

## Estructura básica de una aplicación JavaFX

Cada aplicación JavaFX debe extender la clase `Application` y sobrescribir el método `start()`.

### Código básico:

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

public class JavaFX2 extends Application {

    @Override
    public void start(Stage primaryStage) {
        // Crear un componente
        Label etiqueta = new Label("¡Hola, JavaFX!");

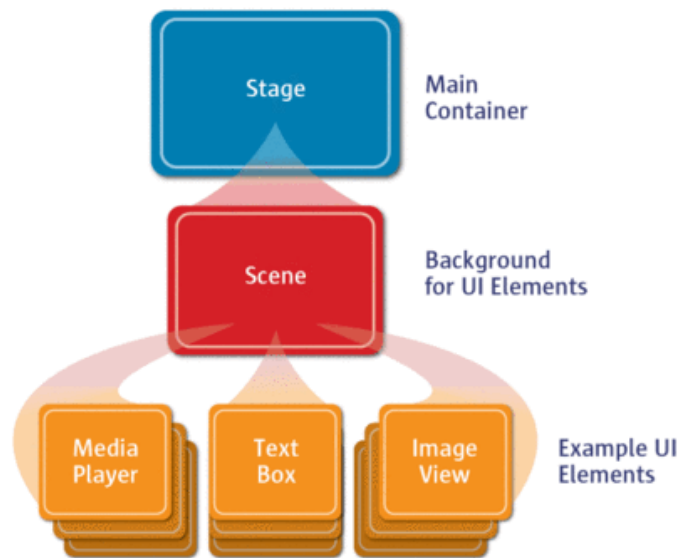
        // Crear el contenedor principal
        StackPane root = new StackPane();
        root.getChildren().add(etiqueta);

        // Crear la escena y agregarla al escenario
        Scene scene = new Scene(root, 400, 200);
        primaryStage.setScene(scene);

        // Configurar el escenario principal
        primaryStage.setTitle("Mi Primera Aplicación JavaFX");
        primaryStage.show();
    }

    public static void main(String[] args) {
        launch(args); // Lanza la aplicación
    }
}
```

# Componentes Básicos



JavaFX tiene muchos componentes predefinidos como botones, etiquetas, campos de texto, etc.

- **Etiqueta (Label):** Muestra texto.

```
Label etiqueta = new Label("Texto estático");
```

- **Botón (Button):** Un botón clicable.

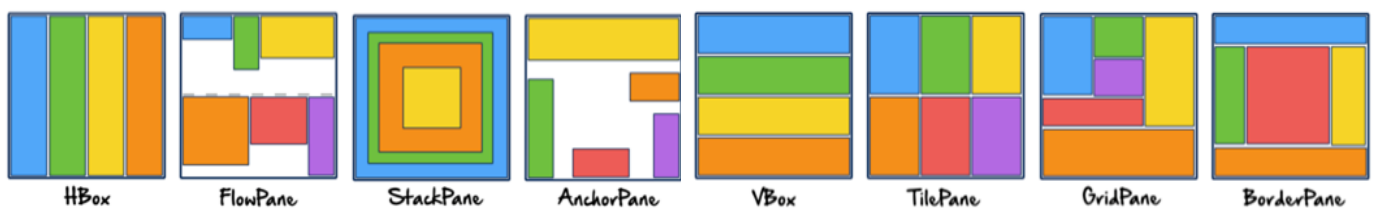
```
Button boton = new Button("Clic aquí");
```

- **Campo de texto (TextField):** Para entrada de texto.

```
TextField campoTexto = new TextField();
```

## Disposición (Layouts)

JavaFX organiza los componentes en contenedores llamados **layouts**.



1. **StackPane:** Apila componentes uno encima de otro.

```
StackPane stackPane = new StackPane();
stackPane.getChildren().add(new Label("Apilado"));
```

2. **HBox**: Alinea componentes horizontalmente.

```
HBox hbox = new HBox(10); // Espaciado de 10 píxeles
hbox.getChildren().addAll(new Label("Izquierda"), new Label("Derecha"));
```

3. **VBox**: Alinea componentes verticalmente.

```
VBox vbox = new VBox(15); // Espaciado de 15 píxeles
vbox.getChildren().addAll(new Label("Arriba"), new Label("Abajo"));
```

4. **BorderPane**: Divide el espacio en cinco regiones (top, bottom, left, right, center).

```
BorderPane borderPane = new BorderPane();
borderPane.setTop(new Label("Encabezado"));
borderPane.setCenter(new Label("Centro"));
```

## Eventos

JavaFX usa **lambdas** o clases para manejar eventos como clics en botones.

### Ejemplo: Manejar el clic de un botón

```
Button boton = new Button("Presiona aquí");
boton.setOnAction(event -> {
    System.out.println("¡Botón presionado!");
});
```

## Uso de CSS en JavaFX

Podes personalizar tus componentes con CSS.

1. **Archivo CSS (styles.css)**:

```
.mi-boton {
    -fx-background-color: lightblue;
    -fx-text-fill: darkblue;
}
```

2. **Aplicar estilos**:

```
boton.getStyleClass().add("mi-boton");  
escena.getStylesheets().add("styles.css");
```