

AJP Micro-Project Report

Travel Management Application

1. Rational:

Advanced Java is designed to develop web-based, network-centric or enterprise applications. It includes the concepts like Servlet, Swing, JDBC, Networking, etc. It is a specialization in specific domain. Most of the applications developed using advance Java uses tow-tier architecture i.e. Client and Server. All the applications that run on Server can be considered as advance Java applications. JEE (advance Java) provides libraries to understand the concept of Client-Server architecture for web- based applications. We can also work with web and application servers such as Apache Tomcat and Glassfish Using these servers, we can understand the working of HTTP protocol. JDBC is one of the major concepts covered in Advance java. JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database.

2. Aims of the Micro-Project:

To Java is an object-oriented programming language. In Advance Java Programming language we will learn to create GUI using Applet and Swing, and even to connect database and store the data dynamically.

❖ Benefits of project are:

1. We can understand a fundamental understanding of how Java works with respect to Event handling.
2. We will learn to use Applets and some basic Swing components.
3. Travel Management enables us to accept input dynamically.
4. This can be implemented in real life system.

3. Course Outcomes Addressed:

1. Develop programs using GUI Framework (AWT and Swing).
2. Develop programs to handle events in Java Programming.
3. Develop programs using database.

4. Literature Review:

The screenshot shows a web application interface with a red header bar containing a large number '1'. Below the header, there are two main sections. The left section is titled 'Customer Detail' and contains several input fields for customer information: Name, Father Name, Phone No, C.N.I.C, Passport No, Passport Issue, Passport Expiry, PNR No, Cost Rate, Ticket Rate, Net Profit, Date, and Address. The right section is titled 'Operators' and contains a search bar with a dropdown menu for 'Re:' (Name, CNIC, Passport No, Phone No) and a search button. Below the search bar are several buttons: Add Customer, Print, Sign Up, Update, Statistics, Log Out, Delete, Contact Us, and Exit. There is also a 'Clear All' button.

The screenshot shows a web application interface for a 'Travel Management System'. The header features a logo on the left and the title 'Travel Management System' in the center. Below the header, there are several sections. The left section contains input fields for customer details: Firstname (Paul), Surname (Parker), Address (145 Godson Road), Post Code (93GM), Telephone (0567577777657), and Email (paul@kings.com). The middle section contains dropdown menus for Departure (DFW Airport), Destination (Canada - 7 Days in Toronto), and Accommodation (None). Below these are checkboxes for Airport Tax, Air Miles Over 20000, Travelling Insurance, and Ext Luggage. The bottom left section contains radio buttons for Standard, Single, Economy, Return, First Class, and Special Needs, along with price fields. The bottom right section contains a summary table with Tax, Subtotal, and Total amounts. The rightmost section is a 'Receipt' area showing the reference number (4993), customer details, and a list of charges (Cost of Car, Tax, Sub Total, Total) with their respective amounts. At the bottom, there are buttons for Total, Reset, Receipt, and Exit.

5. Actual Methodology followed:

1. Finalizing the topic and preparing proposal.
2. Collecting Information:-
Referred various websites of Indian tourism and observed how they manage Tourism process.

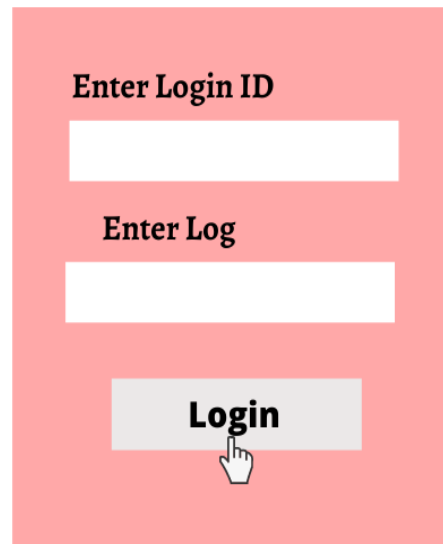
3. Designing of screens and databases

Home Page



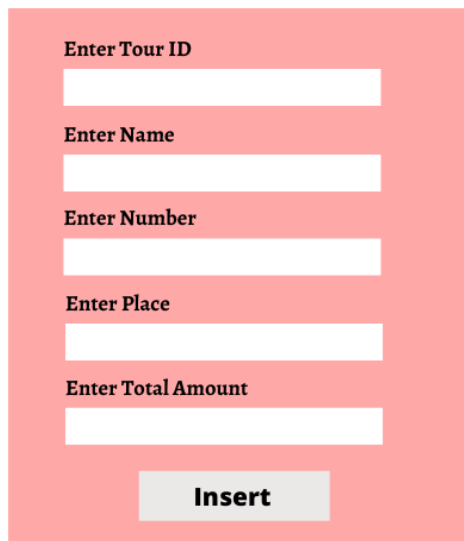
The Home Page UI design features a light red background. It contains three vertically stacked, light gray rectangular buttons with black text. The top button is labeled "Sing up" and has a hand cursor icon pointing at it. The middle button is labeled "Insert". The bottom button is labeled "Delete".

