

UCS1412 – Database Lab Assignment – 10

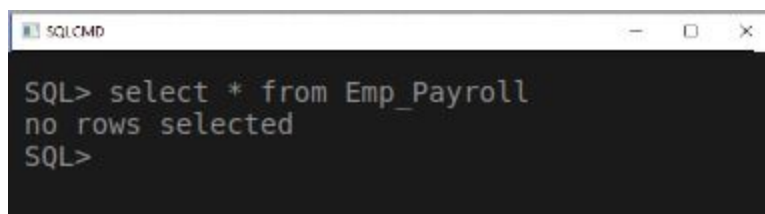
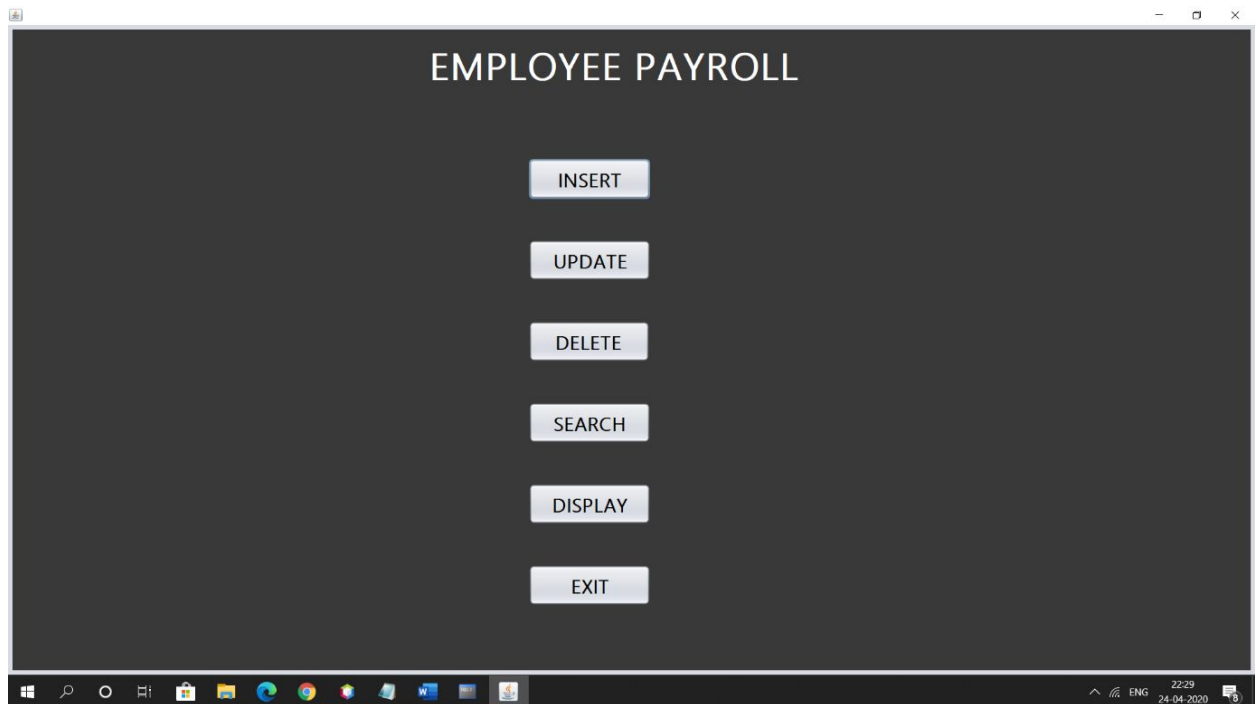
M Gayathri -185001050

Title: Database Application Programming using JDBC

This application maintains the employee payroll and allows insert, update, delete, search and display the details of the employee.

