CSE1007 – JAVA PROGRAMMING

JAVA FX FORM AND JDBC CONNECTION WITH MYSQL

DIGITAL ASSIGNMENT – 02

CHANDRU S 19BCE1878 FACULTY: DR. R RADHA

CRUD Task 1: READ

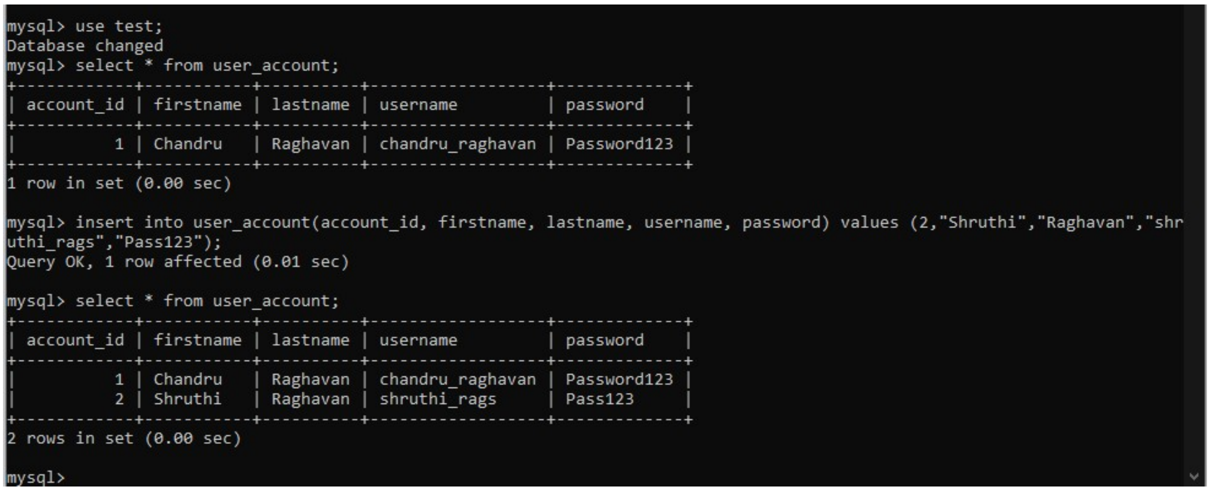
Functionality Involved:

READING from MySQL Database

User inputs username and password. We read the username and password and if exists and matches with values stored in our database, a “Login Successful” message is shown, else “Login unsuccessful, try again” message is shown with the help of a DialogBox.

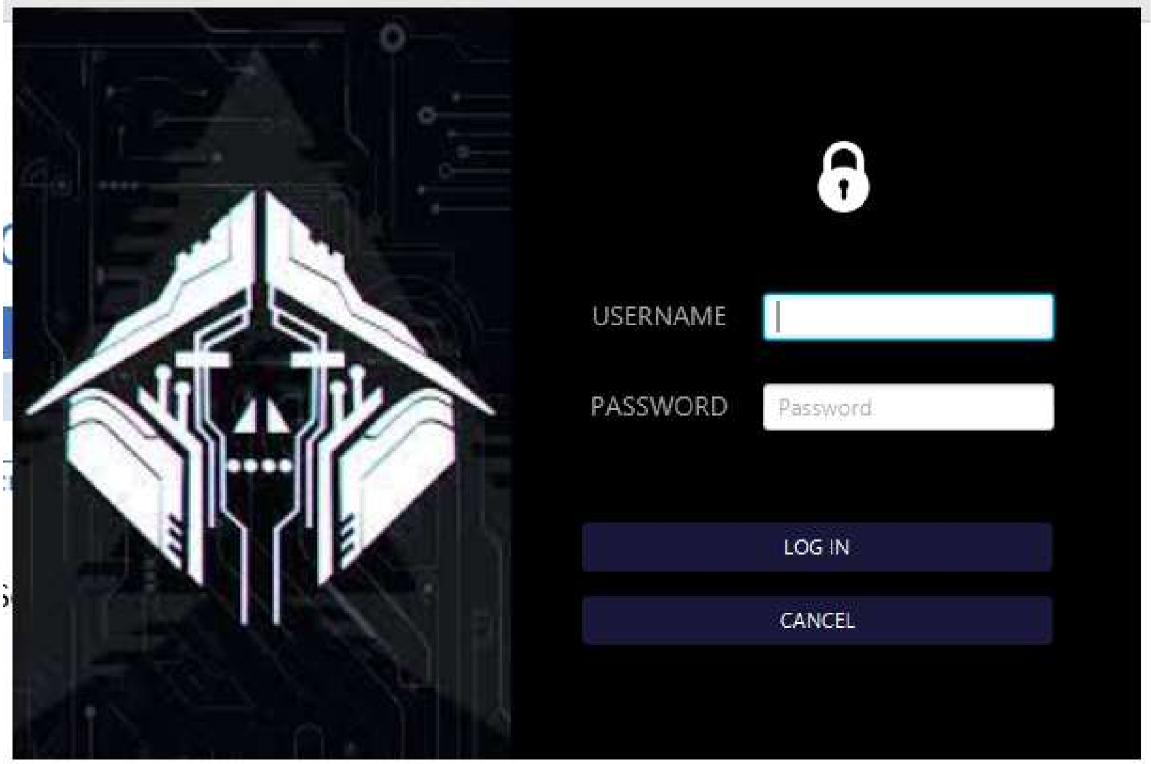
Tools used: Eclipse IDE, MySQL Command Line Prompt, JavaFx Scene Builder

Screenshot of the table in the database:

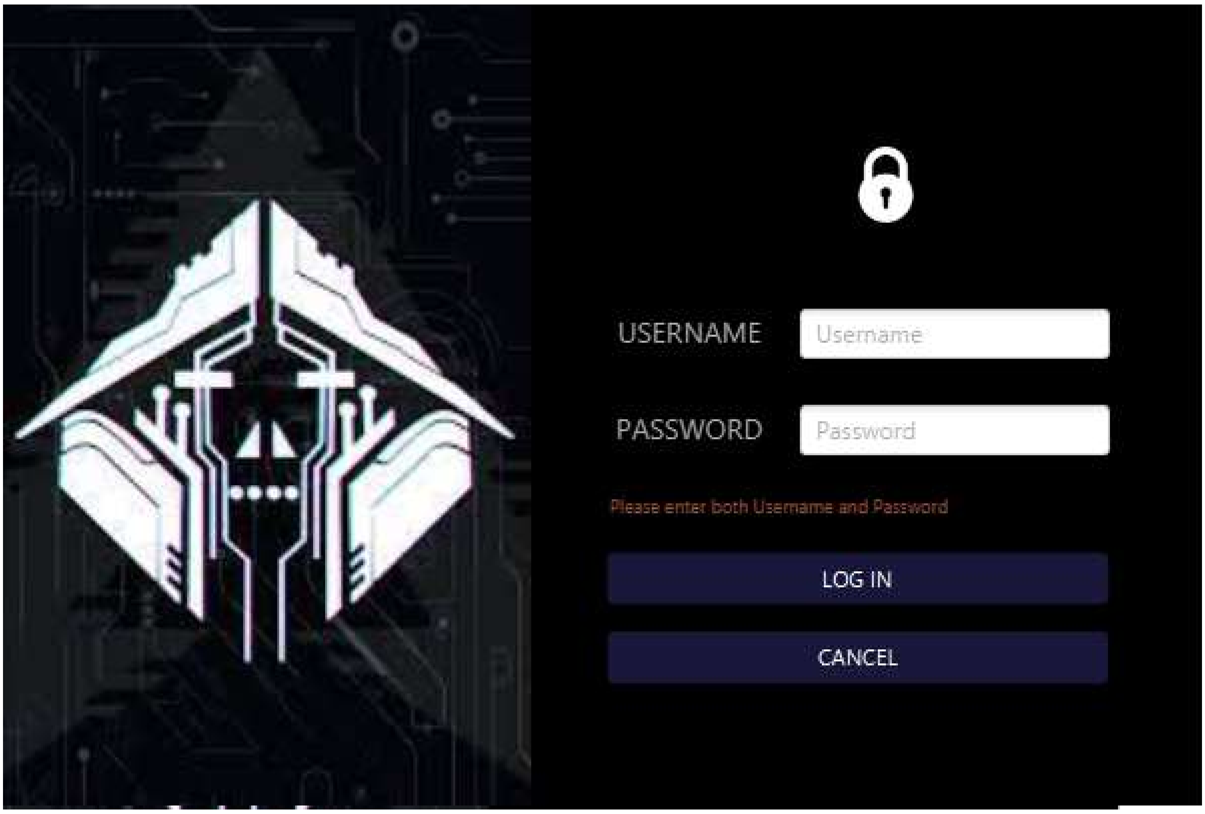


```
mysql> use test;
Database changed
mysql> select * from user_account;
+------------+-----------+----------+------------------+-------------+
| account_id | firstname | lastname | username         | password    |
+------------+-----------+----------+------------------+-------------+
|          1 | Chandru   | Raghavan | chandru_raghavan | Password123 |
+------------+-----------+----------+------------------+-------------+
1 row in set (0.00 sec)

mysql> insert into user_account(account_id, firstname, lastname, username, password) values (2,"Shruthi","Raghavan","shr
uthi_rags","Pass123");
Query OK, 1 row affected (0.01 sec)

mysql> select * from user_account;
+------------+-----------+----------+------------------+-------------+
| account_id | firstname | lastname | username         | password    |
+------------+-----------+----------+------------------+-------------+
|          1 | Chandru   | Raghavan | chandru_raghavan | Password123 |
|          2 | Shruthi   | Raghavan | shruthi_rags     | Pass123     |
+------------+-----------+----------+------------------+-------------+
2 rows in set (0.00 sec)

mysql>
```

Screenshot of the form:

