessageDialog(null, "Input
        Output error during file Reading");
1150.                                     }
1151.
1152.                                     }
1153.                                     }
1154.                                     }
1155.                                     );
1156.
1157.
1158.                                     }

```

1159. }

cateringMangment class:

```
import java.awt.*;
import java.awt.Container;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.GridLayout;
import java.awt.Font;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JDialog;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;
import java.io.*;
import java.util.*;
import javax.swing.*;
public class cateringManagment
{
    BufferedReader buR;
    public PrintWriter ePW=null;
    public void addCat(JFrame window)
    {

        JDialog eDialog = new JDialog();
        eDialog.setVisible(true);
        Container c = eDialog.getContentPane();
        eDialog.setSize(705, 630);
        eDialog.setTitle("Add's the Food Menu for a
Flight");
        eDialog.setLocationRelativeTo(window);
        c.setLayout(null);

        JLabel laEmNam=new JLabel("Enter the Name of
Food Menu:");
```

```

        laEmNam.setBounds(180, 210, 180, 25);
        c.add(laEmNam);
        JTextField tEmNam=new JTextField(30);
        tEmNam.setBounds(370, 210, 180, 25);
        c.add(tEmNam);

        JLabel laEmId=new JLabel("Enter the ID of Food
Item:");
        laEmId.setBounds(180, 250, 160, 25);
        c.add(laEmId);
        JTextField tEmId=new JTextField(30);
        tEmId.setBounds(370, 250, 180, 25);
        c.add(tEmId);

        JLabel laEmCid=new JLabel("Enter the Flight
Number:");
        laEmCid.setBounds(180, 330, 160, 25);
        c.add(laEmCid);
        JTextField tEmCid=new JTextField(30);
        tEmCid.setBounds(370, 330, 180, 25);
        c.add(tEmCid);

        JButton bOK=new JButton("ok");
        bOK.setBounds(370, 490, 50, 25);
        c.add(bOK);
        bOK.setToolTipText("Press OK to Save Data");
        bOK.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent
ae)
            {

if (tEmNam.getText().isEmpty() || tEmCid.getText().isEmpty()
|| tEmId.getText().isEmpty())
            {
                eDialog.setVisible(true);

                JOptionPane.showMessageDialog(null, "Complete all the
Text Fields");
            }
            else
            {
                String nam=tEmNam.getText();
                String id=tEmId.getText();
                String cid=tEmCid.getText();

```

```

        try
        {
            ePW=new PrintWriter(new
FileOutputStream("cateringData.txt",true));
            String emData=nam+"-"+id+"-
"+cid;

            ePW.println(emData);
            ePW.flush();
            ePW.close();

JOptionPane.showMessageDialog(null, "The Record of the
Catering has been added successfully");
            eDialog.setVisible(false);
        }
        catch (FileNotFoundException e)
        {

JOptionPane.showMessageDialog(null, "File not found!");
        }

    }

    }

);

JButton eCancel=new JButton("Cancel");
eCancel.setBounds(430, 490, 80, 25);
c.add(eCancel);
eCancel.setToolTipText("Cancel The Dialog Box");
eCancel.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        eDialog.setVisible(false);
    }
}
);

} //end of addEmp() method.

public void chkCat(JFrame window)
{

```

```

JDialog eCDialog = new JDialog();
    eCDialog.setVisible(true);
    Container ec = eCDialog.getContentPane();
    eCDialog.setSize(630,250);
eCDialog.setTitle("Check the Data of Catering");
    eCDialog.setLocationRelativeTo(window);
    ec.setLayout(null);
JLabel ecEmp=new JLabel("Enter the ID of Food
Menu:");
    ecEmp.setBounds(70, 70, 200, 25);
    ec.add(ecEmp);
JTextField ecTEmp=new JTextField(30);
ecTEmp.setBounds(290, 70, 200, 25);
    ec.add(ecTEmp);
JButton ok=new JButton("ok");
ok.setBounds(290, 100,50, 25);
    ec.add(ok);
ok.setToolTipText("Press ok to Check the record
of Food Menu");
ok.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        String id=ecTEmp.getText();
        try
        {
            buR=new BufferedReader(new
FileReader("cateringData.txt"));
            String l="";
            while((l=buR.readLine())!=null)
            {
                String eArr[]=l.split("-");

                if(eArr[1].equals(id))
                {

                    System.out.println("equal");
                    /*Shows the output dialog*/
                    eCDialog.setVisible(false);
                    JDialog echkDialog=new
JDialog();

                    echkDialog.setVisible(true);
                    Container
c=echkDialog.getContentPane();
                    echkDialog.setSize(400,500);

```

```

        echkDialog.setTitle("Check
The Data of Food Menu");

echkDialog.setLocationRelativeTo(window);
        c.setLayout(null);

        JLabel eNam=new JLabel("Name
of Food Menu:");
        eNam.setBounds(80, 80, 180,
25);
        c.add(eNam);
        JLabel eONam=new
JLabel(eArr[0]);
        eONam.setBounds(200, 80, 180,
25);
        c.add(eONam);

        JLabel eID=new JLabel("ID of
Food Menu:");
        eID.setBounds(80, 120, 180,
25);
        c.add(eID);
        JLabel eOID=new
JLabel(eArr[1]);
        eOID.setBounds(200, 120, 180,
25);
        c.add(eOID);

        JLabel eEOID=new
JLabel("Flight Number:");
        eEOID.setBounds(80, 200, 180,
25);
        c.add(eEOID);
        JLabel eOOID=new
JLabel(eArr[2]);
        eOOID.setBounds(200, 200,
180, 25);
        c.add(eOOID);

        JButton eOK=new
JButton("ok");
        eOK.setBounds(200, 360, 50,
25);
        c.add(eOK);
        eOK.setToolTipText("Exit the
Output Dialog");

```

```

                                eOK.addActionListener(new
ActionListener()
                                {
                                public void
actionPerformed(ActionEvent ae)
                                {

echkDialog.setVisible(false);
                                }
                                }
                                );

                                }
                                else
                                {

JOptionPane.showMessageDialog(null, "The Data of the Food
ID is not Found!");
                                }
                                }
                                buR.close();
                                }
                                catch (IOException io)
                                {

JOptionPane.showMessageDialog(null, "Input Output Error
during File Reading");
                                }

                                }
                                }
                                );

JButton eCancel=new JButton("Cancel");
eCancel.setBounds(350, 100, 80, 25);
ec.add(eCancel);
eCancel.setToolTipText("Cancel The Dialog Box");
eCancel.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        eCDialog.setVisible(false);
    }
}
);

```



