**ADBMS MINIPROJECT ON ETL**

**Team**

**Rishi Joshi (60003140023)**

**Vighnesh Misal (60003140028)**

**Nahush Raichura (60003140034)**

**Project Description**

In computing, Extract, Transform, Load (ETL) refers to a process in database usage and especially in data warehousing that performs:

1. Data extraction – extracts data from homogeneous or heterogeneous data sources
2. Data transformation – transforms the data for storing it in the proper format or structure for the purposes of querying and analysis
3. Data loading – loads it into the final target

We have implemented the process of extraction, transformation and loading. The databases from where data is extracted are Microsoft Access 2016, Oracle Database and MySQL database. The data is then transformed to a normalized form and loaded in a new table in Oracle11g. The front end user interface was implemented using Java Swing Framework.

**Code:**

1. **Etlproject.java**

package etlproject;

import java.sql.\*;

import java.lang.\*;

public class Etlproject

{

//variables of oracle database

static int eid[]=new int[10];

static String ename[]=new String[10];

static float esalary[]=new float[10];

static int ephone[]=new int[10];

static String ebdate[]=new String[10];

static String ecity[]=new String[10];

static String edept\_name[]=new String[10];

static int edept\_id[]=new int[10];

static String egender[]=new String[10];

//variables of mysql database

static int meid[]=new int[10];

static String mename[]=new String[10];

static float mesalary[]=new float[10];

static int mephone[]=new int[10];

static String mebdate[]=new String[10];

static String mecity[]=new String[10];

static String medept\_name[]=new String[10];

static int medept\_id[]=new int[10];

static String megender[]=new String[10];

//variables of msaccess database

static int mseid[]=new int[10];

static String msename[]=new String[10];

static float msesalary[]=new float[10];

static int msephone[]=new int[10];

static String msebdate[]=new String[10];

static String msecity[]=new String[10];

static String msedept\_name[]=new String[10];

static int msedept\_id[]=new int[10];

static String msegender[]=new String[10];

// variable constants of mysql database

private static final String USERNAME="root";

private static final String PASSWORD="";

private static final String CONN\_STRING="jdbc:mysql://localhost:3306/mysqldb";

public static void main(String args[])throws Exception

{

//to extract records from oracle database

try{

//step1 load the driver class

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

//step2 create the connection object

Connection con=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=con.createStatement();

//step4 execute query

ResultSet rs=stmt.executeQuery("select \* from employee");

int i=0;

//storing the database values in arrays

while(rs.next()) {

eid[i]=rs.getInt(1);

ename[i]=rs.getString(2);

esalary[i]=rs.getInt(3);

ephone[i]=rs.getInt(4);

ebdate[i]=rs.getDate(5).toString();

ecity[i]=rs.getString(6);

edept\_name[i]=rs.getString(7);

edept\_id[i]=rs.getInt(8);

egender[i]=rs.getString(9);

i++;

}

//step5 close the connection object

con.close();

}

catch(Exception e)

{

System.out.println(e);

}

finally

{

//to print the data

/\*for(int j=0;j<10;j++)

{

System.out.println(eid[j]+" "+ename[j]+" "+esalary[j]+" "+ephone[j]+" "+ebdate[j]+" "+ecity[j]+" "+edept\_name[j]+" "+edept\_id[j]+" "+egender[j]+" ");

}\*/

}

//to extract data from ms-access database

//System.out.println("\*\*\*\*\*\*MS-ACCESS DATABASE\*\*\*\*\*\*\*\*\*");

try

{

// loading thejdbc odbc driver

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

// creating connection toth data base

Connection con = DriverManager.getConnection("jdbc:odbc:emp","","");

Statement st = con.createStatement();

// create an execute sql command on database

ResultSet msrs = st.executeQuery("Select \* from msaccesstable");

ResultSetMetaData rsmd = msrs.getMetaData();

// this getColumnCount reurn the number of column in the selected table

int numberOfColumns = rsmd.getColumnCount();

// while loop and with while loop code use for print the data

int msi=0;

//to store data from msaccess in array variables

while (msrs.next())