Sing up Page



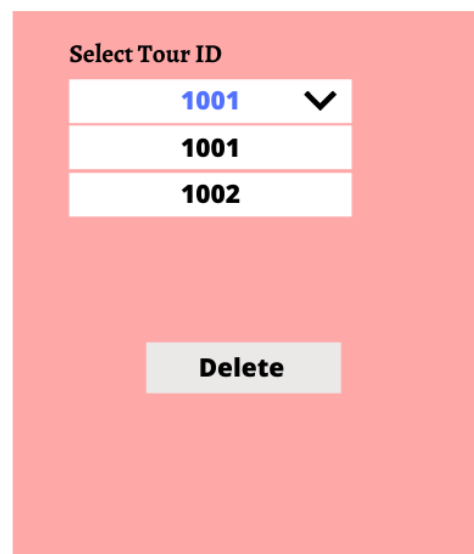
The Sing up Page UI design features a light red background. It contains two white text input fields. The first field is labeled "Enter Login ID" and the second is labeled "Enter Log". Below these fields is a light gray rectangular button with black text labeled "Login", which has a hand cursor icon pointing at it.

Insert Page




The Insert Page UI design features a light red background. It contains five white text input fields, each with a label above it: "Enter Tour ID", "Enter Name", "Enter Number", "Enter Place", and "Enter Total Amount". Below these fields is a light gray rectangular button with black text labeled "Insert".

Delete Page



The Delete Page UI design features a light red background. It contains a dropdown menu labeled "Select Tour ID" with three visible options: "1001" (highlighted in blue), "1001", and "1002". Below the dropdown menu is a light gray rectangular button with black text labeled "Delete".

Database (Design View)

details		
	Field Name	Data Type
	id	Number
	name	Text
	number	Number
	place	Text
	total	Number

4. Coding for business logics:-

```
import java.sql.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.IOException;
public class tour extends JFrame implements ActionListener, ItemListener
{
    int jid=0;
    Connection con;
    JTextField itxtid;
    JTextField itxtname;
    JTextField itxtno;
    JTextField itxtplace;
    JTextField itxttotal;
    JTextField txtn;
    JPasswordField txtpass;
    JButton ibtnInsert,ibtnModify,ibtnDelete,singup,register,home;
    JLabel l1,l2,l3,il1,il2,il3,il4,il5,sl6,inl6,dl6;
    JButton login,insert,delete;
    JComboBox jc;
    public tour()
    {
        l1=new JLabel("<html><i style='color:blue;'>new<br/></i></html>");
        l1=new JLabel();
        l2=new JLabel();
        l3=new JLabel();
        il1=new JLabel("<html><b style='font-size: 15px;'>"+ "Enter Your Tour Id:"+"</b></html>");
        il2=new JLabel("<html><b style='font-size: 15px;'>Enter your Name:</b></html>");
        il3= new JLabel("<html><b style='font-size: 15px;'>Enter your Number:</b></html>");
        il4= new JLabel("<html><b style='font-size: 15px;'>Enter your Place:</b></html>");
        il5=new JLabel("<html><b style='font-size: 15px; padding:25px'>Enter Total Cost</b></html>");
        sl6=new JLabel();
```

```

inl6=new JLabel();
dl6=new JLabel();
txtn= new JTextField(20);
txtn.setVisible(false);
txtpass= new JPasswordField(20);
txtpass.setVisible(false);
itxtid= new JTextField(20);
itxtname=new JTextField(20);
itxtno=new JTextField(20);
itxtplace=new JTextField(20);
itxttotal=new JTextField(20);
login = new JButton("          Login          ");
jc= new JComboBox();
jc.setVisible(false);
login.setVisible(false);
//singup = new JButton("Sing Up");
home = new JButton("          Home          ");
//home.setVisible(false);
register = new JButton("          Register as User          ");
ibtnInsert = new JButton("          Insert Record          ");
//ibtnModify = new JButton("Modify Record");
ibtnDelete = new JButton("          Delete Record          ");
singup = new JButton("          Sing Up          ");
insert = new JButton("          Insert          ");
delete = new JButton("          Delete          ");

ibtnDelete.setVisible(false);
        ibtnInsert.setVisible(false);

        register.setVisible(false);
        il1.setVisible(false);
        home.setVisible(false);
        jc.setVisible(false);
        il2.setVisible(false);
        il3.setVisible(false);
        il4.setVisible(false);
        il5.setVisible(false);
        itxtid.setVisible(false);
        itxtname.setVisible(false);
        itxtno.setVisible(false);
        itxtplace.setVisible(false);
        itxttotal.setVisible(false);
setLayout(new FlowLayout());
//add(sing);
add(singup);
add(insert);
add(delete);
add(home);
add(il1);
add(txtn);
add(il2);

