

# **JAVA AWT BASED- MY CITY INFORMATION- SQL CONNECTIVITY USING JDBC**

*A*

*Report*

*Submitted in partial fulfilment of the  
Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**By**

**D.YASHOVARDHAN (1602-18-737-120)**

**Under the Guidance of**

**B. Leelavathy**



**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

## BONAFIDE CERTIFICATE

This is to certify that this project report titled “MY CITY INFORMATION” is the bonafied mini project work of Mr. D Yasho Vardhan Reddy bearing hall ticket number 1602-18-737-120 under the guidance of B. Leelavathy during 4th sem B.E for the academic year 2019-2020.

External Examiner

Internal Examiner

B.LEELAVATHY

Assistant Professor

Department Of Information

Technology

# MY CITY INFORMATION

## **AIM:**

To create a Java GUI for MY City Information which takes city and different landmarks in city and it also takes details of different services like hotels, metro stations , tourist places . These values are updated in Database so that we can get different services around the landmark in specific city.

These values are to be updated in the database using JDBC connectivity .

## **Abstract:**

Nowadays software application plays a major role in each and every field to improve their carrier. In our state as well as in other states there are many people accessing website if they want to know about city information. It helps user to view city information in easier way. Users can view the data in anytime and anywhere . The idea behind this application is to help people to find different services ,entertainment and other facilities available in

there city and this application is also useful for the tourist to know where are hotels ,transport facilities like metro stations ,tourist places etc.. This application can also be used to find different places in near by area by using current location or landmark. This application can also be used to know rating of different places , timings and other details.

## **Introduction:**

Requirements:

List of Tables:

- 1) MyCity
- 2) Hotels
- 3) TouristPlaces
- 4) MetroStations
- 5) Has\_Metro\_Stations
- 6) Has\_Tourist\_Places
- 7) Has\_Hotels
- 8) Services
- 9) Landmarks

Table Name	Description	Attribute	DataType
MyCity	cityname	cityname	varchar2(15)
Services	Cityname	cityname	varchar2(15)

	Services	services	varchar2(15)
Landmarks	Cityname	cityname	varchar2(15)
	Landmark	landmark	varchar2(15)
MetroStation	Station Name	Station_name	varchar2(25)
	Landmark	landmark	varchar2(25)
	Timmings(open s and closes at)	timmmings	varchar2(20)
Hotels	Hotel Name	hotel_name	varchar2(15)
	Landmark	landmark	varchar2(15)
	Phone Number	phone_number	number(10)
	Rating(between 0 and 10)	rating	number(2)
Tourist Places	Place Name	Place_name	varchar2(15)
	Landmark	landmark	varchar2(15)
	Rating	rating	number(2)
Has Metrostation s	City Name	cityname	varchar2(15)
	Station Name	station_name	varchar2(15)
Has Hotels	City Name	cityname	varchar2(15)
	Hotel Name	hotel_name	varchar2(15)
Has Tourist Places	City Name	cityname	varchar2(15)
	Place Name	place_name	varchar2(15)

## SOFTWARE USED:

Java Eclipse, Oracle 11g Database, Java SE version 7, SQL\*Plus.

**Eclipse:** It is an integrated development environment (IDE) used in computer programming. It contains a base workspace and an extensible plug in system for customizing the environment. The Eclipse software development kit (SDK), which include java development tools is meant for java developers.

**SQL \*plus:** SQL \*plus is a command line tool proprietary to oracle. You can send SQL Queries to the server using the tool. It can also help you format the result of query. SQL is the query language that is used to communicate with the oracle server to access and modify data.

### **Java AWT:**

Java AWT (Abstract Window Toolkit) is an API to develop GUI or window-based

applications in java.

Java AWT components are platform-dependent i.e. components are displayed according

to the view of operating system. AWT is heavyweight i.e. its components are using the

resources of OS.

The java.awt package provides classes for AWT API such as TextField, Label, TextArea,

RadioButton, CheckBox, Choice, List etc.

### **Java Swing:**

Java Swing is a part of Java Foundation Classes (JFC) which was designed for enabling large-scale enterprise development of Java applications. Java Swing is a set of APIs that provides graphical user interface (GUI) for Java programs. Java Swing is also known as Java GUI widget toolkit.

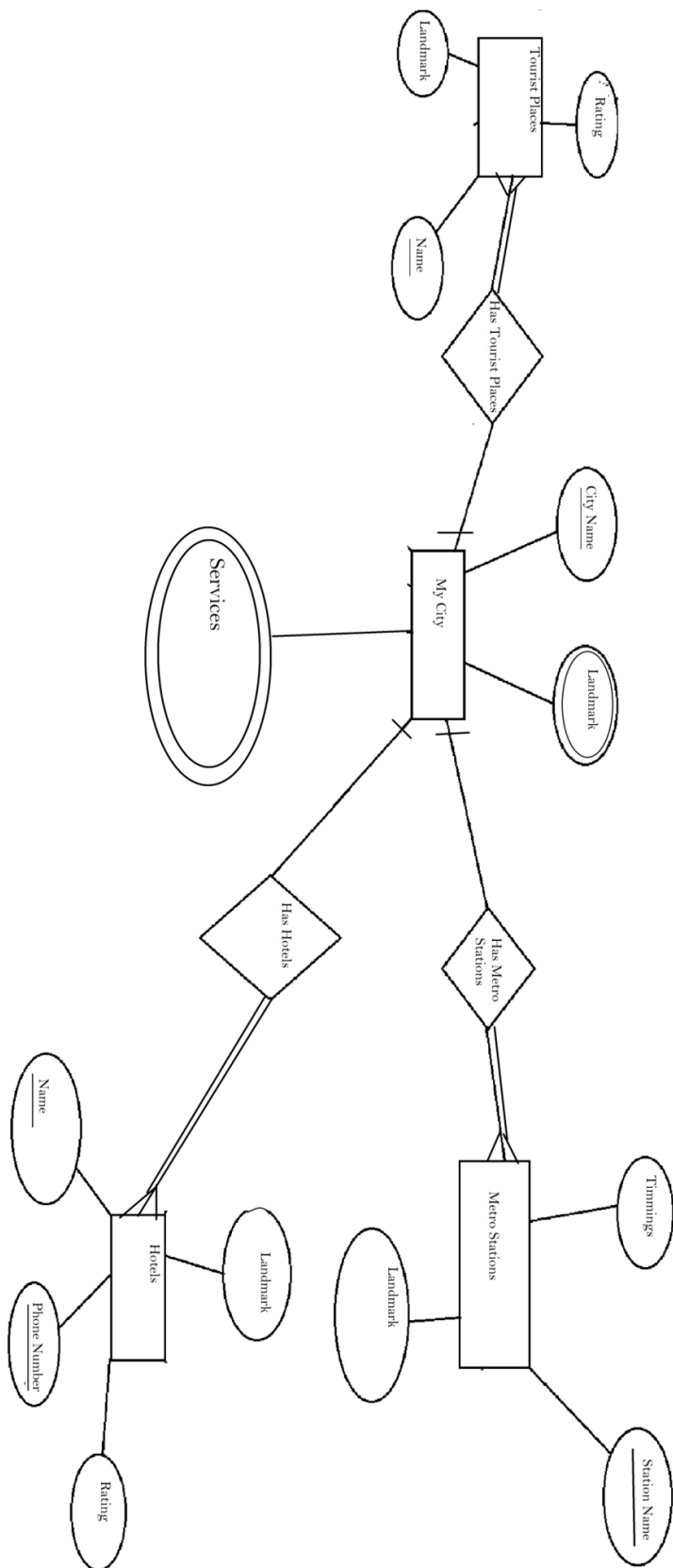
Java Swing or Swing was developed based on earlier APIs called Abstract Windows Toolkit (AWT). Swing provides richer and more sophisticated GUI components than AWT. The GUI

components are ranging from a simple label to complex tree and table. Besides emulating look and feel of various platforms, Swing also provides *the* pluggable look and feel to allow look and feel of Java programs independent from the underlying platform.

## **SQL:**

Structure Query Language(SQL) is a database query language used for storing and managing data in Relational DBMS. SQL was the first commercial language introduced for E.F Codd's Relational model of database. Today almost all RDBMS (MySQL, Oracle, Infomix, Sybase, MS Access) use SQL as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

## **ER DIAGRAM**





## CREATION OF TABLES:

### DDL Commands:

```
SQL> create table mycity(cityname varchar2(15) primary key);
```

Table created.

```
SQL> desc mycity;
```

Name	Null?	Type
-----		
CITYNAME	NOT NULL	VARCHAR2(15)

```
SQL> create table landmarks(cityname varchar2(15),landmark  
varchar2(15),foreign key(cityname) references mycity(cityname),primary  
key(cityname,landmark));
```

Table created.

```
SQL> desc landmarks;
```

Name	Null?	Type
-----		
CITYNAME	NOT NULL	VARCHAR2(15)

LANDMARK

NOT NULL VARCHAR2(15)

```
SQL> create table services(cityname varchar2(15),services varchar2(15)
not null ,foreign key(cityname) references mycity,primary
key(cityname,services));Table created.
```

```
SQL> create table tourist_places(name varchar2(15) primary key,city
varchar2(15) not null, landmark varchar2(15) not null,rating number(2));
```

Table created.

```
SQL> alter table tourist_places add constraint check_rating check(rating
between 1 and 10);
```

Table altered.

```
SQL> desc tourist_places;
```

Name	Null?	Type
NAME	NOT NULL	VARCHAR2(15)
CITY	NOT NULL	VARCHAR2(15)
LANDMARK	NOT NULL	VARCHAR2(15)
RATING		NUMBER(2)

```
SQL> alter table tourist_places rename column name to place_name;
```

Table altered.

```
SQL> create table has_tourist_places(cityname
varchar2(15),place_name varchar2(15) ,foreign key(cityname)
references mycity,foreign key(place_name) references tourist_places);
```

Table created.

```
SQL> desc has_tourist_places;
```

Name	Null?	Type
-----		
CITYNAME		VARCHAR2(15)
PLACE_NAME		VARCHAR2(15)

```
SQL> create table metrostations(station_name varchar2(15) primary
key,landmark varchar2(15) not null,timings varchar2(10));
```

Table created.

```
SQL> desc metrostations;
```

Name	Null?	Type
-----		
STATION_NAME		NOT NULL VARCHAR2(15)
LANDMARK		NOT NULL VARCHAR2(15)
TIMMINGS		VARCHAR2(10)

```
SQL> create table has_metro_stations(cityname
varchar2(15),station_name varchar2(15) ,foreign key(cityname)
references mycity,foreign key(station_name) references metrostations);
```

Table created.

```
SQL> desc has_metro_stations;
```

Name	Null?	Type
-----		
CITYNAME		VARCHAR2(15)
STATION_NAME		VARCHAR2(15)

```
SQL> desc has_metro_stations;
```

Name	Null?	Type
-----		
CITYNAME		VARCHAR2(15)
STATION_NAME		VARCHAR2(15)

```
SQL> create table hotels(hotel_name varchar2(15) primary
key,landmark varchar2(15) not null,phone_number number(10),rating
number(2));
```

Table created.

```
SQL> alter table hotels add constraints ckey_rating check(rating
between 1 and 10);
```

Table altered.

