

Programación de Interfaces Gráficas en Java

Objetivo: Programar aplicaciones básicas con interfaces gráficas usando JavaFX

ELO329: Diseño y Programación Orientados a Objetos

Departamento de Electrónica

Universidad Técnica Federico Santa María

Este material ha sido preparado desde varias fuentes, entre ellas el material del Dr. Paul Fodor, Stony Brook University

AWT, Swing y JavaFx: Su origen

- ❑ En sus orígenes Java introdujo la **AWT** (Abstract Window Toolkit). Ésta “creaba” los objetos delegando su creación y comportamiento a herramientas nativas de la plataforma donde corre la Máquina Virtual Java.
- ❑ Este esquema condujo a problemas por diferencias en distintas plataformas y S.O.
- ❑ La solución fue implementar todos los objetos de la GUI solo con elementos muy básicos y comunes en todas las plataformas. **Así se creó Swing**.
- ❑ Luego surge la guerra de plataformas de software para desarrollar “Rich Internet applications” (RIAs): Adobe Flash y Microsoft Silverlight. En esa época Oracle desarrolló JavaFX.
- ❑ Adobe decidió discontinuar Flash el 2017 y apoyar HTML5. Soporte Flash fue removido de los navegadores en Dic. 2020.
- ❑ **JavaFX** permite desarrollar aplicaciones de escritorio y RIA (aplicaciones que corren en navegador).

JavaFX

- ❑ En este curso usted usará JavaFX
- ❑ JavaFX es una biblioteca usada para construir aplicaciones gráficas en Java
- ❑ JavaFX fue desarrollado por la empresa “See Beyond”, esta empresa fue comprada por Sun Microsystems el año 2005.
- ❑ El año 2009, Oracle compra a Sun Microsystems y continúa el desarrollo de JavaFX.
- ❑ El año 2018 Oracle declara que:
 - Seguirá soportando AWT y Swing hasta al menos 2026
 - Está trabajando con terceras partes para mantener JavaFX como un módulo separado y de código abierto.

Instalación de JavaFX

- ❑ Desde la versión 11 de Java, JavaFX viene como un módulo separado del JDK.
- ❑ Puede acceder a las instrucciones de instalación para los IDEs más populares y para los Sistemas Operativos Windows/Linux en <https://openjfx.io/openjfx-docs/>
- ❑ Revisaremos el proceso de instalación para el IDE IntelliJ IDEA

Instalación de JavaFX

- ❑ Descargue la biblioteca JavaFX desde la página <https://gluonhq.com/products/javafx/> dependiendo de su sistema operativo. Recomiendo bajar la versión Long Term Support (LTS).

Downloads

JavaFX version: 17.0.3 [LTS] Operating System: Windows Architecture: x64 Type: [any]

☐ Include older versions

Looking for older JavaFX 11 or 17 versions? Check 'Include older versions' above.

OS	Version	Architecture	Type	Download
Windows	17.0.3	x64	SDK	More info
Windows	17.0.3	x64	jmods	More info
Windows	17.0.3	x64	Monocle SDK	More info
Javadoc	17.0.3		Javadoc	More info

- ❑ En este punto ya puede compilar y ejecutar sus programas por consola agregando la siguiente instrucción a los comandos javac y java:

--module-path <path/to/javafx-sdk-17/lib> --add-modules javafx.controls

Ej: \$ javac --module-path /home/agustin/tools/javafx-sdk-17/lib --add-modules javafx.controls CloseableStage.java

Compilar por consola con JavaFX

- ❑ Desde la versión 11 de Java, JavaFX es un módulo externo a Java.
- ❑ Para compilar y ejecutar programas con JavaFX es necesario indicar a los comandos `javac` y `java` dónde se encuentran los módulos y controladores que utilizará. Siendo `<path/to/javafx-sdk>` la ubicación de descarga:

- Compilar:

```
javac --module-path <path/to/javafx-sdk-17/lib> --add-modules javafx.controls <archivo>.java
```

- Ejecutar:

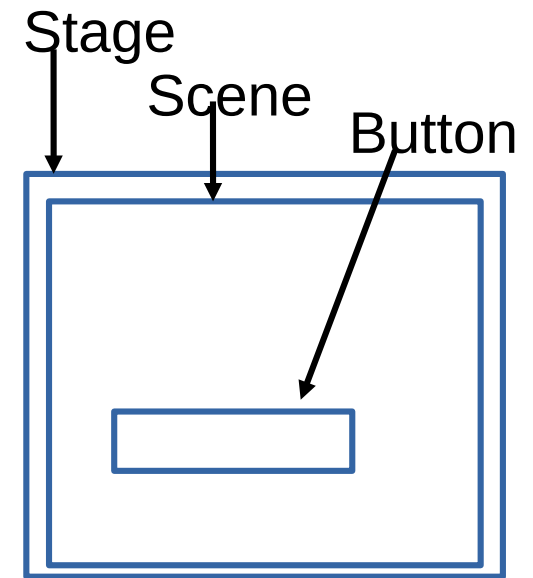
```
java --module-path <path/to/javafx-sdk-17/lib> --add-modules javafx.controls <archivo>
```

JavaFX: Estructura y programación

Estructura Básica de JavaFX

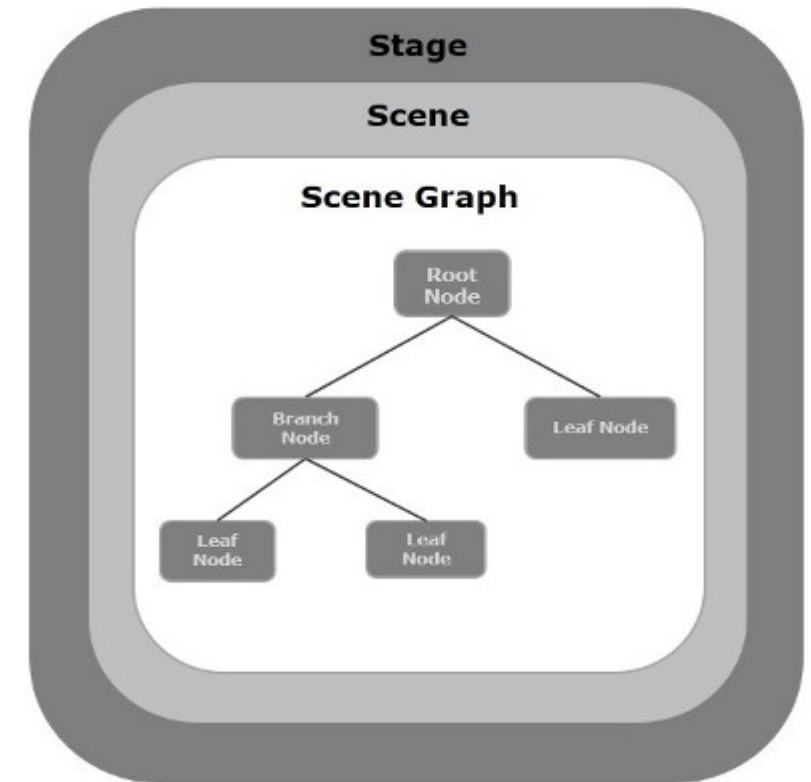
Elementos de una ventana

- ❑ `javafx.application.Application` es el punto de entrada para una aplicación JavaFX
- ❑ `Javafx.stage.Stage` (stage = escenario) es la clase contenedora de nivel superior
 - El escenario primario es construido por la plataforma
- ❑ `javafx.scene.Scene` (escena) es la clase contenedora de todo el contenido en una escena gráfica.
- ❑ `javafx.scene.Node` es la clase base para todos los nodos gráficos de una escena



Estructura Básica de JavaFX

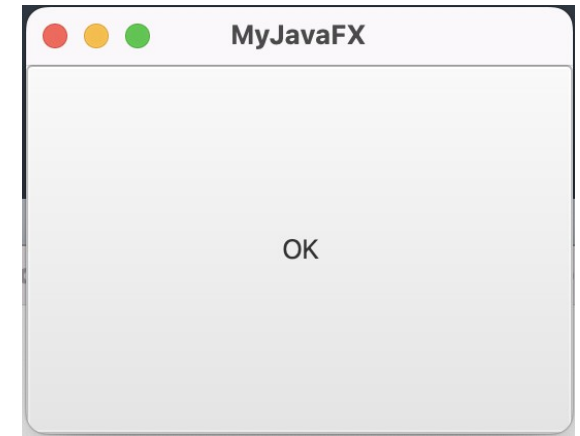
- ❑ Internamente una escena contiene un grafo de Nodos con una estructura similar a la de un estructura de árbol (como el de la figura).
- ❑ Esta estructura es la representación interna del contenido de la escena.
- ❑ Un Nodo es un objeto visual del grafo de una escena.



