LAB 4: JDBC

Duration: 2 Hours

Learning Outcomes

These Labsheet encompasses activities 4A and 4B.

By the end of this lab, students should be able to:

1. Design and develop an interactive web-based application

a. Explain the use of JDBC in Java programming

b. Create JDBC application

**Hardware / Software :**JCreator with the latest version of JDK.

Theory/ Terminologies

* **JDBC** is an [API](http://en.wikipedia.org/wiki/Application_programming_interface) for the [Java programming language](http://en.wikipedia.org/wiki/Java_(programming_language)) that defines how a client may access a [database](http://en.wikipedia.org/wiki/Database). It provides methods for querying and updating data in a database. JDBC is oriented towards [relational databases](http://en.wikipedia.org/wiki/Relational_database_management_system).
* JDBC was first introduced in the [Java 2 Platform, Standard Edition, version 1.1 (J2SE)](http://en.wikipedia.org/wiki/Java_Platform,_Standard_Edition), together with a reference implementation JDBC-to-[ODBC](http://en.wikipedia.org/wiki/Open_Database_Connectivity) bridge, enabling connections to any ODBC-accessible data source in the JVM host environment.

\*Before start doing the JDBC Lab Activity, students are required to **download and installs**:

**1. Java JDK, Netbeans, and XAMPP/WAMP/SQL Server (for MySQL)**

**2. mysql-connector-java-5.x.x-bin.jar (for JDBC Driver)**

Activity 4A

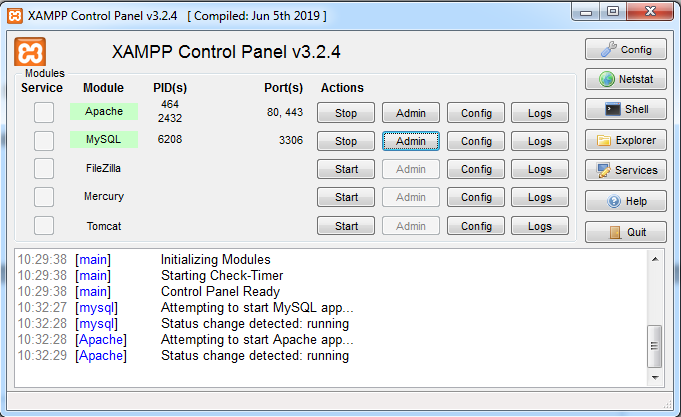
Activity Outcome : Java Swing Application with Database Connection

In this activity, we build a simple Login Application using Swing and we authenticate the application user credentials (username and password) with database username and password.

