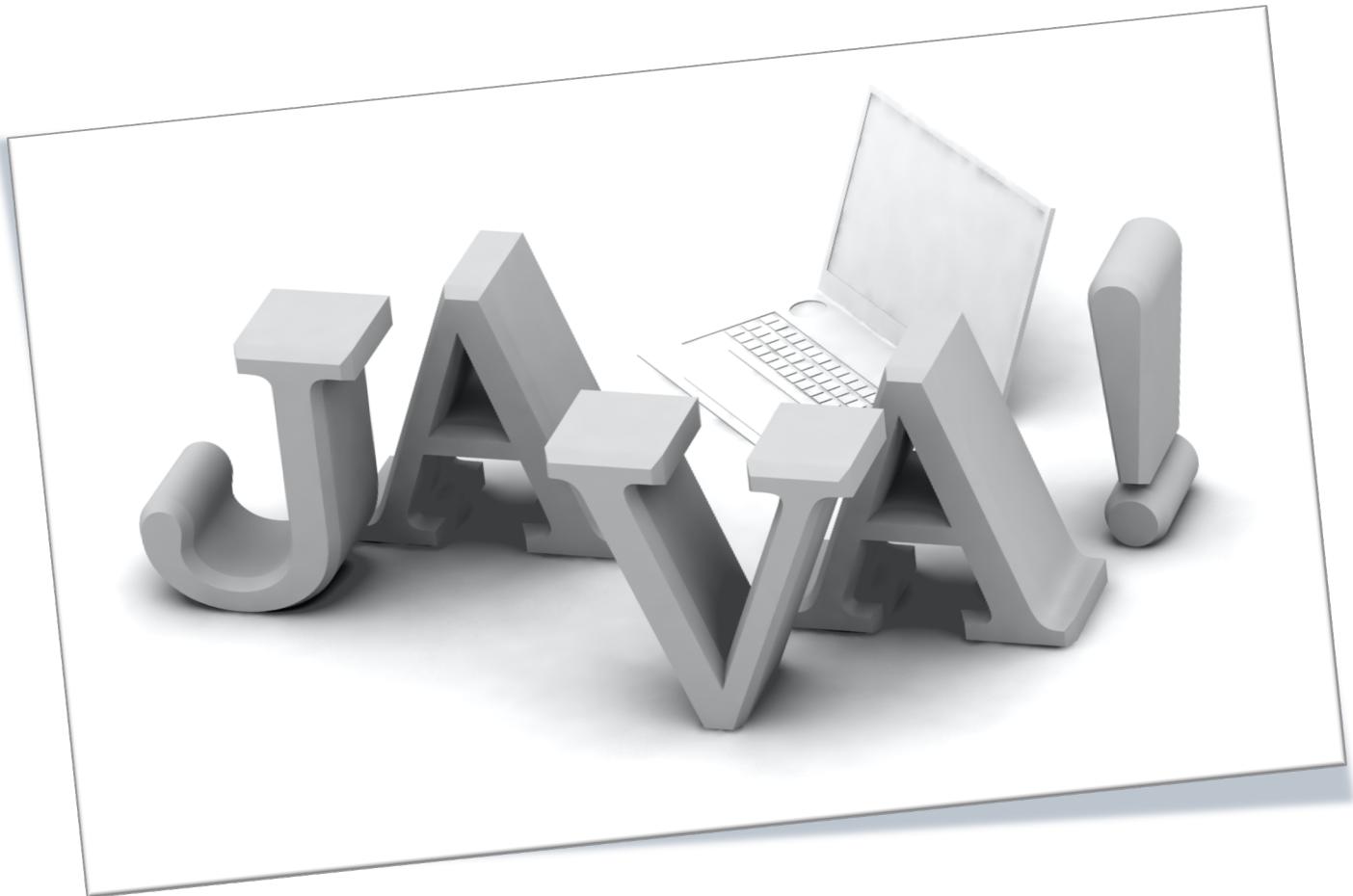


# PRÁCTICAS

## Capítulo 6: Programación gráfica y ficheros



## Unidad 6. Práctica 6.1. Programación gráfica. JavaFX

### Crea un proyecto que se llame Capítulo6

El proyecto debe estar en el paquete org.iesjaroso.daw.capítulo6.

