LAPORAN KEGIATAN PRAKTIKUM MATA KULIAH

PEMOGRAMAN LANJUT

Muhammad Asfar Aslam  
202210370311438  
Informatika

Universitas Muhammadiyah Malang

2023/2024

**Laporan Aplikasi JavaFX: Table\_JavaFx**

1. Deskripsi Aplikasi

Aplikasi ini merupakan contoh sederhana dari penggunaan JavaFX untuk membuat tabel (TableView) yang menampilkan daftar mahasiswa. Pengguna dapat menambahkan entri baru melalui input teks dan tombol "Add". Aplikasi ini juga mendemonstrasikan penggunaan observable list untuk mengelola data tabel.

1. Komponen GUI

* TableView: Menampilkan data mahasiswa dengan kolom Nama, NIM, dan Email.
* Label: "Daftar Mahasiswa" sebagai judul tabel.
* TextFields: Untuk input Nama, NIM, dan Email mahasiswa baru.
* Button "Add": Menambahkan data mahasiswa baru ke dalam tabel.

1. Struktur Kode

Kelas Table\_JavaFx: Merupakan kelas utama yang mengimplementasikan antarmuka Application dari JavaFX.

Metode start(Stage stage): Membangun tata letak GUI dan menampilkan tabel dengan data awal.

Kelas Mahasiswa: Kelas nested (bersarang) yang mewakili entitas mahasiswa dengan properti nama, NIM, dan email.

1. Fungsionalitas Utama

Menampilkan Tabel: Menampilkan daftar mahasiswa dengan kolom-kolom yang sesuai.

Menambah Data Mahasiswa: Pengguna dapat menambahkan mahasiswa baru melalui input teks dan tombol "Add".

Penggunaan Observable List: Menggunakan ObservableList untuk secara dinamis memantau dan memperbarui tampilan tabel saat data berubah.

1. Catatan Pengembangan Selanjutnya

Validasi Input: Tambahkan validasi input untuk memastikan bahwa data yang dimasukkan sesuai dengan format yang diharapkan.

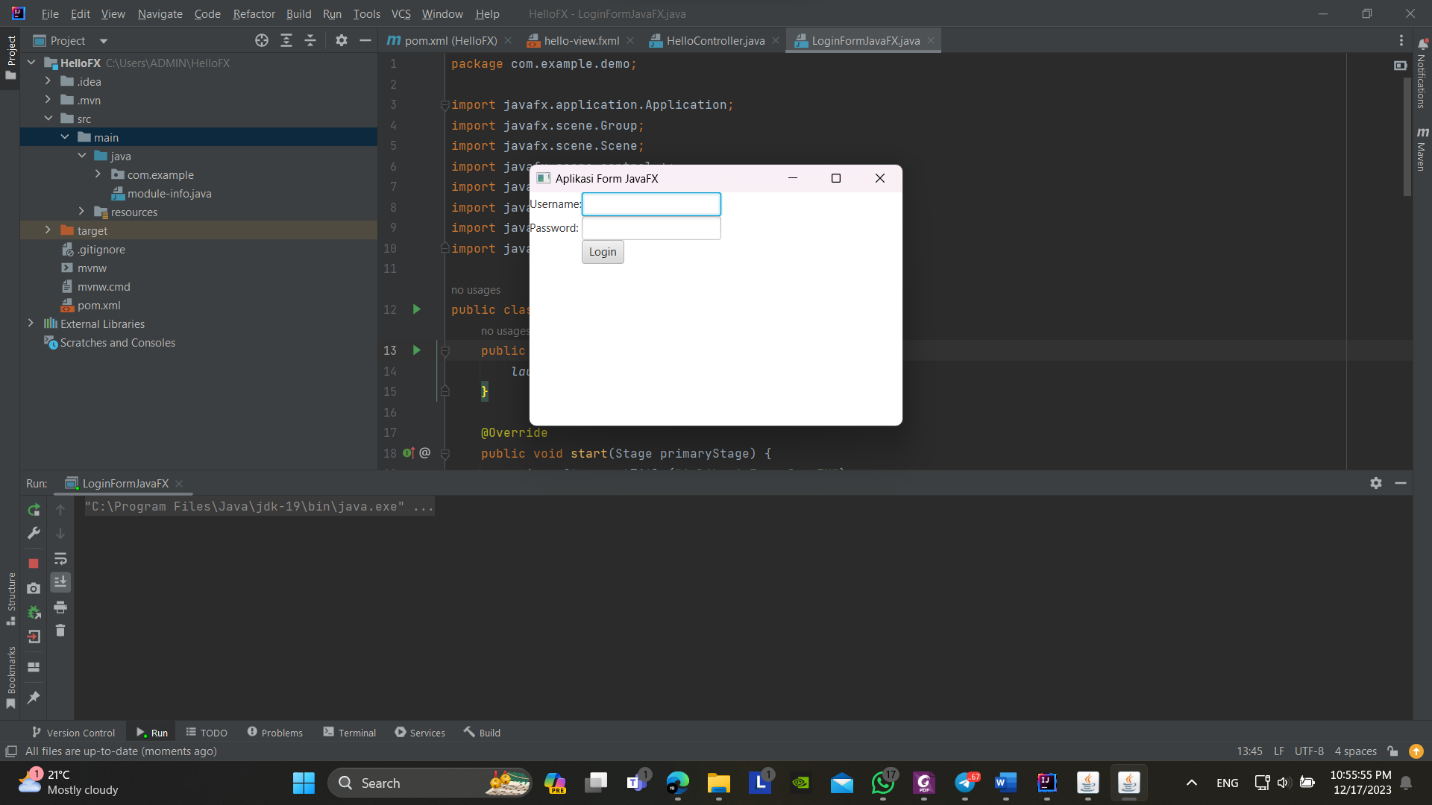
Fungsionalitas Edit dan Hapus: Implementasikan kemampuan untuk mengedit dan menghapus entri mahasiswa dari tabel.