SQL> desc hotels;

Name	Null?	Type
-----		
HOTEL_NAME		NOT NULL VARCHAR2(15)
LANDMARK		NOT NULL VARCHAR2(15)
PHONE_NUMBER		NUMBER(10)
RATING		NUMBER(2)

```
SQL> create table has_hotels(cityname varchar2(15),hotel_name
varchar2(15),foreign key(cityname) references mycity,foreign
key(hotel_name) references hotels);
```

Table created.

SQL> desc has\_hotels;

Name	Null?	Type
-----		
CITYNAME		VARCHAR2(15)
HOTEL_NAME		VARCHAR2(15)

## Java-SQL Connectivity using JDBC:

Java Database Connectivity (JDBC) is an application programming interface (API) for

the programming language Java, which defines how a client may access a database. It is a

Java-based data access technology used for Java database connectivity. It is part of

the Java Standard Edition platform, from Oracle Corporation. It provides methods to

query and update data in a database and is oriented towards relational databases.

The connection to the database can be performed using Java programming (JDBC API)

as:

```
private void connToDb()
{
    try
    {
        Class.forName("oracle.jdbc.driver.OracleDriver");

        connection=DriverManger.getConnection("jdbc:oracle:thin:@localhost:1522:xe","yasho",
        "password");

        statement=connection.createStatement();
    }
    catch(SQLException connectException)
    {
        System.out.println(connectException.getMessage());
    }
}
```

```

        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
    catch(Exception e)
    {
        System.err.println("Unable to find and load driver");
        System.exit(1);
    }
}

```

Thus, the connection from Java to Oracle database is performed and therefore, can be used for updating tables in the database directly.

### **Program For Main GUI:**

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class Home extends JFrame{
    private JPanel panel1;
    private JPanel panel2;
    private JButton btnMycity;
    private JButton btnLandmarks;

```

```

private JButton btnServices;

private JButton btnHotels;

private JButton btnMetrostations;

private JButton btnTouristplaces;

private JButton btnHas_Hotels;

private JButton btnHas_metrostations;

private JButton btnHas_touristplaces;

private JButton btnCustomer;

public Home()
{
    panel1=new JPanel(new GridLayout(3,3,15,15));
    panel2=new JPanel();
    btnMetrostations=new JButton("Metro Stations");
    btnMetrostations.setBackground(Color.orange);
    btnHas_Hotels=new JButton("Has Hotels");
    btnHas_Hotels.setBackground(Color.ORANGE);
    btnHas_metrostations=new JButton("Has Metro Stations");
    btnHas_metrostations.setBackground(Color.orange);
    btnHas_touristplaces=new JButton("Has Tourist Places");
    btnHas_touristplaces.setBackground(Color.ORANGE);
    btnHotels=new JButton("Hotels");
    btnHotels.setBackground(Color.ORANGE);
    btnLandmarks=new JButton("Landmark");
    btnLandmarks.setBackground(Color.ORANGE);
    btnMycity=new JButton("My City");
    btnMycity.setBackground(Color.ORANGE);
    btnServices=new JButton("Services");

```



```

btnServices.setBackground(Color.ORANGE);
btnTouristplaces=new JButton("Tourist Places");
btnTouristplaces.setBackground(Color.ORANGE);
btnCustomer=new JButton("Go To User Interface");
panel1.setBackground(Color.LIGHT_GRAY);
panel2.setBackground(Color.DARK_GRAY);
panel1.add(btnMycity);
panel1.add(btnServices);
panel1.add(btnLandmarks);
panel1.add(btnHotels);
panel1.add(btnMetrostations);
panel1.add(btnTouristplaces);
panel1.add(btnHas_Hotels);
panel1.add(btnHas_metrostations);
panel1.add(btnHas_touristplaces);
panel2.add(btnCustomer);
this.setVisible(true);
this.setSize(600,400);
this.setDefaultCloseOperation(3);
this.add(panel1,BorderLayout.CENTER);
this.add(panel2,BorderLayout.PAGE_END);
btnHas_Hotels.addActionListener(new ActionListener() {

```

```

    @Override

```

```

    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new HasHotelsView();
    }

```

```

        dispose();
    }
});

btnHas_metrostations.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasMetroStationView();
        dispose();
    }

});

btnHas_touristplaces.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasTouristPlacesView();
        dispose();
    }

});

btnHotels.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HotelsView();

```

```

        dispose();
    }
});

btnLandmarks.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new LandmarksView();

        dispose();

    }

});

btnMetrostations.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new MetroStationsView();

        dispose();

    }

});

btnMycity.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new MyCityTableView();

```

```

        dispose();
    }
});

btnServices.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new ServicesView();
        dispose();
    }

});

btnTouristplaces.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new TouristPlacesView();
        dispose();
    }

});

btnCustomer.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new MetroStations();

```

```

        dispose();
    }

});

}

public static void main(String args[])
{
    new Home();
}

}

```

## Program For Insert In Hotels Table:

```

import java.awt.*;
import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

import javax.swing.*;

public class HotelsInsert extends JFrame{

```

```
private JPanel panel;  
private JPanel panel2;  
private JButton btnHome;  
private JLabel lblhotel;  
private JLabel lbllandmark;  
private JLabel lblrating;  
private JLabel lblphonenummer;  
private JTextField txthotel;  
private JTextField txtlandmark;  
private JTextField txtphonenummer;  
private JComboBox< String> cmbratings;  
private JButton btninsert;  
private JButton btnclear;  
private JTextArea txtmessage;  
private JMenu MyCityTable;  
private JMenu Services;  
private JMenu Tourist_Places;  
private JMenu Has_Tourist_Places;  
private JMenu MetroStations;  
private JMenu Has_MetroStations;  
private JMenu Hotels;  
private JMenu Has_Hotels;  
private JMenu Landmarks;  
private JMenuBar menubar;  
private JMenuItem insert1;  
private JMenuItem delete1;  
private JMenuItem view1;
```

```
private JMenuItem insert2;
private JMenuItem delete2;
private JMenuItem view2;
private JMenuItem insert3;
private JMenuItem delete3;
private JMenuItem view3;
private JMenuItem insert4;
private JMenuItem delete4;
private JMenuItem view4;
private JMenuItem insert5;
private JMenuItem delete5;
private JMenuItem view5;
private JMenuItem insert6;
private JMenuItem delete6;
private JMenuItem view6;
private JMenuItem insert7;
private JMenuItem delete7;
private JMenuItem view7;
private JMenuItem insert8;
private JMenuItem delete8;
private JMenuItem view8;
private JMenuItem insert9;
private JMenuItem view9;
private JMenuItem delete9;
private String ratings[] = {"0","1","2","3","4","5","6","7","8","9","10"};
public HotelsInsert()
{
```

```
panel2=new JPanel();
btnHome=new JButton("Home");
panel2.setBackground(Color.DARK_GRAY);
panel=new JPanel(new FlowLayout());
lblhotel=new JLabel("Hotel:");
lbllandmark=new JLabel("Landmark:");
lblphonenumber=new JLabel("Phone Number:");
lblrating=new JLabel("Rating:");
txthotel=new JTextField(20);
txtlandmark=new JTextField(20);
txtphonenumber=new JTextField(10);
cmbratings=new JComboBox<String>(ratings);
btninsert=new JButton("Insert");
btnclear=new JButton("Clear");
txtmessage=new JTextArea(10,40);
menubar=new JMenuBar();
MyCityTable=new JMenu("MyCityTable");
insert1=new JMenuItem("Insert");
view1=new JMenuItem("View");
delete1=new JMenuItem("Delete");
MyCityTable.add(insert1);
MyCityTable.add(view1);
MyCityTable.add(delete1);
Services=new JMenu("Services");
insert2=new JMenuItem("Insert");
view2=new JMenuItem("View");
delete2=new JMenuItem("Delete");
```



```
Services.add(insert2);

Services.add(view2);

Services.add(delete2);

Tourist_Places=new JMenu("Tourist_Places");

insert3=new JMenuItem("Insert");

view3=new JMenuItem("View");

delete3=new JMenuItem("Delete");

Tourist_Places.add(insert3);

Tourist_Places.add(view3);

Tourist_Places.add(delete3);

Has_Tourist_Places=new JMenu("Has_Tourist_Places");

insert4=new JMenuItem("Insert");

view4=new JMenuItem("View");

delete4=new JMenuItem("Delete");

Has_Tourist_Places.add(insert4);

Has_Tourist_Places.add(view4);

Has_Tourist_Places.add(delete4);

MetroStations=new JMenu("MetroStations");

insert5=new JMenuItem("Insert");

view5=new JMenuItem("View");

delete5=new JMenuItem("Delete");