```

```

add(txtpass);
add(login);
add(l3);

//add(register);
add(il1);
add(jc);
add(itxtid);
add(il2);
add(itxtname);
add(il3);
add(itxtno);
add(il4);
add(itxtplace);
add(il5);
add(itxttotal);
add(ibtnInsert);
//add(ibtnModify);
add(ibtnDelete);
add(sl6);
add(inl6);
add(dl6);
singup.addActionListener(this);
login.addActionListener(this);
home.addActionListener(this);
insert.addActionListener(this);
delete.addActionListener(this);
//register.addActionListener(this);
ibtnInsert.addActionListener(this);
//ibtnModify.addActionListener(this);
ibtnDelete.addActionListener(this);
jc.addItemListener(this);
try
{
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    System.out.println("Driver registered successfully");
}
catch(Exception e2)
{
    System.out.println("Error in driver registration...");
}
}

public void itemStateChanged(ItemEvent e)
{
    try
    {
        jid=Integer.parseInt(jc.getSelectedItem().toString());
        Statement st1=con.createStatement();
        ResultSet rs= st1.executeQuery("select name,no,place,total from details where id="+jid);
        while(rs.next())
    }

```

```

        {
            //name=rs.getString(1);
            //no=Integer.parseInt(rs.getString(2));
            //place=rs.getString(3);
            //total=Integer.parseInt(rs.getString(4));
        }
        //txtename.setText(enm);
        //txtsal.setText(Integer.toString(sal));
    }
    catch(SQLException e3)
    {}
}

public void actionPerformed(ActionEvent e)
{
    if(e.getSource() == home){
        //con=DriverManager.getConnection("jdbc:odbc:newc");
        txtn.setVisible(false);
        txtpass.setVisible(false);
        login.setVisible(false);
        home.setVisible(true);
        l1.setVisible(false);
        l2.setVisible(false);
        l3.setVisible(false);
        singup.setVisible(true);
        insert.setVisible(true);
        delete.setVisible(true);
        register.setVisible(false);
        home.setVisible(false);
        ibtnDelete.setVisible(false);
        ibtnInsert.setVisible(false);
        //singup.setVisible(false);
        register.setVisible(false);
        il1.setVisible(false);
        il2.setVisible(false);
        il3.setVisible(false);
        il4.setVisible(false);
        il5.setVisible(false);
        itxtid.setVisible(false);
        itxtname.setVisible(false);
        itxtno.setVisible(false);
        itxtplace.setVisible(false);
        itxttotal.setVisible(false);
        jc.setVisible(false);
        sl6.setVisible(false);
        inl6.setVisible(false);
        dl6.setVisible(false);
    }
}

```

```

if(e.getSource() == delete){
//con=DriverManager.getConnection("jdbc:odbc:newc");
txtn.setVisible(false);
    txtpass.setVisible(false);
    login.setVisible(false);
    home.setVisible(true);
    l1.setVisible(false);
    l2.setVisible(false);
    l3.setVisible(false);
    singup.setVisible(false);
    insert.setVisible(false);
    delete.setVisible(false);
    register.setVisible(false);
    home.setVisible(true);
    ibtnDelete.setVisible(true);
    ibtnInsert.setVisible(false);
    //singup.setVisible(false);
    register.setVisible(false);
    il1.setVisible(true);
    jc.setVisible(true);
    il2.setVisible(false);
    il3.setVisible(false);
    il4.setVisible(false);
    il5.setVisible(false);
    itxtid.setVisible(false);
    itxtname.setVisible(false);
    itxtno.setVisible(false);
    itxtplace.setVisible(false);
    itxttotal.setVisible(false);
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        System.out.println("Driver registered successfully");
        con=DriverManager.getConnection("jdbc:odbc:newc");
        Statement st= con.createStatement();
        ResultSet rs1= st.executeQuery("select id from details");
        while(rs1.next())
        {
            String a=rs1.getString(1);
            jc.addItem(a);
        }
    }
    catch(Exception e2)
    {
        System.out.println("Error in driver registration...");
    }
}

if(e.getSource() == singup){
//con=DriverManager.getConnection("jdbc:odbc:newc");

