

JavaFX en IntelliJ



1. **Introducción**
2. **Proyectos JavaFX con el IDE IntelliJ**
 2. 1. [Descargar el SDK](#)
 2. 2. [Añadir la librería de JavaFX a un proyecto IntelliJ](#)
 2. 3. [Crea un proyecto de JavaFx \(Opción no recomendada\)](#)
 2. 4. [Primer proyecto](#) `JavaFX`
3. **Píldoras informáticas relacionadas**
4. **Fuentes de información**

1. Introducción

Vamos a ver cómo crear una aplicación `JavaFX` en IntelliJ. Necesitamos:

- IntelliJ Ultimate Edition (seguramente con el Community Edition funcione igualmente)
- OpenJDK 17 (seguramente funcionará con una posterior)

2. Proyectos JavaFX con el IDE IntelliJ

Sigue estos pasos para crear un proyecto `JavaFX` utilizando las herramientas IDE para compilarlo y ejecutarlo.

2.1. Descargar el SDK

Descargar el SDK (versión 21 LTS) de `JavaFX` <https://gluonhq.com/products/javafx/> apropiado para tu sistema operativo. Este es mi caso:

Downloads

JavaFX version: 21.0.3 [LTS] | Operating System: Linux | Architecture: x64 | Type: SDK

☐ Include older versions

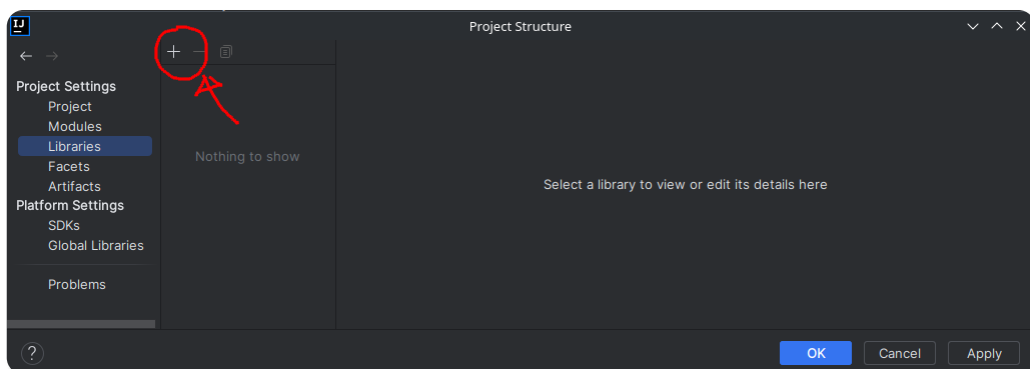
Supported Platforms

OS	Version	Architecture	Type	Download
Linux	21.0.3	x64	SDK	Download [SHA256]

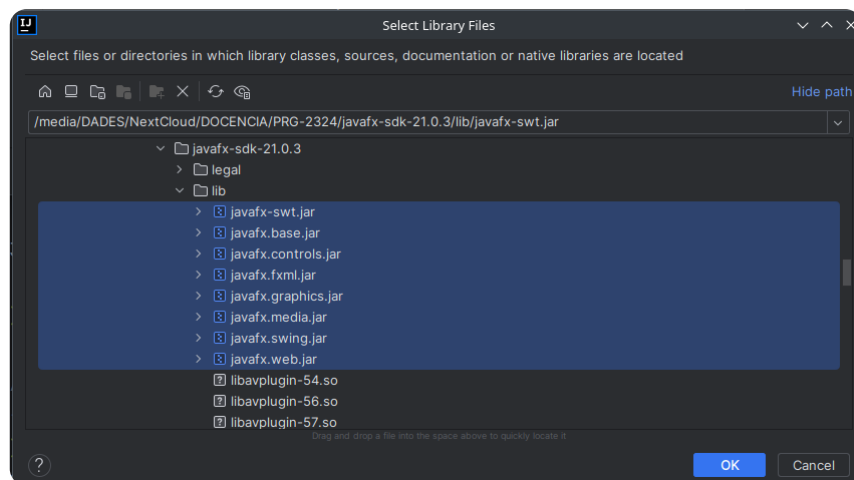
Descomprímelo en la ubicación deseada, por ejemplo, `/Users/your-user/Downloads/javafx-sdk-19`. Es **IMPORTANTÍSIMO** que recuerdes la ruta (`path`) hasta esta librería.