Ejemplos: MyFirstJavaFX

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

public class MyFirstJavaFX extends Application {
    @Override // Override the start method in the Application class
    public void start(Stage primaryStage) {
        // Create a button and place it in the scene
        Button btOK = new Button("OK");
        Scene scene = new Scene(btOK, 200, 250);
        primaryStage.setTitle("MyJavaFX"); // Set the stage title
        primaryStage.setScene(scene); // Place the scene in the stage
        primaryStage.show(); // Display the stage
    }
    /**
     * The main method is only needed for the IDE with limited JavaFX support.
     * Not needed for running from the command line.
     */
    public static void main(String[] args) {
        launch(args);
    }
}
```



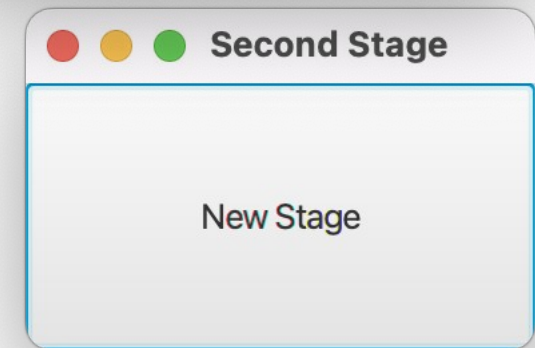
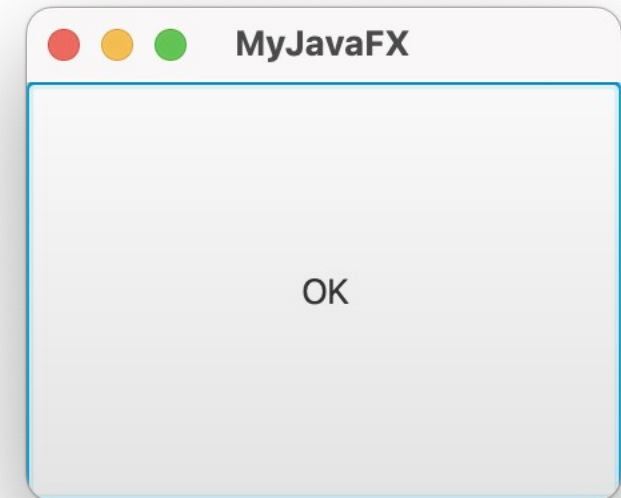
Ejemplos: MultipleStageDemo

// Multiple stages can be added beside the primaryStage

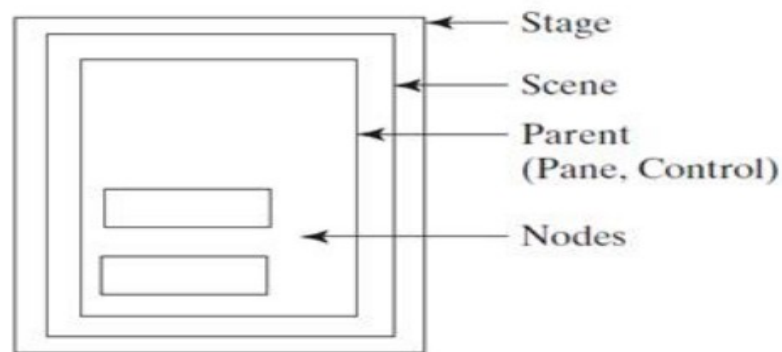
```
public class MultipleStageDemo extends Application {

    @Override // Override the start method in the Application class
    public void start(Stage primaryStage) {
        // Create a scene and place a button in the scene
        Scene scene = new Scene(new Button("OK"), 200, 250);
        primaryStage.setTitle("MyJavaFX"); // Set the stage title
        primaryStage.setScene(scene); // Place the scene in the stage
        primaryStage.show(); // Display the stage
        Stage stage = new Stage(); // Create a new stage
        stage.setTitle("Second Stage"); // Set the stage title
        // Set a scene with a button in the stage
        stage.setScene(new Scene(new Button("New Stage"), 100, 100));
        stage.show(); // Display the stage
    }