{

mseid[msi]=msrs.getInt(1);

msename[msi]=msrs.getString(2);

msesalary[msi]=msrs.getInt(3);

msephone[msi]=msrs.getInt(4);

msebdate[msi]=msrs.getDate(5).toString();

msecity[msi]=msrs.getString(6);

msedept\_name[msi]=msrs.getString(7);

msedept\_id[msi]=msrs.getInt(8);

msegender[msi]=msrs.getString(9).substring(0,1);

msi++;

}

con.close();

}

catch (Exception ex)

{

System.err.print("Exception: ");

System.err.println(ex.getMessage());

}

finally

{

//to print data

/\*

for(int j=0;j<10;j++)

{

System.out.println(mseid[j]+" "+msename[j]+" "+msesalary[j]+" "+msephone[j]+" "+msebdate[j]+" "+msecity[j]+" "+msedept\_name[j]+" "+msedept\_id[j]+" "+msegender[j]+" ");

}\*/

}

//to extract data from mysql(phpmy admin) database

//System.out.println("\*\*\*\*\*\*MYSQL DATABASE\*\*\*\*\*\*\*\*");

try

{

Connection conmysql = DriverManager.getConnection(CONN\_STRING,USERNAME,PASSWORD);

Statement stmt=(Statement)conmysql.createStatement();

String mysqlquery="select \* from emptable";

ResultSet rsmysql = stmt.executeQuery(mysqlquery);

int mi=0;

//to store data from mysql database to array variables

while(rsmysql.next())

{

meid[mi]=rsmysql.getInt("eid");

mename[mi]=rsmysql.getString("ename");

mesalary[mi]=Math.round(rsmysql.getDouble("esalary"));

mephone[mi]=rsmysql.getInt("ephone");

mebdate[mi]=rsmysql.getDate("ebdate").toString();

mecity[mi]=rsmysql.getString("ecity");

medept\_name[mi]=rsmysql.getString("edept\_name");

medept\_id[mi]=rsmysql.getInt("edept\_id");

megender[mi]=rsmysql.getString("egender").substring(0,1).toUpperCase();

mi++;

}

conmysql.close();

}

catch(Exception ef)

{

System.out.println(ef);

}

finally

{

//to print data

/\*

for(int j=0;j<10;j++)

{

System.out.println(meid[j]+" "+mename[j]+" "+mesalary[j]+" "+mephone[j]+" "+mebdate[j]+" "+mecity[j]+" "+medept\_name[j]+" "+medept\_id[j]+" "+megender[j]+" ");

}

\*/

}

//to set and call the form class from the same package

uidesignclass form\_display\_object =new uidesignclass();

form\_display\_object.setVisible(true);

}//MAIN

public void oracle()//copies only oracle database to finaltable

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="select \* from finaltable";

String sql2 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt2 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt2.setInt(1, eid[idx]);

pstmt2.setString(2, ename[idx]);

pstmt2.setFloat(3, esalary[idx]);

pstmt2.setInt(4, ephone[idx]);

pstmt2.setString(5, ebdate[idx]);

pstmt2.setString(6, ecity[idx]);

pstmt2.setString(7, edept\_name[idx]);

pstmt2.setInt(8, edept\_id[idx]);

pstmt2.setString(9,egender[idx]);

pstmt2.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt2.close();

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}//oracle method

public void msaccess()//copies only msaccess database to finaltable

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="select \* from finaltable";

String sql2 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt2 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt2.setInt(1, mseid[idx]);

pstmt2.setString(2, msename[idx]);

pstmt2.setFloat(3, msesalary[idx]);

pstmt2.setInt(4, msephone[idx]);

pstmt2.setString(5, msebdate[idx]);

pstmt2.setString(6, msecity[idx]);

pstmt2.setString(7, msedept\_name[idx]);

pstmt2.setInt(8, msedept\_id[idx]);

pstmt2.setString(9,msegender[idx]);

pstmt2.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt2.close();

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}//msaccess method

public void mysql()//copies only mysql database to finaltable

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="select \* from finaltable";

String sql2 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt2 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt2.setInt(1, meid[idx]);

pstmt2.setString(2, mename[idx]);

pstmt2.setFloat(3, mesalary[idx]);

pstmt2.setInt(4, mephone[idx]);

pstmt2.setString(5, mebdate[idx]);

pstmt2.setString(6, mecity[idx]);

pstmt2.setString(7, medept\_name[idx]);

pstmt2.setInt(8, medept\_id[idx]);

pstmt2.setString(9,megender[idx]);

pstmt2.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt2.close();