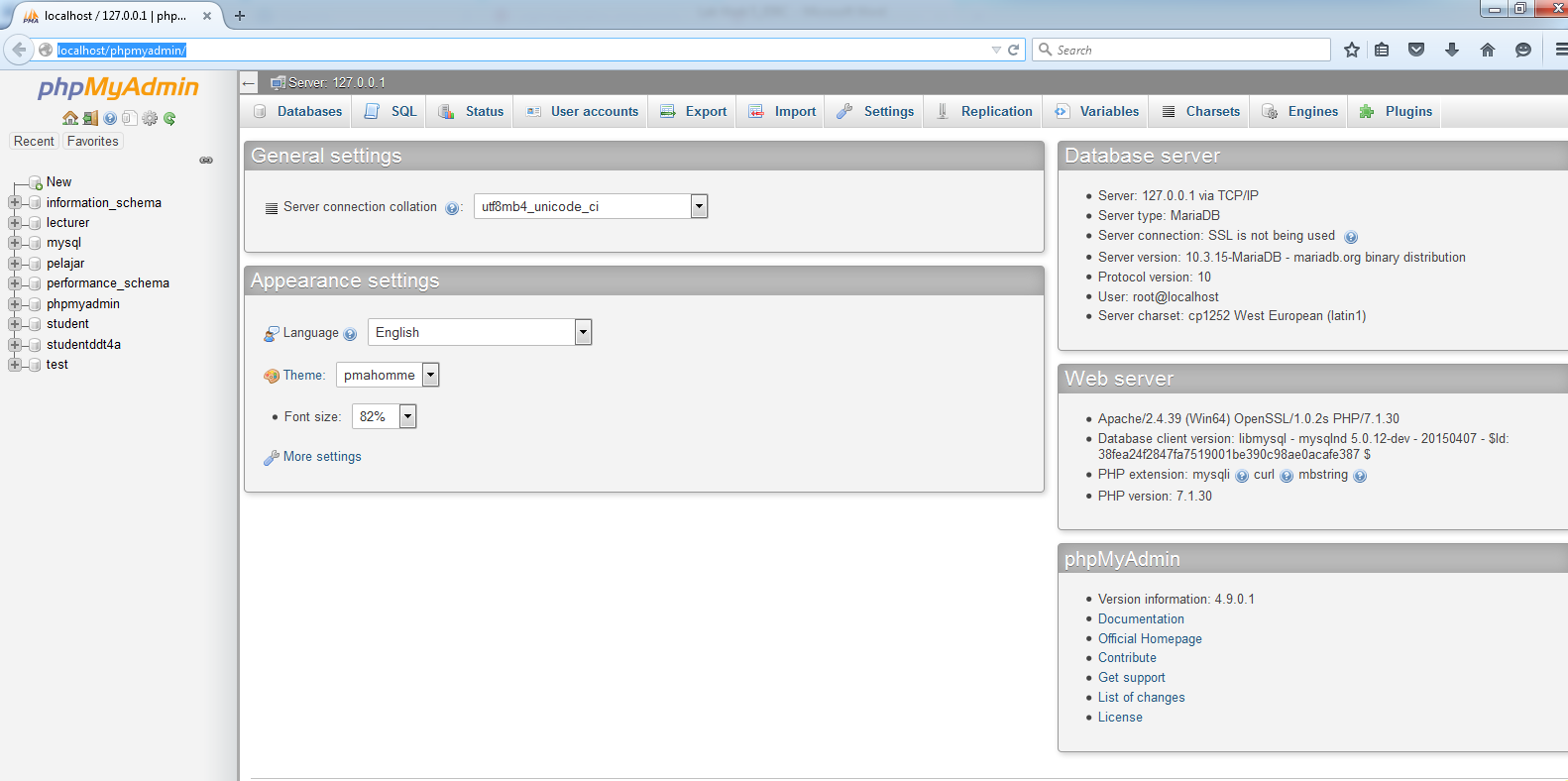
**Procedure:**

# Step 1: Database Setup using XAMPP Control Panel

Open XAMPP Control Panel as below. Click **start** button for Apache & MySQL:



Click the Admin button (MySQL) to open the phpMyAdmin (<http://localhost/phpmyadmin/>)



Let's first create a database with the following SQL statement:

create database swing\_demo;

Now, let's create a student table in the above-created database with the following SQL statement:

CREATE TABLE student

( id int NOT NULL,

name varchar(250) NOT NULL,

password varchar(250)

);

Insert a single record in the above table with the following SQL statement:

INSERT INTO student (id, name, password)

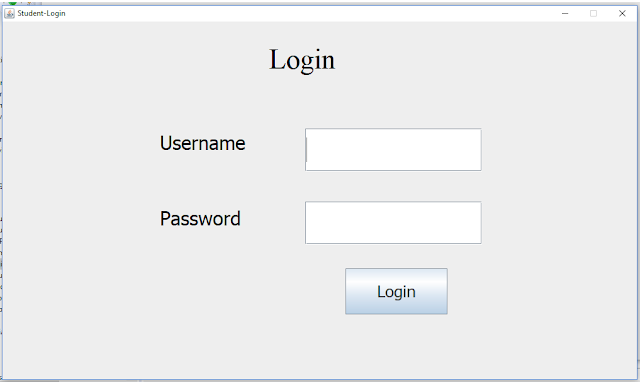
VALUES (1, 'Ramesh', 'Ramesh@123');