Penanganan Error: Perbaiki atau tambahkan penanganan kesalahan (error handling) untuk mengatasi potensi masalah pada aplikasi.

1. Kesimpulan

Aplikasi JavaFX ini menyajikan cara sederhana untuk membuat dan mengelola tabel data dengan antarmuka pengguna yang bersahaja. Dengan beberapa pengembangan lebih lanjut, aplikasi ini dapat menjadi dasar untuk pengembangan sistem informasi lebih lanjut yang melibatkan manipulasi data mahasiswa.

Hasil tampilan :



Code Table\_JavaFx :

package com.example.latihan\_modul\_6;  
  
import javafx.application.Application;  
import javafx.beans.property.SimpleStringProperty;  
import javafx.collections.FXCollections;  
import javafx.collections.ObservableList;  
import javafx.event.ActionEvent;  
import javafx.event.EventHandler;  
import javafx.geometry.Insets;  
import javafx.scene.Group;  
import javafx.scene.Scene;  
import javafx.scene.control.\*;  
import javafx.scene.control.cell.PropertyValueFactory;  
import javafx.scene.layout.HBox;  
import javafx.scene.layout.VBox;  
import javafx.scene.text.Font;  
import javafx.stage.Stage;  
  
public class Table\_JavaFx extends Application {  
 private TableView table = new TableView<>();  
  
 public static void main(String[] args) {  
 *launch*(args);  
 }  
 @Override  
 public void start(Stage stage) {  
 Scene scene = new Scene(new Group());  
  
 stage.setTitle("Test TableView");  
 stage.setWidth(350);  
 stage.setHeight(550);  
  
 final Label label = new Label("Daftar Mahasiswa");  
 label.setFont(new Font("Arial",30));  
 table.setEditable(true);  
  
 TableColumn nameCol = new TableColumn("Nama");  
 TableColumn nimCol = new TableColumn("NIM");  
 TableColumn emailCol = new TableColumn("Email");  
  
 table.getColumns().addAll(nameCol,nimCol,emailCol);  
  
 final VBox vbox = new VBox();  
 vbox.setSpacing(8);  
 vbox.setPadding(new Insets(20,10,10,10));  
 vbox.getChildren().addAll(label,table);  
  
 ((Group)scene.getRoot()).getChildren().addAll(vbox);  
  
 stage.setScene(scene);  
 stage.show();  
  
 final ObservableList<Mahasiwa>data = FXCollections.*observableArrayList*(  
 new Mahasiwa("Asfar","202210370311438","asfaraslam8@gmail.com"),  
 new Mahasiwa("Aslam","202210370311488","aslam8@gmail.com")  
 );  
  
 nameCol.setCellValueFactory(new PropertyValueFactory<Mahasiwa,String>("name"));  
 nimCol.setCellValueFactory(new PropertyValueFactory<Mahasiwa,String>("nim"));  
 emailCol.setCellValueFactory(new PropertyValueFactory<Mahasiwa,String>("email"));  
  
 table.setItems(data);  
  
 final TextField addName = new TextField();  
 addName.setMaxWidth(nameCol.getPrefWidth());  
 addName.setPromptText("Nama Mahasiswa");  
  
 final TextField addNim = new TextField();  
 addNim.setMaxWidth(nimCol.getPrefWidth());  
 addNim.setPromptText("NIM");  
  
 final TextField addEmail = new TextField();  
 addEmail.setMaxWidth(emailCol.getPrefWidth());  
 addEmail.setPromptText("Email");  
  