```

String line,nline;
while((line=dBr.readLine())!=null)
{
    if(line.contains(id))
    {
        nline=line.replace(line,"");
        line=nline;
    }
    if(line.equals(""))
    {
    }
    else
    {
        sBu.append(line+"\n");
        edelDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "Data Relative to ID
is not Found");

        break;
    }
}
dBr.close();
FileWriter f=new FileWriter(new
File("cateringData.txt"),false);
PrintWriter p=new PrintWriter(f);
p.print(sBu.toString());
f.flush();
f.close();
JOptionPane.showMessageDialog(null,
"The Data has been deleted successfully");
edelDialog.setVisible(false);
}

catch(IOException e)
{
    edelDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "Input Output error
during file Reading");
}

}

}

);

```

```

        JButton eCancel=new JButton("Cancel");
        eCancel.setBounds(350, 100, 80, 25);
        c.add(eCancel);
        eCancel.setToolTipText("Cancel The Dialog Box");
        eCancel.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent ae)
            {
                edelDialog.setVisible(false);
            }
        }
    );
}
}

```

1160. ticketManagment:

```

import java.awt.*;
import java.awt.Container;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.GridLayout;
import java.awt.Font;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JDialog;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;
import java.io.*;
import java.util.*;
import javax.swing.*;
public class ticketsData
{
    BufferedReader buR;
    public PrintWriter ePW=null;
    public void addTic(JFrame window)
    {

```

```

JDialog eDialog = new JDialog();
eDialog.setVisible(true);
Container c = eDialog.getContentPane();
eDialog.setSize(705, 630);
eDialog.setTitle("Add's the Tickets for a
Flight:");
eDialog.setLocationRelativeTo(window);
c.setLayout(null);

JLabel laEmNam=new JLabel("Total Number of
Tickets:");
laEmNam.setBounds(180, 210, 180, 25);
c.add(laEmNam);
JTextField tEmNam=new JTextField(30);
tEmNam.setBounds(370, 210, 180, 25);
c.add(tEmNam);

JLabel laEmId=new JLabel("Enter Reserved
Tickets:");
laEmId.setBounds(180, 250, 160, 25);
c.add(laEmId);
JTextField tEmId=new JTextField(30);
tEmId.setBounds(370, 250, 180, 25);
c.add(tEmId);

JLabel laEmCId=new JLabel("Enter the Available
Tickets:");
laEmCId.setBounds(180, 330, 160, 25);
c.add(laEmCId);
JTextField tEmCId=new JTextField(30);
tEmCId.setBounds(370, 330, 180, 25);
c.add(tEmCId);

JLabel laTF1=new JLabel("Enter the Flight
Number:");
laTF1.setBounds(180, 290, 160, 25);
c.add(laTF1);
JTextField latTF1=new JTextField(30);
latTF1.setBounds(370, 290, 180, 25);
c.add(latTF1);

JButton bOK=new JButton("ok");
bOK.setBounds(370, 490, 50, 25);
c.add(bOK);
bOK.setToolTipText("Press OK to Save Data");

```

```

        bOK.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent
ae)
            {

if (tEmNam.getText().isEmpty() || tEmCId.getText().isEmpty()
|| tEmId.getText().isEmpty())
            {
                eDialog.setVisible(true);

                JOptionPane.showMessageDialog(null, "Complete all the
Text Fields");
            }
            else
            {
                String nam=tEmNam.getText();
                String id=tEmId.getText();
                String cid=tEmCId.getText();
                String fl=latTFl.getText();

                try
                {

                    ePW=new PrintWriter(new
FileOutputStream("ticketData.txt", true));
                    String emData=fl+"-"+nam+"-
"+id+"-"+cid;

                    ePW.println(emData);
                    ePW.flush();
                    ePW.close();

                    JOptionPane.showMessageDialog(null, "The Record of the
Tickets has been added successfully");
                    eDialog.setVisible(false);
                }
                catch (FileNotFoundException e)
                {

                    JOptionPane.showMessageDialog(null, "File not found!");
                }

            }

        }
    }
}

```

```

    );

    JButton eCancel=new JButton("Cancel");
    eCancel.setBounds(430, 490, 80, 25);
    c.add(eCancel);
    eCancel.setToolTipText("Cancel The Dialog Box");
    eCancel.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent ae)
        {
            eDialog.setVisible(false);
        }
    });

}

} //end of addEmp() method.

public void chkTick(JFrame window)
{

    JDialog eCDialog = new JDialog();
    eCDialog.setVisible(true);
    Container ec = eCDialog.getContentPane();
    eCDialog.setSize(630,250);
    eCDialog.setTitle("Check the Data of Tickets");
    eCDialog.setLocationRelativeTo(window);
    ec.setLayout(null);
    JLabel ecEmp=new JLabel("Enter the Flight
Number:");
    ecEmp.setBounds(70, 70, 200, 25);
    ec.add(ecEmp);
    JTextField ecTEmp=new JTextField(30);
    ecTEmp.setBounds(290, 70, 200, 25);
    ec.add(ecTEmp);
    JButton ok=new JButton("ok");
    ok.setBounds(290, 100,50, 25);
    ec.add(ok);
    ok.setToolTipText("Press ok to Check the record
of Tickets");
    ok.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent ae)
        {
            String id=ecTEmp.getText();

```

```

        try
        {
            buR=new BufferedReader(new
FileReader("ticketData.txt"));
            String l="";
            while((l=buR.readLine())!=null)
            {
                String eArr[]=l.split("-");
                eCDialog.setVisible(false);
                if(eArr[0].equals(id))
                {

                    JDialog echkDialog=new
JDialog();

                    echkDialog.setVisible(true);
                    Container
c=echkDialog.getContentPane();
                    echkDialog.setSize(400,500);
                    echkDialog.setTitle("Check
The Data of Tickets");

                    echkDialog.setLocationRelativeTo(window);
                    c.setLayout(null);

                    JLabel lNumTic=new
JLabel("Number of Tickets:");
                    lNumTic.setBounds(80, 80,
180, 25);

                    c.add(lNumTic);
                    JLabel lONumTic=new
JLabel(eArr[1]);
                    lONumTic.setBounds(200, 80,
180, 25);

                    c.add(lONumTic);

                    JLabel eID=new
JLabel("Reserved Tickets:");
                    eID.setBounds(80, 120, 180,
25);

                    c.add(eID);
                    JLabel eOID=new
JLabel(eArr[2]);
                    eOID.setBounds(200, 120, 180,
25);

                    c.add(eOID);

```

```

JLabel("Flight Number:");
180, 25);

JLabel(eArr[0]);
180, 25);