MetroStations.add(insert5);

MetroStations.add(view5);

MetroStations.add(delete5);

Has_MetroStations=new JMenu("Has_MetroStations");

insert6=new JMenuItem("Insert");

view6=new JMenuItem("View");
```

```
delete6=new JMenuItem("Delete");
Has_MetroStations.add(insert6);
Has_MetroStations.add(view6);
Has_MetroStations.add(delete6);
Hotels=new JMenu("Hotels");
insert7=new JMenuItem("Insert");
view7=new JMenuItem("View");
delete7=new JMenuItem("Delete");
Hotels.add(insert7);
Hotels.add(view7);
Hotels.add(delete7);
Has_Hotels=new JMenu("Has_Hotels");
insert8=new JMenuItem("Insert");
view8=new JMenuItem("View");
delete8=new JMenuItem("Delete");
Has_Hotels.add(insert8);
Has_Hotels.add(view8);
Has_Hotels.add(delete8);
Landmarks=new JMenu("Landmarks");
insert9=new JMenuItem("Insert");
view9=new JMenuItem("View");
delete9=new JMenuItem("Delete");
Landmarks.add(insert9);
Landmarks.add(view9);
Landmarks.add(delete9);
this.setVisible(true);
this.setSize(1000, 500);
```

```
this.add(panel);

this.setTitle("Hotels : Insert");

this.setJMenuBar(menubar);

this.add(panel2,BorderLayout.PAGE_END);

panel2.add(btnHome);

this.setDefaultCloseOperation(3);

panel.add(lblhotel);

panel.add(txthotel);

panel.add(txthotel);

panel.add(lbllandmark);

panel.add(txtlandmark);

panel.add(lblphonenummer);

panel.add(txtphonenummer);

panel.add(lblrating);

panel.add(cmbratings);

panel.add(btninsert);

panel.add(btnclear);

panel.add(txtmessage);

menubar.add(MyCityTable);

menubar.add(Services);

menubar.add(Tourist_Places);

menubar.add(Has_Tourist_Places);

menubar.add(MetroStations);

menubar.add(Has_MetroStations);

menubar.add(Hotels);

menubar.add(Has_Hotels);

menubar.add(Landmarks);
```

```
btnHome.addActionListener(new ActionListener() {
```

```
    @Override
```

```
    public void actionPerformed(ActionEvent arg0) {
```

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

```
        new Home();
```

```
        dispose();
```

```
    }
```

```
});
```

```
btninsert.addActionListener(new ActionListener() {
```

```
    @Override
```

```
    public void actionPerformed(ActionEvent arg0) {
```

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

```
        try{
```

```
            String hotelname=txthotel.getText();
```

```
            String landmark=txtlandmark.getText();
```

```
            String phonenumber=txtphonenumber.getText();
```

```
            String rating=cmbratings.getSelectedItem()+"";
```

```
            if(hotelname.compareTo("")==0)
```

```
            {
```

```
                JOptionPane.showMessageDialog(null, "Enter  
Hotel Name");
```

```
            }
```

```
            else if(landmark.compareTo("")==0)
```

```

        {
            JOptionPane.showMessageDialog(null, "Enter
Landmark");
        }
        else if(phonenummer.compareTo("")==0 )
        {
            JOptionPane.showMessageDialog(null, "Enter
Phone Number");
        }
        else if(phonenummer.length()!=10)
        {
            JOptionPane.showMessageDialog(null, "Enter
Valid Phone Number of 10 Digits");
        }
        else if(rating.compareTo("0")==0)
        {
            JOptionPane.showMessageDialog(null,
"Rating Can't be Zero 0");
        }
        else
        {

            Class.forName("oracle.jdbc.driver.OracleDriver");

            Connection
con=DriverManager.getConnection(

            "jdbc:oracle:thin:@localhost:1521:xe","yasho","password");

```

```

Statement stmt=con.createStatement();

int rows=stmt.executeUpdate("insert into
hotels values('"+hotelname+"','"+landmark+"','"+phonenummer+"','"+rating+"')");

txtmessage.setText("Inserted "+rows+"
row");

con.commit();

con.close();

}

}

catch(Exception e)

{

    System.out.println(e);

}

}

});

btnclear.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    txthotel.setText(null);

    txtlandmark.setText(null);

    txtphonenummer.setText(null);

    cmbratings.setSelectedIndex(0);

}

```

```

});

insert1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MyCityTableInsert();

        dispose();

    }

});

view1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MyCityTableView();

        dispose();

    }

});

delete1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MyCityTableDelete();

        dispose();

    }

});

```

```

});

insert2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new ServicesInsert();

        dispose();

    }

});

view2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new ServicesView();

        dispose();

    }

});

delete2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new ServicesDelete();

        dispose();

    }

});

```



```

});

insert3.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TouristPlacesInsert();

        dispose();

    }

});

view3.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TouristPlacesView();

        dispose();

    }

});

delete3.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TouristPlacesDelete();

        dispose();

```

```

        }

    });

    insert4.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasTouristPlacesInsert();

            dispose();

        }

    });

    view4.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasTouristPlacesView();

            dispose();

        }

    });

    delete4.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasTouristPlacesDelete();

            dispose();

```

```

        }

});

insert5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MetroStationsInsert();

        dispose();

    }

});

view5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MetroStationsView();

        dispose();

    }

});

delete5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MetroStationsDelete();

        dispose();

    }

});

```

```

        }
    });

insert6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasMetroStationsInsert();
        dispose();
    }

});

view6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasMetroStationView();
        dispose();
    }

});

delete6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasMetroStationDelete();
        dispose();
    }

});

```

```

        }
    });

insert7.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HotelsInsert();
        dispose();

    }

});

view7.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HotelsView();
        dispose();

    }

});

delete7.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HotelsDelete();
        dispose();
    }
});

```

```

        }
    });

    insert8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasHotelsInsert();

            dispose();

        }

    });

    view8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasHotelsView();

            dispose();

        }

    });

    delete8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasHotelsDelete();

            dispose();

```

```

        }

    });

    insert9.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent arg0) {

            // TODO Auto-generated method stub

            new LandmarksInsert();

            dispose();

        }

    });

    view9.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent arg0) {

            // TODO Auto-generated method stub

            new LandmarksView();

            dispose();

        }

    });

    delete9.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent arg0) {

            // TODO Auto-generated method stub

            new LandmarksDelete();

```

```

        dispose();
    }

});

}

public static void main(String[] args) {
    // TODO Auto-generated method stub
    new HotelsInsert();
}

}

```

## **Program For Update Of Hotels Table:**

```

import java.awt.*;
import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.ArrayList;

import javax.swing.*;

```



```
public class HotelsView extends JFrame{  
    private JPanel panel2;  
    private JButton btnHome;  
    private JPanel panel;  
    private JLabel lblselecthotel;  
    private JLabel lblhotel;  
    private JLabel lbllandmark;  
    private JLabel lblphonenumber;  
    private JLabel lblrating;  
    private JComboBox<String> cmbhotels;  
    private JTextField txthotel;  
    private JTextField txtlandmark;  
    private JTextField txtphonenumber;  
    private JTextField txtrating;  
    private JButton btnupdate;  
    private JTextArea txtmessage;  
    private JMenu MyCityTable;  
    private JMenu Services;  
    private JMenu Tourist_Places;  
    private JMenu Has_Tourist_Places;  
    private JMenu MetroStations;  
    private JMenu Has_MetroStations;  
    private JMenu Hotels;  
    private JMenuItem insert9;  
    private JMenuItem view9;  
    private JMenuItem delete9;  
    private JMenu Landmarks;
```

```
private JMenu Has_Hotels;
private JMenuBar menubar;
private JMenuItem insert1;
private JMenuItem delete1;
private JMenuItem view1;
private JMenuItem insert2;
private JMenuItem delete2;
private JMenuItem view2;
private JMenuItem insert3;
private JMenuItem delete3;
private JMenuItem view3;
private JMenuItem insert4;
private JMenuItem delete4;
private JMenuItem view4;
private JMenuItem insert5;
private JMenuItem delete5;
private JMenuItem view5;
private JMenuItem insert6;
private JMenuItem delete6;
private JMenuItem view6;
private JMenuItem insert7;
private JMenuItem delete7;
private JMenuItem view7;
private JMenuItem insert8;
private JMenuItem delete8;
private JMenuItem view8;
private ArrayList hotelsList;
```

```

public HotelsView()
{
    try{

        Class.forName("oracle.jdbc.driver.OracleDriver");

        Connection con=DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe","yasho","password");

        Statement stmt=con.createStatement();
        hotelsList=new ArrayList();
        ResultSet rs=stmt.executeQuery("select hotel_name from hotels");
        while(rs.next())
            hotelsList.add(rs.getString(1));

        con.close();

    }
    catch(Exception e)
    {
        System.out.println(e);
    }
    panel2=new JPanel();
    btnHome=new JButton("Home");
    panel2.setBackground(Color.DARK_GRAY);
    panel=new JPanel(new FlowLayout());

```

```
lblselecthotel=new JLabel("Select Hotel:");
lblhotel=new JLabel("Hotel");
lbllandmark=new JLabel("Landmark:");
lblphonenumber=new JLabel("Phone Number:");
lblrating=new JLabel("Rating:");
cmbhotels=new JComboBox(hotelsList.toArray());
txthotel=new JTextField(20);
txtlandmark=new JTextField(20);
txtphonenumber=new JTextField(10);
txtrating=new JTextField(3);
btnupdate=new JButton("Modify");
txtmessage=new JTextArea(10,40);
menubar=new JMenuBar();
MyCityTable=new JMenu("MyCityTable");
insert1=new JMenuItem("Insert");
view1=new JMenuItem("View");
delete1=new JMenuItem("Delete");
MyCityTable.add(insert1);
MyCityTable.add(view1);
MyCityTable.add(delete1);
Services=new JMenu("Services");
insert2=new JMenuItem("Insert");
view2=new JMenuItem("View");
delete2=new JMenuItem("Delete");
Services.add(insert2);
Services.add(view2);
Services.add(delete2);
```