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}//mysql method

public void oracle\_msaccess()//copies first msaccess and then oracle databsse into finaltable

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="select \* from finaltable";

String sql2 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt2 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt2.setInt(1, mseid[idx]);

pstmt2.setString(2, msename[idx]);

pstmt2.setFloat(3, msesalary[idx]);

pstmt2.setInt(4, msephone[idx]);

pstmt2.setString(5, msebdate[idx]);

pstmt2.setString(6, msecity[idx]);

pstmt2.setString(7, msedept\_name[idx]);

pstmt2.setInt(8, msedept\_id[idx]);

pstmt2.setString(9,msegender[idx]);

pstmt2.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt2.close();

String sql3 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt3 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt3.setInt(1, eid[idx]);

pstmt3.setString(2, ename[idx]);

pstmt3.setFloat(3, esalary[idx]);

pstmt3.setInt(4, ephone[idx]);

pstmt3.setString(5, ebdate[idx]);

pstmt3.setString(6, ecity[idx]);

pstmt3.setString(7, edept\_name[idx]);

pstmt3.setInt(8, edept\_id[idx]);

pstmt3.setString(9, egender[idx]);

pstmt3.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt3.close();

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}//oracle\_msaccess method

public void oracle\_mysql()// copies first oracle database and then mysql databse into finaltable

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="select \* from finaltable";

String sql2 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt2 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt2.setInt(1, eid[idx]);

pstmt2.setString(2, ename[idx]);

pstmt2.setFloat(3, esalary[idx]);

pstmt2.setInt(4, ephone[idx]);

pstmt2.setString(5, ebdate[idx]);

pstmt2.setString(6, ecity[idx]);

pstmt2.setString(7, edept\_name[idx]);

pstmt2.setInt(8, edept\_id[idx]);

pstmt2.setString(9,egender[idx]);

pstmt2.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt2.close();

String sql3 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt3 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt3.setInt(1, meid[idx]);

pstmt3.setString(2, mename[idx]);

pstmt3.setFloat(3, mesalary[idx]);

pstmt3.setInt(4, mephone[idx]);

pstmt3.setString(5, mebdate[idx]);

pstmt3.setString(6, mecity[idx]);

pstmt3.setString(7, medept\_name[idx]);

pstmt3.setInt(8, medept\_id[idx]);

pstmt3.setString(9, megender[idx]);

pstmt3.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt3.close();

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}//oracle\_mysql method

public void msaccess\_mysql()//copies first msaccess and then mysql database into mysql

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="select \* from finaltable";

String sql2 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt2 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt2.setInt(1, mseid[idx]);

pstmt2.setString(2, msename[idx]);

pstmt2.setFloat(3, msesalary[idx]);

pstmt2.setInt(4, msephone[idx]);

pstmt2.setString(5, msebdate[idx]);

pstmt2.setString(6, msecity[idx]);

pstmt2.setString(7, msedept\_name[idx]);

pstmt2.setInt(8, msedept\_id[idx]);

pstmt2.setString(9,msegender[idx]);

pstmt2.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt2.close();

String sql3 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt3 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt3.setInt(1, meid[idx]);

pstmt3.setString(2, mename[idx]);

pstmt3.setFloat(3, mesalary[idx]);

pstmt3.setInt(4, mephone[idx]);

pstmt3.setString(5, mebdate[idx]);

pstmt3.setString(6, mecity[idx]);

pstmt3.setString(7, medept\_name[idx]);

pstmt3.setInt(8, medept\_id[idx]);

pstmt3.setString(9, megender[idx]);

pstmt3.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt3.close();

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}//msaccess\_mysql method

public void oracle\_msaccess\_mysql()//copies first msaccess, second oracle, third mysql database to the finaltable

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="select \* from finaltable";

String sql2 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt2 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt2.setInt(1, mseid[idx]);

pstmt2.setString(2, msename[idx]);

pstmt2.setFloat(3, msesalary[idx]);

pstmt2.setInt(4, msephone[idx]);

pstmt2.setString(5, msebdate[idx]);

pstmt2.setString(6, msecity[idx]);

pstmt2.setString(7, msedept\_name[idx]);

pstmt2.setInt(8, msedept\_id[idx]);

pstmt2.setString(9,msegender[idx]);

pstmt2.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt2.close();