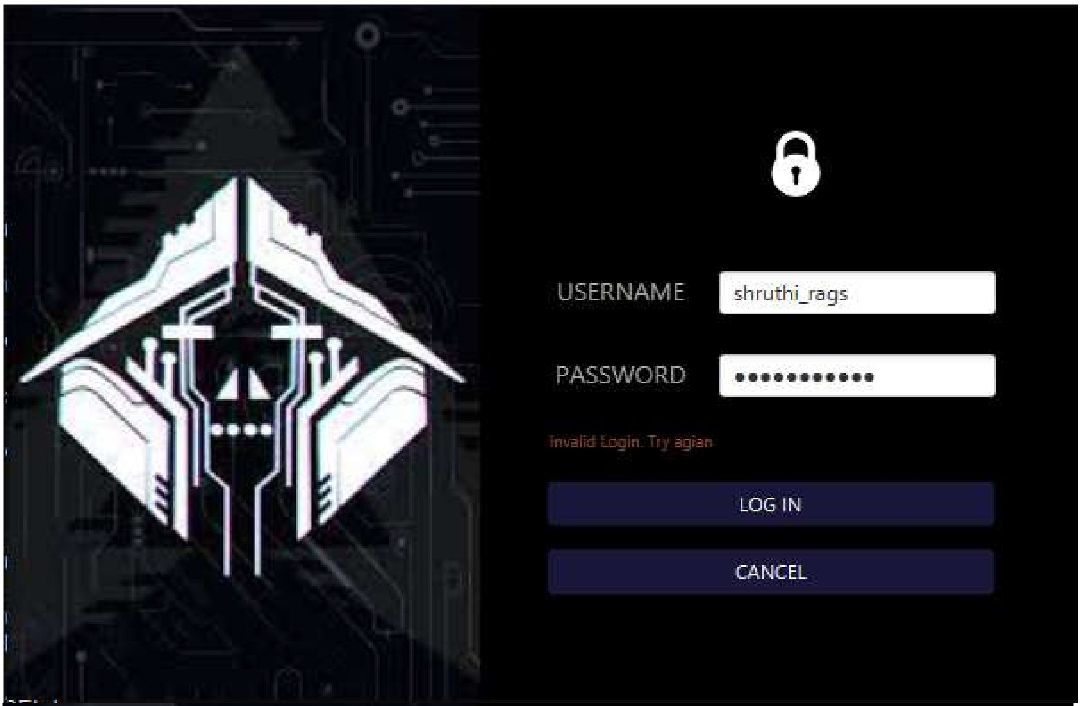


Case 1: No input given, Login Button pressed



Please enter both Username and Password

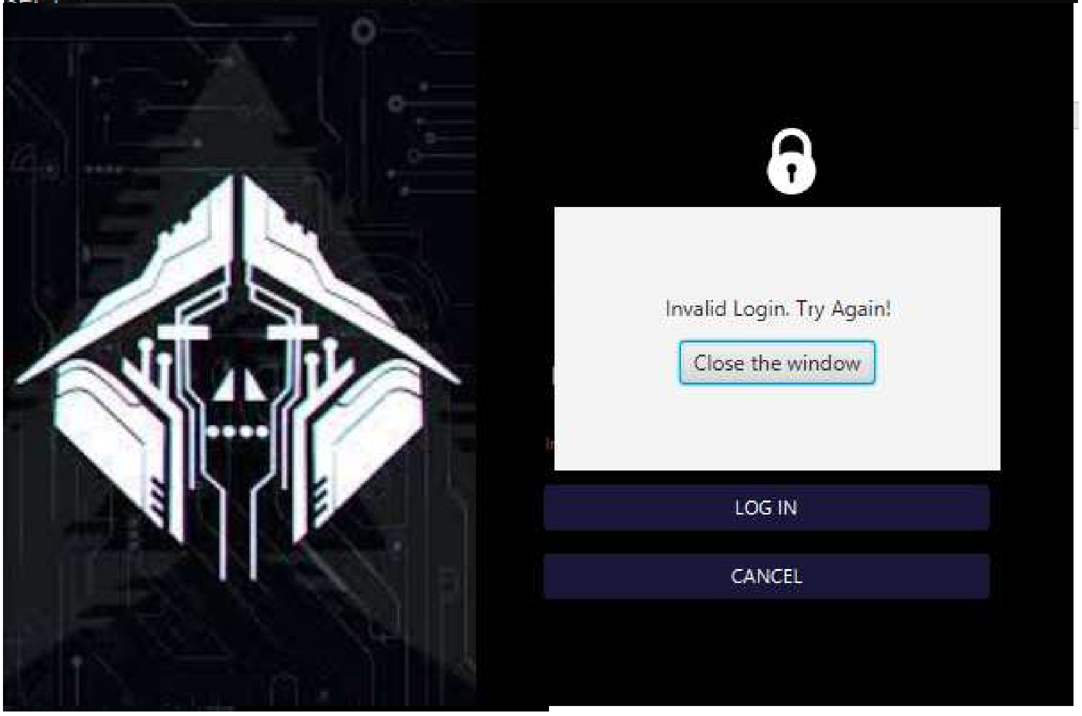
Case 2: Invalid username or passowrd given(here *invalid password* is given, similar case when *invalid username* is given)



USERNAME shruthi_rags

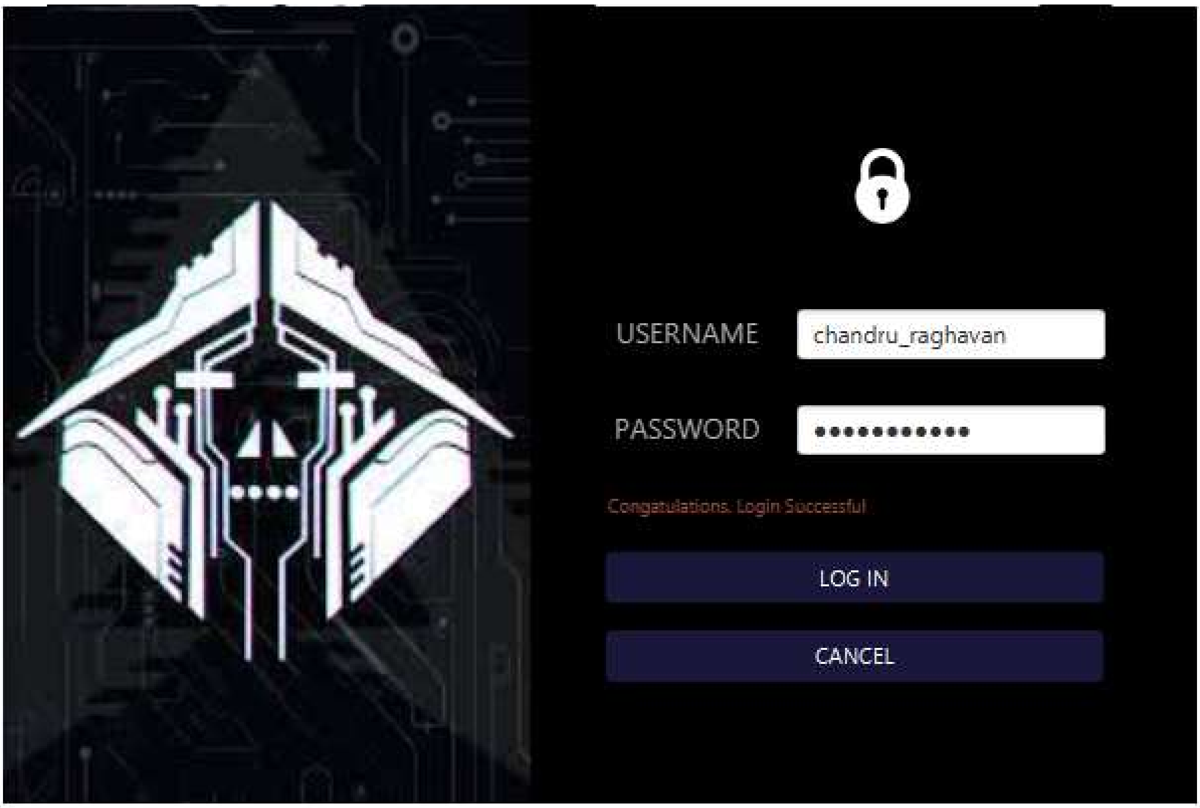
PASSWORD ●●●●●●●●●●●●

Invalid Login. Try agian



Invalid Login. Try Again!

