

## EX-10

### EMPLOYEE PAYROLL APPLICATION

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

```
package ex.pkg10;
```

```
import java.beans.Statement;
```

```
import java.sql.Connection;
```

```
import java.sql.DriverManager;
```

```
import java.sql.PreparedStatement;
```

```
import java.sql.ResultSet;
```

```
import java.sql.SQLException;
```

```
import java.util.logging.Level;
```

```
import java.util.logging.Logger;
```

```
import javax.swing.JOptionPane;
```

```
import javax.swing.JPanel;
```

```
import static org.eclipse.persistence.config.BatchWriting.JDBC;
```

```
/**
```

```
*
```

```
* @author Hari
```

```
*/
```

```
public class operation extends javax.swing.JFrame {
```

```
    /**
```

```
     * Creates new form operation
```

```
    */
```

```
    Connection con;
```

```
    Statement st;
```

```
    PreparedStatement ps,ps1,ps2;
```

```
    ResultSet rs;
```

```
    public operation() {
```

```
        initComponents();
```

```
        try{
```

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

```
            JOptionPane.showMessageDialog(this,"Driver Loaded");
```

```
        try{
```

```
            con =
```

```
            DriverManager.getConnection("jdbc:oracle:thin:@//localhost:1522/oracle","hr","pooja");
```

```
            JOptionPane.showMessageDialog(this, "Connected to oracle database!!");
```

```
        }
```

```

        catch(SQLException ex){
            Logger.getLogger(operation.class.getName()).log(Level.SEVERE, null, ex);
            JOptionPane.showMessageDialog(this,ex.getMessage());
        }
    }
    catch(ClassNotFoundException ex){
        Logger.getLogger(operation.class.getName()).log(Level.SEVERE, null, ex);
        JOptionPane.showMessageDialog(this, ex.getMessage());
    }
}

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

```

## CODE STARTS HERE:

```

private void jinsertActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jinsertActionPerformed
    // TODO add your handling code here:
    switchPanels(insert);
} //GEN-LAST:event_jinsertActionPerformed

private void jupdateActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jupdateActionPerformed
    // TODO add your handling code here:
    switchPanels(update);
} //GEN-LAST:event_jupdateActionPerformed

private void jdeleteActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jdeleteActionPerformed
    // TODO add your handling code here:
    switchPanels(delete);
} //GEN-LAST:event_jdeleteActionPerformed

private void jMenuItem4ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem4ActionPerformed
    // TODO add your handling code here:
    System.exit(0);
}

```

```

} //GEN-LAST:event_jMenuItem4ActionPerformed

private void jMenuItem5ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jMenuItem5ActionPerformed
    // TODO add your handling code here:
    new About().setVisible(true);
} //GEN-LAST:event_jMenuItem5ActionPerformed

private void resetActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_resetActionPerformed
    // TODO add your handling code here:
    clearText();
} //GEN-LAST:event_resetActionPerformed

private void jinsert1ActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_jinsert1ActionPerformed
    // TODO add your handling code here:
    try{
        String query= "INSERT INTO EMP_PAYROLL(eid,ename,dob,sex,designation,bpay)
VALUES(?,?,?,?,?,?,?)";
        ps = con.prepareStatement(query);
        ps.setString(1, empid.getText());
        ps.setString(2, empname.getText());
        ps.setString(3, empdob.getText());
        ps.setString(4, empsex.getSelectedItem().toString());
        ps.setString(5, empdesig.getText());
        ps.setString(6, empbp.getText());
        String proc="call netpay_calc(?,?)";
        ps1=con.prepareStatement(proc);
        ps1.setString(1, empid.getText());
        ps1.setString(2, empbp.getText());
        if(ps.executeUpdate()>0)
        {
            ps1.executeQuery();
            JOptionPane.showMessageDialog(null, "Employee Details Inserted
Successfully\nNet Pay calculated");
            clearText();
        }
        else
        {
            JOptionPane.showMessageDialog(null, "Employee Details Insertion Failed!!!");
        }
    }
    catch(SQLException ex){
        Logger.getLogger(operation.class.getName()).log(Level.SEVERE,null,ex);
        JOptionPane.showMessageDialog(this, ex.getMessage());
    }
}