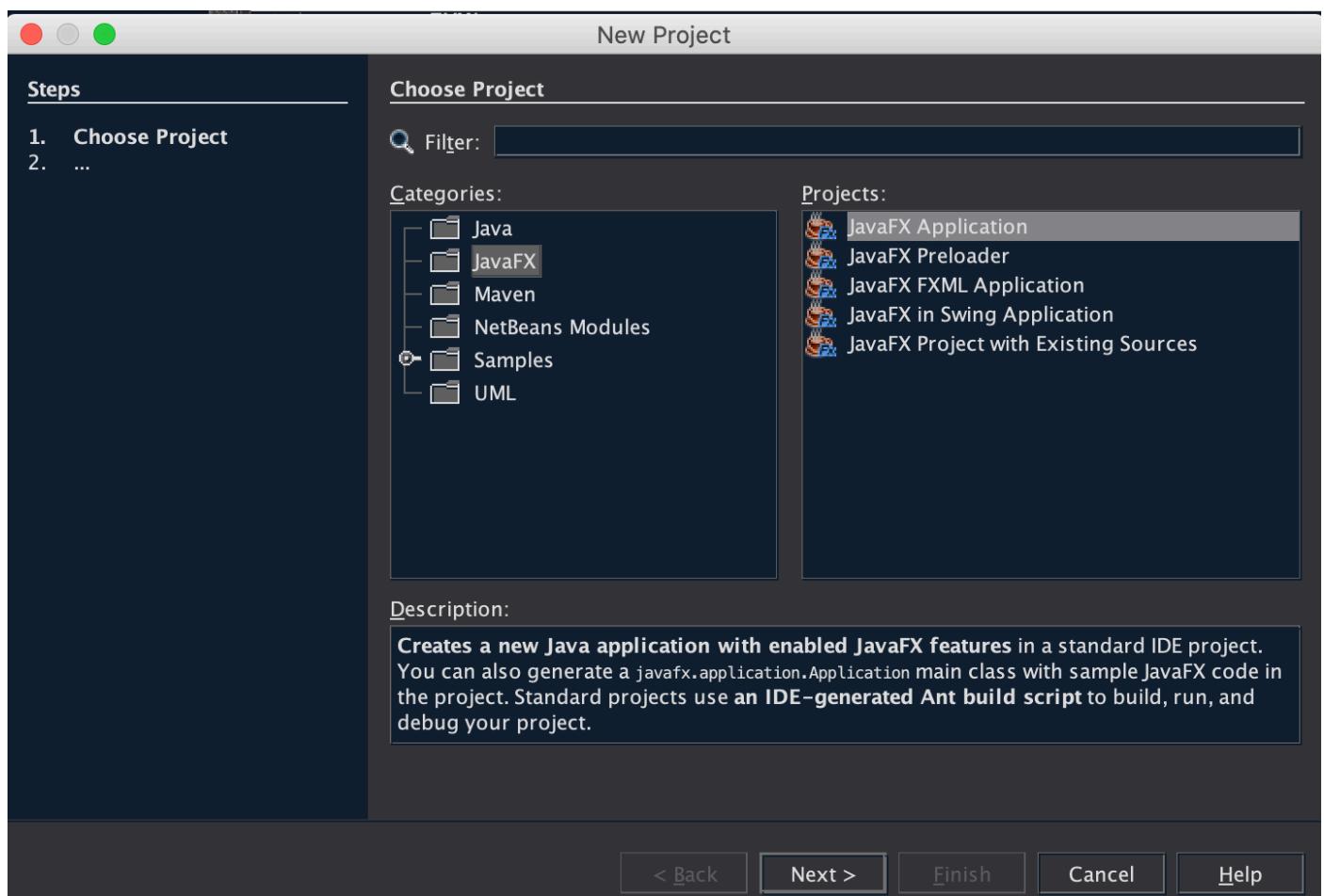
Organiza los ejercicios en paquetes si lo consideras necesario.