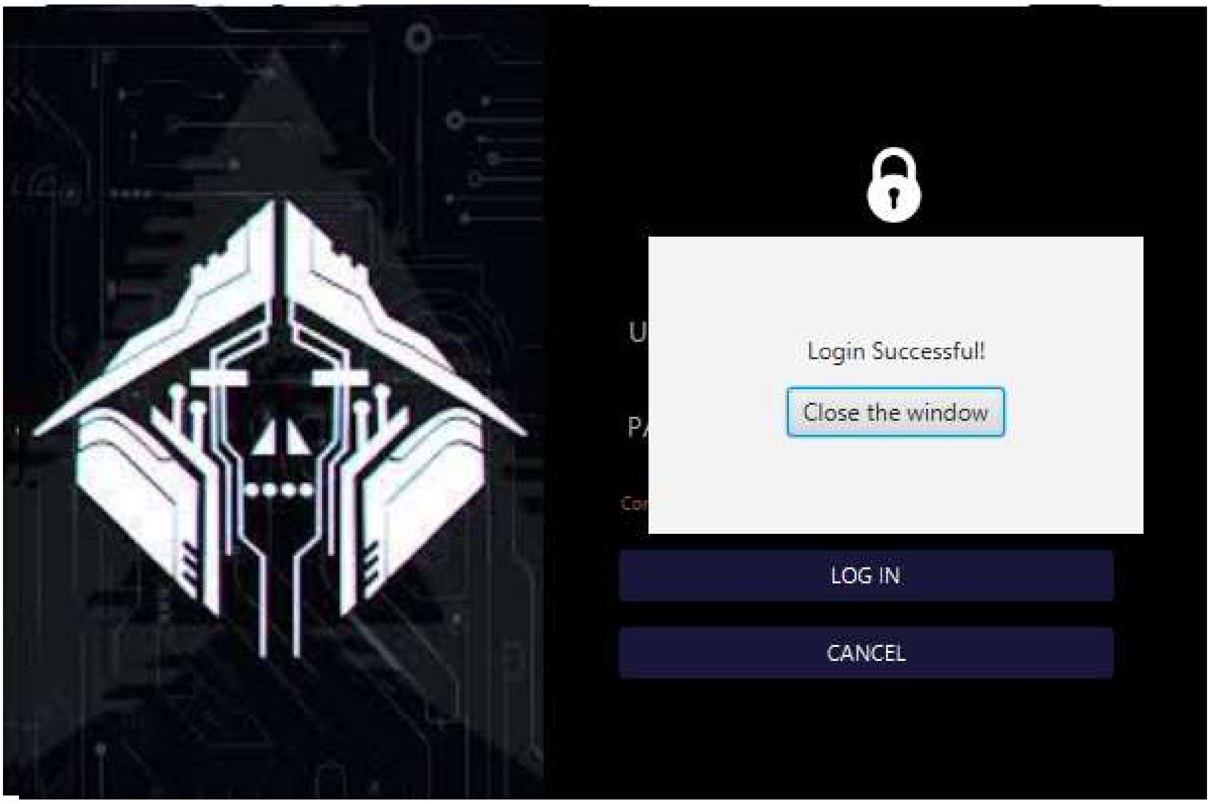
Case 3: Successful Login – both password and username is right.



USERNAME chandru_raghavan

PASSWORD ●●●●●●●●●●●●

Congratulations. Login Successful



Login Successful!

Code:

i. Main.

```
package application;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.stage.Stage;
import javafx.stage.StageStyle;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.layout.BorderPane;


public class Main extends Application {
	@Override
	public void start(Stage primaryStage) {
		try {
			Parent root = 
FXMLLoader.load(getClass().getResource("login.fxml"));
//			primaryStage.setTitle("Hello World");
			primaryStage.initStyle(StageStyle.UNDECORATED);
			primaryStage.setScene(new Scene(root));
			primaryStage.show();
		} catch(Exception e) {
			e.printStackTrace();
		}
	}

	public static void main(String[] args) {
		launch(args);
	}
	  }
```

ii. LoginController.java

```
package application;

import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.PasswordField;
import javafx.scene.control.TextField;
import javafx.stage.Stage;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;

import javafx.event.*;
import javafx.fxml.FXML;

public class LoginController {
	@FXML
	private Button cancelButton;
	@FXML
	private Label loginMessageLabel;
	@FXML
	private TextField usernameTextField;
	@FXML
	private PasswordField enterPasswordField;
	
	
	
	public void loginButtonOnAction() {
		
		if(usernameTextField.getText().isBlank() == false &&
enterPasswordField.getText().isBlank() == false)
		{
			validateLogin();
		}
		else
		{
			loginMessageLabel.setText("Please enter both
Username and Password");
		}
		
	}
	
	public void cancelButtonOnAction(ActionEvent event)
	{
		Stage stage = (Stage)
cancelButton.getScene().getWindow();
```

```
        stage.close();

    }

    public void validateLogin()
    {
        DatabaseConnection connectNow = new DatabaseConnection();
        Connection connectDB = connectNow.getConnection();

        String verifyLogin = "Select count(1) from user_account
where username = '"+usernameTextField.getText()+"' and password
='"+enterPasswordField.getText()+"'";
        try
        {
            Statement statement = connectDB.createStatement();
            ResultSet rs = statement.executeQuery(verifyLogin);
            while(rs.next())
            {
                if(rs.getInt(1)>=1)
                {

    loginMessageLabel.setText("Congatulations. Login Successful");
                    DialogBox.display("Welcome","Login
Successful!");
                }
                else
                {
                    loginMessageLabel.setText("Invalid Login.
Try agian");

                    DialogBox.display("Sorry","Invalid Login.
Try Again!");
                }
            }
        }
        catch(Exception e)
        {
            e.printStackTrace();
            e.getCause();
        }

    }

      }
```