```



```

        txtn.setVisible(true);
        txtpass.setVisible(true);
        login.setVisible(true);
        home.setVisible(true);
        insert.setVisible(false);
        delete.setVisible(false);
        ibtnDelete.setVisible(false);
        ibtnInsert.setVisible(false);
        singup.setVisible(false);
        register.setVisible(false);
        il1.setVisible(false);
        il2.setVisible(false);
        il3.setVisible(false);
        il4.setVisible(false);
        il5.setVisible(false);
        itxtid.setVisible(false);
        itxtname.setVisible(false);
        itxtno.setVisible(false);
        itxtplace.setVisible(false);
        itxttotal.setVisible(false);
        l1.setText("<html><b style='font-size: 15px; padding:25px'>Enter  User
ID<b></html>");
        l2.setText("<html><b style='font-size: 15px; padding:25px'>Enter
Password<b></html>");
        l1.setVisible(true);
        l2.setVisible(true);
        l3.setVisible(true);

    }

    if(e.getSource() == insert){
        il1.setVisible(true);
        il2.setVisible(true);
        il3.setVisible(true);
        il4.setVisible(true);
        il5.setVisible(true);
        itxtid.setVisible(true);
        itxtname.setVisible(true);
        itxtno.setVisible(true);
        itxtplace.setVisible(true);
        itxttotal.setVisible(true);
        ibtnInsert.setVisible(true);
        home.setVisible(true);
        singup.setVisible(false);
        insert.setVisible(false);
        delete.setVisible(false);
    }

    if(e.getSource() == login){

```

```

txtn.setVisible(true);
    txtpass.setVisible(true);
    login.setVisible(true);
    home.setVisible(true);

    ibtnDelete.setVisible(false);
    ibtnInsert.setVisible(false);
    singup.setVisible(false);
    register.setVisible(false);
    il1.setVisible(false);
    il2.setVisible(false);
    il3.setVisible(false);
    il4.setVisible(false);
    il5.setVisible(false);
    itxtid.setVisible(false);
    itxtname.setVisible(false);
    itxtno.setVisible(false);
    itxtplace.setVisible(false);
    itxttotal.setVisible(false);
    l1.setText("Enter User ID");
    l2.setText("Enter Password");
    l1.setText("Enter User ID");
    l2.setText("Enter Password");
    String uid= txtn.getText();
    String pwd1=txtpass.getText();
    try
    {
        con=DriverManager.getConnection("jdbc:odbc:newc");
        PreparedStatement ps= con.prepareStatement("select count(*) from login where userid=? and
        password=?");
        ps.setString(1,uid);
        ps.setString(2,pwd1);
        ResultSet rs=ps.executeQuery();
        int n=0;
        while(rs.next())
        {
            n=Integer.parseInt(rs.getString(1));
        }
        if(n!=0)
            sl6.setText("Logged in successfully");
        else
            sl6.setText("Not a valid user");

        con.close();

    }
    catch(SQLException e1)
    {}
    }

```

```

if(e.getSource() == ibtnInsert){
    try
    {
        int id=Integer.parseInt(itxtid.getText());
        String name=itxtname.getText();
        int no=Integer.parseInt(itxtno.getText());
        String place=itxtplace.getText();
        int total=Integer.parseInt(itxttotal.getText());
        con=DriverManager.getConnection("jdbc:odbc:newc");
        PreparedStatement ps= con.prepareStatement("insert into details values(?,?,?,?,?)");
        ps.setInt(1,id);
        ps.setString(2,name);
        ps.setInt(3,no);
        ps.setString(4,place);
        ps.setInt(5,total);
        ps.executeUpdate();
        inl6.setVisible(true);
        inl6.setText("Record Successfully inserted");
        con.close();
    }
    catch(SQLException e1)
    {
        inl6.setText("Error in inserting Record");
    }
}

if(e.getSource() == ibtnDelete){
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        System.out.println("Driver registered successfully");
        con=DriverManager.getConnection("jdbc:odbc:newc");
        Statement st= con.createStatement();
        ResultSet rs1= st.executeQuery("select id from details");
        while(rs1.next())
        {
            String a=rs1.getString(1);
            jc.addItem(a);
        }
    }
    catch(Exception e2)
    {
        System.out.println("Error in driver registration...");
    }
    try
    {
        con=DriverManager.getConnection("jdbc:odbc:newc");
        l3.setVisible(true);
        l3.setText("Select your tour ID");
        dl6.setVisible(true);
    }
}