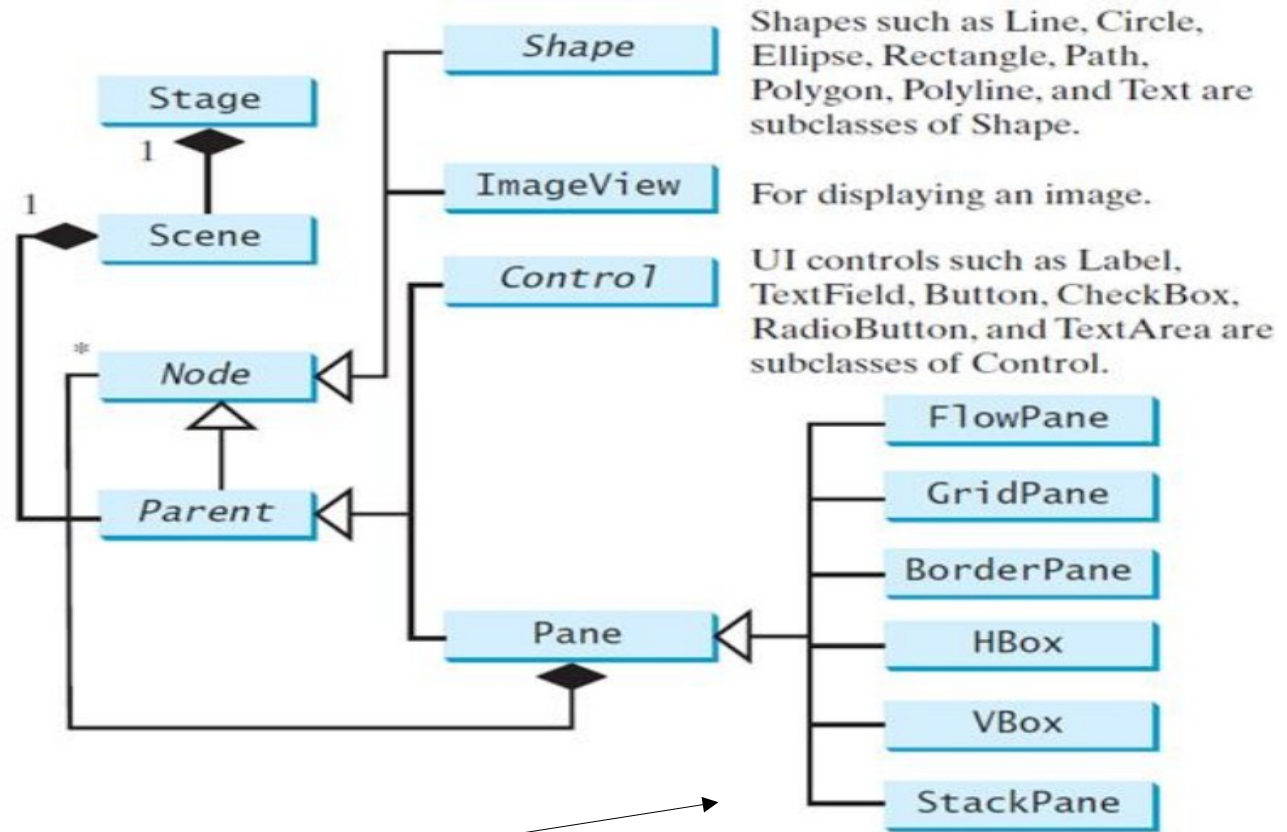
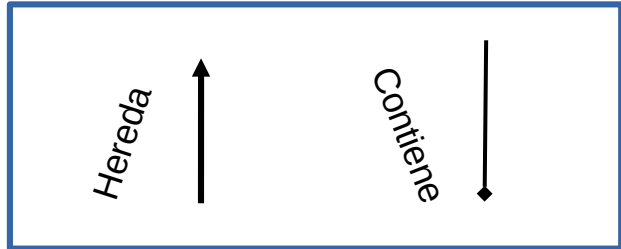
    public static void main(String[] args) {
        launch(args);
    }
}
```



Panes (cristales), UI Controls, y Shapes



Notación



“cristales” con organización específica del espacio.

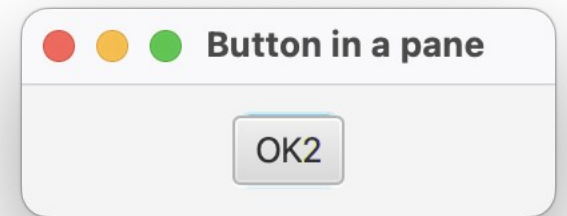
Layout Panes

- ❑ JavaFX provee varios tipos de “cristales” para organizar el despliegue de nodos (objetos gráficos) de un contenedor; por ejemplo, una escena.

Clase	Descripción
Pane	Clase Base para otros Layout Panes. Contiene el método getChildren() para retornar la lista de nodos en el pane (cristal). Esta clase puede ser usada directamente cuando se requiere ubicar nodos en posiciones absolutas.
StackPane	Pone los nodos (objetos gráficos) uno sobre otro en el centro del pane.
FlowPane	Pone los nodos de izquierda a derecha y de arriba a abajo.
GridPane	Pone los nodos en celdas de una grilla de dos dimensiones
BorderPane	Ubica los nodos en regiones: Top, Right, Bottom, Left, y Centro.
HBox	Ubica los nodos en una fila única horizontal.
Vbox	Ubica los nodos en una única columna vertical.

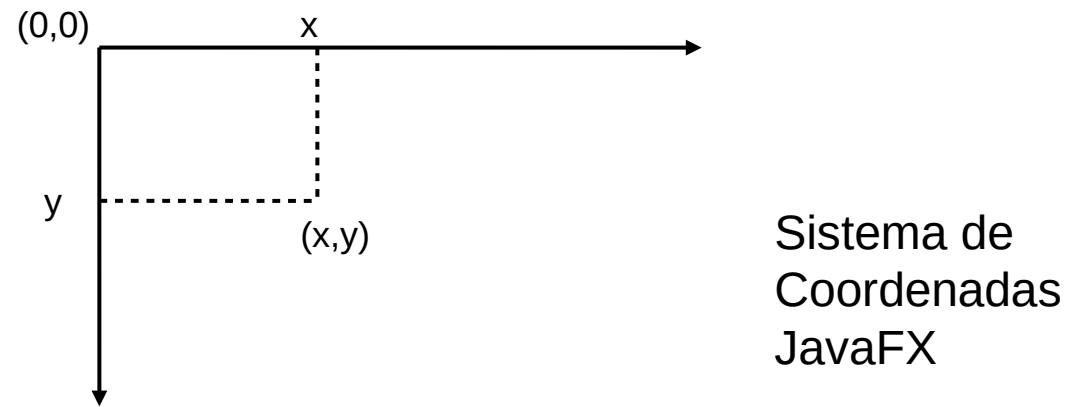
Ejemplo: ButtonInPane