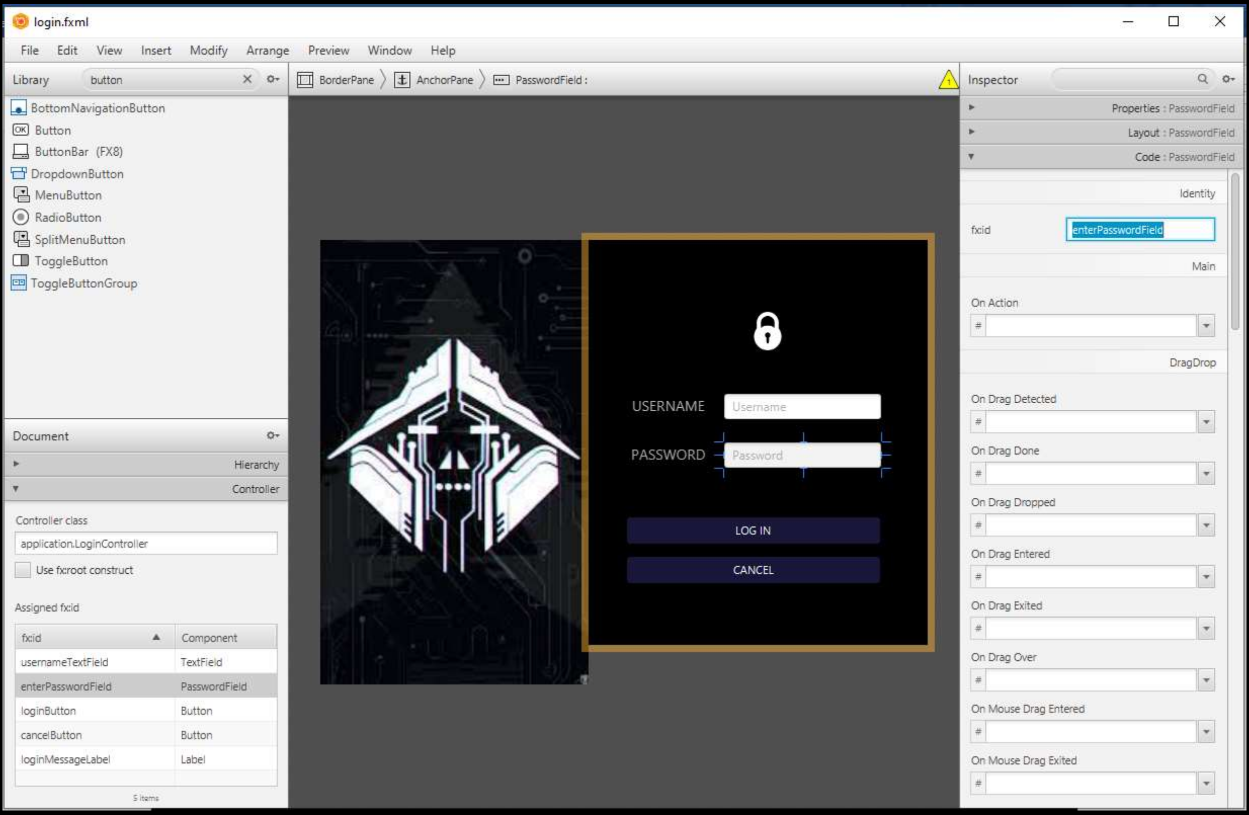
iii. DatabaseConnection.java

```
package application;

import java.sql.Connection;
import java.sql.DriverManager;

public class DatabaseConnection {
	public Connection databaseLink;

	public Connection getConnection() {
//		String databaseName = "test";
		String databaseUser = "root";
		String databasePassword = "root";
		String url ="jdbc:mysql://localhost:3306/test";
		try
		{
			Class.forName("com.mysql.cj.jdbc.Driver");
			databaseLink = 
DriverManager.getConnection(url,databaseUser,databasePassword);
		}
		catch(Exception e)
		{
			e.printStackTrace();
			e.getCause();
		}
		return databaseLink;
	}

}
```

iv. DialogBox.java

```
package application;
import javafx.stage.*;
import javafx.scene.*;
import javafx.scene.layout.*;
import javafx.scene.control.*;
import javafx.geometry.*;

public class DialogBox {

	public static void display(String title, String message)
	{
        Stage window = new  Stage();
        window.initModality(Modality.APPLICATION_MODAL);
        window.setTitle(title);
        window.setMinHeight(150);
        window.setMinWidth(250);
        window.initStyle(StageStyle.UNDECORATED);

        Label label = new Label();
        label.setText(message);
        Button closeButton = new Button("Close the window");
        closeButton.setOnAction(e->window.close());

        VBox layout = new VBox(10);
        layout.getChildren().addAll(label, closeButton);
        layout.setAlignment(Pos.CENTER);

        Scene scene = new Scene(layout);
        window.setScene(scene);
        window.showAndWait();
    }

}
```

SCENEBUILDER:



fx:id: enterPasswordField

Controller class: application.LoginController

☐ Use fx:root construct

Assigned fx:id

fx:id ▲	Component
usernameTextField	TextField
enterPasswordField	PasswordField
loginButton	Button
cancelButton	Button
loginMessageLabel	Label

5 items

CRUD Task 2: CREATE

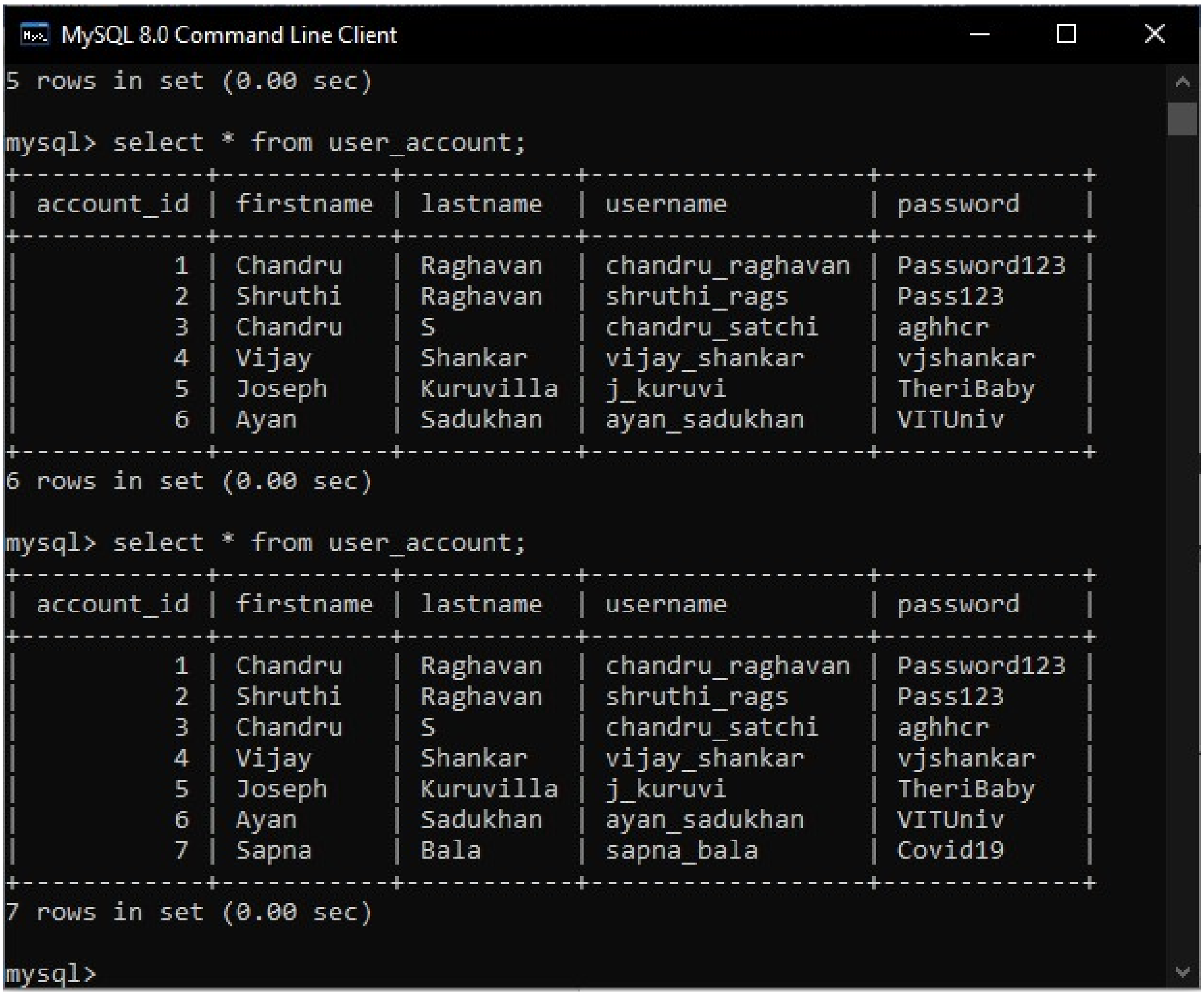
Functionality Involved:

Writing data into an existing table in the MYSQL Database.

