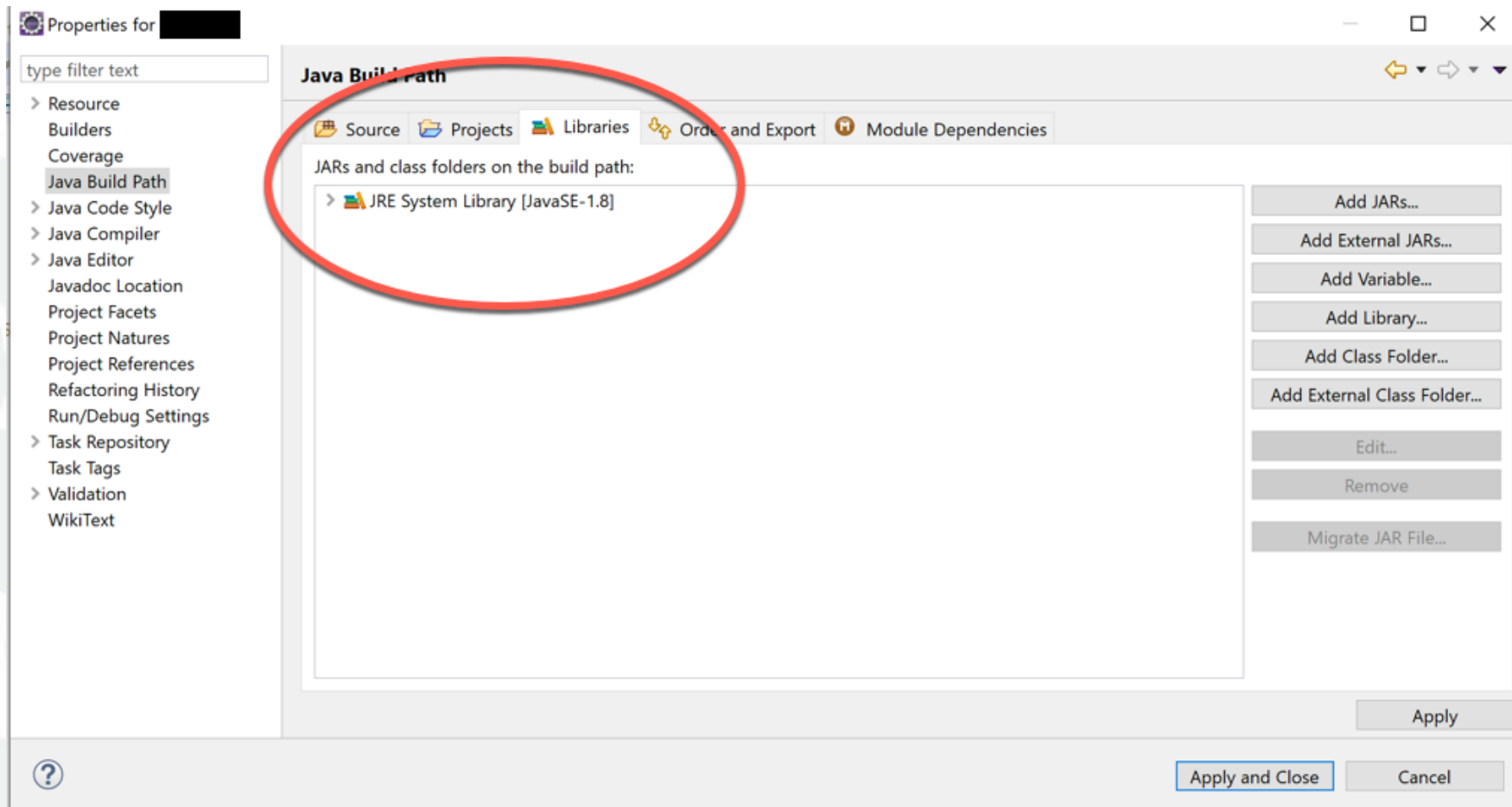