```
Tourist_Places=new JMenu("Tourist_Places");
insert3=new JMenuItem("Insert");
view3=new JMenuItem("View");
delete3=new JMenuItem("Delete");
Tourist_Places.add(insert3);
Tourist_Places.add(view3);
Tourist_Places.add(delete3);

Has_Tourist_Places=new JMenu("Has_Tourist_Places");
insert4=new JMenuItem("Insert");
view4=new JMenuItem("View");
delete4=new JMenuItem("Delete");
Has_Tourist_Places.add(insert4);
Has_Tourist_Places.add(view4);
Has_Tourist_Places.add(delete4);

MetroStations=new JMenu("MetroStations");
insert5=new JMenuItem("Insert");
view5=new JMenuItem("View");
delete5=new JMenuItem("Delete");
MetroStations.add(insert5);
MetroStations.add(view5);
MetroStations.add(delete5);

Has_MetroStations=new JMenu("Has_MetroStations");
insert6=new JMenuItem("Insert");
view6=new JMenuItem("View");
delete6=new JMenuItem("Delete");
Has_MetroStations.add(insert6);
Has_MetroStations.add(view6);
```

```
Has_MetroStations.add(delete6);

Hotels=new JMenu("Hotels");

insert7=new JMenuItem("Insert");

view7=new JMenuItem("View");

delete7=new JMenuItem("Delete");

Hotels.add(insert7);

Hotels.add(view7);

Hotels.add(delete7);

Has_Hotels=new JMenu("Has_Hotels");

insert8=new JMenuItem("Insert");

view8=new JMenuItem("View");

delete8=new JMenuItem("Delete");

Has_Hotels.add(insert8);

Has_Hotels.add(view8);

Has_Hotels.add(delete8);

this.setVisible(true);

this.setSize(1000, 500);

this.add(panel);

this.setTitle("Hotels : View");

this.setJMenuBar(menubar);

this.add(panel2,BorderLayout.PAGE_END);

panel2.add(btnHome);

this.setDefaultCloseOperation(3);

panel.add(lblselecthotel);

panel.add(cmbhotels);

panel.add(lblhotel);

panel.add(txthotel);
```

```
panel.add(lbllandmark);
panel.add(txtlandmark);
panel.add(lblphonenumber);
panel.add(txtphonenumber);
panel.add(lblrating);
panel.add(txtrating);
panel.add(btnupdate);
panel.add(txtmessage);
menubar.add(MyCityTable);
menubar.add(Services);
menubar.add(Tourist_Places);
menubar.add(Has_Tourist_Places);
menubar.add(MetroStations);
menubar.add(Has_MetroStations);
menubar.add(Hotels);
menubar.add(Has_Hotels);
Landmarks=new JMenu("Landmarks");
menubar.add(Landmarks);

insert9=new JMenuItem("Insert");
view9=new JMenuItem("View");
delete9=new JMenuItem("Delete");
Landmarks.add(insert9);
Landmarks.add(view9);
Landmarks.add(delete9);
btnHome.addActionListener(new ActionListener() {
```

```

@Override

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    new Home();

    dispose();

}

});

insert9.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    new LandmarksInsert();

    dispose();

}

});

view9.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    new LandmarksView();

    dispose();

}

});

delete9.addActionListener(new ActionListener() {

```



```

@Override

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    new LandmarksDelete();

    dispose();

}

```

```

});

```

```

cmbhotels.addActionListener(new ActionListener() {

```

```

@Override

```

```

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    try{

```

```

        Class.forName("oracle.jdbc.driver.OracleDriver");

```

```

        Connection con=DriverManager.getConnection(

```

```

            "jdbc:oracle:thin:@localhost:1521:xe","yasho","password");

```

```

        Statement stmt=con.createStatement();

```

```

        ResultSet rs=stmt.executeQuery("select * from
hotels where hotel_name='"+cmbhotels.getSelectedItem()+"");

```

```

        while(rs.next())

```

```

        {
            txthotel.setText(rs.getString(1));
            txtlandmark.setText(rs.getString(2));
            txtphonenummer.setText(rs.getString(3));
            txtrating.setText(rs.getString(4));
        }

        con.close();

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

btnupdate.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent arg0) {
        // TODO Auto-generated method stub

        if(txthotel.getText().compareTo("")==0)
            JOptionPane.showMessageDialog(null,"Enter
Hotel Name");

        else if(txtlandmark.getText().compareTo("")==0)
            JOptionPane.showMessageDialog(null, "Enter
Landmark");

        else
            if(txtphonenummer.getText().compareTo("")==0)

```

```
OptionPane.showMessageDialog(null, "Enter  
Phone Number");
```

```
else if(txtrating.getText().compareTo("")==0)  
OptionPane.showMessageDialog(null, "Enter  
Rating");
```

```
else  
{  
try{
```

```
Class.forName("oracle.jdbc.driver.OracleDriver");
```

```
Connection  
con=DriverManager.getConnection(  

```

```
"jdbc:oracle:thin:@localhost:1521:xe","yasho","password");
```

```
Statement  
stmt=con.createStatement();
```

```
String  
hotelname=txthotel.getText();
```

```
String  
landmark=txtlandmark.getText();
```

```
String  
phonenummer=txtphonenummer.getText();
```

```
String  
rating=txtrating.getText();
```

```
int  
rows=stmt.executeUpdate("update hotels set  
hotel_name='"+hotelname+"',landmark='"+landmark+"',phone_number='"+phonenumbe  
r+"',rating='"+rating+" where hotel_name='"+cmbhotels.getSelectedItemAt()+"");
```

```
txtmessage.setText("Updated  
"+rows+" row");
```

```

                                con.commit();
                                hotelsList=new ArrayList();
                                ResultSet
rs=stmt.executeQuery("select hotel_name from hotels");
                                while(rs.next())

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

                                int
count=cmbhotels.getItemCount();

                                for(int i=0;i<count;i++)

                                cmbhotels.removeItemAt(0);

                                for(int
i=0;i<hotelsList.size();i++)

                                cmbhotels.addItem(hotelsList.get(i)+"");

                                con.close();

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

                                }

                                });

                                insert1.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new MyCityTableInsert();

    dispose();

}

});

view1.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new MyCityTableView();

    dispose();

}

});

delete1.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new MyCityTableDelete();

    dispose();

}

});

insert2.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new ServicesInsert();

    dispose();

}

});

view2.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new ServicesView();

    dispose();

}

});

delete2.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new ServicesDelete();

    dispose();

}

});

insert3.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new TouristPlacesInsert();

    dispose();

}

});

view3.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new TouristPlacesView();

    dispose();

}

});

delete3.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new TouristPlacesDelete();

    dispose();

}

});

insert4.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HasTouristPlacesInsert();

    dispose();

}

});

view4.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HasTouristPlacesView();

    dispose();

}

});

delete4.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HasTouristPlacesDelete();

    dispose();

}

});

insert5.addActionListener(new ActionListener() {

```



```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new MetroStationsInsert();

    dispose();

}

});

view5.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new MetroStationsView();

    dispose();

}

});

delete5.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new MetroStationsDelete();

    dispose();

}

});

insert6.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HasMetroStationsInsert();

    dispose();

}

});

view6.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HasMetroStationView();

    dispose();

}

});

delete6.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HasMetroStationDelete();

    dispose();

}

});

insert7.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HotelsInsert();

    dispose();

}

});

view7.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HotelsView();

    dispose();

}

});

delete7.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub
    new HotelsDelete();

    dispose();

}

});

insert8.addActionListener(new ActionListener() {

```

```

        @Override
        public void actionPerformed(ActionEvent e) {
            // TODO Auto-generated method stub
            new HasHotelsInsert();
            dispose();
        }
    });
    view8.addActionListener(new ActionListener() {

        @Override
        public void actionPerformed(ActionEvent e) {
            // TODO Auto-generated method stub
            new HasHotelsView();
            dispose();
        }
    });
    delete8.addActionListener(new ActionListener() {

        @Override
        public void actionPerformed(ActionEvent e) {
            // TODO Auto-generated method stub
            new HasHotelsDelete();
            dispose();
        }
    });
}

```

```

        public static void main(String[] args) {

            // TODO Auto-generated method stub

            new HotelsView();

        }

    }

```

### **Program for Delete Operation in Hotels Table:**

```

import java.awt.*;
import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.ArrayList;

import javax.swing.*;

public class HotelsDelete extends JFrame{

    private JPanel panel2;

    private JButton btnHome;

    private JPanel panel;

    private JLabel lblselecthotel;

```

```
private JLabel lblhotel;  
private JLabel lbllandmark;  
private JLabel lblphonenumber;  
private JLabel lblrating;  
private JComboBox<String> cmbhotels;  
private JTextField txthotel;  
private JTextField txtlandmark;  
private JTextField txtphonenumber;  
private JTextField txtrating;  
private JButton btndelete;  
private JTextArea txtmessage;  
private JMenu MyCityTable;  
private JMenu Services;  
private JMenu Tourist_Places;  
private JMenu Has_Tourist_Places;  
private JMenu MetroStations;  
private JMenu Has_MetroStations;  
private JMenu Hotels;  
private JMenu Has_Hotels;  
private JMenuItem insert9;  
private JMenuItem view9;  
private JMenuItem delete9;  
private JMenu Landmarks;  
private JMenuBar menubar;  
private JMenuItem insert1;  
private JMenuItem delete1;  
private JMenuItem view1;
```