Debemos considerar en realizar las prácticas en diferentes proyectos

**NOTA.** Junto a estas prácticas se entrega como recurso una serie de proyectos realizados en JavaFX. Se recomienda probarlos con NetBeans y familiarizarse con la estructura de un proyecto JavaFX

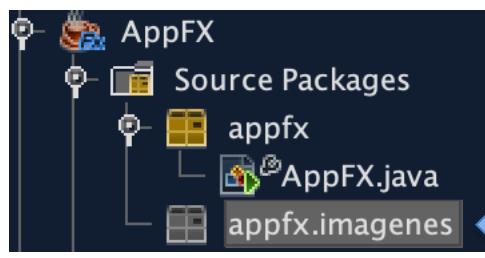
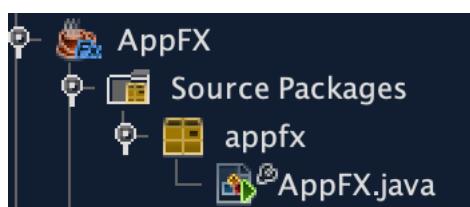
Se van a realizar 2 tipos de ejercicios, los primeros usando solo JavaFX, sin necesidad de utilizar una plantilla FXML y sin aplica el estilo arquitectónico Modelo-Vista-Controlador, y los segundos creando plantillas FXML con el programa SceneBuilder y aplicando el estilo arquitectónico MVC.

### Proyectos con solo JavaFX.



Al pasar a la siguiente ventana hacer clic en **Finish**.

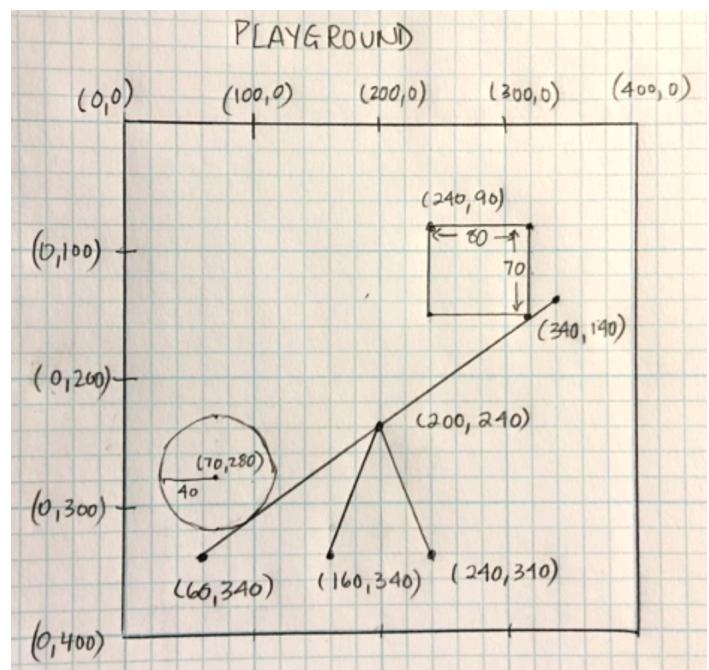
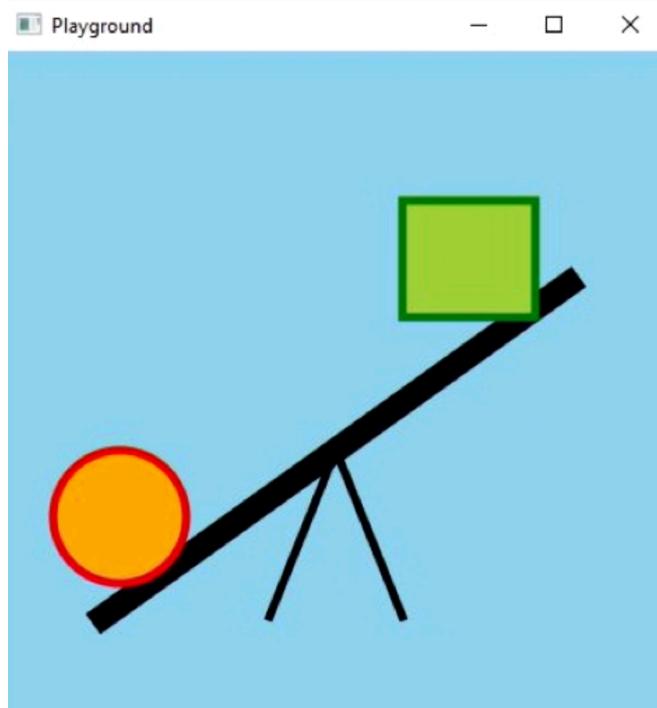
Estructura del proyecto:



Se puede crear una carpeta para almacenar imágenes

## Ejercicios con SOLO JavaFX.

**Ejercicio 1. Dibujar.** Crea la siguiente figura usando las formas básicas de JavaFX.

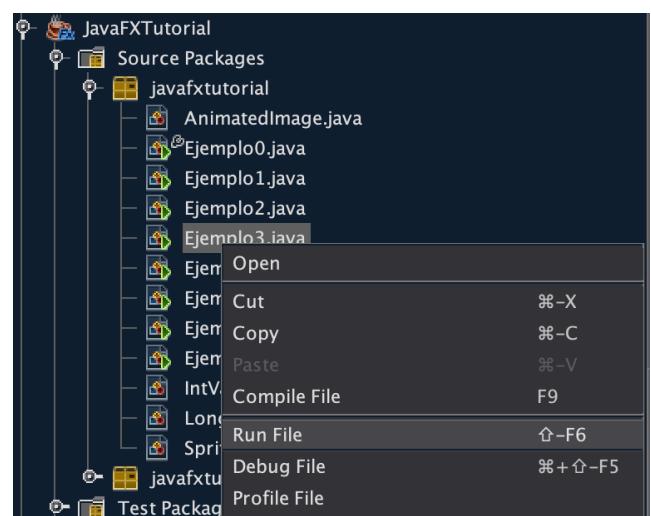


Para ello nos hacemos con la imagen un boceto (Sketch), en dónde están la posición (x,y) de cada figura (Rectángulo, círculo) sus valores.

Ten en cuenta:

1. El fondo de la ventana se puede cambiar.
2. Las formas básicas se caracterizan por que tienen trazo o contorno (stroke) y relleno (fill). Las propiedades más importante del trazo son el estilo, grosor, y color. Y del relleno la propiedad más importante es el color.
3. Fíjate que para colocar los objetos en la ventana partes de la posición (0, 0) que está en el extremo izquierdo superior. Tener esto claro es fundamental.
4. Se recomienda usar GraphicsContext y Canvas. Mirar la presentación y los ejemplos aportados. En este caso las coordenadas de las figuras se debe tener en cuenta en la posición (0, 0) del canvas. El canvas se coloca en la ventana (puede existir más de uno) y después los objetos se van colocando en el canvas.
5. En un mismo proyecto podemos tener varios documentos ejecutables (con diversos main). Para ejecutar uno, hacemos clic con el botón derecho del ratón sobre el fichero que queremos ejecutar y hacemos clic en Run File o pulsamos MAY+F6.  
Se recomienda esta estructura de proyecto para realizar la entrega.  
También es interesante por si necesitamos otros documentos o la entrega es por partes. Si hemos acabado una parte de un ejercicio podemos ir numerando. Observa que realmente lo que cambia de un archivo a otro es el método public void start(Stage primaryStage) {}
6. Recomendamos crear una carpeta images en el proyecto para introducir las imágenes. Ten cuidado con la ruta de las imágenes o el fichero no se ejecutará. Para este ejercicio no hace falta pero para los siguientes si lo necesitaremos.