**Step 2: Open Netbeans IDE:**

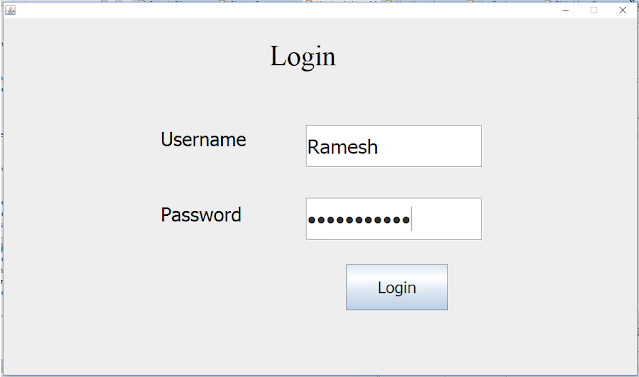
* New Project -> choose Java -> Java Application -> insert the project name: **swing-registration-from-example**
* In order to connect our Java program with the MySQL database, we need to include MySQL JDBC driver which is a JAR file, namely mysql-connector-java-8.0.13-bin.jar. **Find libraries** folder under the project that you have created; **right-click -> Add Jar Folder -> browse** the location of the driver **mysql-connector-java-5.x.x-bin.jar** -> open.
* To open database creation, go to **Services** -> **Database** icon-> **right-click** -> **New Connection** -> **Driver: MySQL (Connector/J Driver) -> Driver Files: the location where the mysql-connector-java-5.x.x-bin.jar is saved -> Next -> Test Connection.**
* Let's write a code to develop user login form using Java Swing APIs. Save a java class as **UserLogin** and type the following sample Java code.

|  |
| --- |
| import java.awt.Color;  import java.awt.EventQueue;  import java.awt.Font;  import java.awt.event.ActionEvent;  import java.awt.event.ActionListener;  **//Step 1 : import sql package**  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.SQLException;  import javax.swing.JButton;  import javax.swing.JFrame;  import javax.swing.JLabel;  import javax.swing.JOptionPane;  import javax.swing.JPanel;  import javax.swing.JPasswordField;  import javax.swing.JTextField;  import javax.swing.border.EmptyBorder;  public class UserLogin extends JFrame {  private static final long serialVersionUID = 1L;  private JTextField textField;  private JPasswordField passwordField;  private JButton btnNewButton;  private JLabel label;  private JPanel contentPane;  /\*\*  \* Launch the application.  \*/  public static void main(String[] args) {  EventQueue.invokeLater(new Runnable() {  public void run() {  try {  UserLogin frame = new UserLogin();  frame.setVisible(true);  } catch (Exception e) {  e.printStackTrace();  }  }  });  }  /\*\*  \* Create the frame.  \*/  public UserLogin() {  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  setBounds(450, 190, 1014, 597);  setResizable(false);  contentPane = new JPanel();  contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));  setContentPane(contentPane);  contentPane.setLayout(null);  // Create a Login label  JLabel lblNewLabel = new JLabel("Login");  lblNewLabel.setForeground(Color.BLACK);  lblNewLabel.setFont(new Font("Times New Roman", Font.PLAIN, 46));  lblNewLabel.setBounds(423, 13, 273, 93);  contentPane.add(lblNewLabel);  //Create username and password text fields  textField = new JTextField();  textField.setFont(new Font("Tahoma", Font.PLAIN, 32));  textField.setBounds(481, 170, 281, 68);  contentPane.add(textField);  textField.setColumns(10);  passwordField = new JPasswordField();  passwordField.setFont(new Font("Tahoma", Font.PLAIN, 32));  passwordField.setBounds(481, 286, 281, 68);  contentPane.add(passwordField);  //Create username and password labels  JLabel lblUsername = new JLabel("Username");  lblUsername.setBackground(Color.BLACK);  lblUsername.setForeground(Color.BLACK);  lblUsername.setFont(new Font("Tahoma", Font.PLAIN, 31));  lblUsername.setBounds(250, 166, 193, 52);  contentPane.add(lblUsername);  JLabel lblPassword = new JLabel("Password");  lblPassword.setForeground(Color.BLACK);  lblPassword.setBackground(Color.CYAN);  lblPassword.setFont(new Font("Tahoma", Font.PLAIN, 31));  lblPassword.setBounds(250, 286, 193, 52);  contentPane.add(lblPassword);  btnNewButton = new JButton("Login");  btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 26));  btnNewButton.setBounds(545, 392, 162, 73);  btnNewButton.addActionListener(new ActionListener() {  public void actionPerformed(ActionEvent e) {  String userName = textField.getText();  String password = passwordField.getText();  try {  **//Step 2 : Register/Load JDBC (MySQL) driver**  Class.forName("com.mysql.jdbc.Driver");    **//Step 3 : Open/Establish a connection**  Connection connection = (Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/swing\_demo",  "root", "root");  **//Step 4 : Execute a query using suitable statements object**  PreparedStatement st = (PreparedStatement) connection  .prepareStatement("Select name, password from student where name=? and password=?");  st.setString(1, userName);  st.setString(2, password);  **//Step 5 : Process the results/Extract data from result set**  ResultSet rs = st.executeQuery();  if (rs.next()) {  dispose();  UserHome ah = new UserHome(userName);  ah.setTitle("Welcome");  ah.setVisible(true);  JOptionPane.showMessageDialog(btnNewButton, "You have successfully logged in");  } else {  JOptionPane.showMessageDialog(btnNewButton, "Wrong Username & Password");  }  } catch (SQLException sqlException) {  sqlException.printStackTrace();  }  }  });  contentPane.add(btnNewButton);  label = new JLabel("");  label.setBounds(0, 0, 1008, 562);  contentPane.add(label);  }  } |