```
private JMenuItem insert2;
private JMenuItem delete2;
private JMenuItem view2;
private JMenuItem insert3;
private JMenuItem delete3;
private JMenuItem view3;
private JMenuItem insert4;
private JMenuItem delete4;
private JMenuItem view4;
private JMenuItem insert5;
private JMenuItem delete5;
private JMenuItem view5;
private JMenuItem insert6;
private JMenuItem delete6;
private JMenuItem view6;
private JMenuItem insert7;
private JMenuItem delete7;
private JMenuItem view7;
private JMenuItem insert8;
private JMenuItem delete8;
private JMenuItem view8;
private ArrayList hotelList;
public HotelsDelete()
{
    try{
```

```
        Class.forName("oracle.jdbc.driver.OracleDriver");
```

```

        Connection con=DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe","yasho","password");

        Statement stmt=con.createStatement();

        hotelList=new ArrayList();

        ResultSet rs=stmt.executeQuery("select hotel_name from hotels");
        while(rs.next())
            hotelList.add(rs.getString(1));

        con.close();

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

    panel2=new JPanel();
    btnHome=new JButton("Home");
    panel2.setBackground(Color.DARK_GRAY);
    panel=new JPanel(new FlowLayout());
    lblselecthotel=new JLabel("Select Hotel:");
    lblhotel=new JLabel("Hotel");
    lbllandmark=new JLabel("Landmark:");
    lblphonenumber=new JLabel("Phone NUmber:");
    lblrating=new JLabel("Rating:");

```



```
cmbhotels=new JComboBox(hotelList.toArray());
txthotel=new JTextField(20);
txtlandmark=new JTextField(20);
txtphonenummer=new JTextField(10);
txtrating=new JTextField(3);
btndelete=new JButton("Delete");
txtmessage=new JTextArea(10,40);
menubar=new JMenuBar();
MyCityTable=new JMenu("MyCityTable");
insert1=new JMenuItem("Insert");
view1=new JMenuItem("View");
delete1=new JMenuItem("Delete");
MyCityTable.add(insert1);
MyCityTable.add(view1);
MyCityTable.add(delete1);
Services=new JMenu("Services");
insert2=new JMenuItem("Insert");
view2=new JMenuItem("View");
delete2=new JMenuItem("Delete");
Services.add(insert2);
Services.add(view2);
Services.add(delete2);
Tourist_Places=new JMenu("Tourist_Places");
insert3=new JMenuItem("Insert");
view3=new JMenuItem("View");
delete3=new JMenuItem("Delete");
Tourist_Places.add(insert3);
```

```
Tourist_Places.add(view3);
Tourist_Places.add(delete3);
Has_Tourist_Places=new JMenu("Has_Tourist_Places");
insert4=new JMenuItem("Insert");
view4=new JMenuItem("View");
delete4=new JMenuItem("Delete");
Has_Tourist_Places.add(insert4);
Has_Tourist_Places.add(view4);
Has_Tourist_Places.add(delete4);
MetroStations=new JMenu("MetroStations");
insert5=new JMenuItem("Insert");
view5=new JMenuItem("View");
delete5=new JMenuItem("Delete");
MetroStations.add(insert5);
MetroStations.add(view5);
MetroStations.add(delete5);
Has_MetroStations=new JMenu("Has_MetroStations");
insert6=new JMenuItem("Insert");
view6=new JMenuItem("View");
delete6=new JMenuItem("Delete");
Has_MetroStations.add(insert6);
Has_MetroStations.add(view6);
Has_MetroStations.add(delete6);
Hotels=new JMenu("Hotels");
insert7=new JMenuItem("Insert");
view7=new JMenuItem("View");
delete7=new JMenuItem("Delete");
```

```
Hotels.add(insert7);
Hotels.add(view7);
Hotels.add(delete7);
Has_Hotels=new JMenu("Has_Hotels");
insert8=new JMenuItem("Insert");
view8=new JMenuItem("View");
delete8=new JMenuItem("Delete");
Has_Hotels.add(insert8);
Has_Hotels.add(view8);
Has_Hotels.add(delete8);
this.setVisible(true);
this.setSize(1000, 500);
this.add(panel);
this.setTitle("Hotels : Delete");
this.setJMenuBar(menubar);
this.setDefaultCloseOperation(3);
this.add(panel2,BorderLayout.PAGE_END);
panel2.add(btnHome);
panel.add(lblselecthotel);
panel.add(cmbhotels);
panel.add(lblhotel);
panel.add(txthotel);
panel.add(lbllandmark);
panel.add(txtlandmark);
panel.add(lblphonenumber);
panel.add(txtphonenumber);
panel.add(lblrating);
```

```

panel.add(txtrating);
panel.add(btndelete);
panel.add(txtmessage);
menubar.add(MyCityTable);
menubar.add(Services);
menubar.add(Tourist_Places);
menubar.add(Has_Tourist_Places);
menubar.add(MetroStations);
menubar.add(Has_MetroStations);
menubar.add(Hotels);
menubar.add(Has_Hotels);
txthotel.setEditable(false);
txtlandmark.setEditable(false);
txtphonenummer.setEditable(false);
txtrate.setEditable(false);
Landmarks=new JMenu("Landmarks");
menubar.add(Landmarks);

insert9=new JMenuItem("Insert");
view9=new JMenuItem("View");
delete9=new JMenuItem("Delete");
Landmarks.add(insert9);
Landmarks.add(view9);
Landmarks.add(delete9);
btnHome.addActionListener(new ActionListener() {

```

```

@Override

```

```

        public void actionPerformed(ActionEvent arg0) {
            // TODO Auto-generated method stub
            new Home();
            dispose();
        }
    });

```

```

insert9.addActionListener(new ActionListener() {

```

```

    @Override

```

```

    public void actionPerformed(ActionEvent arg0) {
        // TODO Auto-generated method stub
        new LandmarksInsert();
        dispose();
    }

```

```

});

```

```

view9.addActionListener(new ActionListener() {

```

```

    @Override

```

```

    public void actionPerformed(ActionEvent arg0) {
        // TODO Auto-generated method stub
        new LandmarksView();
        dispose();
    }

```

```

});

```

```

delete9.addActionListener(new ActionListener() {

```

```

@Override

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    new LandmarksDelete();

    dispose();

}

});

cmbhotels.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent arg0) {

    // TODO Auto-generated method stub

    try{

        Class.forName("oracle.jdbc.driver.OracleDriver");

        Connection con=DriverManager.getConnection(

            "jdbc:oracle:thin:@localhost:1521:xe","yasho","password");

        Statement stmt=con.createStatement();

        ResultSet rs=stmt.executeQuery("select * from

hotels where hotel_name='"+cmbhotels.getSelectedItem()+"'");

        while(rs.next())

        {

```

```

        txthotel.setText(rs.getString(1));
        txtlandmark.setText(rs.getString(2));
        txtphonenummer.setText(rs.getString(3)+ "");
        txtrating.setText(rs.getString(4)+ "");
    }

    con.close();

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

}

});

btndelete.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent arg0) {
        // TODO Auto-generated method stub
        try{

            Class.forName("oracle.jdbc.driver.OracleDriver");

            Connection con=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","yasho","password");

            Statement stmt=con.createStatement();

```

```

        int rows=stmt.executeUpdate("delete from hotels
where hotel_name='"+cmbhotels.getSelectedItem()+"");

        System.out.println("yes");

        txtmessage.setText("Deleted "+rows+" rows");

        con.commit();

        hotelList=new ArrayList();

        ResultSet rs=stmt.executeQuery("select hotel_name
from hotels");

        while(rs.next())

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

        int count=cmbhotels.getItemCount();
        for(int i=0;i<count;i++)

            cmbhotels.removeItemAt(0);

        for(int i=0;i<hotelList.size();i++)
        {

            cmbhotels.addItem(hotelList.get(i)+"");

        }


        con.close();

    }

    catch(Exception e)

    {

        System.out.println(e);

    }

}

```



```

});

insert1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MyCityTableInsert();

        dispose();

    }

});

view1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MyCityTableView();

        dispose();

    }

});

delete1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new MyCityTableDelete();

        dispose();

    }

});

```

```

});

insert2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new ServicesInsert();

        dispose();

    }

});

view2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new ServicesView();

        dispose();

    }

});

delete2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new ServicesDelete();

        dispose();

    }

}

```

```

});

insert3.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TouristPlacesInsert();

        dispose();

    }

});

```

```

view3.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TouristPlacesView();

        dispose();

    }

});

```

```

delete3.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TouristPlacesDelete();

        dispose();

    }

});

```

```

        }

    });

    insert4.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasTouristPlacesInsert();

            dispose();

        }

    });

    view4.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasTouristPlacesView();

            dispose();

        }

    });

    delete4.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasTouristPlacesDelete();

            dispose();

```

```

        }

    });

    insert5.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new MetroStationsInsert();

            dispose();

        }

    });

    view5.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new MetroStationsView();

            dispose();

        }

    });

    delete5.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new MetroStationsDelete();

            dispose();

```

```

        }
    });

insert6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasMetroStationsInsert();
        dispose();
    }

});

view6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasMetroStationView();
        dispose();
    }

});

delete6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new HasMetroStationDelete();
        dispose();
    }

});

```

```

        }

    });

    insert7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HotelsInsert();

            dispose();

        }

    });

    view7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HotelsView();

            dispose();

        }

    });

    delete7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HotelsDelete();

            dispose();

```

```

        }

    });

    insert8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasHotelsInsert();

            dispose();

        }

    });

    view8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasHotelsView();

            dispose();

        }

    });

    delete8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new HasHotelsDelete();

            dispose();

```



```

    }

    });

}

public static void main(String[] args) {

    // TODO Auto-generated method stub

    new HotelsDelete();