MODULES IN THE APPLICATION->

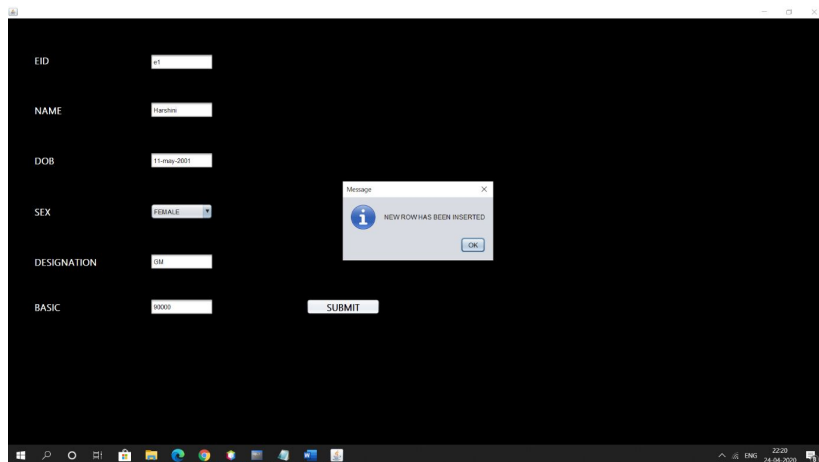
Basic Front End



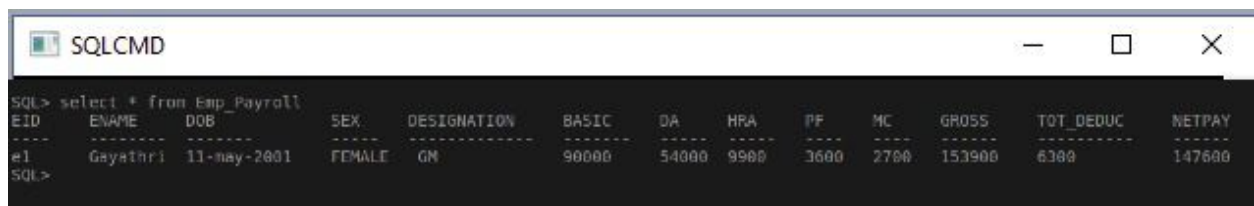
Insert Module

Allows insertion of an employee details into the database table Emp_Payroll and calculates the net pay with basic using a procedure .

It also shows an error message if we try to insert another employee with the same id since it is the primary key of the table. After insert it will be redirected to the first page.



The screenshot shows a web form titled "Insert Module" with the following fields: EID (text), NAME (text), DOB (date), SEX (dropdown menu), DESIGNATION (text), and BASIC (text). A "SUBMIT" button is located at the bottom right. A message box is displayed in the center, stating "NEW ROW HAS BEEN INSERTED" with an "OK" button.

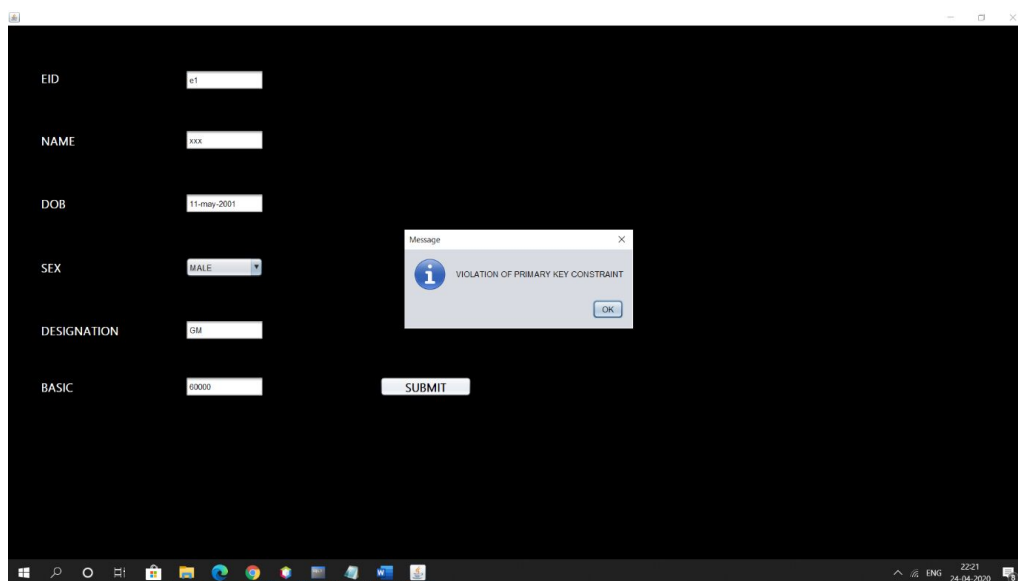


```
SQL> select * from Emp_Payroll
```

EID	ENAME	DOB	SEX	DESIGNATION	BASIC	DA	HPA	PF	MC	GROSS	TOT_DEDUC	NETPAY
e1	Gayathri	11-may-2001	FEMALE	GM	90000	54000	9900	3600	2700	153900	6300	147600

```
SQL>
```

Insertion of e1 that is already present shows a pop up window.