JLabel("Available Tickets:");
25);

JLabel(eArr[3]);
180, 25);

JButton("ok");
25);

Output Dialog");
ActionListener()
{
    public void
    actionPerformed(ActionEvent ae)
    {
        echkDialog.setVisible(false);
    }
}

else
{

//JOptionPane.showMessageDialog(null,"The Data of the
ticket ID is not Found!");

```

```

        }
    }
    buR.close();
}
catch(IOException io)
{

JOptionPane.showMessageDialog(null,"Input Output Error
during File Reading");
}

}
}
);

JButton eCancel=new JButton("Cancel");
eCancel.setBounds(350, 100, 80, 25);
ec.add(eCancel);
eCancel.setToolTipText("Cancel The Dialog Box");
eCancel.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        eCDialog.setVisible(false);
    }
});

}

}

```

1161. securityStaff class:

```

import java.awt.*;
import java.awt.Container;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.GridLayout;
import java.awt.Font;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JDialog;
import javax.swing.JFrame;

```



```

import javax.swing.JLabel;
import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;
import java.io.*;
import java.util.*;
import javax.swing.*;
public class secData
{
    BufferedReader buR;
    public FileWriter pFW=null;
    public PrintWriter pPW=null;
    public void addSecData(JFrame mainframe)
    {
        JDialog pDialog = new JDialog();
        pDialog.setVisible(true);
        Container c = pDialog.getContentPane();
        pDialog.setSize(765, 730);
        pDialog.setTitle("Add the Data of Security
Personal");
        pDialog.setLocationRelativeTo(mainframe);
        c.setLayout(null);

        JLabel laPaNam=new JLabel("Security Employe:");
        laPaNam.setBounds(180, 210, 180, 25);
        c.add(laPaNam);
        JTextField tPaNam=new JTextField(30);
        tPaNam.setBounds(370, 210, 180, 25);
        c.add(tPaNam);

        JLabel laPaId=new JLabel("ID of Security
Employe:");
        laPaId.setBounds(180, 250, 160, 25);
        c.add(laPaId);
        JTextField tPaId=new JTextField(30);
        tPaId.setBounds(370, 250, 180, 25);
        c.add(tPaId);

        JLabel laPaFl=new JLabel("Enter the Phone
Number:");
        laPaFl.setBounds(180, 290, 160, 25);
        c.add(laPaFl);
        JTextField tPaFl=new JTextField(30);

```

```

tPaFl.setBounds(370, 290, 180, 25);
c.add(tPaFl);

JLabel laPaAi=new JLabel("Enter the Address:");
laPaAi.setBounds(180, 330, 160, 25);
c.add(laPaAi);
JTextField tPaAi=new JTextField(30);
tPaAi.setBounds(370, 330, 180, 25);
c.add(tPaAi);

JLabel laPatm=new JLabel("Hours of Working
HH:MM/HH:MM");
laPatm.setBounds(180, 370, 160, 25);
c.add(laPatm);
JTextField tPatm=new JTextField(30);
tPatm.setBounds(370, 370, 180, 25);
c.add(tPatm);

JButton bOK=new JButton("ok");
bOK.setBounds(370, 450, 50, 25);
c.add(bOK);
bOK.setToolTipText("Press OK to Save Data");
bOK.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent
ae)
    {

if(tPaNam.getText().isEmpty()||tPaAi.getText().isEmpty()|
|tPatm.getText().isEmpty()||tPaFl.getText().isEmpty()||tP
aId.getText().isEmpty())
    {
        pDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "Complete all the
Text Fields");
    }
    else
    {
        String nam=tPaNam.getText();
        String id=tPaId.getText();
        String ai=tPaAi.getText();
        String fl=tPaFl.getText();
        String tm=tPatm.getText();

```

```

        try
        {
            // pFW=new FileWriter(new
File("passangerData.txt"));

            pPW=new PrintWriter(new
FileOutputStream("securityData.txt",true));
            String paData=nam+"-"+id+"-
"+ai+"-"+fl+"-"+tm;

            pPW.println(paData);
            pPW.flush();
            pPW.close();

JOptionPane.showMessageDialog(null, "The Record of the
Security Employee has been added successfully");
            pDialog.setVisible(false);
        }
        catch(FileNotFoundException e)
        {
            pDialog.setVisible(true);
JOptionPane.showMessageDialog(null, "File not found!");
        }

    }

}

);

JButton pCancel=new JButton("Cancel");
pCancel.setBounds(430, 450, 80, 25);
c.add(pCancel);
pCancel.setToolTipText("Cancel The Dialog Box");
pCancel.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        pDialog.setVisible(false);
    }
});

}

//end of method addPassangerData()

public void chkSecurity(JFrame window)
{

```

```

JDialog pCDialog = new JDialog();
    pCDialog.setVisible(true);
    Container pc = pCDialog.getContentPane();
    pCDialog.setSize(630,250);
    pCDialog.setTitle("Check the Data of Security
Data");
        pCDialog.setLocationRelativeTo(window);
        pc.setLayout(null);
        JLabel pcEmp=new JLabel("ID Number of Security
Personal:");
        pcEmp.setBounds(70, 70, 200, 25);
        pc.add(pcEmp);
        JTextField pcTPas=new JTextField(30);
        pcTPas.setBounds(290, 70, 200, 25);
        pc.add(pcTPas);
        JButton ok=new JButton("ok");
        ok.setBounds(290, 100,50, 25);
        pc.add(ok);
        ok.setToolTipText("Press ok to Check the record
of Security Personal");
        ok.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent ae)
            {
                String id=pcTPas.getText();
                try
                {
                    buR=new BufferedReader(new
FileReader("securityData.txt"));
                    String l="";
                    while((l=buR.readLine())!=null)
                    {
                        String pArr[]=l.split("-");

                        if(pArr[1].equals(id))
                        {

                            /*Shows the output dialog*/
                            pCDialog.setVisible(false);
                            JDialog pchkDialog=new
JDialog();

                            pchkDialog.setVisible(true);
                            Container
c=pchkDialog.getContentPane();

```

```

pchkDialog.setSize(400,500);
pchkDialog.setTitle("Check
The Data of Passanger");

pchkDialog.setLocationRelativeTo(window);
c.setLayout(null);

JLabel pNam=new JLabel("Name
:");
pNam.setBounds(80, 80, 180,
25);
c.add(pNam);
JLabel pONam=new
JLabel(pArr[0]);
pONam.setBounds(200, 80, 180,
25);
c.add(pONam);

