|  |
| --- |
| **Lab12: Database Connectivity** |

# Name: Muhammad Zain Momin Roll:02-134222-014 Class:BSCS-2B

Designing and implementing Java programs that deal with:

**JDBC <-> ODBC**

JDBC

Java Database Connectivity (JDBC) provides a standard library for accessing databases. The JDBC API contains number of interfaces and classes that are extensively helpful while communicating with a database.

The java.sql package

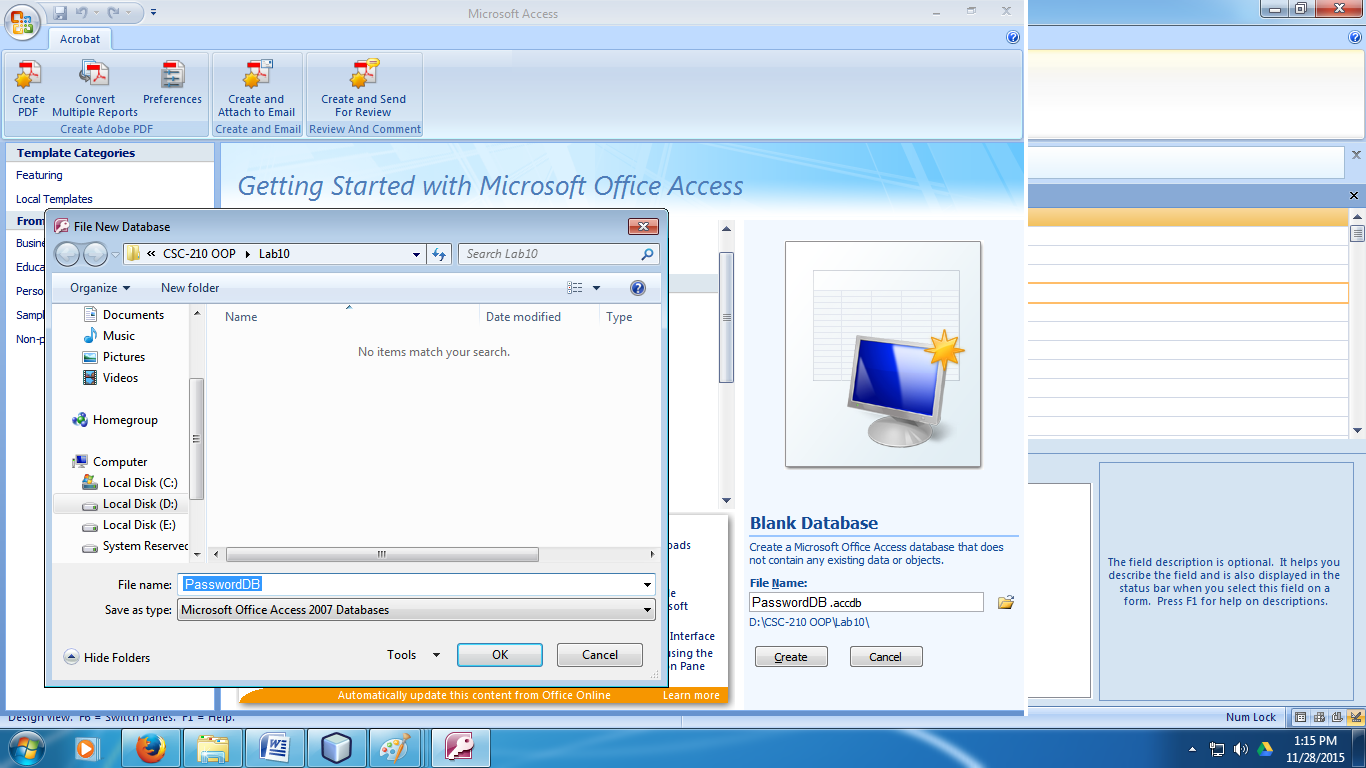
The java.sql package contains basic & most of the interfaces and classes. You automatically get this package when you download the J2SE™. You have to import this package whenever you want to interact with a relational database.

Conneting with Microsoft Access

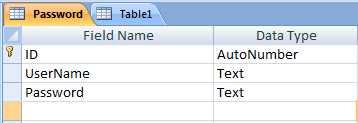
In this handout, we will learn how to connect & communicate with Microsoft Access Database.