```

```

    }

    }//GEN-LAST:event_jinsert1ActionPerformed

    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jButton3ActionPerformed
        // TODO add your handling code here:
        deid.setText("");
    } //GEN-LAST:event_jButton3ActionPerformed

    private void jdelete3ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jdelete3ActionPerformed
        // TODO add your handling code here:
        try{
            String query= "DELETE FROM EMP_PAYROLL WHERE eid=?";
            ps = con.prepareStatement(query);
            ps.setString(1, deid.getText());
            if(ps.executeUpdate(>0)
            {
                JOptionPane.showMessageDialog(null, "Employee Details
Deleted\nSuccessfully");
                deid.setText("");
            }
            else
            {
                JOptionPane.showMessageDialog(null, "Employee Details Deletion\nFailed!!!");
            }
        }
        catch(SQLException ex){
            Logger.getLogger(operation.class.getName()).log(Level.SEVERE,null,ex);
            JOptionPane.showMessageDialog(this, ex.getMessage());
        }
    } //GEN-LAST:event_jdelete3ActionPerformed

    private void jupdate1ActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_jupdate1ActionPerformed
        // TODO add your handling code here:
        try{
            String query= "UPDATE EMP_PAYROLL SET
ename=?,dob=?,sex=?,designation=?,bpay=? WHERE eid=?";
            ps = con.prepareStatement(query);
            ps.setString(6, empid1.getText());
            ps.setString(1, empname1.getText());
            ps.setString(2, empdob1.getText());
            ps.setString(3, empsex1.getSelectedItem().toString());
            ps.setString(4, empdesig1.getText());

```

```

        ps.setString(5, empbp1.getText());
        String proc="call netpay_calc(?,?)";
        ps2=con.prepareStatement(proc);
        ps2.setString(1, empid1.getText());
        ps2.setString(2, empbp1.getText());
        if(ps.executeUpdate()>0)
        {
            ps2.executeQuery();
            JOptionPane.showMessageDialog(null, "Employee Details Updated
Successfully");
            clearTextupdate();
        }
        else
        {
            JOptionPane.showMessageDialog(null, "Employee Details Updation Failed!!!");
        }
    }
    catch(SQLException ex){
        Logger.getLogger(operation.class.getName()).log(Level.SEVERE,null,ex);
        JOptionPane.showMessageDialog(this, ex.getMessage());
    }
}
}
//GEN-LAST:event_jupdate1ActionPerformed

```

```

private void reset2ActionPerformed(java.awt.event.ActionEvent evt)
{
//GEN-FIRST:event_reset2ActionPerformed
    // TODO add your handling code here:
    clearSearch();
}
//GEN-LAST:event_reset2ActionPerformed

```

```

private void jsearchActionPerformed(java.awt.event.ActionEvent evt)
{
//GEN-FIRST:event_jsearchActionPerformed
    // TODO add your handling code here:
    try{
        String query= "SELECT * FROM EMP_PAYROLL WHERE eid=?";
        ps=con.prepareStatement(query);
        ps.setString(1, empid2.getText());
        rs=ps.executeQuery();
        if(rs.next())
        {
            empname2.setText(rs.getString(2));
            empdob2.setText(rs.getString(3));
            empsex2.setSelectedItem(rs.getString(4));
            empdesig2.setText(rs.getString(5));
            empbp2.setText(rs.getString(6));
            da2.setText(rs.getString(7));
            hra2.setText(rs.getString(8));
        }
    }
}
}

```

```

        pf2.setText(rs.getString(9));
        mc2.setText(rs.getString(10));
        gross2.setText(rs.getString(11));
        tot_ded2.setText(rs.getString(12));
        jnpay.setText(rs.getString(13));
        JOptionPane.showMessageDialog(null, "Employee Search Successfully");
    }
    else
    {
        JOptionPane.showMessageDialog(null, "Employee Search Failed!!!");
    }
}
catch(SQLException ex){
    Logger.getLogger(operation.class.getName()).log(Level.SEVERE,null,ex);
    JOptionPane.showMessageDialog(this, ex.getMessage());
}
}
}

GEN-LAST:event_jsearchActionPerformed

private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt)
{
GEN-FIRST:event_jMenuItem1ActionPerformed
    // TODO add your handling code here:
    switchPanels(search);
}
GEN-LAST:event_jMenuItem1ActionPerformed

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

## OTHER USED METHODS:

```

private void switchPanels(JPanel panel)
{
    jLayeredPane1.removeAll();
    jLayeredPane1.add(panel);
    jLayeredPane1.repaint();
    jLayeredPane1.revalidate();
}

```