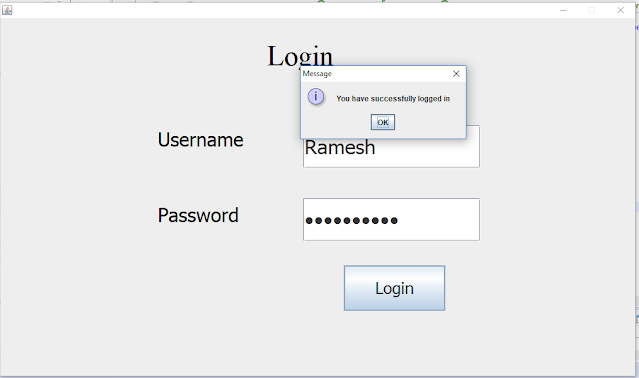
* Let’s compile and run the **UserLogin.java** file and the below screenshot shows the step by step execution of this application.
  + **User Login**



* + **Enter User Login Details**



* + **User Login Success**



Activity 4B

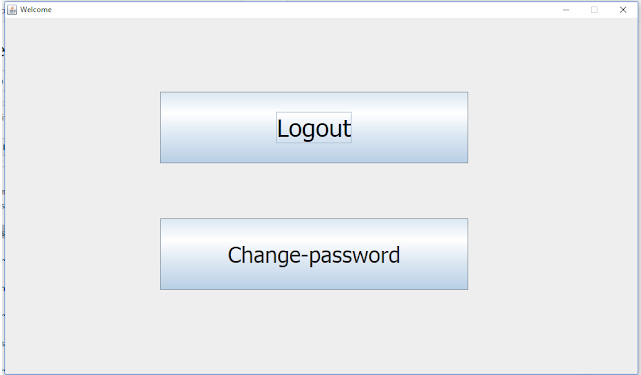
Activity Outcome : Java Swing Application with Database Connection

By using same database and project from Activity 4A, create java class to develop Change Password Form to change the current password into a new password as output below.

**Procedure:**

**Step 1: Develop User Home Page**

* Let's write a code to develop user home page using Java Swing APIs. Save a java class as **UserHome** and type the following sample Java code.
* After user login successfully, this page will be displayed.