2.2. Añadir la librería de JavaFX a un proyecto IntelliJ

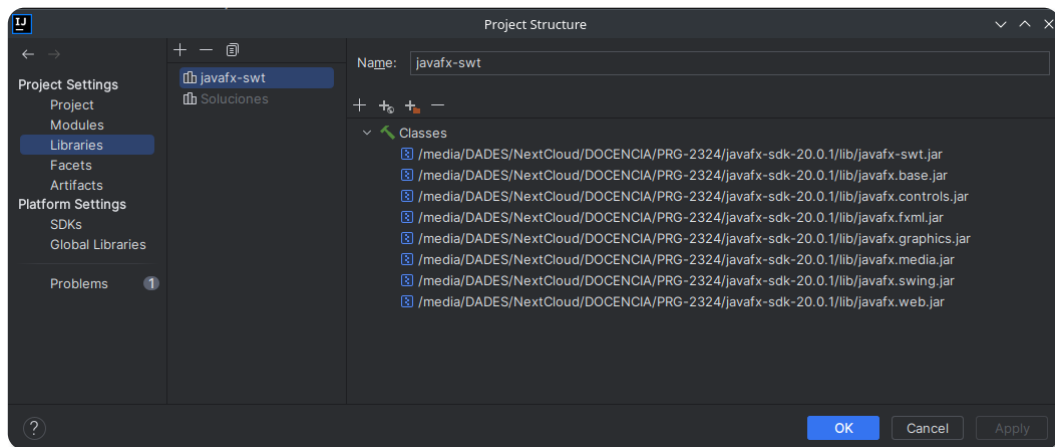
En `File` -> `Project Structure` -> `Libraries`



A continuación busca la carpeta que descargaste en el punto anterior y selecciona todos los archivos con extensión `*.jar` que incluye.