String sql3 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt3 = conn.prepareStatement(sql2);

for (int idx=0;idx<10;idx++)

{

pstmt3.setInt(1, eid[idx]);

pstmt3.setString(2, ename[idx]);

pstmt3.setFloat(3, esalary[idx]);

pstmt3.setInt(4, ephone[idx]);

pstmt3.setString(5, ebdate[idx]);

pstmt3.setString(6, ecity[idx]);

pstmt3.setString(7, edept\_name[idx]);

pstmt3.setInt(8, edept\_id[idx]);

pstmt3.setString(9, egender[idx]);

pstmt3.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt3.close();

String sql4 = "INSERT INTO finaltable VALUES (?,?,?,?,?,?,?,?,?)";

//PreparedStatement pstmt1 = conn.prepareStatement(sql1);

PreparedStatement pstmt4 = conn.prepareStatement(sql4);

for (int idx=0;idx<10;idx++)

{

pstmt4.setInt(1, meid[idx]);

pstmt4.setString(2, mename[idx]);

pstmt4.setFloat(3, mesalary[idx]);

pstmt4.setInt(4, mephone[idx]);

pstmt4.setString(5, mebdate[idx]);

pstmt4.setString(6, mecity[idx]);

pstmt4.setString(7, medept\_name[idx]);

pstmt4.setInt(8, medept\_id[idx]);

pstmt4.setString(9, megender[idx]);

pstmt4.executeUpdate();

} // End of for loop

//pstmt1.close();

pstmt4.close();

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}//oracle\_msaccess\_mysql method

public void truncatetable()

{

try

{

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

//step2 create the connection object

Connection conn=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","SYSTEM","dba");

//step3 create the statement object

Statement stmt=conn.createStatement();

String sql1="truncate table finaltable";

stmt.executeQuery(sql1);

conn.close();

}

catch(Exception exc)

{

System.err.println(exc);

}

}

}//CLASS

**2.uidesignclass.java**

package etlproject;

import javax.swing.JOptionPane;

public class uidesignclass extends javax.swing.JFrame {

boolean check1=false,check2=false,check3=false;

boolean flag1,flag2,flag3;

public uidesignclass() {

initComponents();

}

@SuppressWarnings("unchecked")

private void initComponents() {

jLabel1 = new javax.swing.JLabel();

jCheckBox1 = new javax.swing.JCheckBox();

jCheckBox2 = new javax.swing.JCheckBox();

jCheckBox3 = new javax.swing.JCheckBox();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jLabel2 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jButton4 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jLabel1.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel1.setText("select the databases from where you want to extract data");

jLabel1.setToolTipText("");

jCheckBox1.setText("oracle ");

jCheckBox1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jCheckBox1ActionPerformed(evt);

}

});

jCheckBox2.setText("msaccess");

jCheckBox2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jCheckBox2ActionPerformed(evt);

}

});

jCheckBox3.setText("mysql");

jCheckBox3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jCheckBox3ActionPerformed(evt);

}

});

jButton1.setText("EXTRACT DATA");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("TRANSFORM DATA");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

jButton3.setText("LOAD DATA ");

jButton3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton3ActionPerformed(evt);

}

});

jLabel3.setFont(new java.awt.Font("Tahoma", 1, 11)); // NOI18N

jLabel3.setText("Click to truncate finaltable");

jButton4.setText("TRUNCATE");

jButton4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton4ActionPerformed(evt);

}

});

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(36, 36, 36)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jCheckBox3)

.addComponent(jCheckBox2)

.addComponent(jCheckBox1)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton3, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

.addGroup(layout.createSequentialGroup()

.addGap(291, 291, 291)

.addComponent(jLabel2))

.addGroup(layout.createSequentialGroup()

.addGap(38, 38, 38)

.addComponent(jLabel3)

.addGap(18, 18, 18)

.addComponent(jButton4)))

.addContainerGap(34, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jCheckBox1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jCheckBox2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jCheckBox3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jButton1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton3)

.addGap(14, 14, 14)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3)

.addComponent(jButton4))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel2)

.addContainerGap(73, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

if(check1==true)

{

//oracle connectivity

}

if(check2==true)

{

//msaccess connectivity

}

if(check3==true)

{

//mysql connectivity

}

if(check1==true && check2==false && check3==false)

{

flag1=true;