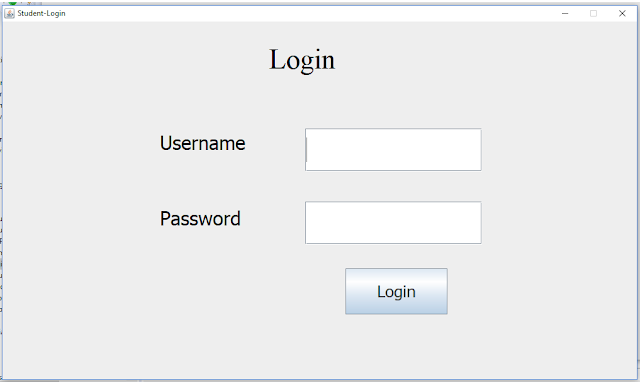
|  |
| --- |
| import java.awt.Color;  import java.awt.EventQueue;  import java.awt.Font;  import java.awt.event.ActionEvent;  import java.awt.event.ActionListener;  import javax.swing.JButton;  import javax.swing.JFrame;  import javax.swing.JOptionPane;  import javax.swing.JPanel;  import javax.swing.UIManager;  import javax.swing.border.EmptyBorder;  public class UserHome extends JFrame {  private static final long serialVersionUID = 1 L;  private JPanel contentPane;  /\*\*  \* Launch the application.  \*/  public static void main(String[] args) {  EventQueue.invokeLater(new Runnable() {  public void run() {  try {  UserHome frame = new UserHome();  frame.setVisible(true);  } catch (Exception e) {  e.printStackTrace();  }  }  });  }  public UserHome() {  }  /\*\*  \* Create the frame.  \*/  public UserHome(String userSes) {  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  setBounds(450, 190, 1014, 597);  setResizable(false);  contentPane = new JPanel();  contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));  setContentPane(contentPane);  contentPane.setLayout(null);  JButton btnNewButton = new JButton("Logout");  btnNewButton.setForeground(new Color(0, 0, 0));  btnNewButton.setBackground(UIManager.getColor("Button.disabledForeground"));  btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 39));  btnNewButton.addActionListener(new ActionListener() {  public void actionPerformed(ActionEvent e) {  int a = JOptionPane.showConfirmDialog(btnNewButton, "Are you sure?");  // JOptionPane.setRootFrame(null);  if (a == JOptionPane.YES\_OPTION) {  dispose();  UserLogin obj = new UserLogin();  obj.setTitle("Student-Login");  obj.setVisible(true);  }  dispose();  UserLogin obj = new UserLogin();  obj.setTitle("Student-Login");  obj.setVisible(true);  }  });  btnNewButton.setBounds(247, 118, 491, 114);  contentPane.add(btnNewButton);  JButton button = new JButton("Change-password\r\n");  button.setBackground(UIManager.getColor("Button.disabledForeground"));  button.addActionListener(new ActionListener() {  public void actionPerformed(ActionEvent e) {  ChangePassword bo = new ChangePassword(userSes);  bo.setTitle("Change Password");  bo.setVisible(true);  }  });  button.setFont(new Font("Tahoma", Font.PLAIN, 35));  button.setBounds(247, 320, 491, 114);  contentPane.add(button);  }  } |

**Step 2: Develop Change Password Form**

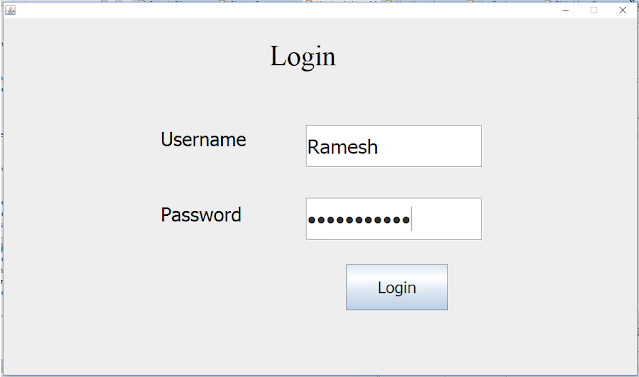
* Let's write a code to develop change password page using Java Swing APIs. Save a java class as **ChangePassword** and type the following sample Java code.