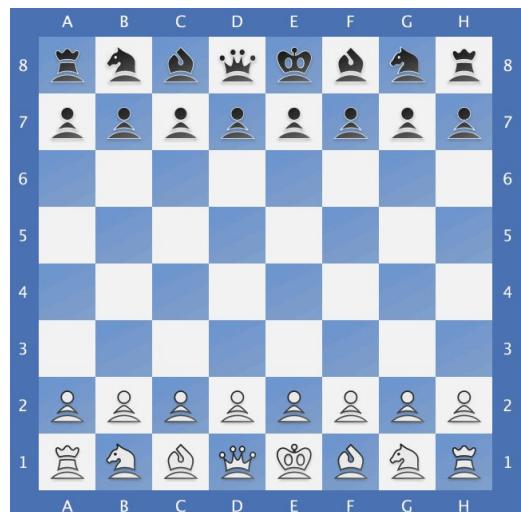
**Animación.** Anima el círculo y el cuadrado, haciendo que se muevan y cambien de color. Usa para ello AnimationTimer que tienes en los ejemplos.



**Ejercicio 2. Ajedrez.** Crea un tablero ajedrez que incluya la numeración de cada casillas y con las piezas usando JavaFX.

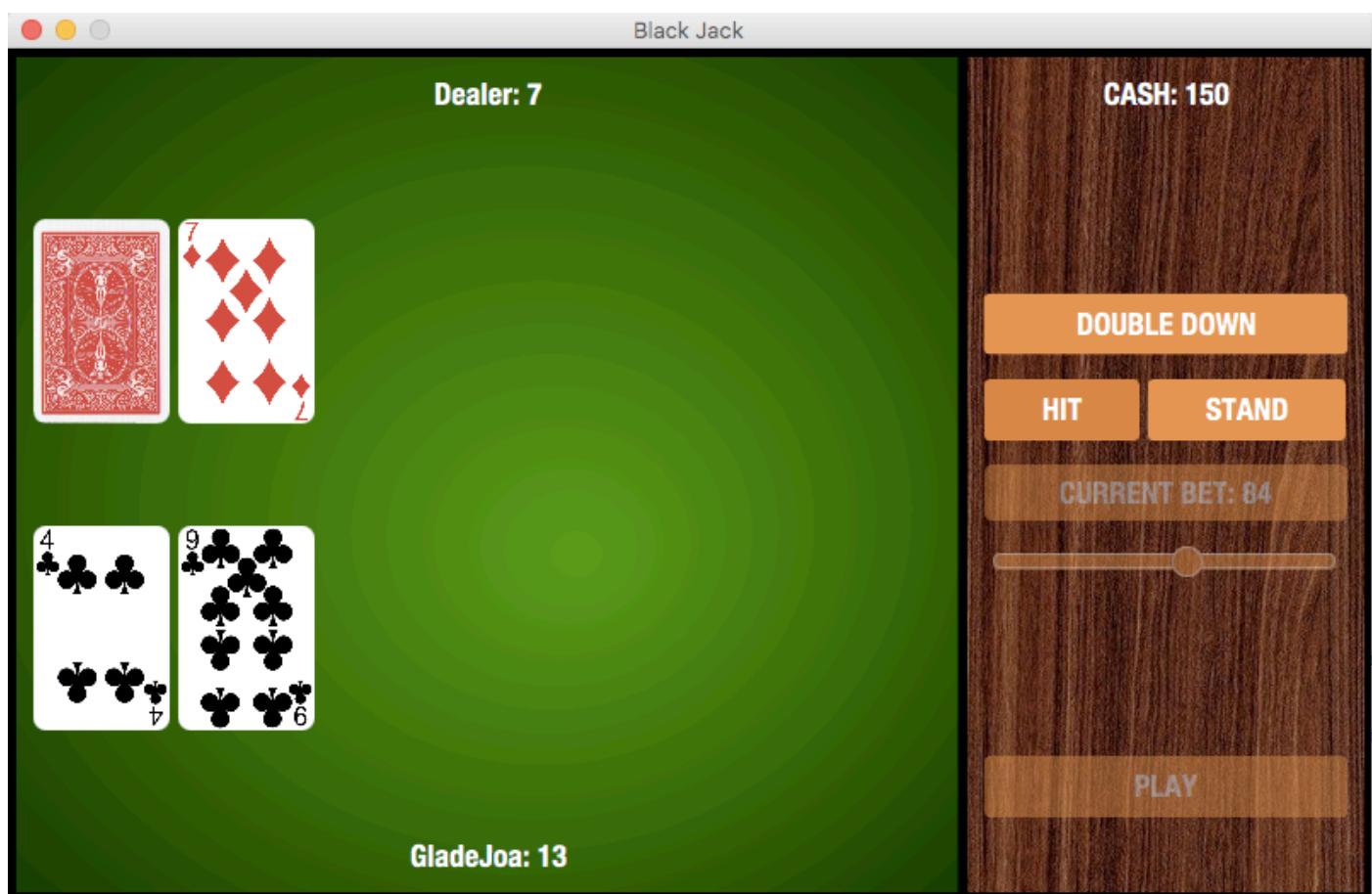
Consideraciones:

- Se recomienda hacer un boceto (sketch) antes de comenzar el proyecto.
- Comienza por el canvas, estableciendo un tamaño y color de fondo.
- El tablero lo crearemos usando un bucle y objetos rectángulos con un relleno.
- Las coordenadas de las casilla, letras y números, se crearán usando objetos texto y con un bucle.
- Las piezas, están disponibles como recurso, en la carpeta images\ajedrez



**Animación.** Anima una pieza moviéndose de una casilla a otra.

**Ejercicio 3. Blackjack.** Crea una ventana usando JavaFX parecida a la imagen:



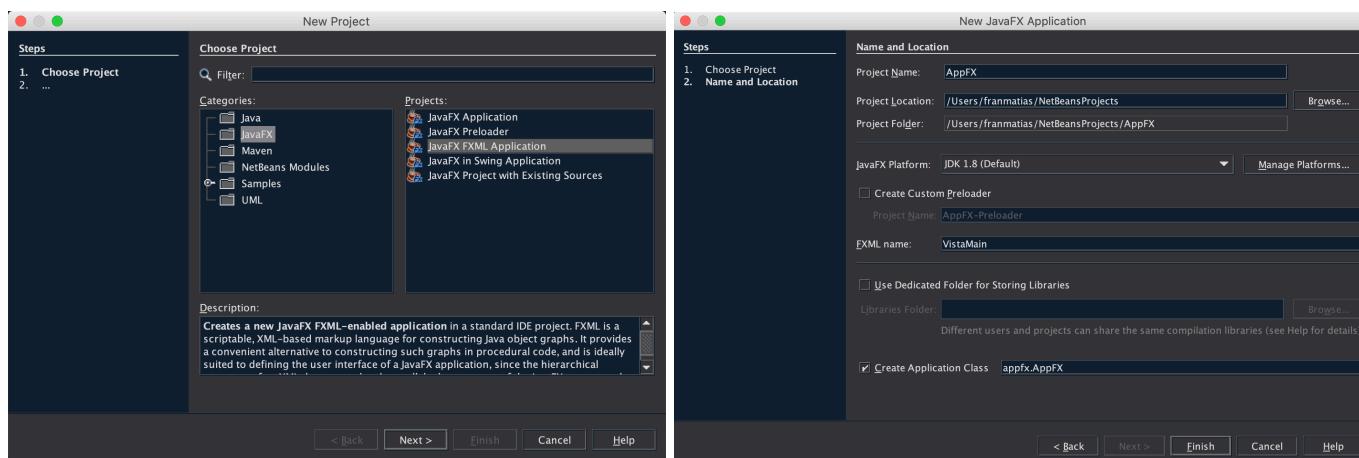
Las imágenes de las cartas saldrán de forma aleatoria.