A simple user registration form is designed where we collect Account Id, First name, Last name, username and password from the user and write it as a new entry in the table in our database.

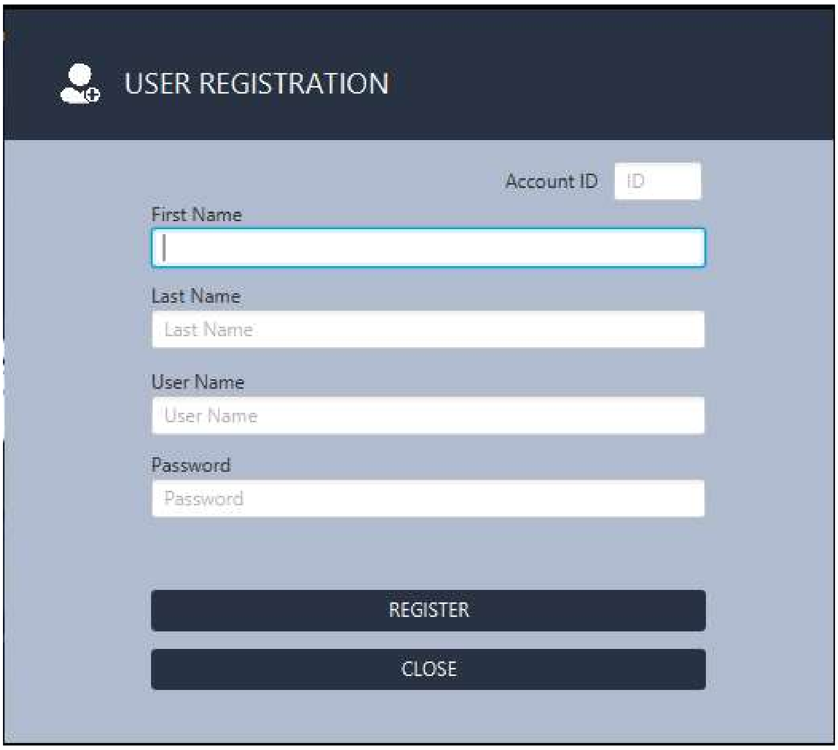
Tools used: Eclipse IDE, MySQL Command Line Prompt, JavaFx Scene Builder

Database Screenshot –

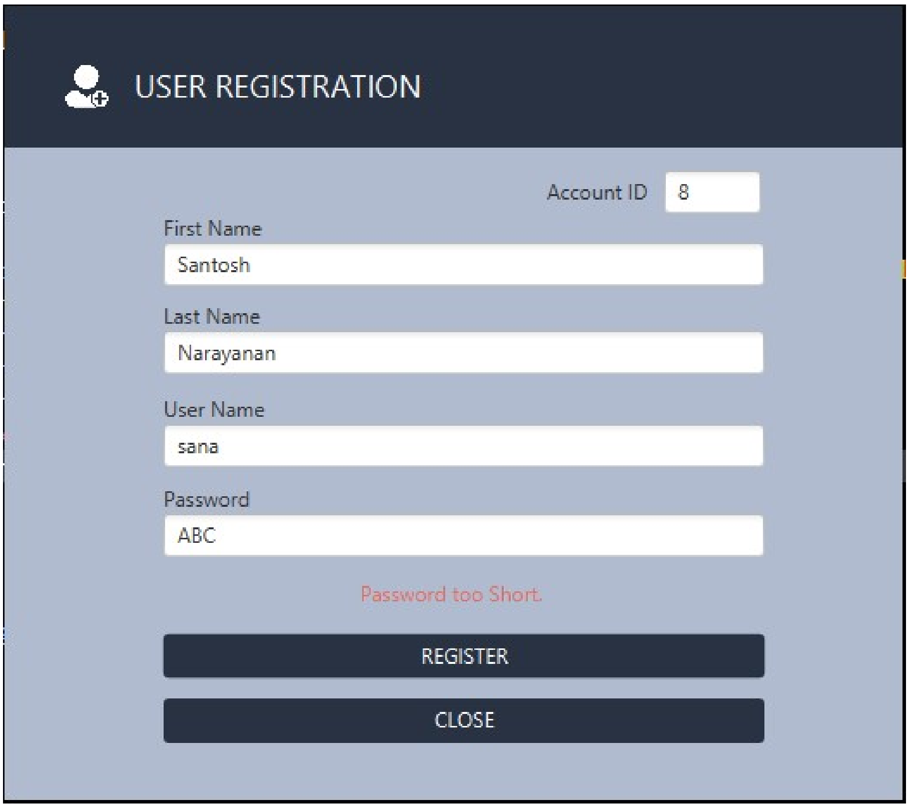


```
5 rows in set (0.00 sec)

mysql> select * from user_account;
+------------+----------+-----------+-----------------+------------+
| account_id | firstname | lastname  | username        | password   |
+------------+----------+-----------+-----------------+------------+
|          1 | Chandru  | Raghavan  | chandru_raghavan | Password123 |
|          2 | Shruthi  | Raghavan  | shruthi_rags    | Pass123    |
|          3 | Chandru  | S         | chandru_satchi  | aghhcr     |
|          4 | Vijay    | Shankar   | vijay_shankar   | vjshankar  |
|          5 | Joseph   | Kuruvilla | j_kuruvi        | TheriBaby  |
|          6 | Ayan     | Sadukhan  | ayan_sadukhan   | VITUniv    |
+------------+----------+-----------+-----------------+------------+
6 rows in set (0.00 sec)

mysql> select * from user_account;
+------------+----------+-----------+-----------------+------------+
| account_id | firstname | lastname  | username        | password   |
+------------+----------+-----------+-----------------+------------+
|          1 | Chandru  | Raghavan  | chandru_raghavan | Password123 |
|          2 | Shruthi  | Raghavan  | shruthi_rags    | Pass123    |
|          3 | Chandru  | S         | chandru_satchi  | aghhcr     |
|          4 | Vijay    | Shankar   | vijay_shankar   | vjshankar  |
|          5 | Joseph   | Kuruvilla | j_kuruvi        | TheriBaby  |
|          6 | Ayan     | Sadukhan  | ayan_sadukhan   | VITUniv    |
|          7 | Sapna    | Bala      | sapna_bala      | Covid19    |
+------------+----------+-----------+-----------------+------------+
7 rows in set (0.00 sec)

mysql>
```

User Registration screen snippets:



Case: When password is too short.