|  |
| --- |
| import java.awt.Color;  import java.awt.EventQueue;  import java.awt.Font;  import java.awt.event.ActionEvent;  import java.awt.event.ActionListener;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.PreparedStatement;  import java.sql.SQLException;  import javax.swing.JButton;  import javax.swing.JFrame;  import javax.swing.JLabel;  import javax.swing.JOptionPane;  import javax.swing.JPanel;  import javax.swing.JTextField;  import javax.swing.border.EmptyBorder;  public class ChangePassword extends JFrame {  private static final long serialVersionUID = 1L;  private JPanel contentPane;  private JTextField textField;  private JLabel lblEnterNewPassword;  /\*\*  \* Launch the application.  \*/  public static void main(String[] args) {  EventQueue.invokeLater(new Runnable() {  public void run() {  try {  } catch (Exception e) {  e.printStackTrace();  }  }  });  }  /\*\*  \* Create the frame.  \*/  public ChangePassword(String name) {  setBounds(450, 360, 1024, 234);  setResizable(false);  contentPane = new JPanel();  contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));  setContentPane(contentPane);  contentPane.setLayout(null);  textField = new JTextField();  textField.setFont(new Font("Tahoma", Font.PLAIN, 34));  textField.setBounds(373, 35, 609, 67);  contentPane.add(textField);  textField.setColumns(10);  JButton btnSearch = new JButton("Enter");  btnSearch.addActionListener(new ActionListener() {  public void actionPerformed(ActionEvent e) {  String pstr = textField.getText();  try {  System.out.println("update password name " + name);  System.out.println("update password");  Class.forName("com.mysql.jdbc.Driver");  Connection con = (Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/swing\_demo",  "root", "root");  PreparedStatement st = (PreparedStatement) con  .prepareStatement("Update student set password=? where name=?");  st.setString(1, pstr);  st.setString(2, name);  st.executeUpdate();  JOptionPane.showMessageDialog(btnSearch, "Password has been successfully changed");  } catch (SQLException sqlException) {  sqlException.printStackTrace();  }  }  });  btnSearch.setFont(new Font("Tahoma", Font.PLAIN, 29));  btnSearch.setBackground(new Color(240, 240, 240));  btnSearch.setBounds(438, 127, 170, 59);  contentPane.add(btnSearch);  lblEnterNewPassword = new JLabel("Enter New Password :");  lblEnterNewPassword.setFont(new Font("Tahoma", Font.PLAIN, 30));  lblEnterNewPassword.setBounds(45, 37, 326, 67);  contentPane.add(lblEnterNewPassword);  }  } |

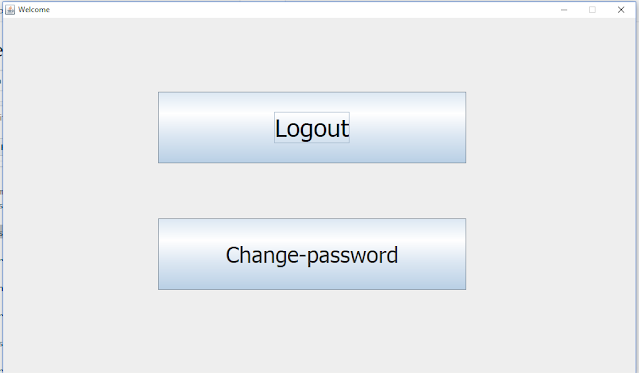
* Let’s compile and run the **UserLogin.java** file and the below screenshot shows the step by step execution of this application.
  + **User Login**



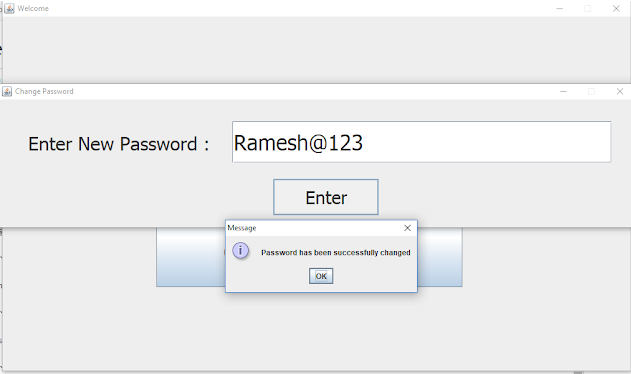
* + **Enter User Login Details**



* + **User Home Page**



* + **User Change Password**



* + **User Logout**