```

```

PreparedStatement ps= con.prepareStatement("delete from details where id=?");
ps.setInt(1,jid);
ps.executeUpdate();
dl6.setText("Record Sucessfully deleted");
con.close();
}
catch(SQLException e1)
{
dl6.setText("Error in deleting Rechord");
}
}
}
public static void main(String args[])
{
    tour f1= new tour();
    f1.setSize(285,430);
    f1.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    f1.getContentPane().setBackground(Color.PINK);
    f1.setVisible(true);
}
}

```

5. Testing with various inputs

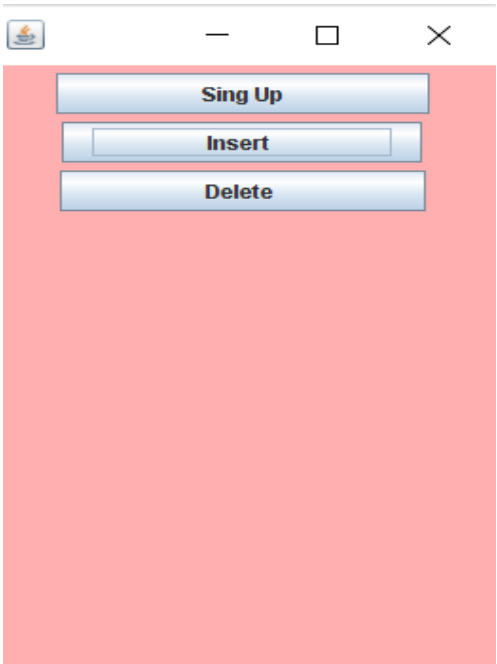
6. Implementation

6. Resources Required:

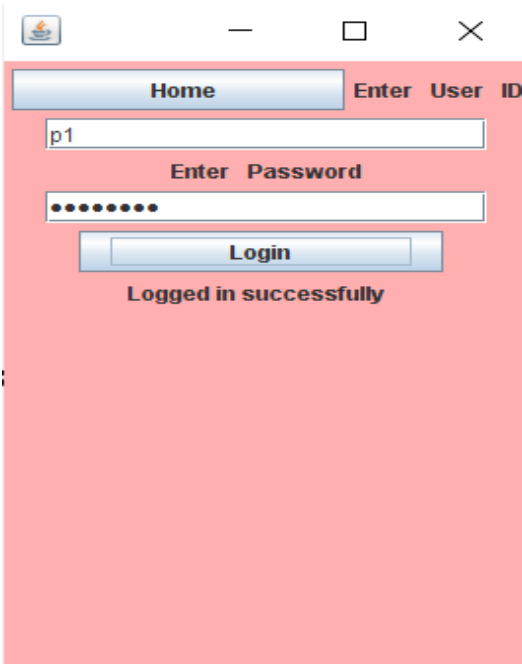
Sr.no	Name of Resources	Specifications	Qty	Remarks
1	Computer System	Intel corei3 7 th gen, Windows 10	1	-
2	Software	JDK 1.6 & MS Access 2007	-	-
3.	References	AJP Notes	-	-

7. Output of the Moicro-Project:

Home page



Sing up page



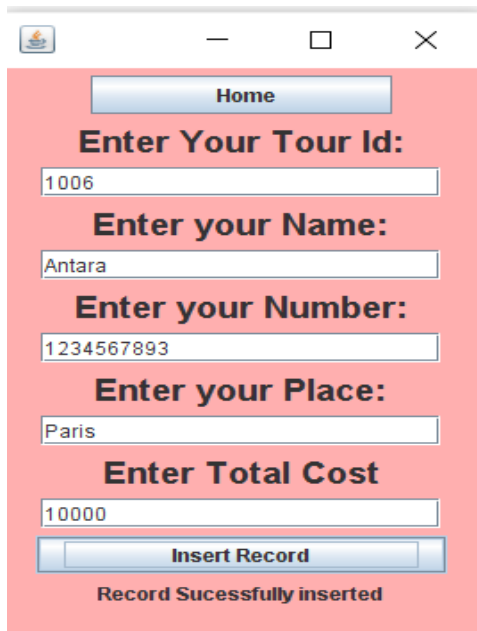
Database Login Details

<div><div></div>details</div> <div><div></div>login</div>			
	userid	password	Add New Field
	p1	tanvi123	
	p2	aarya123	
*			