Account ID 8

First Name
Santosh

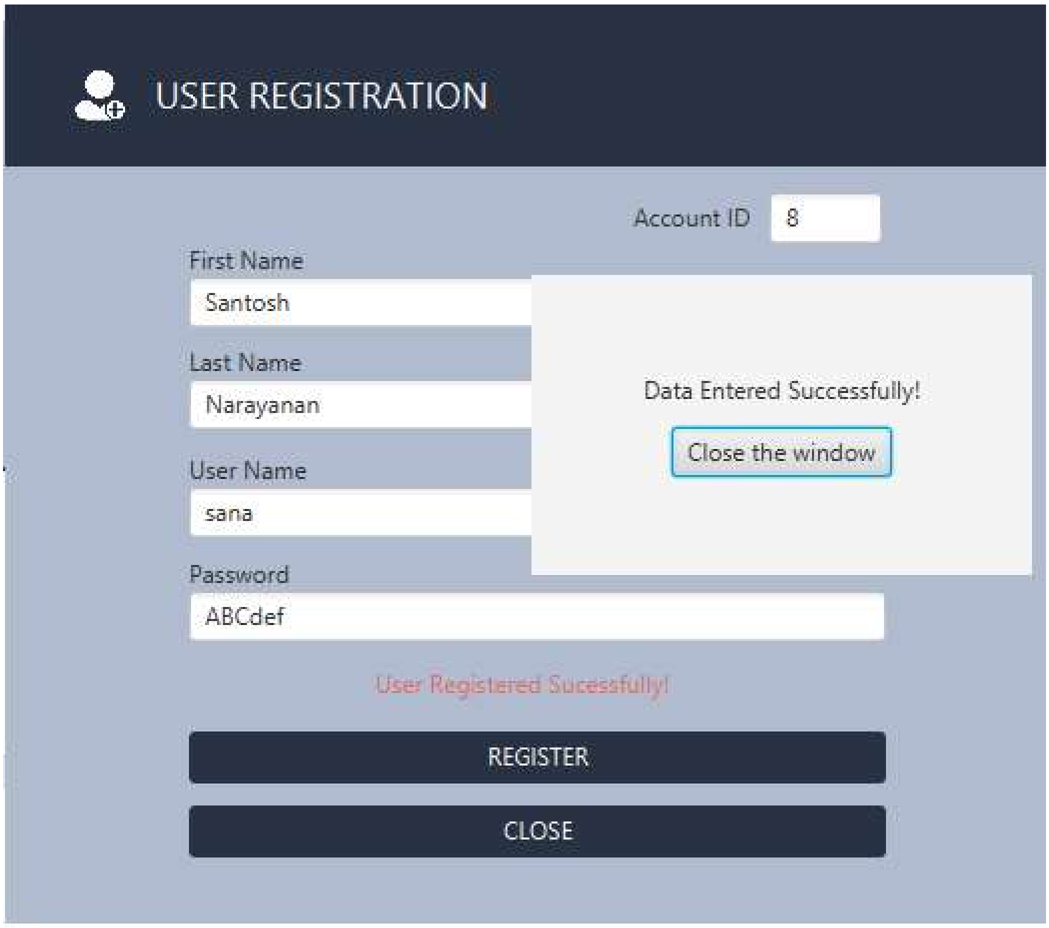
Last Name
Narayanan

User Name
sana

Password
ABC

Password too Short.

Case: Successfully data entry.



Account ID 8

First Name Santosh

Last Name Narayanan

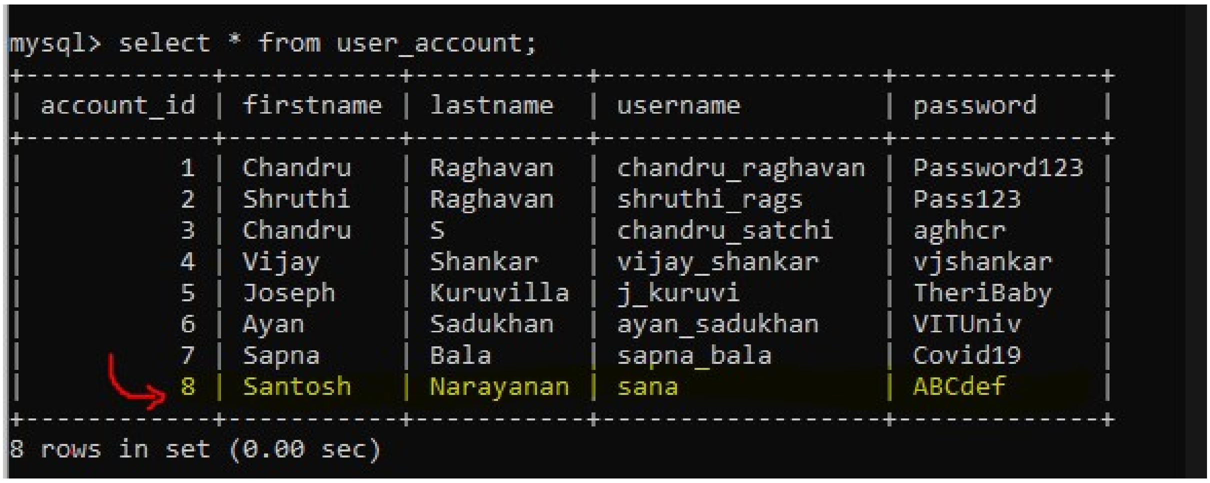
User Name sana

Password ABCdef

Data Entered Successfully!

User Registered Sucessfully!