The screenshot shows the same web form as before, but with the EID field set to "e1" and the NAME field set to "xxx". A message box is displayed in the center, stating "VIOLATION OF PRIMARY KEY CONSTRAINT" with an "OK" button.

EID

e2

NAME

Jack

DOB

18-apr-1990

SEX

MALE

DESIGNATION

Senior Manager

BASIC

60000

SUBMIT

Message

NEW ROW HAS BEEN INSERTED

OK

Another employee e3 is inserted and the results are shown in the command prompt.

EID

e3

NAME

Yanini

DOB

22-apr-1975

SEX

FEMALE

DESIGNATION

HR

BASIC

60000

SUBMIT

Message

NEW ROW HAS BEEN INSERTED

OK

SQL> select * from Emp Payroll												
EID	FNAME	DOB	SEX	DESIGNATION	BASIC	DA	HRA	PF	MC	GROSS	TOT_DEDUC	NETPAY
e1	Gayathri	11-may-2001	FEMALE	GM	90000	54000	9900	3500	2700	153900	6300	147600
e2	Jack	18-apr-1990	MALE	Senior Manager	60000	36000	6600	2400	1800	102600	4200	98400
e3	Yanini	22-apr-1975	FEMALE	HR	60000	36000	5600	2400	1800	102600	4200	98400

Update Module

This option allows to modify the details of existing employees and the updation of basic pay will also calculate the new net pay.

After updating it will be redirected to the first page.

Updating an employee whose id is not present will be an error. Updation of e3.

UPDATE

ENTER ID IN THE TEXT BOX BELOW TO UPDATE: e3

ENAME: Yamini

DOB: 16-apr-1975

SEX: FEMALE

DESIGNATION: HR

BASIC: 66000

UPDATE

Message: UPDATED! OK

Basic pay is changed from 60000 to 66000, hence resulting in a change in the other columns of pay and net pay as well.

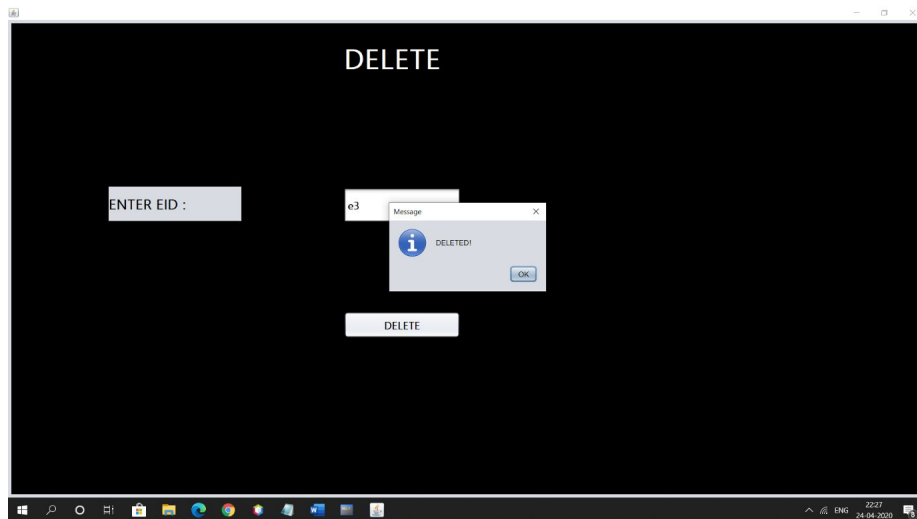
```
SQL> select * from Emp_Payroll
```

EID	ENAME	DOB	SEX	DESIGNATION	BASIC	DA	HRA	PF	MC	GROSS	TOT_DEDUC	NETPAY
c1	Gayathri	11 may 2001	FEMALE	GM	90000	54000	9900	3600	2700	153900	6300	147600
e2	Jack	16-apr-1990	MALE	Senior Manager	60000	36000	6600	2400	1800	102600	4200	98400
e3	Yamini	22 apr 1975	FEMALE	HR	66000	39600	7200	2640	1980	112800	4620	108240

```
SQL>
```

Delete Module

This module deletes the existing record and will show an error if the employee id is not present.