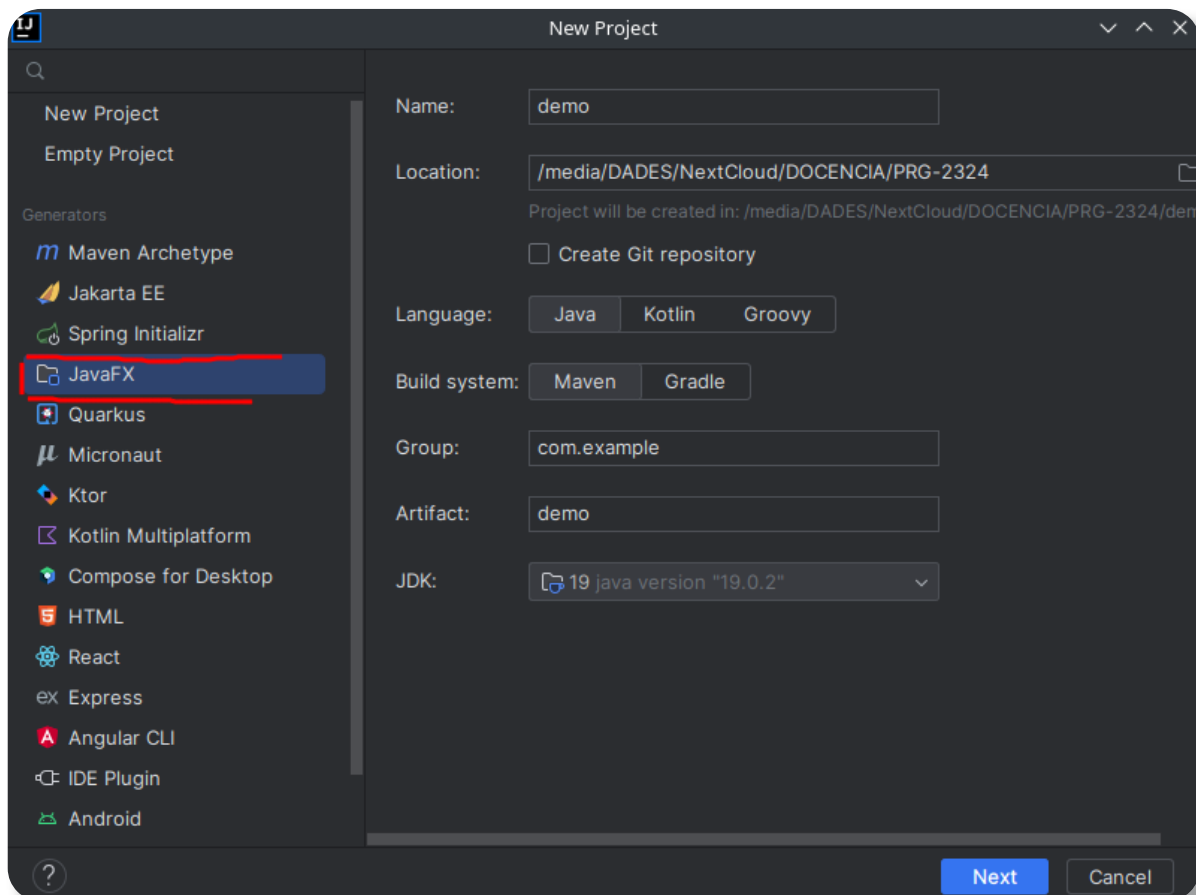


Deberías terminar con algo similar a esto:



2.3. Crea un proyecto de JavaFx (Opción no recomendada)

Cuando creas un proyecto en IntelliJ, el mismo IDE ofrece un tipo de proyecto JavaFx:



Importante No recomiendo esta opción para empezar porque "oculta/facilita" alguna de las cosas que necesitas entender.

2.4. Primer proyecto JavaFX

Una vez creado el proyecto Java, o configurado tal y como hemos visto en los puntos anteriores, comprobaremos que el IDE reconoce las clases de `JavaFX`, para ello vamos a crear nuestra primera clase con interfaz gráfica con el siguiente código:

Comenzaremos por:

1. Heredar la clase principal de la clase `Application`.
2. Importar las clases de la librería.
3. Implementar el método abstracto, añadir el `launch()` al `main()` y añadir los imports necesarios.

```
1 | import javafx.application.Application;
```

```

2  import javafx.scene.Parent;
3  import javafx.scene.Scene;
4  import javafx.stage.Stage;
5
6  import javafx.scene.layout.StackPane;
7  import javafx.scene.text.Text;
8
9  public class HolaFx extends Application {
10
11      private Parent createContent() {
12          return new StackPane(new Text("Hola Mundo"));
13      }
14
15      @Override
16      public void start(Stage stage) throws Exception {
17          stage.setScene(new Scene(createContent(), 400, 400));
18          stage.show();
19      }
20
21      public static void main(String[] args) {
22          launch(args);
23      }
24  }

```

Si ejecutamos ahora mismo el código, aunque no tiene ningún error de compilación, obtendremos el siguiente error de ejecución:

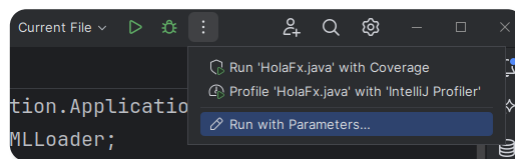
```

1  /usr/lib/jvm/java-1.19.0-openjdk-amd64/bin/java -javaagent:/opt/idea-IU-
232.9921.47/lib/idea_rt.jar=33389:/opt/idea-IU-232.9921.47/bin -Dfile.encoding=UTF-8 -
Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath
/media/DADES/NextCloud/DOCENCIA/PRG-
2324/JavaFXTest/out/production/JavaFXTest:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-
sdk-21.0.3/lib/javafx-swt.jar:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-
21.0.3/lib/javafx.base.jar:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-
21.0.3/lib/javafx.controls.jar:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-
21.0.3/lib/javafx.fxml.jar:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-
21.0.3/lib/javafx.graphics.jar:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-
21.0.3/lib/javafx.media.jar:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-
21.0.3/lib/javafx.swing.jar:/media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-
21.0.3/lib/javafx.web.jar HolaFx
2  Error: JavaFX runtime components are missing, and are required to run this application
3
4  Process finished with exit code 1

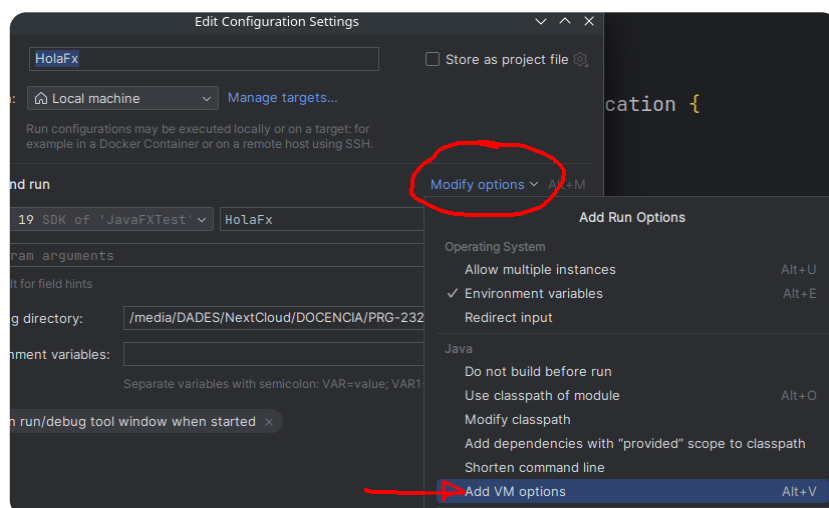
```

Para resolver este error, debemos editar los parámetros de ejecución del proyecto:

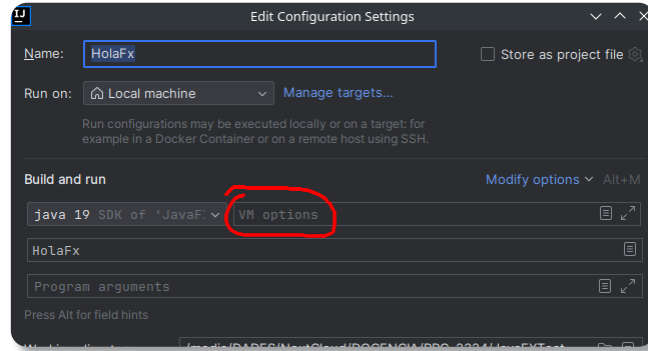
1. Elegimos la opción `Run with Parameters...`



2. Ahora desplegamos el menú `Modify options`, y añadimos las opciones de VM (máquina virtual):



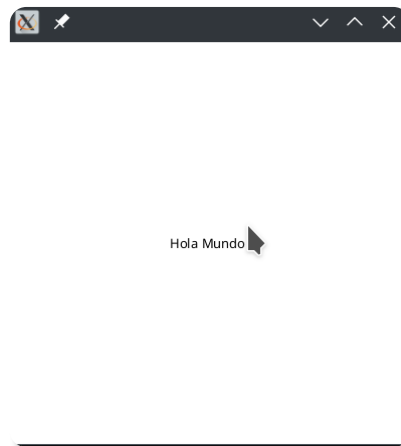
3. Dentro de esta opción que aparece ahora:



Añadiremos los siguientes parámetros:

```
1 --module-path /media/DADES/NextCloud/DOCENCIA/PRG-2324/javafx-sdk-21.0.3/lib/ --
  add-modules=javafx.controls
2 ##OJO! la ruta detrás de --module-path debe ser la ruta que apunte a donde tienes
  tu librería de JavaFX
```

Ahora, si volvemos a ejecutar el mismo proyecto veremos algo similar a esto:



3. Píldoras informáticas relacionadas

- <https://www.youtube.com/playlist?list=PLNjWMbvTjAljLRW2qyuc4DEgFVW5YFRSR>
- <https://www.youtube.com/playlist?list=PLaxZkGILWHGUWZxuadN3j7KKalCRlh5->

4. Fuentes de información

- Apuntes de Jose Antonio Diaz-Alejo
- <https://github.com/openjfx/openjfx-docsopen>
- <https://github.com/openjfx/samples>
- [FXDocs](#)
- <https://openjfx.io/openjfx-docs/>
- <https://docs.oracle.com/javase/8/javafx/user-interface-tutorial>
- <https://github.com/JonathanGiles/scenic-view>