Updated Table “**user-account”** in Database “**test**”



```
mysql> select * from user_account;
+------------+-----------+-----------+-------------------+-------------+
| account_id | firstname | lastname  | username          | password    |
+------------+-----------+-----------+-------------------+-------------+
|          1 | Chandru   | Raghavan  | chandru_raghavan  | Password123 |
|          2 | Shruthi   | Raghavan  | shruthi_rags      | Pass123     |
|          3 | Chandru   | S         | chandru_satchi    | aghhcr      |
|          4 | Vijay     | Shankar   | vijay_shankar     | vjshankar   |
|          5 | Joseph    | Kuruvilla | j_kuruvi          | TheriBaby   |
|          6 | Ayan      | Sadukhan  | ayan_sadukhan     | VITUniv     |
|          7 | Sapna     | Bala      | sapna_bala        | Covid19     |
|          8 | Santosh   | Narayanan | sana              | ABCdef      |
+------------+-----------+-----------+-------------------+-------------+
8 rows in set (0.00 sec)
```

CODE:

i. **RegisterController.java**

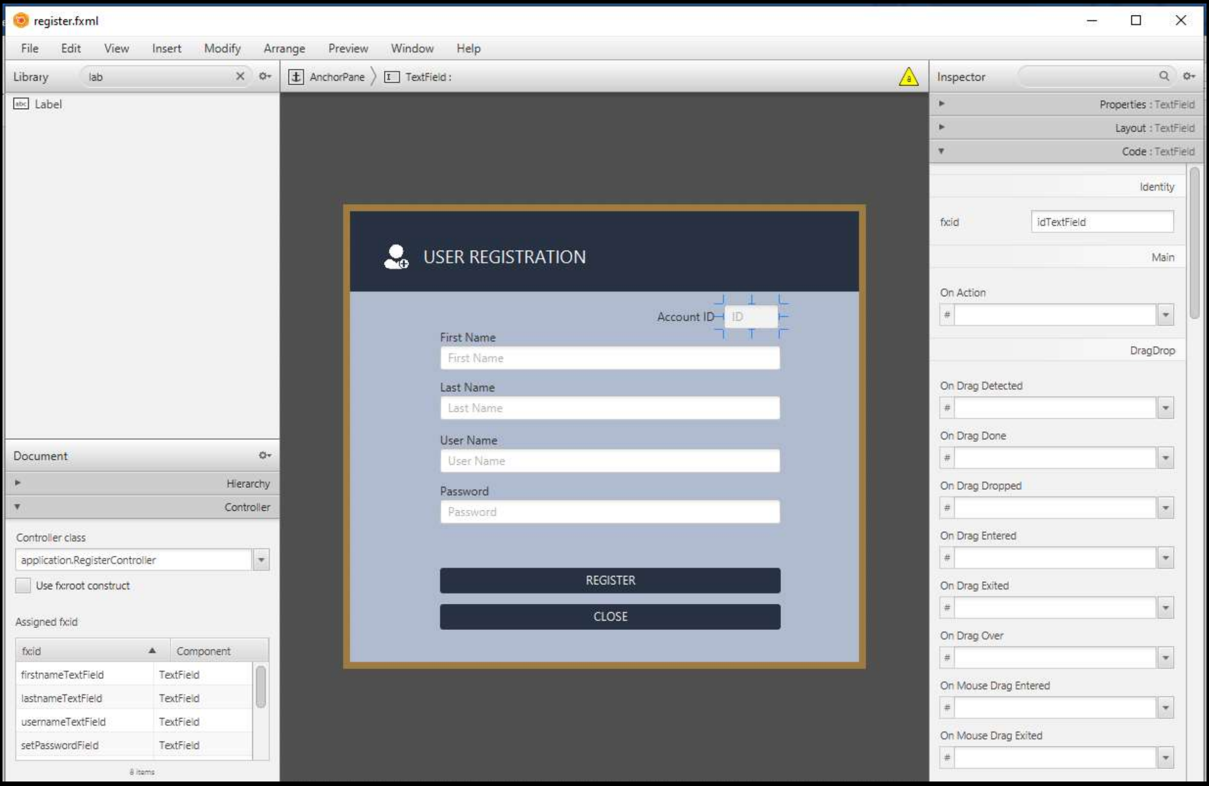
```
 package application;
import javafx.application.Platform;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.TextField;
import javafx.stage.Stage;

import java.sql.Connection;
import java.sql.Statement;




public class RegisterController {

	@FXML
	private Button closeButton;
	@FXML
	private Button registerButton;
	@FXML
	private Label registrationMessageLabel;
	@FXML
	private TextField setPasswordField;
	@FXML
	private TextField firstnameTextField;
	@FXML
	private TextField lastnameTextField;
	@FXML
	private TextField usernameTextField;
	@FXML
	private TextField idTextField;


	public void registerButtonOnAction(ActionEvent event)
	{

		registerUser();
	}

	public void closeButtonOnAction(ActionEvent event)
	{
		Stage stage = (Stage) closeButton.getScene().getWindow();
		stage.close();
		Platform.exit();
```

```
      }

      public void registerUser()
      {
           if(setPasswordField.getText().length()>=5)
           {
                DatabaseConnection connectNow = new
DatabaseConnection();
                Connection connectDB = connectNow.getConnection();

                String id = idTextField.getText();
                String firstname = firstnameTextField.getText();
                String lastname = lastnameTextField.getText();
                String username = usernameTextField.getText();
                String password = setPasswordField.getText();

                String insertFields = "INSERT INTO
user_account(account_id,firstname,lastname,username,password) VALUES
(";
                String insertValues = id+"','"+firstname +"','"+
lastname+"','"+ username+"','"+ password+"');";
                String insertToRegister = insertFields +
insertValues;

                System.out.println(insertToRegister);
                try
                {
                      Statement statement =
connectDB.createStatement();
                      statement.executeUpdate(insertToRegister);
                      registrationMessageLabel.setText("User
Registered Sucessfully!");
                      DialogBox.display("Congrats","Data Entered
Successfully!");
                      idTextField.setText("");
                      firstnameTextField.setText("");
                      lastnameTextField.setText("");
                      usernameTextField.setText("");
                      setPasswordField.setText("");

                }
                catch(Exception e)
                {
                      e.printStackTrace();
                      e.getCause();
                }
           }
           else
           {
```

```
                    registrationMessageLabel.setText("Password too
Short.");
              }
       }




}
```

DatabaseConnector.java and other Files are same as the above Task – 1

SCENEBUILDER



Library: lab

fx:id: idTextField

Controller class: application.RegisterController

□ Use fx:root construct

Assigned fx:id

fx:id	Component
firstnameTextField	TextField
lastnameTextField	TextField
usernameTextField	TextField
setPasswordField	TextField