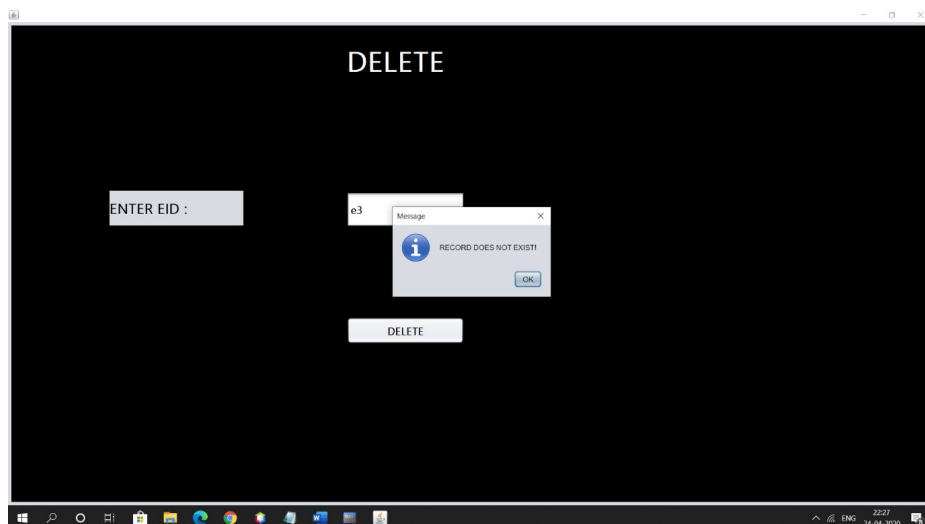


The screenshot shows a window titled "SQLCMD" displaying the results of two SQL queries. The first query shows the current state of the "Emp Payroll" table, which includes three records: e1 (Gayathri), e2 (Jack), and e3 (Yamini). The second query shows the state of the table after deleting record e3, which now only contains two records: e1 (Gayathri) and e2 (Jack).

EID	ENAME	DOB	SEX	DESIGNATION	BASIC	DA	HRA	PF	MC	GROSS	TOT DEDUC	NETPAY
e1	Gayathri	11-may-2001	FEMALE	GM	90000	54000	9900	3600	2700	153900	6300	147600
e2	Jack	16-apr-1990	MALE	Senior Manager	60000	36000	6600	2400	1800	102600	4200	98400
e3	Yamini	22-apr-1975	FEMALE	HR	66000	39600	7200	2640	1900	112800	4620	108240

EID	ENAME	DOB	SEX	DESIGNATION	BASIC	DA	HRA	PF	MC	GROSS	TOT DEDUC	NETPAY
e1	Gayathri	11-may-2001	FEMALE	GM	90000	54000	9900	3600	2700	153900	6300	147600
e2	Jack	16-apr-1990	MALE	Senior Manager	60000	36000	6600	2400	1800	102600	4200	98400

Trying to delete a record that is not present will show an error.



Search Module

This module displays the searched record and shows a popup window if the employee id is not present. Back button will redirect to the first page.

The screenshot shows a web application window titled "SEARCH FOR A RECORD". It features a search form with a label "ENTER EID :", a text input field containing "e2", and a "SEARCH" button. Below the form is a table displaying employee records. The table has columns: ENAME, DOB, SEX, DESIGNATION, BASIC, DA, HRA, PF, MC, GROSS, DEDUCTION, and NETPAY. The first row of data shows an employee named "Jack" with a DOB of "15-apr-1990", SEX of "MALE", DESIGNATION of "Senior Manager", and various salary components. A "BACK" button is located at the bottom of the form area.