JLabel pID=new JLabel("ID of
Employee:");
pID.setBounds(80, 120, 180,
25);
c.add(pID);
JLabel pOID=new
JLabel(pArr[1]);
pOID.setBounds(200, 120, 180,
25);
c.add(pOID);

JLabel pDep=new JLabel("Phone
Number");
pDep.setBounds(80, 160, 180,
25);
c.add(pDep);
JLabel pODep=new
JLabel(pArr[3]);
pODep.setBounds(200, 160,
180, 25);
c.add(pODep);

JLabel pDOJ=new
JLabel("Address:");
pDOJ.setBounds(80, 200, 180,
25);
c.add(pDOJ);
JLabel pODOJ=new
JLabel(pArr[2]);

```

```

180, 25);

of Working:");
25);

JLabel (pArr[4]);
180, 25);

JButton("ok");
25);

Output Dialog");
ActionListener()
{
    public void
    actionPerformed(ActionEvent ae)
    {
        pchkDialog.setVisible(false);
    }
};

}
else
{

//JOptionPane.showMessageDialog(null,"The Data of the ID
is not Found!");

}
}
buR.close();
}
catch(IOException io)
{

```

```

JOptionPane.showMessageDialog(null, "Input Output Error
during File Reading");
    }

    }
}
);

JButton pCancel=new JButton("Cancel");
pCancel.setBounds(350, 100, 80, 25);
pc.add(pCancel);
pCancel.setToolTipText("Cancel The Dialog Box");
pCancel.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        pCDialog.setVisible(false);
    }
}
);

} //end of method chkPass()

public void delSecurity(JFrame window)
{
    JDialog pdelDialog = new JDialog();
    pdelDialog.setVisible(true);
    Container c = pdelDialog.getContentPane();
    pdelDialog.setSize(630,250);
    pdelDialog.setTitle("Delete the Data of Security
Employee");
    pdelDialog.setLocationRelativeTo(window);
    c.setLayout(null);

    JLabel dPass=new JLabel("ID Number:");
    dPass.setBounds(70, 70, 200, 25);
    c.add(dPass);
    JTextField dTPass=new JTextField(30);
    dTPass.setBounds(290, 70, 200, 25);
    c.add(dTPass);
    JButton dButton=new JButton("ok");
    dButton.setBounds(290, 100,50, 25);
    c.add(dButton);
    dButton.setToolTipText("Press ok to delete the
Record!");
}

```

```

dButton.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        if(dTPass.getText().isEmpty())
        {
            pdelDialog.setVisible(true);

            JOptionPane.showMessageDialog(null, "Empty
Fields, Please Enter Valid ID");
        }
        else
        {
            String id=dTPass.getText();

            try
            {
                BufferedReader dBr=new
BufferedReader(new FileReader("securityData.txt"));
                StringBuffer sBu=new
StringBuffer();

                String line,nline;
                while((line=dBr.readLine())!=null)
                {
                    if(line.contains(id))
                    {
                        nline=line.replace(line, "");
                        line=nline;
                    }
                    if(line.equals(""))
                    {
                    }
                    else
                    {
                        sBu.append(line+"\n");
                        pdelDialog.setVisible(true);

                        //JOptionPane.showMessageDialog(null, "Data Relative to
ID is not Found");

                        break;
                    }
                }
                dBr.close();
                FileWriter f=new FileWriter(new
File("securityData.txt"), false);

```



```

        PrintWriter p=new PrintWriter(f);
        p.print(sBu.toString());
        f.flush();
        f.close();
        JOptionPane.showMessageDialog(null,
"The Data has been deleted successfully");
        pdelDialog.setVisible(false);
    }

    catch(IOException e)
    {
        pdelDialog.setVisible(true);

JOptionPane.showMessageDialog(null, "Input Output error
during file Reading");
    }

    }
}
);

JButton pCancel=new JButton("Cancel");
pCancel.setBounds(350, 100, 80, 25);
c.add(pCancel);
pCancel.setToolTipText("Cancel The Dialog Box");
pCancel.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        pdelDialog.setVisible(false);
    }
}
);

} // end of method delPass().
}

```

1162. passwordFile class:

```

1163. import java.io.BufferedReader;
1164. import java.io.*;
1165.
1166.
1167. public class passwordFile{
1168.     public String u;
1169.     public String p;

```

```

1170.      {
1171.      try
1172.      {
1173.          BufferedReader istrm=new
      BufferedReader(new FileReader("pass.txt"));
1174.          u=istrm.readLine();
1175.          p=istrm.readLine();
1176.      }
1177.      catch(FileNotFoundException e)
1178.      {
1179.          System.out.println("File Not
      Found");
1180.      }
1181.      catch(IOException f)
1182.      {
1183.          System.out.println("IO");
1184.      }
1185.  }
1186.  }

```

1187. changePasword class:

```

1188.  import java.awt.*;
1189.  import java.io.*;
1190.  import java.util.*;
1191.  import javax.swing.*;
1192.  import java.awt.event.ActionListener;
1193.  import java.awt.event.ActionEvent;
1194.  public class changePasword
1195.  {
1196.      public void c_pass(JFrame window)
1197.      {
1198.          JDialog cPDialog = new JDialog();
1199.          cPDialog.setVisible(true);
1200.          Container c =
      cPDialog.getContentPane();
1201.          cPDialog.setSize(630,400);
1202.          cPDialog.setTitle("Change
      Password");
1203.
      cPDialog.setLocationRelativeTo(window);
1204.          c.setLayout(null);
1205.

```

```

1206.          JLabel oPwd=new JLabel("Enter the
old Password:");
1207.          oPwd.setBounds(70, 70, 200, 25);
1208.          c.add(oPwd);
1209.          JTextField oTPwd=new JTextField(30);
1210.          oTPwd.setBounds(290, 70, 200, 25);
1211.          c.add(oTPwd);
1212.
1213.          JLabel nPwd=new JLabel("Enter the
new Password");
1214.          nPwd.setBounds(70,100,200,25);
1215.          c.add(nPwd);
1216.          JTextField nTPwd=new JTextField(30);
1217.          nTPwd.setBounds(290, 100, 200, 25);
1218.          c.add(nTPwd);
1219.
1220.          JButton cPButton=new JButton("ok");
1221.          cPButton.setBounds(290, 130,50, 25);
1222.          c.add(cPButton);
1223.          cPButton.setToolTipText("Press ok to
change Password");
1224.
1225.          cPButton.addActionListener(new
ActionListener()
1226.          {
1227.              public void
actionPerformed(ActionEvent ae)
1228.              {
1229.                  try
1230.                  {
1231.                      BufferedReader istrm=new
BufferedReader(new FileReader("pass.txt"));
1232.                      String
usr=istrm.readLine();
1233.                      String
opds=istrm.readLine();
1234.                      istrm.close();
1235.
System.out.println(oTPwd.getText());
1236.                      PrintWriter
o_stream=null;
1237.                      try
1238.                      {
1239.                          if((oTPwd.getText().equals(opds)))
1240.                          {

```