Create Database

In start create a database “PasswordDB” using Microsoft Access. Create one table named “Password”. The schema of the table is shown in the picture.

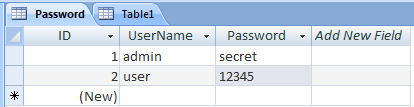


Save the database in some folder. (Your database will be saved as an .accdb file)



Save the above table with a name "Password"

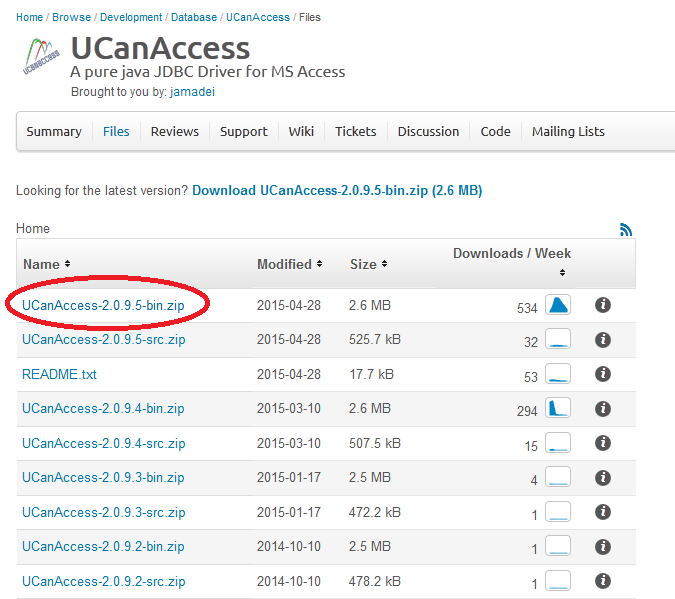
Add the following records into Person table as shown below.



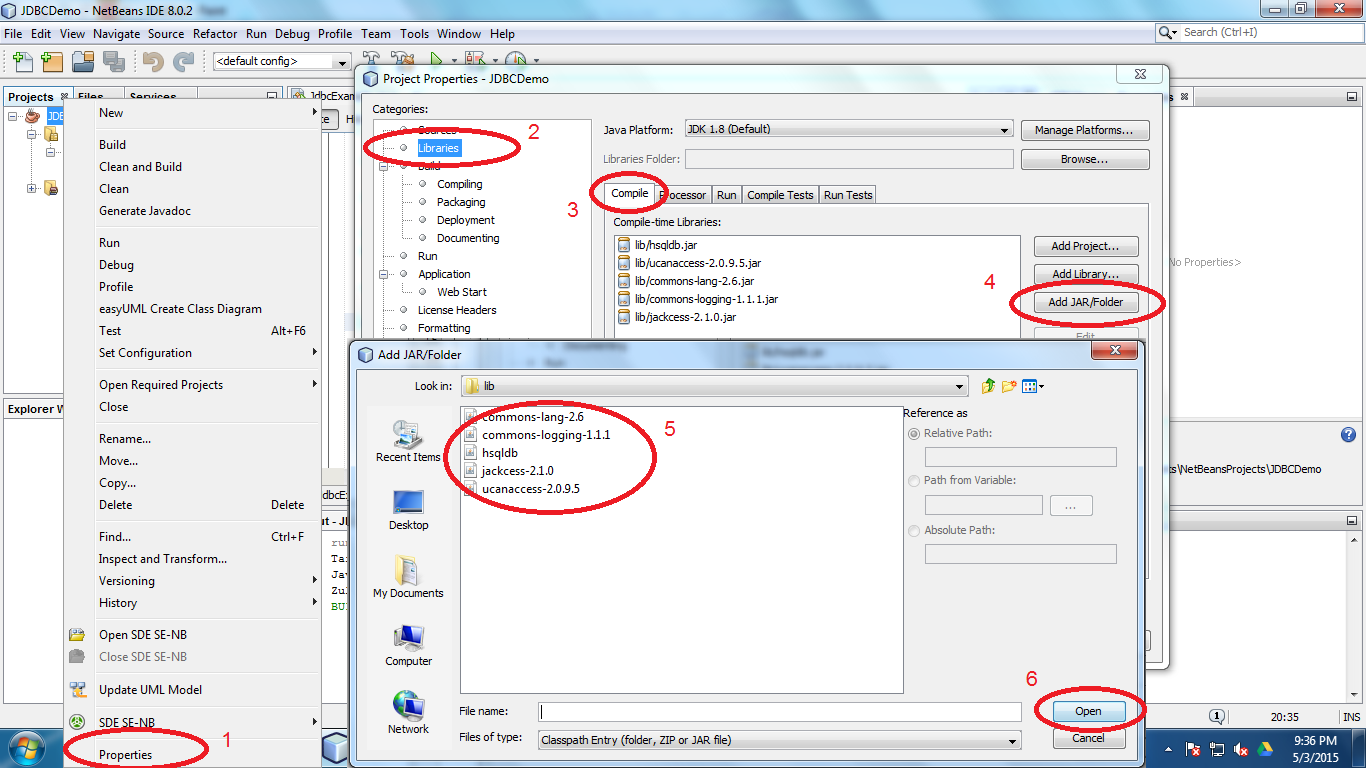
Using ucanaccess Library to Setup Database Connectivity