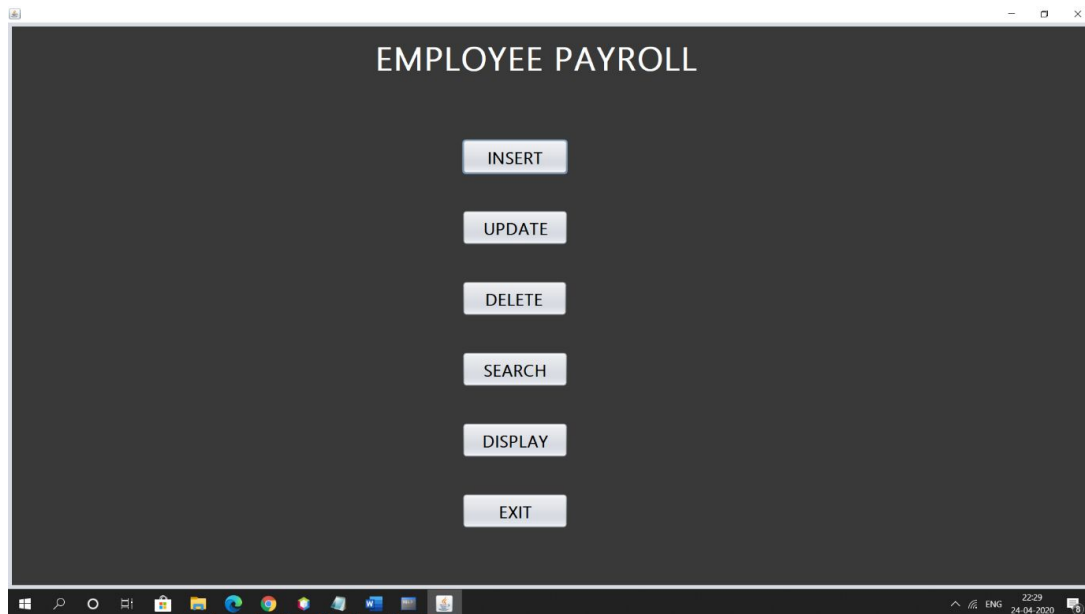
ENAME	DOB	SEX	DESIGNATION	BASIC	DA	HRA	PF	MC	GROSS	DEDUCTION	NETPAY
Jack	15-apr-1990	MALE	Senior Manager	60000.0	36000.0	6600.0	2400.0	1800.0	102600.0	4200.0	98400.0

The screenshot shows the same web application window, but the text input field now contains "e5". A message box is displayed in the center of the screen, titled "Message", with an information icon and the text "RECORD NOT FOUND!". The "OK" button is visible on the message box. The table below the form is partially visible, showing the same columns as in the previous screenshot.

ENAME	DOB	SEX	DESIGNATION	BASIC	DA	HRA	PF	MC	GROSS	DEDUCTION	NETPAY
-------	-----	-----	-------------	-------	----	-----	----	----	-------	-----------	--------

Exit Module

Exit option terminates the program.



SQL CODE→

drop table Emp_Payroll;

create table Emp_Payroll(eid varchar2(20) primary key,ename char(20),dob char(20),sex char(20),designation char(20),basic number(10),

da number(10),hra number(10),pf number(10),mc number(10),gross number(10),tot_deduc number(10),netpay number(10));

rem create procedure

create or replace procedure comp(id in Emp_Payroll.eid%type,pay in Emp_Payroll.basic%type) is

d number(10);

hr number(10);

p_f number(10);

m_c number(10);

gr number(10);

ded number(10);

np number(10);

BEGIN

d:=(60*pay)/100;

hr:=(11*pay)/100;

p_f:=(4*pay)/100;

m_c:=(3*pay)/100;

gr:=pay+d+hr;

ded:=p_f+m_c;

np:=gr-ded;

update Emp_Payroll set basic=pay,da=d,hra=hr,pf=p_f,mc=m_c,gross=gr,tot_deduc=ded, netpay=np where eid=id;

END;

JAVA APPLICATION 10(Extracted from netbeans 10)

src file contents-->Under java application 10->

Javaapplication10.java(CONTENTS)

```
package javaapplication10;

import java.sql.*;

/*
 * @author Gayathri
 */

public class JavaApplication10
{
    public static void main(String[] args)
    {
        Connection conn=null;

        try
        { String driverName="oracle.jdbc.driver.OracleDriver";

            Class.forName(driverName);

            conn=DriverManager.getConnection("jdbc:oracle:thin:@DESKTOP-N6BHVVF:1521:XE",
"gayu", "gayu123");

        }

        catch (ClassNotFoundException e)

        { System.out.println("could not find conn db driver"+e.getMessage());}

        catch(SQLException e)

        { System.out.println("could not connect to db"+e.getMessage()); }

    }
}
```

IMPLEMENTATION CODE->

MODULE CODES→

If insert button is clicked

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
insertop insert=new insertop();  
insert.setVisible(true);  
}
```

If update button is clicked

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
updateop update=new updateop();  
update.setVisible(true);  
}
```

If delete button is clicked

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
deleteop delete=new deleteop();  
delete.setVisible(true);  
}
```

If search button is clicked

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {  
searchop search=new searchop();  
search.setVisible(true);  
}
```

If display button is clicked

```
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {  
displayop display=new displayop();  
display.setVisible(true);  
}
```

If exit button is clicked

```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {  
this.dispose();  
}
```

