Para impedir que os usuários selecionem uma data posterior à data do computador local em um DatePicker no SceneBuilder, você precisará definir uma validação no código Java. Aqui está um exemplo de como você pode fazer isso:

java

Copy code

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.DatePicker;

import javafx.scene.layout.StackPane;

import javafx.stage.Stage;

import java.time.LocalDate;

public class Main extends Application {

@Override

public void start(Stage primaryStage) {

DatePicker datePicker = new DatePicker();

// Define a data máxima como a data do computador local

datePicker.setMaxDate(LocalDate.now());

StackPane root = new StackPane();

root.getChildren().add(datePicker);

primaryStage.setScene(new Scene(root, 300, 250));

primaryStage.show();

}

public static void main(String[] args) {

launch(args);

}

}

Neste exemplo, datePicker.setMaxDate(LocalDate.now()); define a data máxima que pode ser selecionada como a data atual do computador local. Isso garante que os usuários não possam selecionar datas no futuro.

No SceneBuilder, você pode configurar os componentes visuais, como o DatePicker, mas a lógica de programação, como essa validação, precisa ser implementada no código Java. Depois de criar o layout no SceneBuilder, você pode carregar o arquivo FXML em seu aplicativo Java e definir os manipuladores de eventos e outras lógicas de programação necessárias.

Parte superior do formulário

Parte inferior do formulário