* Download the ucanaccess from the following link:

http://sourceforge.net/projects/ucanaccess/files/

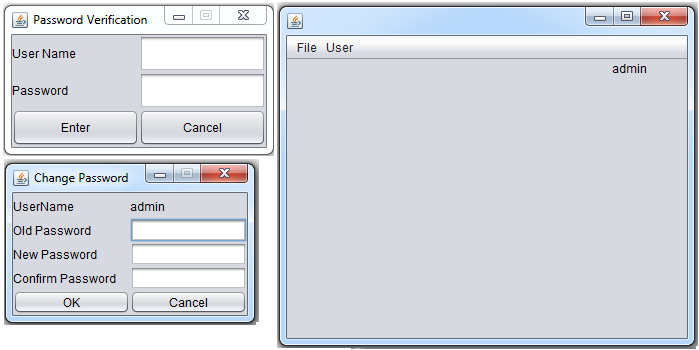


* Extract library files to your netbeans folder
* Add Library files to Project and follow these steps



Retrieving Data from ResultSet

The following example demonstrates the usage of all above explained steps. In this code example, we connect with the PasswordDB database, the one we have created earlier, and then execute the simple SQL SELECT query on Password table, and then process the query results. This example also demonstrates how we can connect multiple forms with each other.



**Password Verification Form:**

Double-click on the Enter JButton—this should take you in the Source code *btnEnterActionPerformed* method. Under the comment line, type the following code:

|  |
| --- |
| // TODO add your handling code here:  Connecting2DB c = new Connecting2DB();  String myPass=String.valueOf(txtPassword.getPassword());  if( c.matchPassword(txtUserName.getText(), myPass)){  MainFrame mf = new MainFrame();  mf.setUser(txtUserName.getText());  mf.show();  dispose();  }  else  System.out.println("Not matched..\nTry again"); |

**Main Form:**

Double-click on the Change Password Menu—this should take you in the Source code *menuChgPasswordActionPerformed* method. Under the comment line, type the following code:

|  |
| --- |
| // TODO add your handling code here:  frmChangePassword cp = new frmChangePassword();  cp.setUserName(lblUser.getText());  cp.show(); |

Double-click on the Logout Menu—this should take you in the Source code *menuLogoutActionPerformed* method. Under the comment line, type the following code:

|  |
| --- |
| // TODO add your handling code here:  frmPassword frmpass = new frmPassword();  frmpass.show();  dispose(); |

**Change Password Form:**

Double-click on the OK JButton—this should take you in the Source code *btnOKActionPerformed* method. Under the comment line, type the following code:

|  |
| --- |
| // TODO add your handling code here:  Connecting2DB c = new Connecting2DB();  String oldPass=String.valueOf(oldPassword.getPassword());  String userName = lblUserName.getText();  if( c.searchPassword(userName,oldPass)){  System.out.println("Match found");  String newpassword = String.valueOf(newPassword.getPassword());  String confirmnewpassword = String.valueOf  (confirmNewPassword.getPassword());  if(newpassword.equals(confirmnewpassword))  System.out.println("Calling updatepassword......");  c.updatePassword(userName,String.valueOf(newPassword.getPassword()));  }  else  System.out.println("Not matched..\nTry again"); |

Double-click on the Enter JButton—this should take you in the Source code btnEnterActionPerformed method. Delete the comment line and type the following code:

|  |
| --- |
| dispose(); |

The following class Connecting2DB provides the functionality to connect to database (PasswordDb.accdb) and execute the SQL commands (Select and Update) to search for the password and match it with the user name and the password. If user wants to change the password then it also updates the new password.