```

private void clearText()
{
    empid.setText("");
    empname.setText("");
    empdob.setText("");
    empsex.setSelectedIndex(0);
    empdesig.setText("");
    empbp.setText("");
}

```

```

private void clearTextupdate()
{

```

```

        empid1.setText("");
        empname1.setText("");
        empdob1.setText("");
        empsex1.setSelectedIndex(0);
        empdesig1.setText("");
        empbbp1.setText("");
    }
    private void clearSearch()
    {
        empid2.setText("");
        empname2.setText("");
        empdob2.setText("");
        empsex2.setSelectedIndex(0);
        empdesig2.setText("");
        empbbp2.setText("");
        jnpay.setText("");
        da2.setText("");
        hra2.setText("");
        pf2.setText("");
        mc2.setText("");
        gross2.setText("");
        tot_ded2.setText("");
    }

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JTextField da2;
private javax.swing.JTextField deid;
private javax.swing.JPanel delete;
private javax.swing.JTextField empbbp;
private javax.swing.JTextField empbbp1;
private javax.swing.JTextField empbbp2;
private javax.swing.JTextField empdesig;
private javax.swing.JTextField empdesig1;
private javax.swing.JTextField empdesig2;
private javax.swing.JTextField empdob;
private javax.swing.JTextField empdob1;
private javax.swing.JTextField empdob2;
private javax.swing.JTextField empid;
private javax.swing.JTextField empid1;
private javax.swing.JTextField empid2;
private javax.swing.JTextField empname;
private javax.swing.JTextField empname1;
private javax.swing.JTextField empname2;
private javax.swing.JComboBox<String> empsex;
private javax.swing.JComboBox<String> empsex1;
private javax.swing.JComboBox<String> empsex2;

```



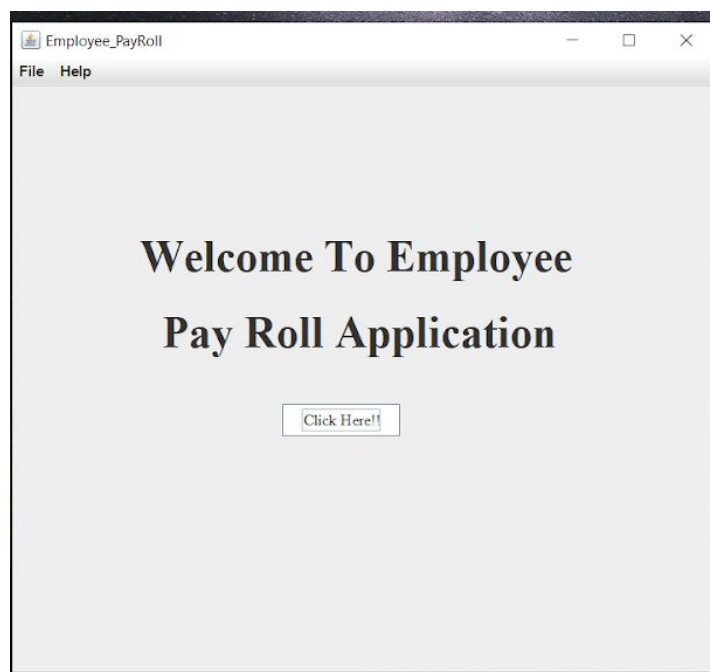
```
private javax.swing.JTextField gross2;
private javax.swing.JTextField hra2;
private javax.swing.JPanel insert;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton3;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel17;
private javax.swing.JLabel jLabel18;
private javax.swing.JLabel jLabel19;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel22;
private javax.swing.JLabel jLabel23;
private javax.swing.JLabel jLabel24;
private javax.swing.JLabel jLabel25;
private javax.swing.JLabel jLabel26;
private javax.swing.JLabel jLabel27;
private javax.swing.JLabel jLabel28;
private javax.swing.JLabel jLabel29;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel30;
private javax.swing.JLabel jLabel31;
private javax.swing.JLabel jLabel32;
private javax.swing.JLabel jLabel33;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JLayeredPane jLayeredPane1;
private javax.swing.JMenu jMenu1;
private javax.swing.JMenu jMenu4;
private javax.swing.JMenuBar jMenuBar1;
private javax.swing.JMenuItem jMenuItem1;
private javax.swing.JMenuItem jMenuItem4;
private javax.swing.JMenuItem jMenuItem5;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
```