```
public class ButtonInPane extends Application {  
  
    @Override // Override the start method in the Application class  
    public void start(Stage primaryStage) {  
        // Create a scene and place a button in the scene  
        StackPane pane = new StackPane();  
        pane.getChildren().add(new Button("OK"));  
        Scene scene = new Scene(pane, 200, 50);  
        primaryStage.setTitle("Button in a pane"); // Set the stage title  
        primaryStage.setScene(scene); // Place the scene in the stage  
        primaryStage.show(); // Display the stage  
    }  
  
    public static void main(String[] args) {  
        launch(args);  
    }  
}
```



Despliegue de alguna Forma (Shape)

- Notar la convención para los sistemas de coordenadas en programación, $(0,0)$ está en la esquina superior izquierda.

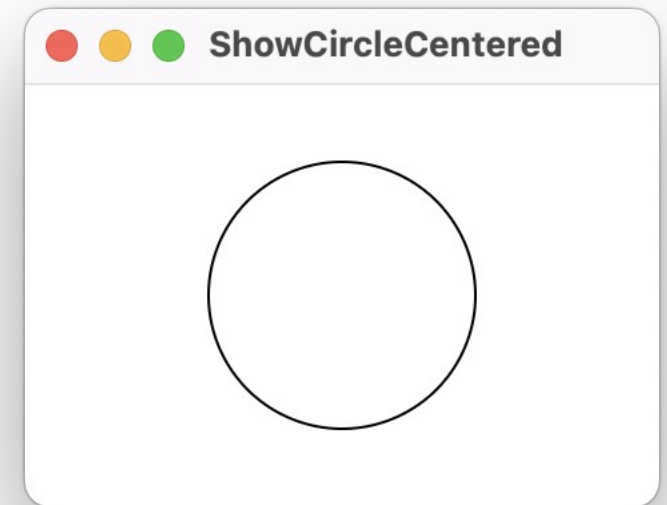


- Veamos cómo dibujar un círculo (formas de este tipo se ocuparán en Tarea 2)

Ejemplo: ShowCircleCentered