```

1241.                                     o_stream=new
    PrintWriter(new FileOutputStream("pass.txt"));
1242.
    o_stream.println("admin");
1243.
    o_stream.println(nTPwd.getText());
1244.
    o_stream.flush();
1245.
    o_stream.close();
1246.
    JOptionPane.showMessageDialog(null,"Password is
    Changed Successfully");
1247.
    cPDialog.setVisible(false);
1248.                                     }
1249.                                     else
1250.                                     {
1251.
    JOptionPane.showMessageDialog(null,"Password did
    not match!");
1252.                                     }
1253.                                     }
1254.
    catch(FileNotFoundException e)
1255.    {
1256.
    JOptionPane.showMessageDialog(null,"File Not
    Found");
1257.    }
1258.
1259.
1260.    }
1261.    catch(IOException f)
1262.    {
1263.
    JOptionPane.showMessageDialog(null,"Input Output
    error During File Reading!");
1264.    }
1265.
1266.
1267.
1268.    }
1269.    }
1270.    );
1271.

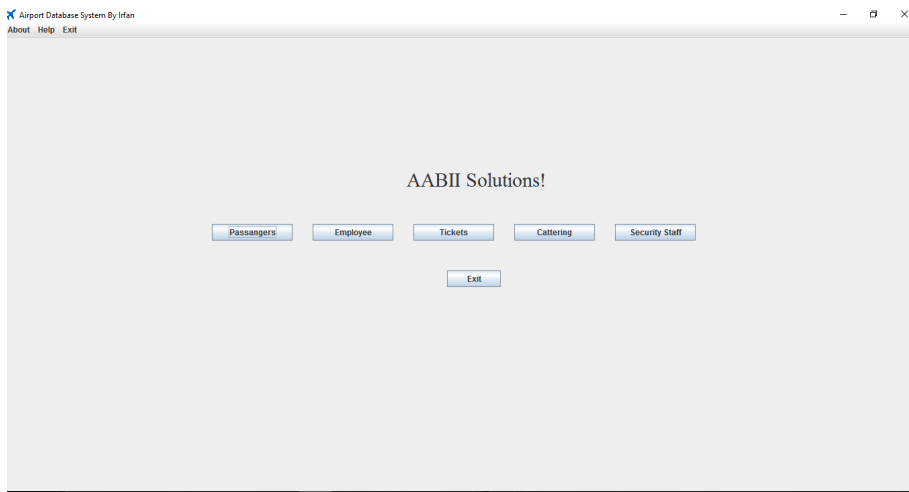
```

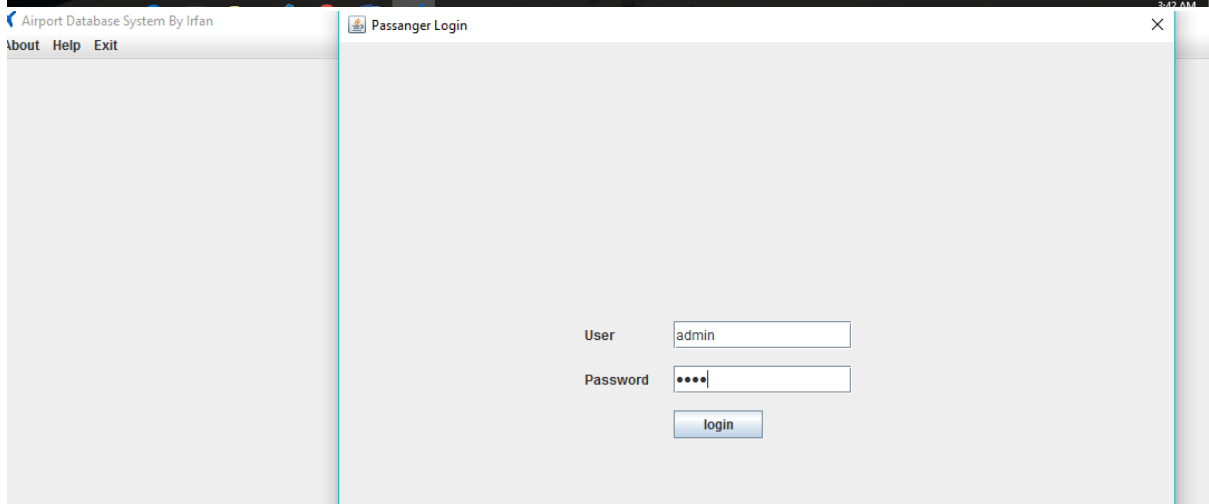
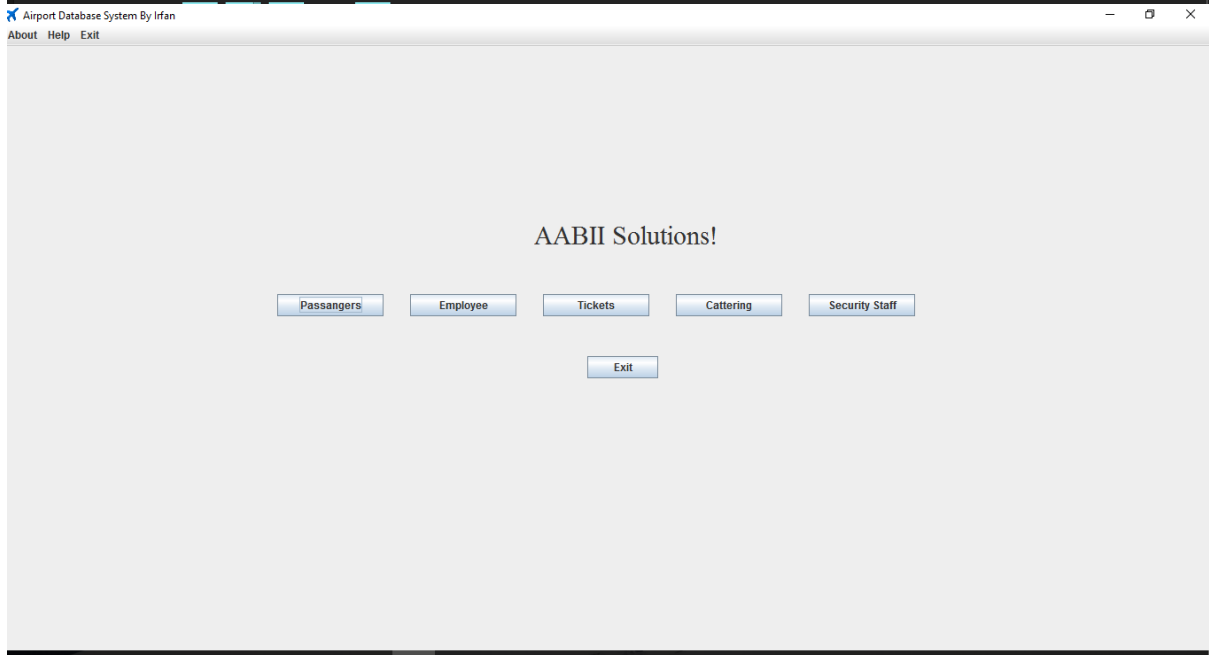
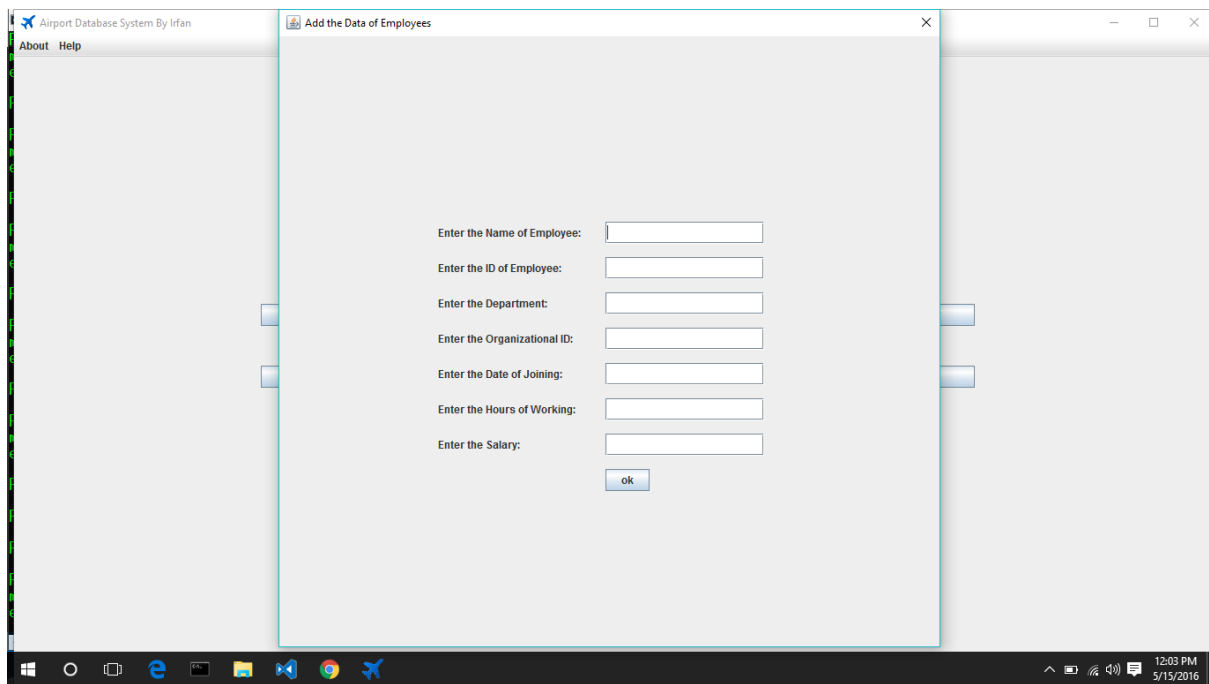
```