```

private javax.swing.JPanel jPanel4;
private javax.swing.JPanel jPanel5;
private javax.swing.JPopupMenu.Separator jSeparator1;
private javax.swing.JPopupMenu.Separator jSeparator2;
private javax.swing.JMenuItem jdelete;
private javax.swing.JButton jdelete3;
private javax.swing.JMenuItem jinsert;
private javax.swing.JButton jinsert1;
private javax.swing.JTextField jnpay;
private javax.swing.JButton jsearch;
private javax.swing.JMenuItem jupdate;
private javax.swing.JButton jupdate1;
private javax.swing.JTextField mc2;
private javax.swing.JTextField pf2;
private javax.swing.JButton reset;
private javax.swing.JButton reset2;
private javax.swing.JPanel search;
private javax.swing.JTextField tot_ded2;
private javax.swing.JPanel update;
private javax.swing.JButton ureset;
// End of variables declaration//GEN-END:variables
}

```

## SAMPLE SCREENSHOTS OF APPLICATION



Message X



Driver Loaded

OK

Message X



Connected to oracle database!!

OK

Employee Pay Roll

File Help

Insert Ctrl-I

Update Ctrl-U

Delete Ctrl-D

Search Ctrl-S

Exit Alt-F4

## Insert

Employee ID

Employee Name

DOB

Sex

Select any ▼

Designation

Basic Pay

Insert

Reset

Employee Pay Roll

File Help

### Insert

Employee ID	<input type="text" value="E001"/>
Employee Name	<input type="text" value="Pooja"/>
DOB	<input type="text" value="03-may-2000"/>
Sex	<input type="text" value="F"/> ▼
Designation	<input type="text" value="Manager"/>
Basic Pay	<input type="text" value="50000"/>


Employee Pay Roll

File Help

### Insert

Employee ID	<input type="text" value="E001"/>
Employee Name	<input type="text" value="Pooja"/>
DOB	<input type="text" value="03-may-2000"/>
Sex	<input type="text" value="F"/> ▼
Designation	<input type="text" value="Manager"/>
Basic Pay	<input type="text" value="50000"/>

Message

 **Employee Details Inserted Successfully**  
Net Pay calculated

```
SQL> set linesize 700;
SQL> select * from emp_payroll;
```

EID	ENAME	DOB	SEX	DESIGNATIO	BPAY	DA	HRA	PF	MC	GROSS	TOT_DED	NET_PAY
E001	Pooja	03-MAY-00 12.00.00.0000000 AM	F	Manager	50000	30000	5500	2000	1500	85500	3500	82000

Employee Pay Roll

File Help

## Update

Employee ID	<input type="text" value="E001"/>
Employee Name	<input type="text" value="Pooja"/>
DOB	<input type="text" value="03-may-00"/>
Sex	<input type="text" value="F"/> ▼
Designation	<input type="text" value="CEO"/>
Basic Pay	<input type="text" value="75000"/>

Message



Employee Details Updated Successfully

OK