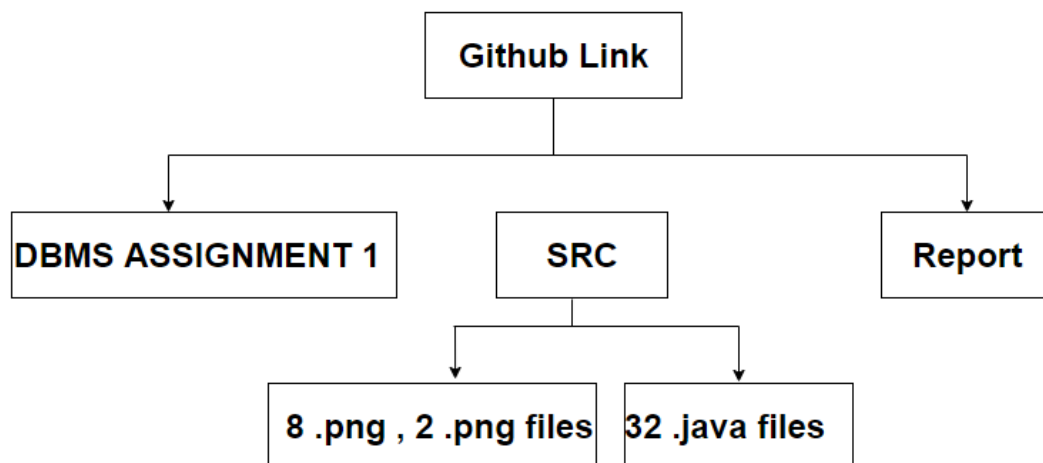
}

}

```

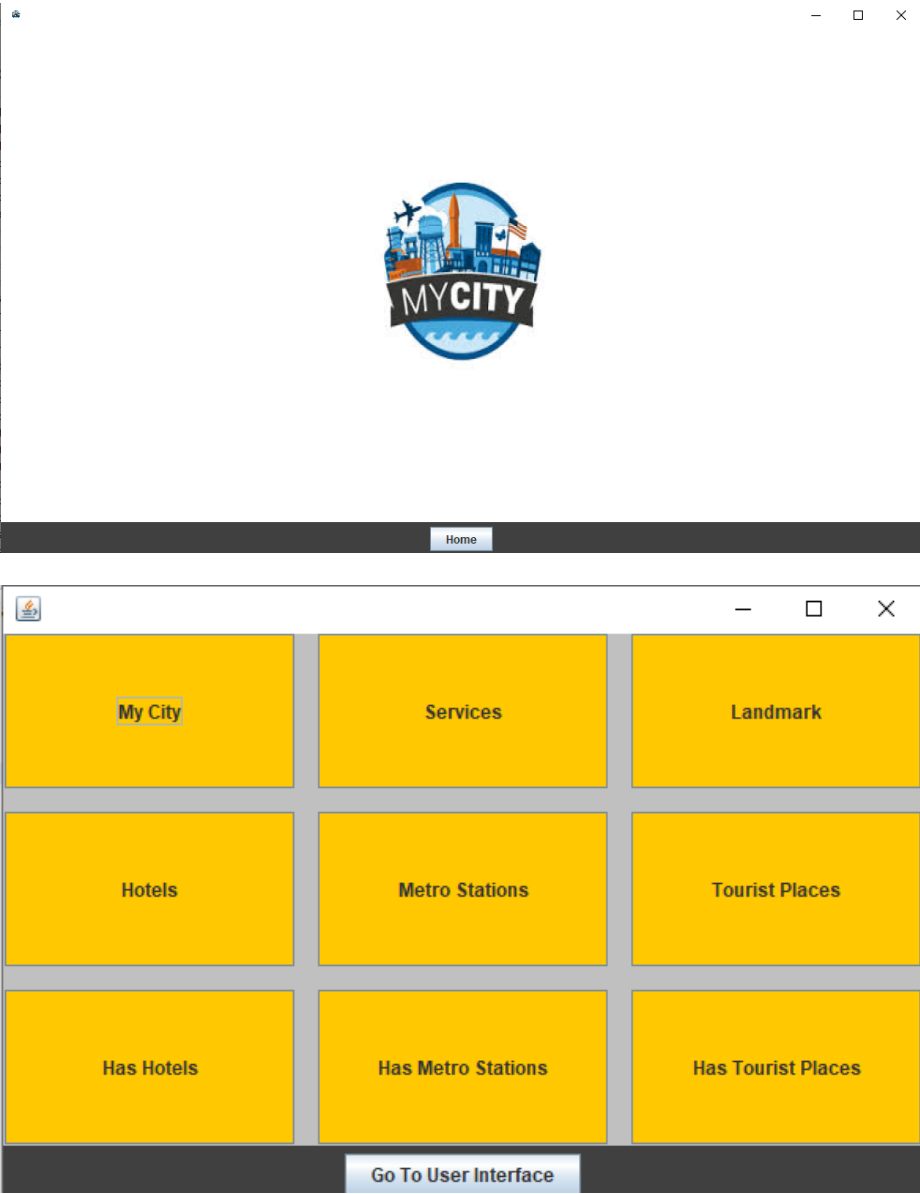
## GITHUB LINK AND FOLDER STRUCTURE:

<https://github.com/dyashovardhanreddy/MYCITYINFORMATION>



## OUTPUT :

GUI For Main Frame:



Error Messages:

Hotels : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Hotel:  Landmark:  Phone Number:  Rating:

Message

Enter Valid Phone Number of 10 Digits

Home

Hotels : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Hotel:  Landmark:  Phone Number:  Rating:

Message

Rating Can't be Zero 0

Home

TouristPlaces : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Place Name:  Landmark:  Rating:

java.sql.SQLException: ORA-12899: value too large for column "YASHO"."TOURIST\_PLACES"."PLACE\_NAME" (actual: 21, maximum: 15)

Home

## Java GUI Insert in Hotels:

Hotels : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Hotel: MyChoice Landmark: M.J.Market Phone Number: 5656545492 Rating: 7 Insert

Inserted 1 row

Clear

Home

```
Run SQL Command Line

SQL> select * from hotels;

HOTEL_NAME  LANDMARK  PHONE_NUMBER  RATING
-----
Oyo         Abids     3214567897    9
Revolt2.0   Jubilee Hills  4569871236    8

SQL> select * from hotels;

HOTEL_NAME  LANDMARK  PHONE_NUMBER  RATING
-----
Oyo         Abids     3214567897    9
Revolt2.0   Jubilee Hills  4569871236    8
MyChoice    M.J.Market  5656545492    7

SQL>
```

## Update GUI for Hotels Table:

The screenshot shows a web application window titled "Hotels : View". It has a navigation bar with links: MyCityTable, Services, Tourist\_Places, Has\_Tourist\_Places, MetroStations, Has\_MetroStations, Hotels, Has\_Hotels, and Landmarks. Below the navigation bar, there are input fields for "Select Hotel:" (a dropdown menu with "MyChoice" selected), "Hotel:" (a text input with "MyChoice"), "Landmark:" (a text input with "M.J.Market"), "Phone Number:" (a text input with "5656545697"), and "Rating:" (a text input). A "Modify" button is located below the "Hotel:" field. A message "Updated 1 row" is displayed in a white box. At the bottom of the window, there is a "Home" button.

## DataBase Before And After Update in Hotels:

```
Run SQL Command Line

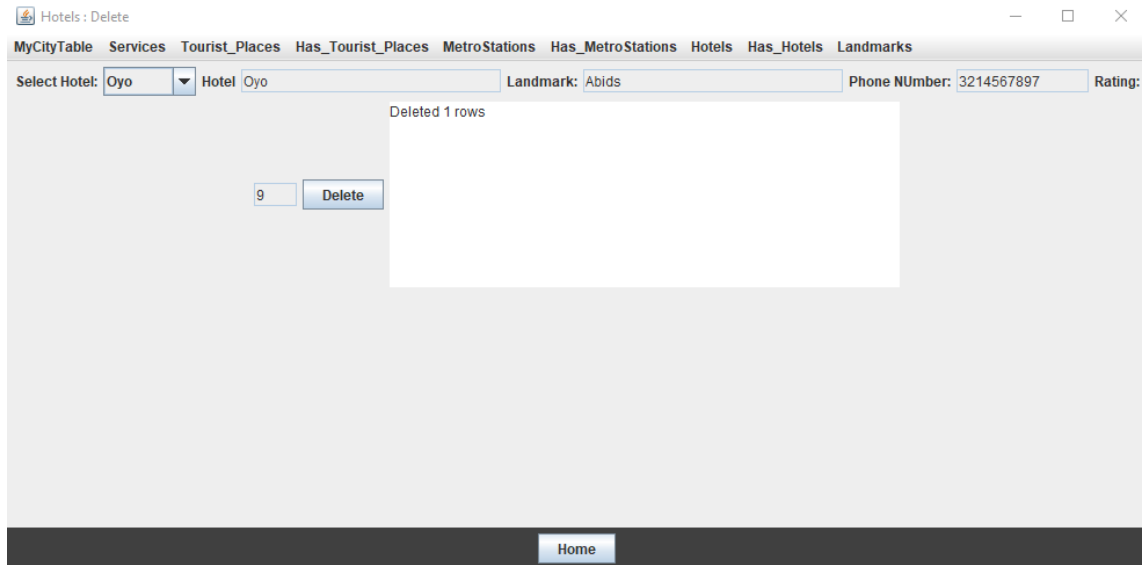
SQL> select * from hotels;

HOTEL_NAME      LANDMARK      PHONE_NUMBER      RATING
-----
Oyo              Abids         3214567897        9
Revolt2.0        Jubilee Hills 4569871236        8
MyChoice         M.J.Market    5656545492        7

SQL> select * from hotels;

HOTEL_NAME      LANDMARK      PHONE_NUMBER      RATING
-----
Oyo              Abids         3214567897        9
Revolt2.0        Jubilee Hills 4569871236        8
MyChoice         M.J.Market    5656545697        7
```

## GUI for Delete operation on Hotels Table:



## DataBase Before and After Delete:

```
Run SQL Command Line

SQL> select * from hotels;

HOTEL_NAME      LANDMARK      PHONE_NUMBER      RATING
-----
Oyo              Abids         3214567897        9
Revolt2.0       Jubilee Hills 4569871236        8
MyChoice        M.J.Market    5656545697        7

SQL> select * from hotels;

HOTEL_NAME      LANDMARK      PHONE_NUMBER      RATING
-----
Oyo              Abids         3214567897        9
Revolt2.0       Jubilee Hills 4569871236        8
```

## Other GUI For View, Insert, Delete:

### My City Table:

MyCity Table View

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Select City:  CityName:

MyCity Table : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

CityName:   Message:

1 row inserted successfully

# Run SQL Command Line

```
jaipur
kochi
kolkata

CITYNAME
-----
lucknow
mumbai
new delhi
noida
visakhapatnam
vizag

17 rows selected.