1272.         JButton cPCancel=new
        JButton("Cancel");
1273.         cPCancel.setBounds(350, 130,80, 25);
1274.         c.add(cPCancel);
1275.         cPCancel.setToolTipText("TO Exit the
        Change Password Dialog");
1276.         cPCancel.addActionListener(new
        ActionListener()
1277.         {
1278.             public void
        actionPerformed(ActionEvent ae)
1279.             {
1280.                 cPDialog.setVisible(false);
1281.             }
1282.         }
1283.         );
1284.
1285.     }
1286.
1287.
1288. }

```

Screenshots/Outputs:





Enter the Name of Passanger:

Enter the ID of Passanger:


Enter the Flight Number:


Enter the Airline:

Enter the Date

Enter the time of Flight

Message ×

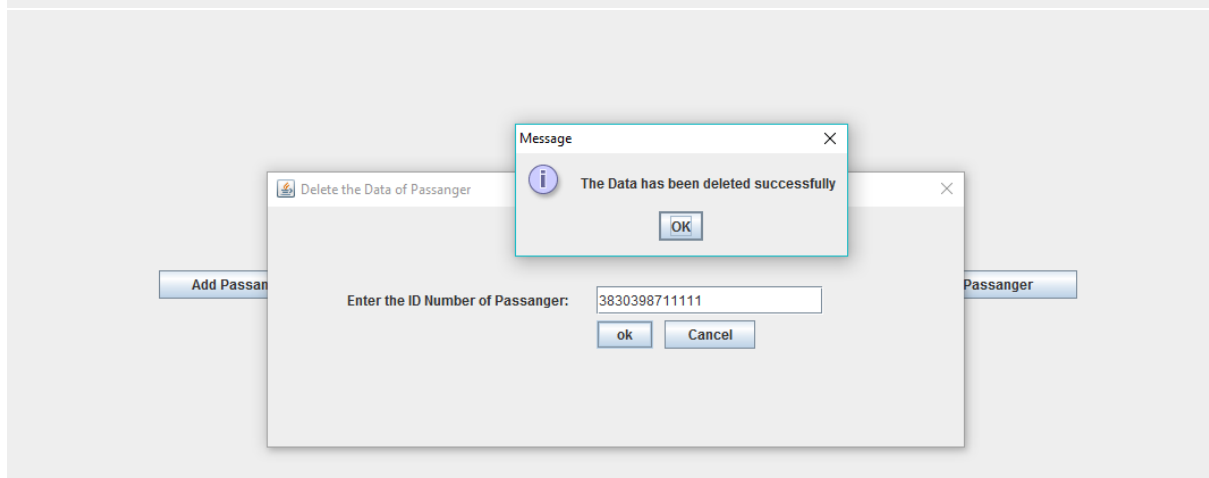
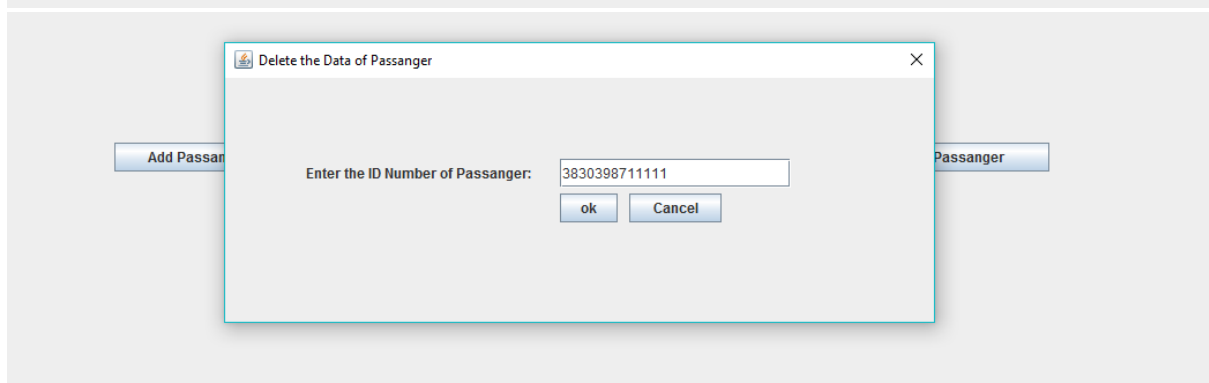
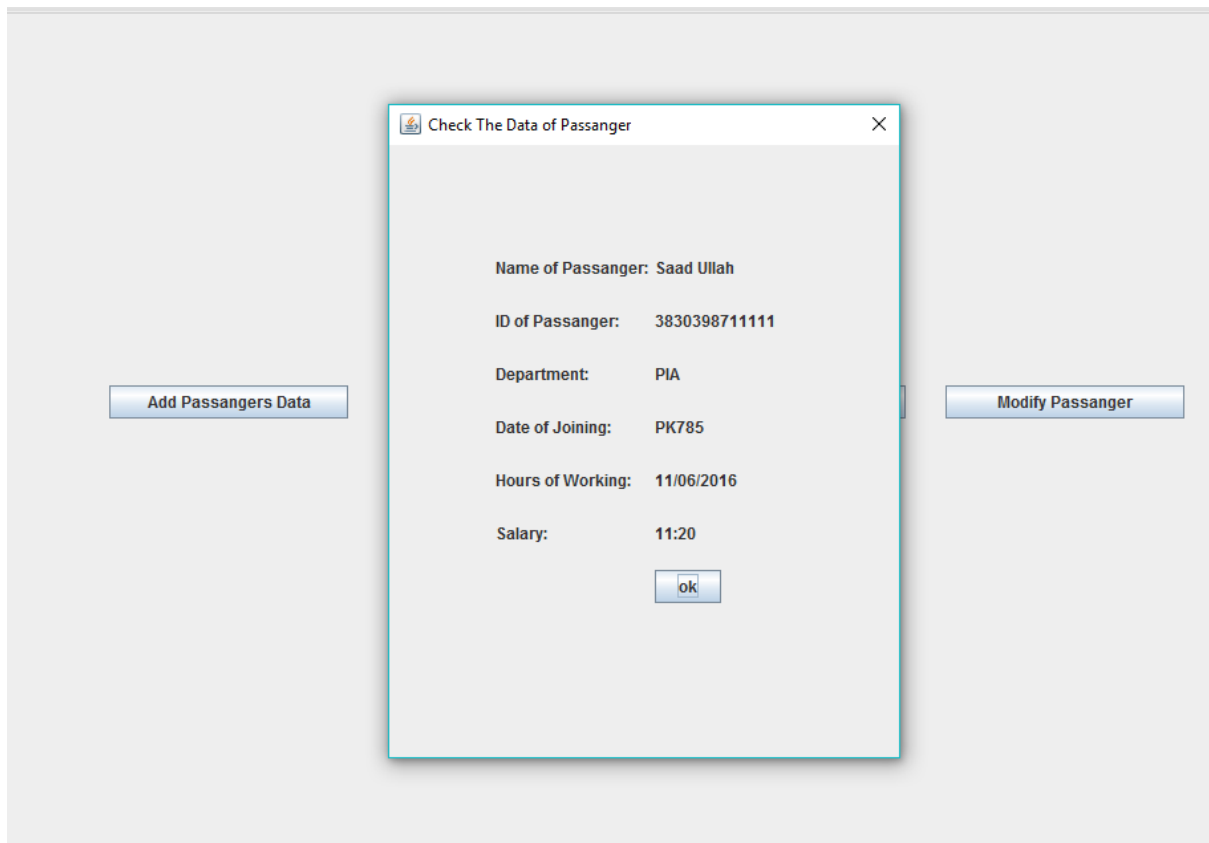
 The Record of the Passanger has been added successfully

 Check the Data of Passanger ×

Enter the ID Number of Passanger:

Add Passan

Passanger



Add Employee Data

Check Employee Data

Delete Employee Data

Modify Employee

Exit

Add the Data of Employees

Enter the Name of Employee:

Azhar

Enter the ID of Employee:

123456

Enter the Department:

Technical

Enter the Organizational ID:

TC321

Enter the Date of Joining:

11/05/2002

Enter the Hours of Working:

09

Enter the Salary:

60000

ok

Cancel

Press OK to Save Data

Check the Data of Employee

Enter the ID Number of Employee:

123456

ok

Cancel

Press ok to Check the record of Employee

Add Employee Data

Check The Data of Employee

Name of Employee:

Azhar

ID of Employee:

123456

Department:

Technical

Organizational ID:

TC321

Hours of Working:

09

Date of Joining:

11/05/2002

Salary:

60000

ok

Modify Employee

Add Employee

Delete the Data of Employee

Enter the ID Number of Employee:

123456

ok

Employee