```
SQL> select * from emp_payroll;
```

EID	ENAME	DOB	SEX	DESIGNATIO	BPAY	DA	HRA	PF	MC	GROSS	TOT_DED	NET_PAY
E001	Pooja	03-MAY-00 12.00.00.0000000 AM	F	CEO	75000	45000	8250	3000	2250	128250	5250	123000

Employee Pay Roll

File Help

### Search

Employee ID

Employee Name

DOB

Sex

Designation

Basic Pay

DA  HRA  PF

MC  GROSS  TOT\_DED

Net Pay

Employee Pay Roll

File Help

### Search

Employee ID

Employee Name

DOB

Sex

Designation


Basic Pay

DA  HRA  PF

MC  GROSS  TOT\_DED

Net Pay

Message

 Employee Search Successfully

Employee Pay Roll

File Help

Insert Ctrl-I  
Update Ctrl-U  
Delete Ctrl-D  
Search Ctrl-S  
Exit Alt-F4

## Search

Employee ID

Employee Name

DOB

Sex

Designation

Basic Pay

DA  HRA  PF

MC  GROSS  TOT\_DED

Net Pay

Search Reset

Employee Pay Roll

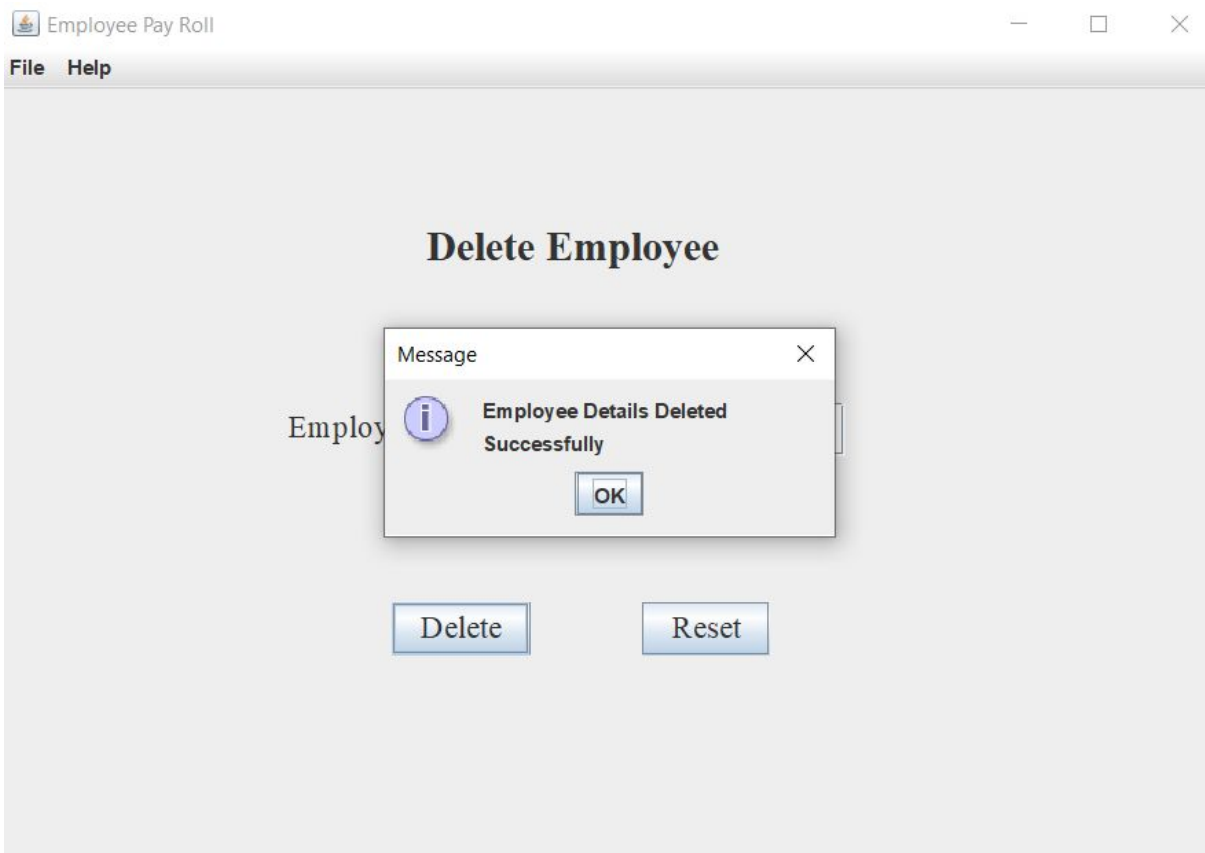
File Help

Insert Ctrl-I  
Update Ctrl-U  
Delete Ctrl-D  
Search Ctrl-S  
Exit Alt-F4

## Delete Employee

Employee ID

Delete Reset



```
SQL> select * from emp_payroll;  
  
no rows selected  
  
SQL>
```



## BACKEND CODE:

### *rem:creating a table*

```
create table EMP_PAYROLL
(
    EID VARCHAR2(5) constraint EID_PK primary key,
    ENAME VARCHAR2(10),
    DOB DATA,
    SEX VARCHAR2(17),
    DESIGNATION VARCHAR2(10),
    BPAY NUMBER,
    DA NUMBER,
    HRA NUMBER,
    PF NUMBER,
    MC NUMBER,
    GROSS NUMBER,
    TOT_DED NUMBER,
    NET_PAY NUMBER
);
```

### *rem:procedure to calculate net pay salary*

```
CREATE OR REPLACE procedure netpay_calc(empid IN varchar2,basic IN number) IS
pda number;
phra number;
ppf number;
pmc number;
pgross number;
td number;
np number;
begin
pda := 0.60*basic;
phra := 0.11*basic;
ppf :=0.04*basic;
pmc := 0.03*basic;
pgross:=basic+pda+phra;
td := ppf+pmc;
np := pgross-td;
update emp_payroll set net_pay =
np,da=pda,hra=phra,pf=ppf,mc=pmc,gross=pgross,tot_ded=td where eid = empid;
end;
/
```