JOptionPane.showMessageDialog(this, "Data extracted from Oracle");

//call oracleonly();

}

if(check1==false && check2==true && check3==false)

{

flag1=true;

JOptionPane.showMessageDialog(this, "Data extracted from MSAccess");

//call msaccessonly();

}

if(check1==false && check2==false && check3==true)

{

flag1=true;

JOptionPane.showMessageDialog(this, "Data extracted from Mysql");

//call myysql();

}

if(check1==true && check2==true && check3==false)

{

flag1=true;

JOptionPane.showMessageDialog(this, "Data extracted from Oracle and MSAccess");

//call oracle\_msaccess();

}

if(check1==true && check2==false && check3==true)

{

flag1=true;

JOptionPane.showMessageDialog(this, "Data extracted from Oracle and Mysql");

//call Oracle\_mysql();

}

if(check1==false && check2==true && check3==true) {

flag1=true;

JOptionPane.showMessageDialog(this, "Data extracted from MSAccess and mysql");

//call msaccess\_mysql();

}

if(check1==true && check2==true && check3==true) {

flag1=true;

JOptionPane.showMessageDialog(this, "Data extracted from oracle, MSAccess and Excel");

//call oracle\_msaccess\_mysql();

}

if(check1==false && check2==false && check3==false) {

JOptionPane.showMessageDialog(this, "No data souce found to extract data");

//call nothing;

}

private void jCheckBox1ActionPerformed(java.awt.event.ActionEvent evt) {

if(jCheckBox1.isSelected()==true) {

check1=true;

}

}

private void jCheckBox2ActionPerformed(java.awt.event.ActionEvent evt) {

if(jCheckBox2.isSelected()==true) {

check2=true;

}

}

private void jCheckBox3ActionPerformed(java.awt.event.ActionEvent evt) {

if(jCheckBox3.isSelected()==true) {

check3=true;

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

if(flag1==true) {

if(check1==true && check2==false && check3==false) {

flag2=true;

JOptionPane.showMessageDialog(this, "Data from Oracle transformed in Standard form");

}

if(check1==false && check2==true && check3==false) {

flag2=true;

JOptionPane.showMessageDialog(this, "Data from MSAccess transformed in Standard form");

}

if(check1==false && check2==false && check3==true) {

flag2=true;

JOptionPane.showMessageDialog(this, "Data from Mysql transformed in Standard form");

}

if(check1==true && check2==true && check3==false) {

flag2=true;

JOptionPane.showMessageDialog(this, "Data from Oracle and MSAccess transformed in Standard form");

}

if(check1==true && check2==false && check3==true) {

flag2=true;

JOptionPane.showMessageDialog(this, "Data from Oracle and Mysql transformed in Standard form");

}

if(check1==false && check2==true && check3==true) {

flag2=true;

JOptionPane.showMessageDialog(this, "Data from MSAccess and mysql transformed in Standard form"); }

if(check1==true && check2==true && check3==true) {

flag2=true;

JOptionPane.showMessageDialog(this, "Data from oracle, MSAccess and Excel transformed in Standard form");

}

if(check1==false && check2==false && check3==false) {

JOptionPane.showMessageDialog(this, "No data souce found to transform data");

}

}

if(flag1==false) {

JOptionPane.showMessageDialog(this, "PLEASE EXTRACT THE DATA SOURCE FIRST");

}

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

Etlproject myobj=new Etlproject();

if(flag1==true && flag2==true){

if(check1==true && check2==false && check3==false){

flag3=true;

JOptionPane.showMessageDialog(this, "Data from Oracle copied to finaltable in the standard form");

myobj.oracle();

}

if(check1==false && check2==true && check3==false) {

flag3=true;

JOptionPane.showMessageDialog(this, "Data from MSAccess copied to finaltable in the standard form ");

myobj.msaccess();

}

if(check1==false && check2==false && check3==true) {

flag3=true;

JOptionPane.showMessageDialog(this, "Data from Mysql copied to finaltable in the standard form");

myobj.mysql();

}

if(check1==true && check2==true && check3==false){

flag3=true;

JOptionPane.showMessageDialog(this, "Data from Oracle and MSAccess copied to finaltable in the standard form");

myobj.oracle\_msaccess();

}

if(check1==true && check2==false && check3==true) {

flag3=true;