Busca las cartas en internet en formato .png o .svg (preferible este último).

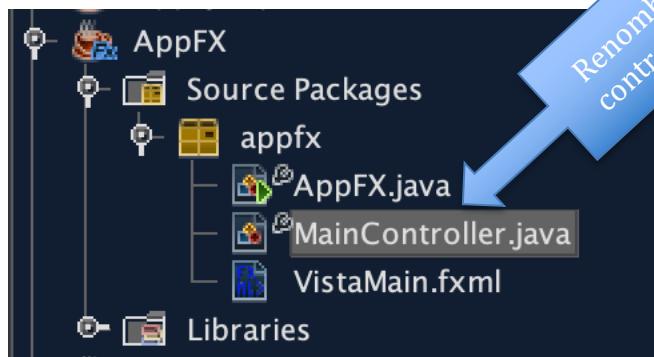
No será interactivo, es decir, no tendrá ninguna funcionalidad, solo mostrará el entorno.

## Proyectos con JavaFX+SceneBuilder y usando MVC.

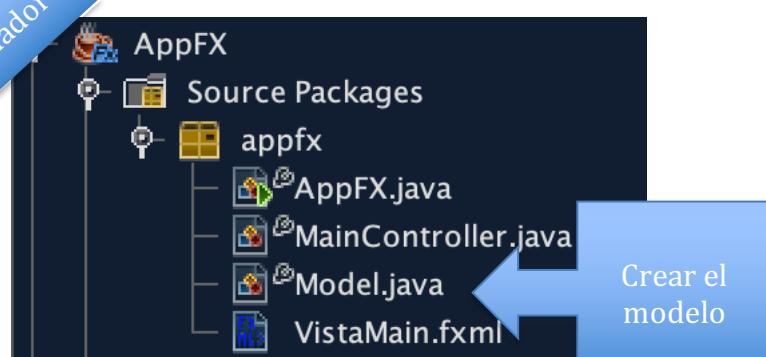
Proyectos usando SceneBuilder y el estilo arquitectónico Modelo-Vista-Controlador(MVC)



Estructura del proyecto:



Renombrar controlador



Crear una nueva clase para crear el Modelo:

## Ejercicios con usando JavaFX + SceneBuilder + MVC.

### Ejercicio 4. Diccionario.

Diseña una aplicación que permita insertar, buscar, eliminar, modificar palabras con su definición, es decir, un diccionario.

#### Consideraciones:

- Se recomienda usar un Map para almacenar las palabras con su definición. Esto estaría almacenado en el Modelo (Model). No olvides crear la clase del modelo. Investiga los tutoriales como se relacionan el modelo, la vista y el controlador.
- Crea en el modelo las acciones: añadir, buscar, eliminar, modificar
- Diseña una ventana que permita realizar todas las acciones.
- Utiliza un GridLayout para colocar todos los elementos
- Se recomienda utilizar el componente TableView usando una ObservableList. Recomendamos el siguiente videotutorial: <https://www.youtube.com/watch?v=2ZZ7UOGjY54>

### Ejercicio 5. Agenda de contactos tipo teléfono.

Se desea hacer una aplicación que permita llevar la gestión de una agenda. (Agenda de contactos) Piensa que se debe permitir insertar, buscar, eliminar y modificar personas de la agenda.

Este proyecto debe incluir:

- Componente menú.
- Ventanas modales.
- Y los controles:
 

<input checked="" type="checkbox"/> Label	<input checked="" type="checkbox"/> TextArea	<input checked="" type="checkbox"/> DatePicker
<input checked="" type="checkbox"/> TextField	<input checked="" type="checkbox"/> Button	<input checked="" type="checkbox"/> ImageView
<input checked="" type="checkbox"/> CheckBox	<input checked="" type="checkbox"/> Spinner	
<input checked="" type="checkbox"/> MenuBar	<input checked="" type="checkbox"/> ChoiceBox	
<input checked="" type="checkbox"/> Slider		