 final Button addButton = new Button("Add");  
 addButton.setOnAction(new EventHandler<ActionEvent>() {  
 @Override  
 public void handle(ActionEvent actionEvent) {  
 data.add(new Mahasiwa(  
 addName.getText(),  
 addNim.getText(),  
 addEmail.getText()  
 ));  
 addName.clear();  
 addNim.clear();  
 addEmail.clear();  
 }  
 });  
 final HBox hboxInput = new HBox();  
 hboxInput.getChildren().addAll(addName,addNim,addEmail,addButton);  
 hboxInput.setSpacing(10);  
  
  
 vbox.getChildren().addAll(hboxInput);  
 }  
  
 public static class Mahasiwa{  
 private final SimpleStringProperty name;  
 private final SimpleStringProperty nim;  
 private final SimpleStringProperty email;  
  
 public Mahasiwa(String name, String nim, String email) {  
 this.name = new SimpleStringProperty (name);  
 this.nim = new SimpleStringProperty(nim);  
 this.email = new SimpleStringProperty(email);  
 }  
  
 public String getName() {  
 return name.get();  
 }  
  
 public void setName(String fName) {  
 name.set(fName);  
 }  
  
 public String getNim() {  
 return nim.get();  
 }  
  
 public void setNim(String fNim) {  
 nim.set(fNim);  
 }  
  
 public String getEmail() {  
 return email.get();  
 }  
  
 public void setEmail(String fEmail) {  
 email.set(fEmail);  
 }  
 }  
}

**Laporan Aplikasi JavaFX: LoginFormJavaFX**

1. Deskripsi Aplikasi

Aplikasi ini adalah formulir login sederhana yang dibangun menggunakan JavaFX. Pengguna diminta untuk memasukkan nama pengguna (Username) dan kata sandi (Password) mereka. Setelah menekan tombol "Login," aplikasi memberikan pesan hasil yang menyatakan apakah login berhasil atau gagal.

2. Komponen GUI

* Label "Username": Untuk menunjukkan area input nama pengguna.
* TextField "textFieldUsername": Untuk memasukkan nama pengguna.
* Label "Password": Untuk menunjukkan area input kata sandi.
* PasswordField "passwordField": Untuk memasukkan kata sandi.
* Button "Login": Untuk memulai proses login.
* Text "message": Untuk menampilkan pesan hasil login.

3. Struktur Kode

Kelas LoginFormJavaFX: Merupakan kelas utama yang mengimplementasikan antarmuka Application dari JavaFX.

Metode start(Stage primaryStage): Membangun tata letak GUI formulir login dan menangani logika login.

Aksi Tombol Login (loginButton.setOnAction(e -> {...})): Memeriksa apakah nama pengguna dan kata sandi yang dimasukkan sesuai dengan nilai yang diharapkan.

4. Fungsionalitas Utama

Validasi Login: Memeriksa apakah nama pengguna dan kata sandi yang dimasukkan sesuai dengan nilai yang diharapkan ("user" dan "pass").

Memberikan Pesan Login: Memberikan pesan berwarna hijau jika login berhasil dan pesan berwarna merah jika login gagal.

5. Catatan Pengembangan Selanjutnya

-Penyimpanan Data Pengguna: Tambahkan mekanisme penyimpanan dan pemeriksaan data pengguna dari sumber eksternal (misalnya, database).

-Validasi Input: Perbarui aplikasi dengan validasi input tambahan untuk meningkatkan keamanan.

