

# Aula 10

Introdução ao JavaFX



# O que é JavaFX?

- Biblioteca para interfaces gráficas modernas em Java.
- Substitui Swing, com suporte a animações, CSS e mais.
- Estrutura básica:
  - Classe `Application`
  - Método `start(Stage)`
  - `Scene` e `Stage`



# Exemplo Básico

```
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Label;
import javafx.stage.Stage;

public class BasicJavaFX extends Application {
    @Override
    public void start(Stage primaryStage) {
        Label label = new Label("Bem-vindo ao JavaFX.");
        Scene scene = new Scene(label, 200, 100);
        primaryStage.setTitle("Exemplo Básico");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```



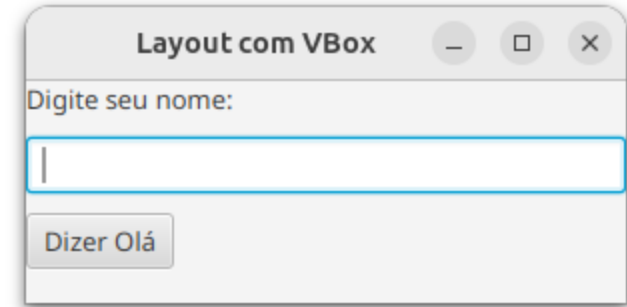
# Compilação e Execução

- Requer instalação da biblioteca javafx:
  - `sudo apt install openjfx`
- A compilação exige inclusão explícita da biblioteca:
  - `javac --module-path /usr/share/openjfx/lib --add-modules javafx.controls BasicJavaFX.java`
- Bem como a execução:
  - `java --module-path /usr/share/openjfx/lib`
  - `--add-modules javafx.controls BasicJavaFX`

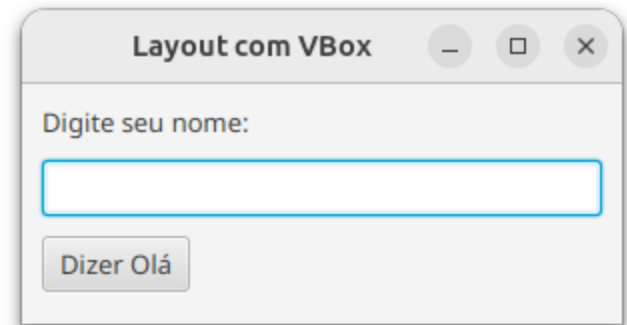


# Layouts em JavaFX

- Tipos:
  - VBox: Vertical
  - HBox: Horizontal
- Propriedades:
  - Espaçamento: VBox (10)
  - Padding: `setPadding(new Insets(15))`
- Exemplo (LayoutJavaFX):
  - Label, TextField, Button alinhados verticalmente.



Sem padding



Com padding



# Eventos

- Eventos:
  - `setOnAction(event -> {...})`
- Exemplo (EventJavaFX)

```
botao.setOnAction(event -> {  
    String nome = nomeField.getText();  
    resultado.setText("Olá, " + (nome.isEmpty() ? "Mundo" : nome) + "!");  
});
```



# Estilização com CSS

- Inline:
  - `setStyle("-fx-background-color: #4CAF50;")`
- CSS Externo:
  - `scene.getStylesheets().add("styles.css")`
  - #id para IDs, .class para classes



# Outros controles de Entrada/Saída

- Lista de Controles (JavaFXInputsDemo):
  - TextField, PasswordField, TextArea
  - CheckBox, RadioButton (com ToggleGroup)
  - ComboBox, ChoiceBox, Slider
  - Spinner, DatePicker, ColorPicker
- Saída:
  - Exibir valores com ScrollPane.





# Listas Interativas

- ListView + ObservableList:
  - Dados dinâmicos: `items.add()`, `items.remove()`
- Eventos:
  - Adicionar, atualizar, remover itens.
  - Sincronizar com TextField via `selectedItemProperty`.
- Exemplo (JavaFXListDemo)