SQL> select * from mycity;

CITYNAME
-----
Ahmedabad
Bengaluru
Bhopal
Bhubaneswar
Chandigarh
chennai
gurugram
hyderabad
indore
jaipur
kochi

CITYNAME
-----
kolkata
lucknow
mumbai
new delhi
noida
visakhapatnam
vizag

18 rows selected.
```



MyCity Table Delete

MyCityTable

Services

Tourist\_Places

Has\_Tourist\_Places

MetroStations

Has\_MetroStations

Hotels

Has\_Hotels

Landmarks

Select City:

Ahmedabad

CityName:

Delete

Home

```
Run SQL Command Line
indore
jaipur
kochi

CITYNAME
-----
kolkata
lucknow
mumbai
new delhi
noida
visakhapatanam
vizag

18 rows selected.

SQL> select * from mycity;

CITYNAME
-----
Bengaluru
Bhopal
Bhubaneswar
Chandigarh
chennai
gurugram
hyderabad
indore
jaipur
kochi
kolkata

CITYNAME
-----
lucknow
mumbai
new delhi
noida
visakhapatanam
vizag

17 rows selected.
```

## Services Table:

Services : View

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Select CityName: Bengaluru View Service Availiable:

Home

Services : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

CityName:  Service:

Services : Delete

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

CityName:  Service:

## Land Marks Table:

Landmarks : View

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Cityname:  Landmark:  CityName:  Landmark:

Landmarks : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Cityname:  Landmark:

```

Run SQL Command Line
SQL> select * from landmarks;

CITYNAME      LANDMARK
-----
hyderabad     Abids
hyderabad     Ameerpet
hyderabad     Dilsukh Nagar
hyderabad     Golconda
hyderabad     Hussian sagar
hyderabad     Jubilee Hills
hyderabad     L.B.Nagar
hyderabad     Lakdikapool
hyderabad     M.J.Market
hyderabad     Oldcity
mumbai        Andeeri East

CITYNAME      LANDMARK
-----
mumbai        Andeeri West

12 rows selected.

SQL> select * from landmarks;

CITYNAME      LANDMARK
-----
hyderabad     Abids
hyderabad     Ameerpet
hyderabad     Bapu Ghat
hyderabad     Dilsukh Nagar
hyderabad     Golconda
hyderabad     Hussian sagar
hyderabad     Jubilee Hills
hyderabad     L.B.Nagar
hyderabad     Lakdikapool
hyderabad     M.J.Market
hyderabad     Oldcity

CITYNAME      LANDMARK
-----
mumbai        Andeeri East
mumbai        Andeeri West

13 rows selected.

```

Landmarks : Delete

MyCityTable

Services

Tourist\_Places

Has\_Tourist\_Places

MetroStations

Has\_MetroStations

Hotels

Has\_Hotels

Landmarks

Cityname:

Landmark:

CityName:

Landmark:

Delete

Home

#### Run SQL Command Line

```
hyderabad      Abids
hyderabad      Ameerpet
hyderabad      Bapu Ghat
hyderabad      Dilsukh Nagar
hyderabad      Golconda
hyderabad      Hussian sagar
hyderabad      Jubilee Hills
hyderabad      L.B.Nagar
hyderabad      Lakdikapool
hyderabad      M.J.Market
hyderabad      Oldcity

CITYNAME      LANDMARK
-----
mumbai        Andeeri East
mumbai        Andeeri West
mumbai        Dhobi Ghat

14 rows selected.

SQL> select * from landmarks;

CITYNAME      LANDMARK
-----
hyderabad      Abids
hyderabad      Ameerpet
hyderabad      Bapu Ghat
hyderabad      Dilsukh Nagar
hyderabad      Golconda
hyderabad      Hussian sagar
hyderabad      Jubilee Hills
hyderabad      L.B.Nagar
hyderabad      Lakdikapool
hyderabad      M.J.Market
hyderabad      Oldcity

CITYNAME      LANDMARK
-----
mumbai        Andeeri East
mumbai        Andeeri West

13 rows selected.
```

### Metro Stations Table:

Metro Station: View

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Select StationName: Ameerpet M.S StationName: Ameerpet M.S LandMark: Ameerpet Timmings: 7a.m-11.30p.m Update

Home

MetroStation: Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Station Name: L.B.Nagar M.S Landmark: L.B.Nagar Timmings: 5.30a.m-11p.m Insert Clear

Home

#### Run SQL Command Line

```
SQL> select * from metrostations;
```

STATION_NAME	LANDMARK	TIMMINGS
Gandhi Bhavan M.S	M.J.Market	7a.m-11.30p.m
Lakdikapool M.S	Lakdikapool	5.30a.m-11p.m
Nampally M.S	Nampally	6a.m-10.30p.m
Ameerpet M.S	Ameerpet	7a.m-11.30p.m

```
SQL> select * from metrostations;
```

STATION_NAME	LANDMARK	TIMMINGS
Gandhi Bhavan M.S	M.J.Market	7a.m-11.30p.m
Lakdikapool M.S	Lakdikapool	5.30a.m-11p.m
Nampally M.S	Nampally	6a.m-10.30p.m
L.B.Nagar M.S	L.B.Nagar	5.30a.m-11p.m
Ameerpet M.S	Ameerpet	7a.m-11.30p.m

Metro Station: Delete

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Select StationName: **L.B.Nagar M.S** StationName: L.B.Nagar M.S LandMark: L.B.Nagar Timmings: 5.30a.m-11p.m **Delete**

**Home**

#### Run SQL Command Line

```
SQL> select * from metrostations;
```

STATION_NAME	LANDMARK	TIMMINGS
Gandhi Bhavan M.S	M.J.Market	7a.m-11.30p.m
Lakdikapool M.S	Lakdikapool	5.30a.m-11p.m
Nampally M.S	Nampally	6a.m-10.30p.m
Ameerpet M.S	Ameerpet	7a.m-11.30p.m
<b>L.B.Nagar M.S</b>	<b>L.B.Nagar</b>	<b>5.30a.m-11p.m</b>