```
public class ShowCircleCentered extends Application {
    @Override // Override the start method in the Application class
    public void start(Stage primaryStage) {
        // Create a pane to hold the circle
        Pane pane = new Pane();
        // Create a circle and set its properties
        Circle circle = new Circle();
        circle.centerXProperty().bind(pane.widthProperty().divide(2));
        circle.centerYProperty().bind(pane.heightProperty().divide(2));
        circle.setRadius(50);
        circle.setStroke(Color.BLACK);
        circle.setFill(Color.WHITE);
        pane.getChildren().add(circle); // Add circle to the pane
        // Create a scene and place it in the stage
        Scene scene = new Scene(pane, 200, 200);
        primaryStage.setTitle("ShowCircleCentered"); // Set the stage title
        primaryStage.setScene(scene); // Place the scene in the stage
        primaryStage.show(); // Display the stage
    }

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



Ejemplo: Clases Image e ImageView

```
public class ShowImage extends Application {
    @Override
    public void start(Stage primaryStage) {
// Create a pane to hold the image views
        Pane pane = new HBox(10);
        pane.setPadding(new Insets(5, 5, 5, 5));
        Image image = new Image("header-usm.png");
        pane.getChildren().add(new ImageView(image));
        ImageView imageView2 = new ImageView(image);
        imageView2.setFitHeight(100);
        imageView2.setFitWidth(100);
        imageView2.setRotate(90);
        pane.getChildren().add(imageView2);
        Scene scene = new Scene(pane);
        primaryStage.setTitle("ShowImage");
        primaryStage.setScene(scene);
        primaryStage.show();
    }

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

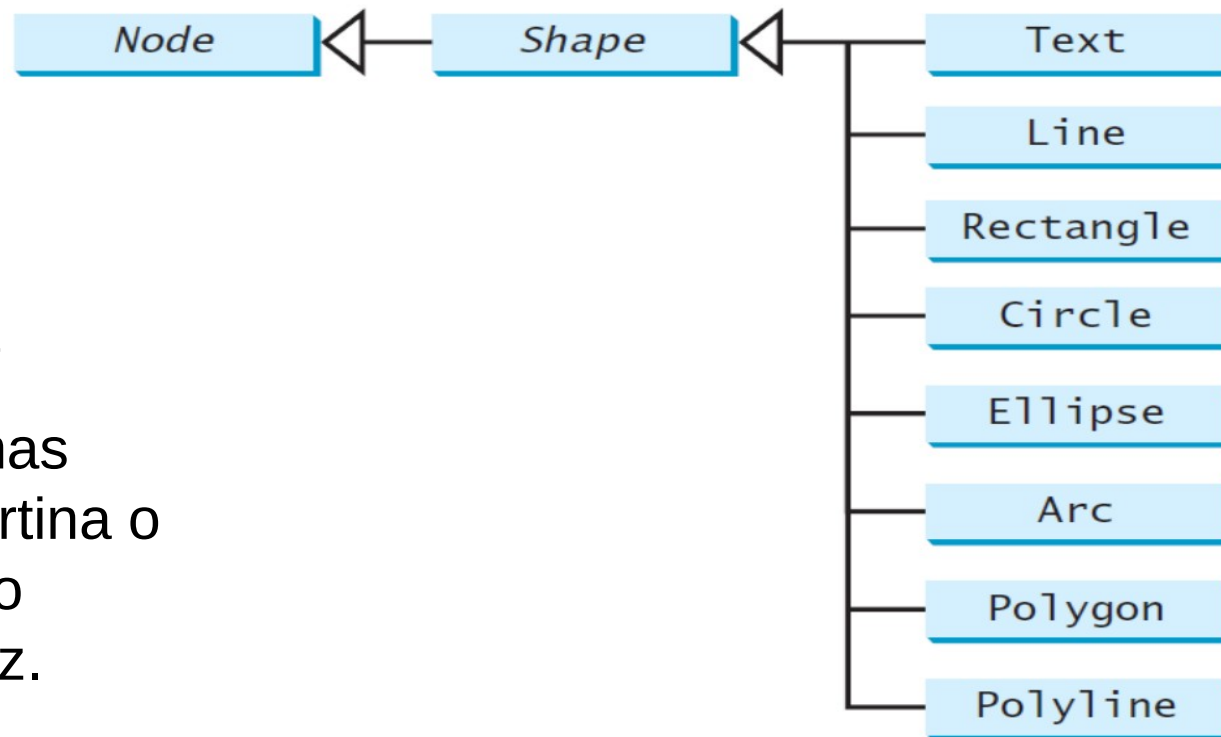


Formas (Shapes)

- ❑ JavaFX provee varias clases para representar textos, líneas, círculos, rectángulos, elipses, arcos, polígonos, polígonos abiertos (polylines).

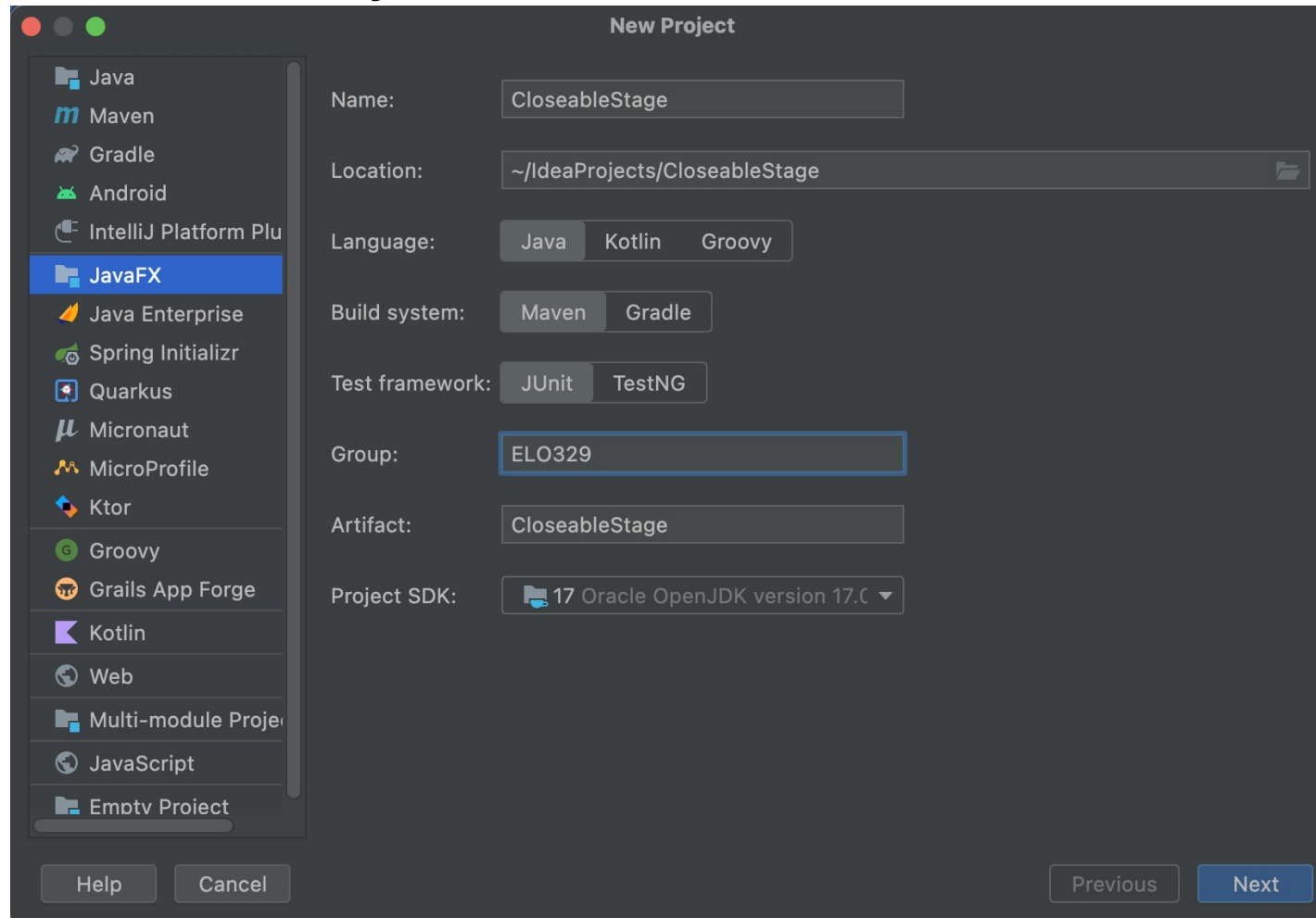