/\*

\* Reads the fields of Person Table from PersonalInfo database and print the screen.

\*/

public class Connecting2DB {

public Connection setConnection(){

String dataSourceName="database/PasswordDB.accdb";

String dir = System.getProperty("user.dir");

String url = "jdbc:ucanaccess://"+dir+"/" + dataSourceName;

Connection con=null;

try {

con = DriverManager.getConnection(url);

}

catch(Exception sqlEx){

System.out.println(sqlEx);

}

return con;

}

public boolean matchPassword(String user, String pass){

boolean successful =false;

try {

Connection con = setConnection();

Statement st = con.createStatement();

String sql =

"SELECT \* FROM Password where username = '"+user+"'";

ResultSet rs = st.executeQuery(sql);

while(rs.next()){

String userName = rs.getString("UserName");

String password = rs.getString("Password");

if(user.equals(userName) && pass.equals(password))

successful = true;

else

successful = false;

}

con.close();

}

catch(Exception sqlEx){

System.out.println(sqlEx);

}

return successful;

}

public boolean searchPassword(String user, String pass){

boolean found =false;

try {

Connection con = setConnection();

Statement st = con.createStatement();

String sql = "SELECT \* FROM Password where username ='"+user+"' and password = '"+pass+"'";

ResultSet rs = st.executeQuery(sql);

while(rs.next()){

String userName = rs.getString("UserName");

String password = rs.getString("Password");

if(pass.equals(password))

found = true;

else

found = false;

}

con.close();

}catch(Exception sqlEx){

System.out.println(sqlEx);

}

return found;

}

public void updatePassword(String user, String pass){

try {

Connection con = setConnection();

PreparedStatement ps = con.prepareStatement(

"UPDATE Password SET password = ? WHERE username = ? ");

ps.setString(1,pass);

ps.setString(2,user);

ps.executeUpdate();

ps.close();

}

catch(Exception sqlEx){

System.out.println(sqlEx);

}

}

}

**Note:**

We can use the following hard coded Database path but it makes it difficult to run application on

different machines unless the path to the database is modified according the location of the

database file.

String url = "jdbc:ucanaccess://C:/Users/INTEL/Documents/NetBeansProjects/prjPassword/database/PasswordDB.accdb";

**Exercise:**

Create an application for the student enrollment system where student is enrolled in the university. Create proper dataset to store information for a student an apply CRUD oerations over the students enrillement application. Also have the appropriate designing and login system for admin.

# Code:

### **import java.sql.Connection;**

### **import java.sql.DriverManager;**

### **import java.sql.PreparedStatement;**

### **import java.sql.ResultSet;**

### **import java.sql.SQLException;**

### **public class Dbconnector {**

### **private static final String DBloc="jdbc:ucanaccess://C:\\Users\\spring2018.BUKC\\Desktop\\lab13\\LAB13.accdb";**

### **private Connection conn;**

### **private PreparedStatement prepState;**

### **private ResultSet rSet;**

### 

### **public Dbconnector(){**

### **try {**

### **conn=DriverManager.getConnection(DBloc);**

### **System.out.println("Connection Established !!");**

### **} catch (SQLException e) {**

### **System.out.println("e");**

### **System.out.println("Connection Failed");**

### **}**

### **}**

### **//method for selection queries**

### **public ResultSet RunSelect(String query){**

### **try {**

### **prepState=conn.prepareStatement(query);**

### **rSet= prepState.executeQuery();//this method is only for select statements**

### 

### **} catch (SQLException e) {**

### **System.out.println(e);**

### **System.out.println("Record error");**

### **}**

### **return rSet;**

### **}**

### **public void RunDML(String query){**

### **try {**

### **prepState=conn.prepareStatement(query);**

### **prepState.executeUpdate();**

### **} catch (Exception e) {**

### **System.out.println(e);**

### **System.out.println("Error RunDML");**

### 

### **}**

### **}**

### **}**

# Output:

# 