```
SQL> select * from metrostations;
```

STATION_NAME	LANDMARK	TIMMINGS
Gandhi Bhavan M.S	M.J.Market	7a.m-11.30p.m
Lakdikapool M.S	Lakdikapool	5.30a.m-11p.m
Nampally M.S	Nampally	6a.m-10.30p.m
Ameerpet M.S	Ameerpet	7a.m-11.30p.m

## Tourist Places Table:



Tourist Places: View

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Select PlaceName: **Birla Temple** PlaceName: Birla Temple LandMark: Lakdikapool Rating: 9 **Update**

**Home**

TouristPlaces: Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Place Name: Nehru Zoo Park Landmark: Bahadurpura Rating: 10 **Insert** **Clear**

Insert 1 row

**Home**

Tourist Places: Delete

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Select PlaceName: **Nehru Zoo Park** PlaceName: Nehru Zoo Park LandMark: Bahadurpura Rating: 10 **Delete**

**Home**

## Has Hotels Table:

Has Hotels : View

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

City:  Hotels:  City Name:  Hotel Name:

[Home](#)

Has Hotels : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

CityName:  Hotel:

[Home](#)

Has Hotels : Delete
—
□
×

[MyCityTable](#)
[Services](#)
[Tourist\\_Places](#)
[Has\\_Tourist\\_Places](#)
[MetroStations](#)
[Has\\_MetroStations](#)
[Hotels](#)
[Has\\_Hotels](#)
[Landmarks](#)

Select CityName:

Select Hotel:

Cityname:

Hotel:

Delete

Home

## Has Metro Stations Table:

Has MetroStation : View
—
□
×

[MyCityTable](#)
[Services](#)
[Tourist\\_Places](#)
[Has\\_Tourist\\_Places](#)
[MetroStations](#)
[Has\\_MetroStations](#)
[Hotels](#)
[Has\\_Hotels](#)
[Landmarks](#)

CityName:

MetroStations:

Cityname:

Metro Station:

Home

Has MetroStation : Insert

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

CityName:  MetroStation:

Has MetroStation : Delete

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

CityName:  MetroStation:  Cityname:  MetroStation:

## Has Tourist Places Table:

Has Tourist Places : View

MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

Select CityName:  Placename Available:  Cityname:  PlaceName:

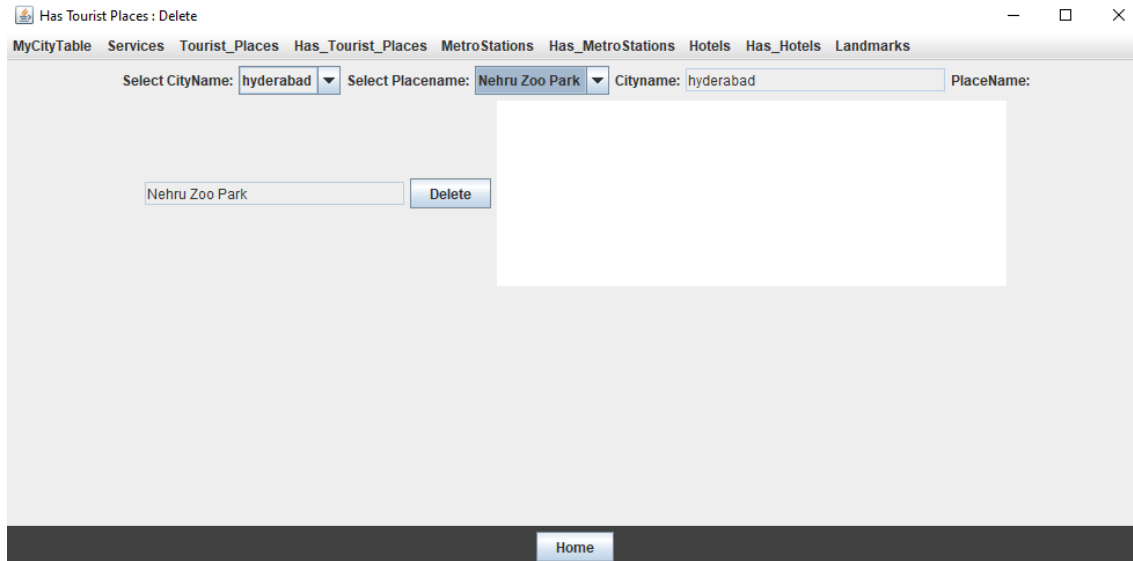
Home

Has Tourist Places : Insert

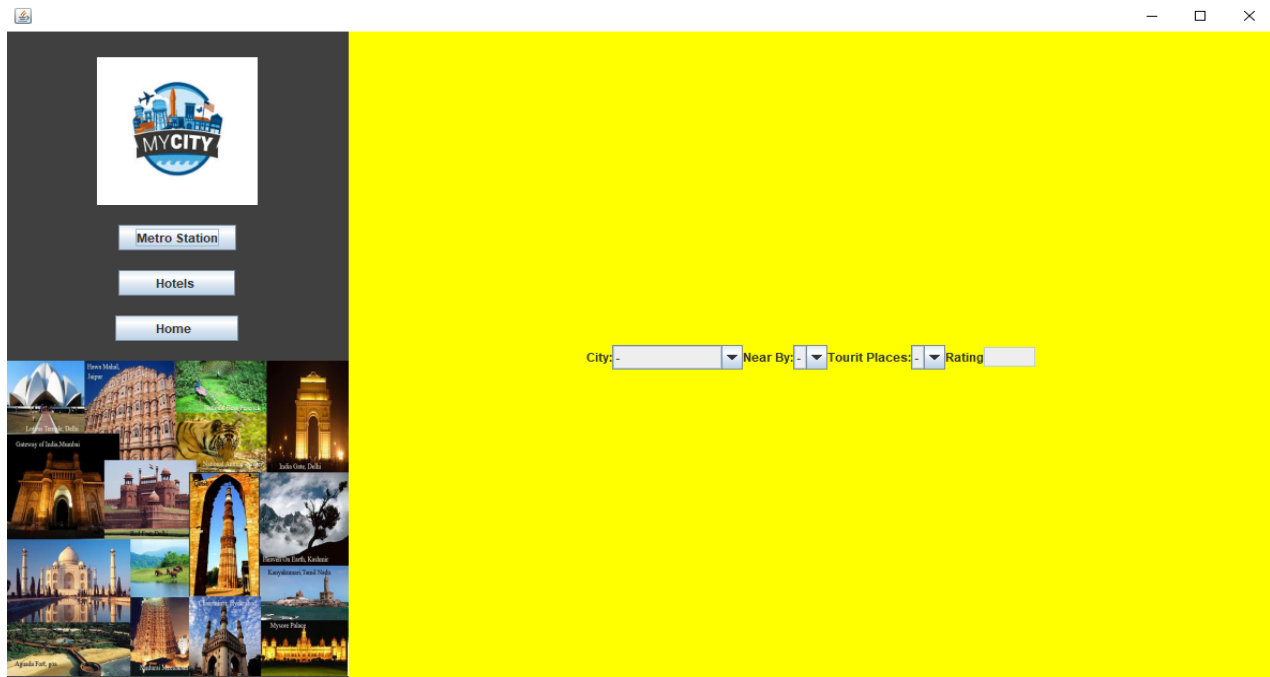
MyCityTable Services Tourist\_Places Has\_Tourist\_Places MetroStations Has\_MetroStations Hotels Has\_Hotels Landmarks

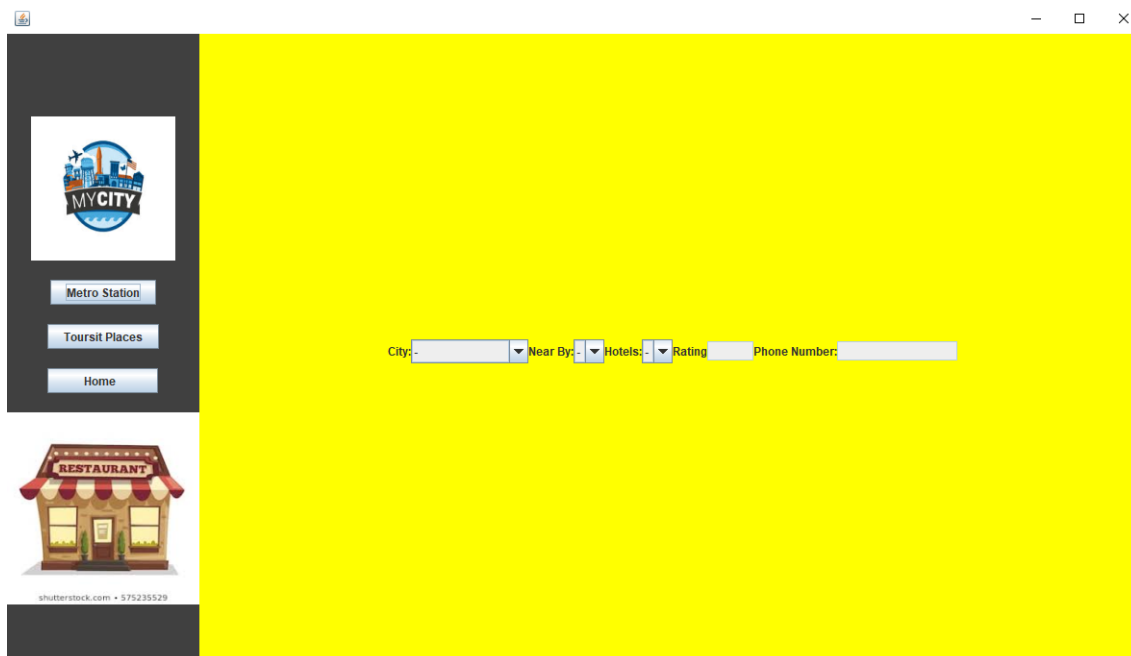
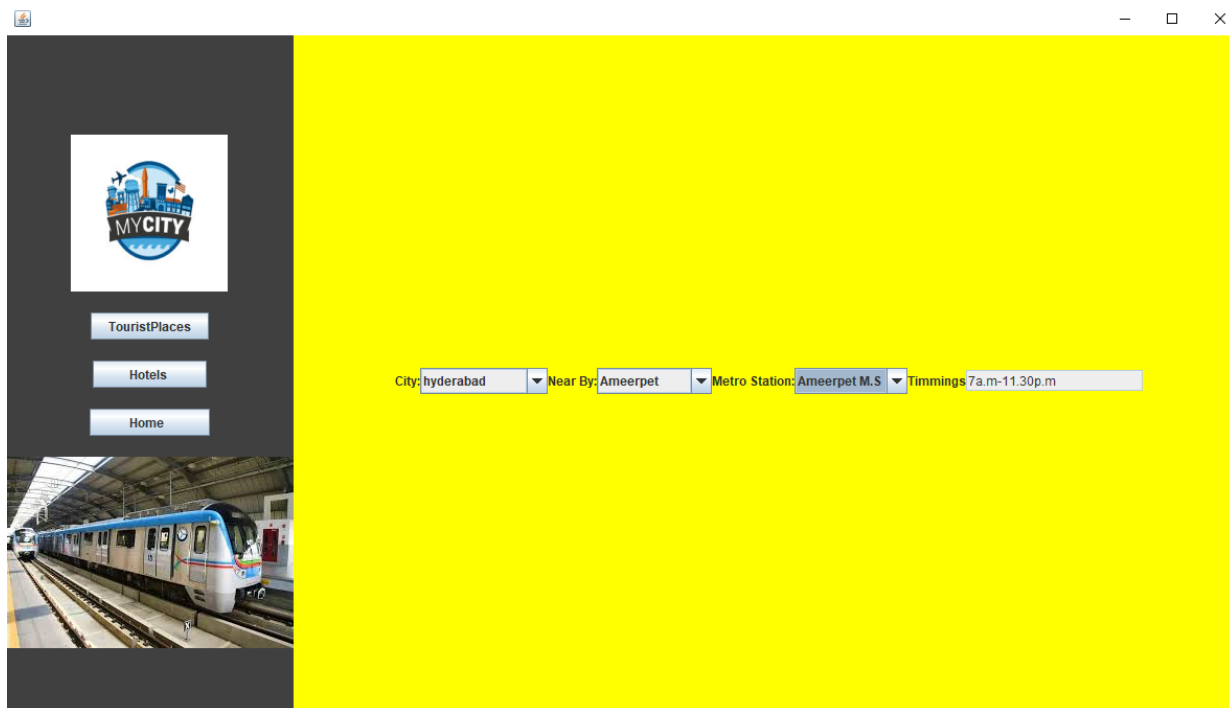
CityName:  Placename:

Home



## Application Interfaces for this project:





## **DISCUSSION AND FUTURE WORK!**

This project tells about different services like metro stations , hotels , restaurants , tourist places in the specific area . In future it can be further developed by adding more number of services and also we can add a feature of getting distance from specific landmark to metro stations , hotels , restaurants , tourist places and many more services.

## **REFERENCES:**

<https://www.decodejava.com/what-is-jdbc.htm>

<https://docs.oracle.com/javase/8/docs/api/>

<https://www.tutorialspoint.com/swing/index.htm>