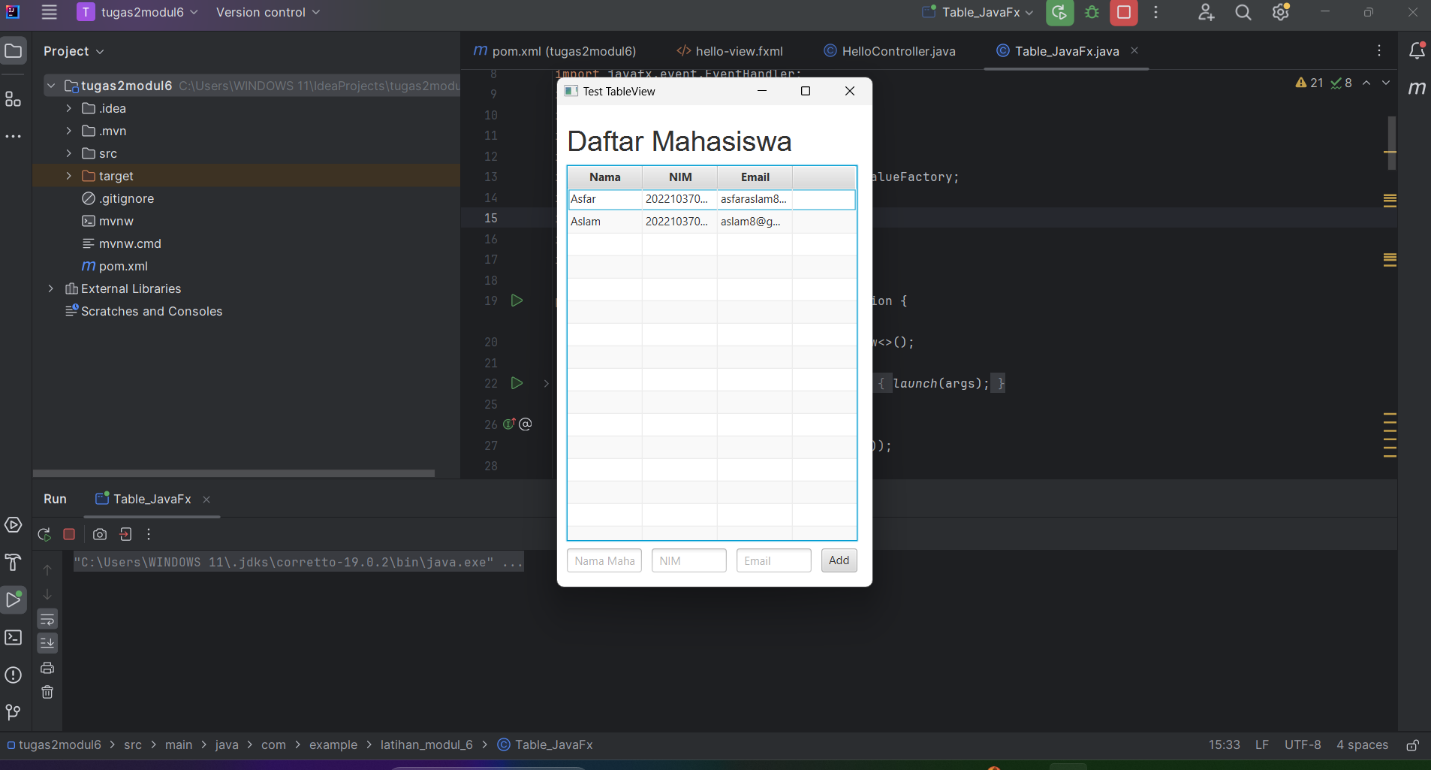
-Pengaturan Keamanan: Implementasikan mekanisme keamanan yang lebih kuat, seperti penyandian kata sandi.

-Perbaikan UI: Sesuaikan tata letak dan gaya GUI untuk peningkatan estetika dan pengalaman pengguna.

6. Kesimpulan

Aplikasi LoginFormJavaFX memberikan dasar untuk formulir login sederhana dengan menggunakan JavaFX. Pengembangan lebih lanjut dapat melibatkan peningkatan keamanan, manajemen pengguna, dan peningkatan antarmuka pengguna untuk menciptakan pengalaman login yang lebih baik.

Hasil Tampilan Seperti ini :



Code LoginFormJavaFX :

package com.example.demo;  
  
import javafx.application.Application;  
import javafx.scene.Group;  
import javafx.scene.Scene;  
import javafx.scene.control.\*;  
import javafx.scene.layout.GridPane;  
import javafx.scene.paint.Color;  
import javafx.scene.text.Text;  
import javafx.stage.Stage;  
  
public class LoginFormJavaFX extends Application {  
 public static void main(String[] args) {  
 *launch*(args);  
 }  
  
 @Override  
 public void start(Stage primaryStage) {  
 primaryStage.setTitle("Aplikasi Form JavaFX");  
  
 Scene scene = new Scene(new Group(), 400, 250);  
  
 Label labelUsername = new Label("Username:");  
 TextField textFieldUsername = new TextField();  
  
 Label labelPassword = new Label("Password:");  
 PasswordField passwordField = new PasswordField();  
  
 Button loginButton = new Button("Login");  
 Text message = new Text();  
  
 GridPane grid = new GridPane();  
 grid.add(labelUsername, 0, 0);  
 grid.add(textFieldUsername, 1, 0);  
 grid.add(labelPassword, 0, 1);  
 grid.add(passwordField, 1, 1);  
 grid.add(loginButton, 1, 2);  
 grid.add(message, 1, 3);  
  
 loginButton.setOnAction(e -> {  
 String username = textFieldUsername.getText();  
 String password = passwordField.getText();  
  
 if (username.equals("user") && password.equals("pass")) {  
 message.setFill(Color.*GREEN*);  
 message.setText("Login Berhasil!");  
 } else {  
 message.setFill(Color.*RED*);  
 message.setText("Login Gagal. Coba lagi!");  
 }  
 });  
  
 Group root = (Group) scene.getRoot();  
 root.getChildren().add(grid);  
  
 primaryStage.setScene(scene);  
 primaryStage.show();  
}  
}