Database Tour Details

<div><div></div>details</div>						
	id	name	number	place	total	Ac
	1001	Tanvi	1234567897	Malaysia	10000	
	1002	Sanskruti	1234567896	Maldivis	10000	
	1003	Purva	1234567898	New York	10000	
	1004	Latika	1234567895	Maldivs	20000	
	1005	Jhanvi	1234567894	Paris	20000	
*						

Insert Page



Home

Enter Your Tour Id:
1006

Enter your Name:
Antara

Enter your Number:
1234567893

Enter your Place:
Paris

Enter Total Cost
10000

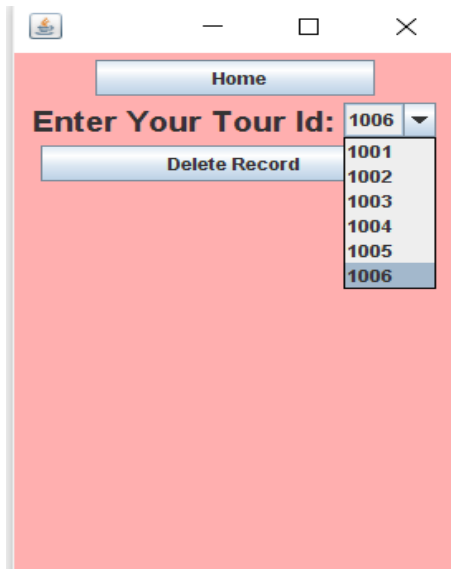
Insert Record

Record Sucessfully inserted

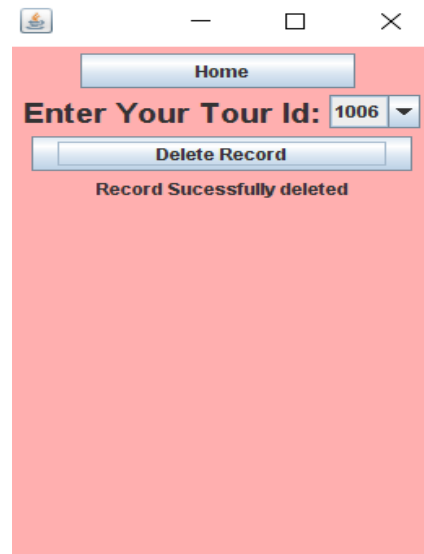
Record Inserted in Database

details						
	id	name	number	place	total	Add New Field
	1001	Tanvi	1234567897	Malaysia	10000	
	1002	Sanskruti	1234567896	Maldivis	10000	
	1003	Purva	1234567898	New York	10000	
	1004	Latika	1234567895	Maldivs	20000	
	1005	Jhanvi	1234567894	Paris	20000	
	1006	Antara	1234567893	Paris	10000	
*						

Delete Page



A screenshot of a web application window titled "Delete Page". The window has a red background. At the top, there is a "Home" button. Below it, the text "Enter Your Tour Id:" is followed by a dropdown menu showing a list of tour IDs: 1001, 1002, 1003, 1004, 1005, and 1006. The ID 1006 is currently selected. Below the dropdown is a "Delete Record" button.



A screenshot of the same web application window after a successful deletion. The dropdown menu is now closed, and the text "Record Sucessfully deleted" is displayed below the "Delete Record" button. The "Delete Record" button is still visible.

Database Deleted Page

details					
	id	name	number	place	total
	1001	Tanvi	1234567897	Malaysia	10000
	1002	Sanskruti	1234567896	Maldivis	10000
	1003	Purva	1234567898	New York	10000
	1004	Latika	1234567895	Maldivs	20000
	1005	Jhanvi	1234567894	Paris	20000
*					

8. Skill Develop:

We learned to develop Application using Advance java and store & retrieve information using SQL commands and Microsoft Access

9. Application

Used as real world application for Tour booking system Application

10. Names of Team Members with Roll No: -

Sr. No.	Roll no	Name
1.	54	Tanvi Vijay Chiman