- Ver ejemplos:
 - ShowText.java
 - ShowLines.java
 - ShowRectangle.java

Usando este tipo de formas podremos dibujar una cortina o lámpara simple que luego animaremos en la interfaz.



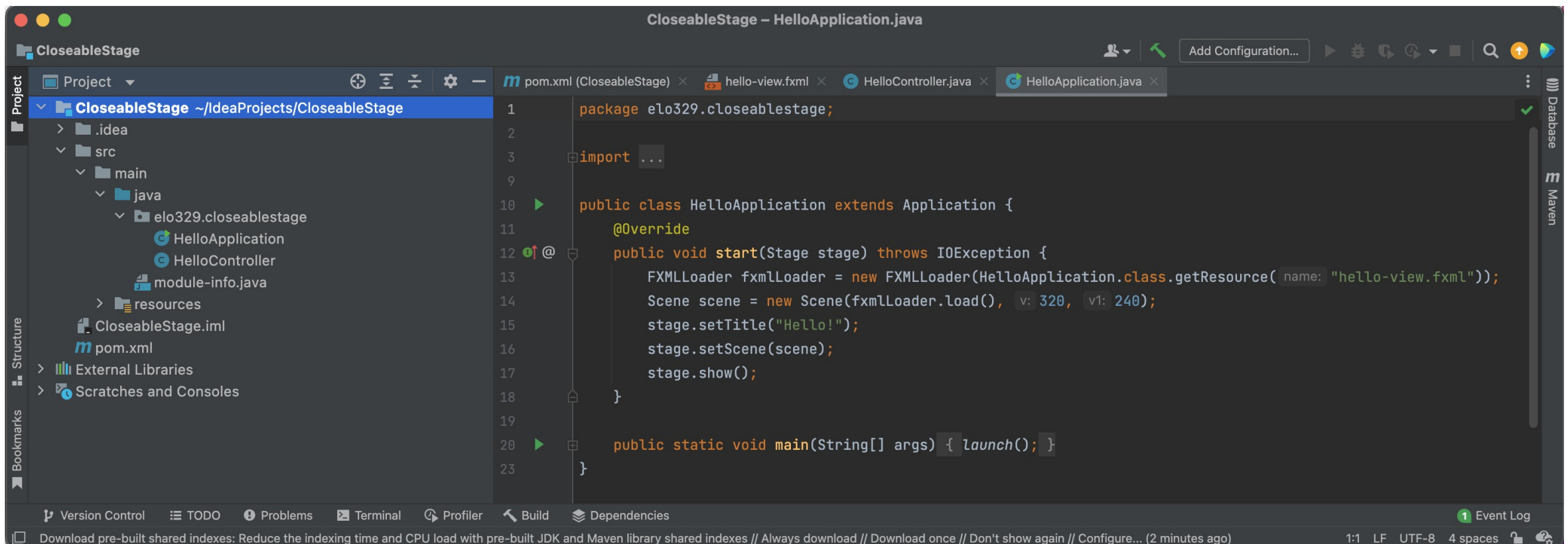
Aplicación JavaFX en IDE IntelliJ IDEA

- ❑ Paso 1: Abra su IDE IntelliJ IDEA y cree un nuevo proyecto JavaFX agregando su nombre y ubicación



Aplicación JavaFX en IDE IntelliJ IDEA

- ❑ Paso 2: No seleccione dependencias y ya puede compilar su primer programa.



Scene Builder

- ❑ JavaFX provee un entorno para creación de interfaces de usuario: el Scene Builder.
- ❑ Este se puede descargar desde la página <https://gluonhq.com/products/scene-builder/>
- ❑ Se motivará que usted desarrolle su propia interfaz gráfica codificando en Java con JavaFX para que usted vea la relación entre clases y el resultado obtenido.
- ❑ Superada la etapa previa, usted puede usar Scene Builder pero no será exigido en este curso.
- ❑ También se puede integrar con los otros IDEs disponibles