NAVIGATING BUTTON CODES->

If SUBMIT button is clicked

```
String eid=jTextField1.getText();
String name=jTextField2.getText();
String dob=jTextField3.getText();
String sex=jComboBox1.getSelectedItem().toString();
String desig=jTextField4.getText();
float basic=Float.parseFloat(jTextField5.getText());
try{
String msg="select * from Emp_Payroll";
PreparedStatement s=conn.prepareStatement(msg);
ResultSet res = s.executeQuery();
while(res.next()) {
String data=res.getString(1);
if(data.equals(eid))
{
JOptionPane.showMessageDialog(null,"VIOLATION OF PRIMARY KEY CONSTRAINT");
this.dispose();
}
}
s.close();
res.close();
PreparedStatement st=conn.prepareStatement("insert into
Emp_Payroll(eid,ename,dob,sex,designation,basic) values(?,?,?,?,?,?)");
st.setString(1,eid);st.setString(2,name);st.setString(3,dob);st.setString(4,sex);st.setString(5,desig);
st.setFloat(6, basic);
int rs=st.executeUpdate();
System.out.println(rs+" status");
st.close();
PreparedStatement ss=conn.prepareStatement("call comp(?,?)");
ss.setString(1,eid);
ss.setFloat(2, basic);
int rss=ss.executeUpdate();
ss.close();
JOptionPane.showMessageDialog(null,"NEW ROW HAS BEEN INSERTED");
this.dispose();
} catch(SQLException e)
{ e.printStackTrace(); }
```

If UPDATE Button is clicked

```
try {
String id=jTextField1.getText();
String msg="update Emp_Payroll set ename='"+jTextField2.getText()+"
',dob='"+jTextField3.getText()+"",
sex='"+jComboBox1.getSelectedItem().toString()+"",
designation='"+jTextField4.getText()+"
where eid='"+jTextField1.getText()+"''";
PreparedStatement st=conn.prepareStatement(msg);
int res=st.executeUpdate();
System.out.println(res+" rest status");
st.close();
PreparedStatement s=conn.prepareStatement("call comp(?,?)");
s.setString(1,id);
s.setFloat(2,Float.parseFloat(jTextField5.getText()));
int rs=s.executeUpdate();
s.close();
if(res==1)
{OptionPane.showMessageDialog(null,"UPDATED!");
this.dispose();
} else
{
OptionPane.showMessageDialog(null,"RECORD NOT FOUND!");
this.dispose();
}
}
catch (SQLException ex) {
ex.printStackTrace();
}
```

If DELETE Button is clicked

```
try {
String id=jTextField1.getText();
String msg="delete from Emp_Payroll where eid='"+id+"''";
PreparedStatement st=conn.prepareStatement(msg);
int res=st.executeUpdate();
System.out.println(res+" rest status");
st.close();
if(res==1)
OptionPane.showMessageDialog(null,"DELETED!");
else
OptionPane.showMessageDialog(null,"RECORD DOES NOT EXIST!");
} catch (SQLException ex)
{ ex.printStackTrace(); }
this.dispose();
```

If SEARCH Button is clicked

```
try {
String id=jTextField1.getText();
String msg="select * from Emp_Payroll where eid='"+id+"'";
PreparedStatement st=conn.prepareStatement(msg);
ResultSet rs = st.executeQuery();
if (rs.next())
{Object data[]={rs.getString(2),rs.getString(3),rs.getString(4),rs.getString(5),rs.getFloat(6),rs.getFloat(7),
                rs.getFloat(8),rs.getFloat(9),rs.getFloat(10),rs.getFloat(11),rs.getFloat(12),
                rs.getFloat(13)
                };
DefaultTableModel tb=(DefaultTableModel)jTable1.getModel();
tb.addRow(data);
}
else
JOptionPane.showMessageDialog(null,"RECORD NOT FOUND!");
st.close();
} catch (SQLException ex) {
ex.printStackTrace();
}
```

If BACK Button is clicked

```
this.dispose();
```

If DISPLAY Button is clicked

```
try {
String msg="select * from Emp_Payroll";
PreparedStatement st=conn.prepareStatement(msg);
ResultSet rs = st.executeQuery();
while(rs.next())
{
Object data[] = {rs.getString(1),rs.getString(2),rs.getString(3),rs.getString(4),rs.getString(5),rs.getFloat(6)
                ,rs.getFloat(7),rs.getFloat(8),rs.getFloat(9),rs.getFloat(10),rs.getFloat(11),rs.getFloat(12),
                rs.getFloat(13)};
DefaultTableModel tb=(DefaultTableModel)jTable1.getModel();
tb.addRow(data);
}
st.close();
}
catch (SQLException ex) {
ex.printStackTrace();
}
```