Add the Data of Employees

Message

The Record of the Employee has been added successfully

OK

Enter the Name of Employee: azhar

Enter the ID of Employee: 1234567

Enter the Department: Accounts

Enter the Organizational ID: AC85

Enter the Date of Joining: 12/03/2015

Enter the Hours of Working: 03

Enter the Salary: 10000

ok


Delete the Data of Employee

Enter the ID Number of Employee: 1234567

ok Cancel

Add's the Tickets for a Flight: X

Message X

 The Record of the Tickets has been added successfully

OK


Total Number of Tickets:

Enter Reserved Tickets:

Enter the Flight Number:

Enter the Available Tickets:

ok Cancel

 Check The Data of Tickets X

Number of Tickets: 330

Reserved Tickets: 300

Flight Number: PK787

Available Tickets: 30

ok


AABII Solutions!

Add Food

Check Foods

Delete Food

Change Password

 Airport Database System By Irfan

About Help Exit

 Add's the Food Menu for a Flight

Message



The Record of the Catering has been added successfully

OK

Enter the Name of Food Menu:

French

Enter the ID of Food Item:

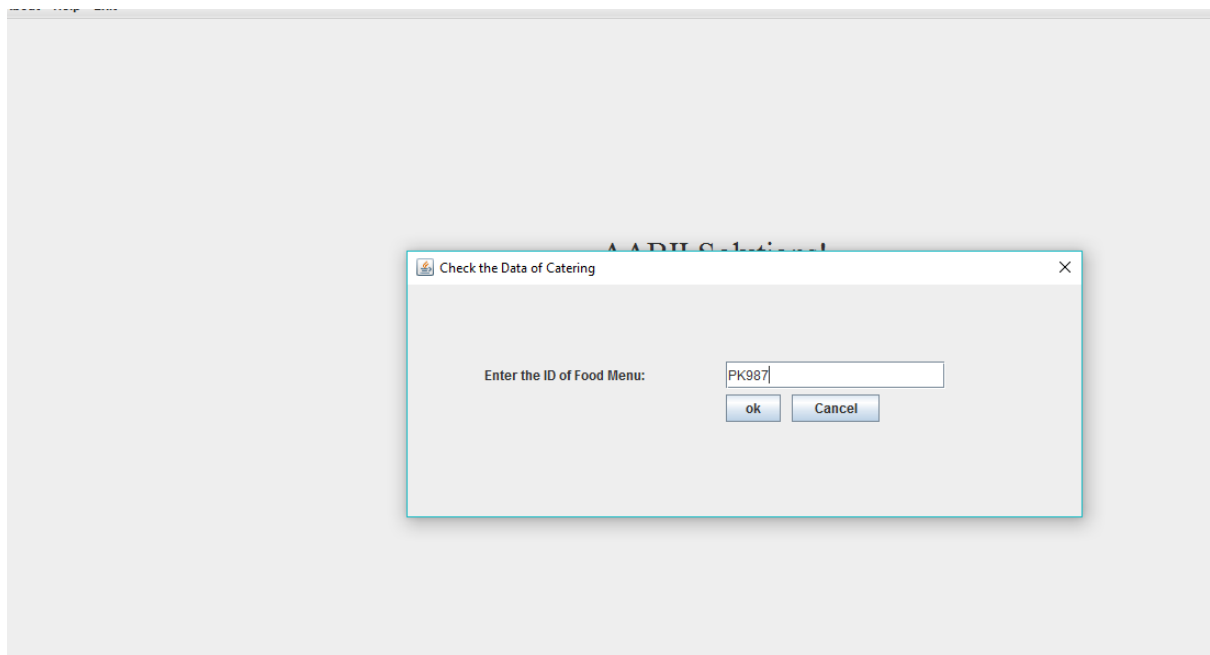
F1325

Enter the Flight Number:

PK987

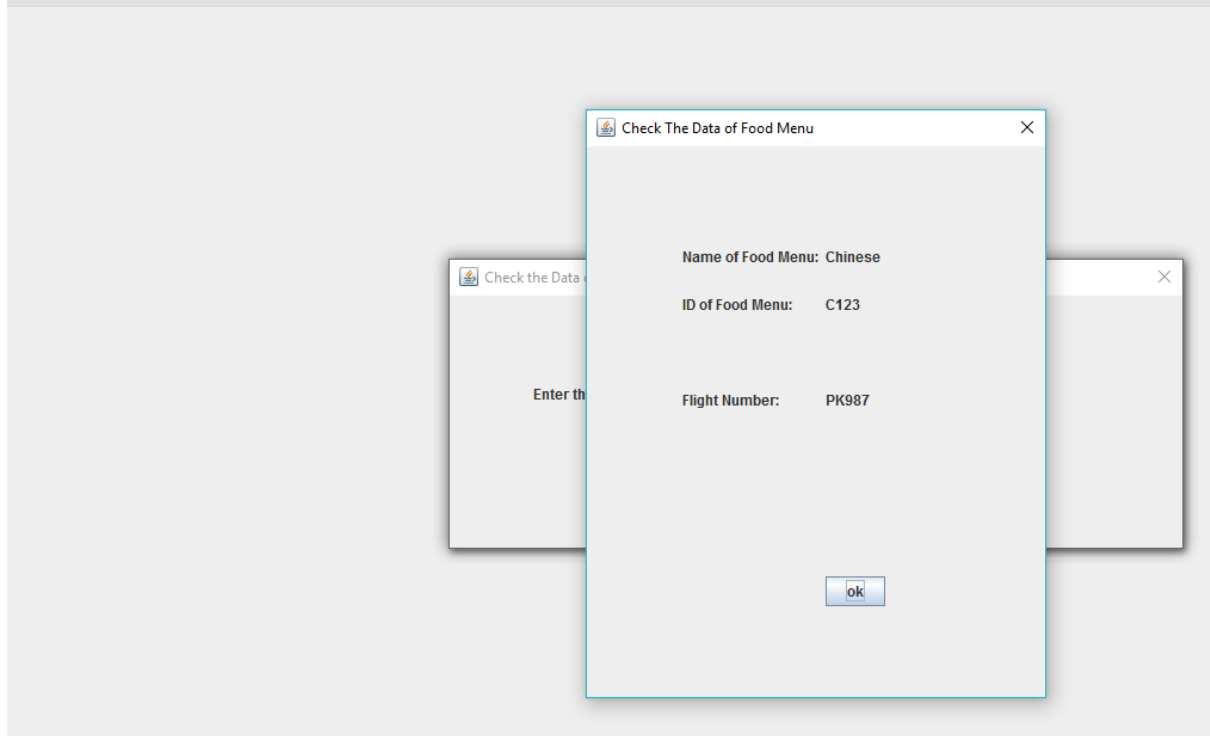
ok

Cancel



Airport Database System By Irfan

About Help Exit



Delete the Data of Food Menu

Enter the ID Number of Food Menu:

ok Cancel

Delete the Data of Food Menu

Enter the ID Number of Food Menu:

Message

The Data has been deleted successfully

OK


AABII Solutions!

Add Security

Check Security

Delete Security

Message

 The Record of the Security Employee has been added successfully

OK

Security Employee: Malik

ID of Security Employee: 125

Enter the Phone Number: 0300

Enter the Address: Lahore

Hours of Working HH:MM/... 11:00/12:00

ok Cancel

Name : Malik

ID of Employee: 125

Phone Number 0300

Address: Lahore

Hours of Working: 11:00/12:00

ok