JOptionPane.showMessageDialog(this, "Data from Oracle and Mysql copied to finaltable in the standard form");

myobj.oracle\_mysql();

}

if(check1==false && check2==true && check3==true){

flag3=true;

JOptionPane.showMessageDialog(this, "Data from MSAccess and mysql copied to finaltable in the standard form");

myobj.msaccess\_mysql();

}

if(check1==true && check2==true && check3==true) {

flag3=true;

JOptionPane.showMessageDialog(this, "Data from oracle, MSAccess and Excel copied to finaltable in the standard form");

myobj.oracle\_msaccess\_mysql();

}

if(check1==false && check2==false && check3==false){

JOptionPane.showMessageDialog(this, "NO DATA SOURCE FOUND TO BE LOADED");

}

}

if(flag1==true && flag2==false) {

JOptionPane.showMessageDialog(this, "PLEASE TRANSFORM THE DATA BEFORE LOADING");

}

if(flag1==false && flag2==false){

JOptionPane.showMessageDialog(this, "PLEASE EXTRACT THE DATA BEFORE LOADING");

}

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

Etlproject myobject=new Etlproject();

myobject.truncatetable();

JOptionPane.showMessageDialog(this, "Finaltable truncated");

}

public static void main(String args[]){

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new uidesignclass().setVisible(true);

}

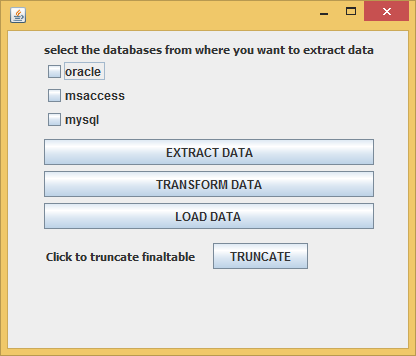
});

}

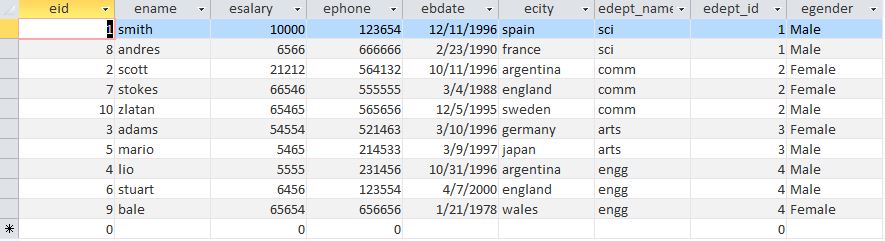
}

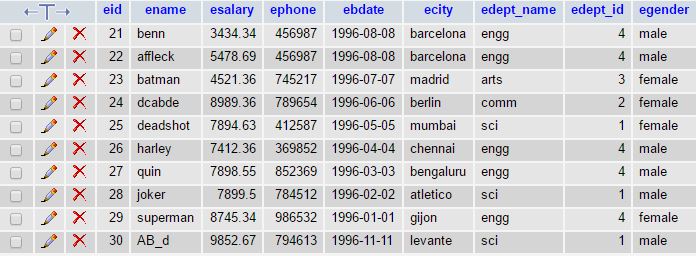
**Screenshots:**

**User Interface(UI)**

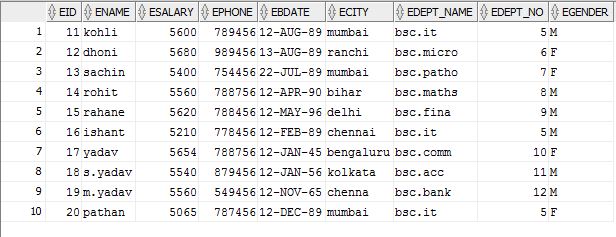


**Microsoft Access 2016 database data**

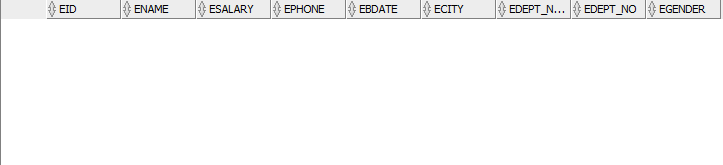


**MySQL database**

**Oracle Database data**



**Oracle Database before loading**



**Oracle Database after loading MySQL and MS Access data**