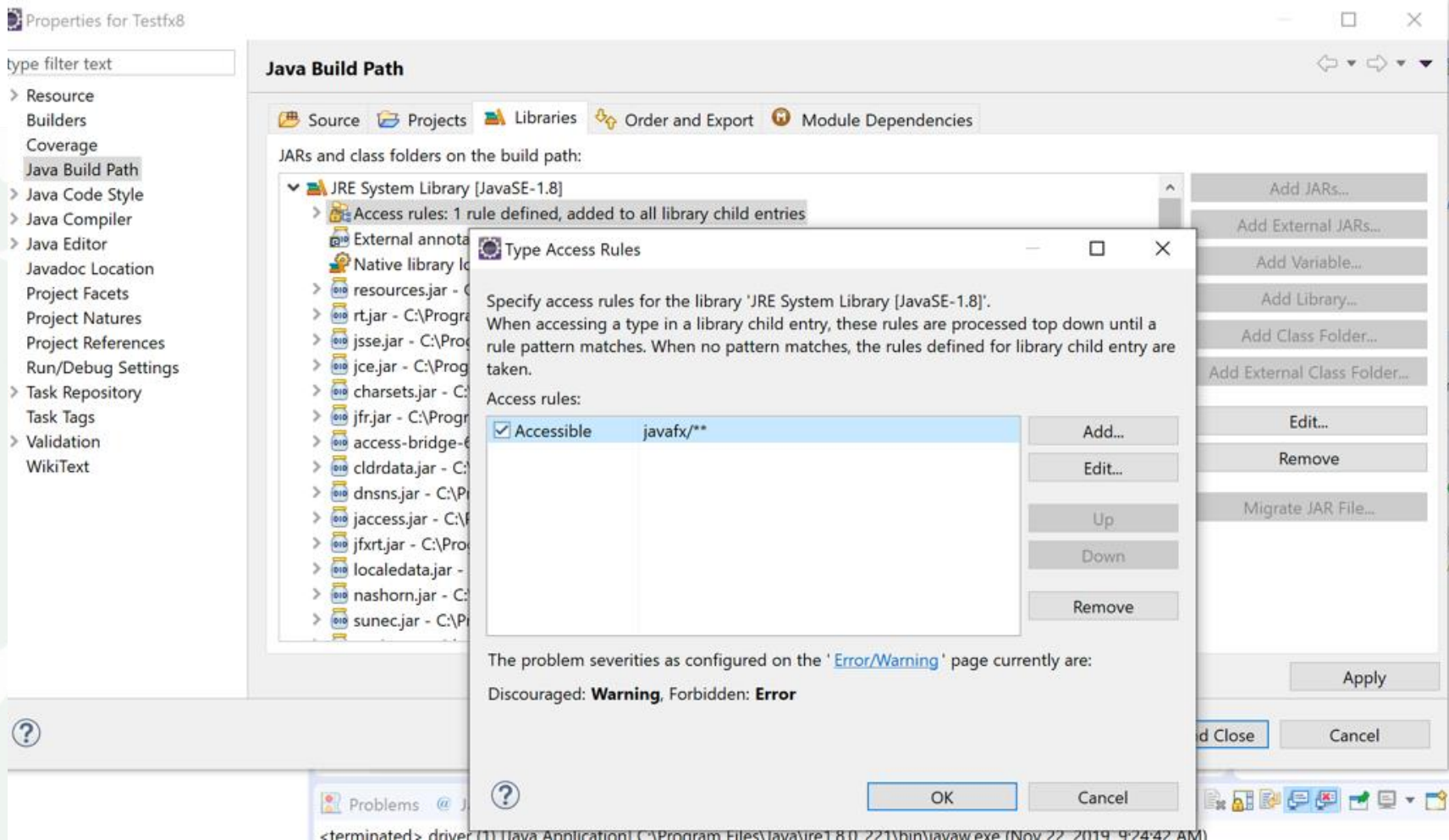
Installing JavaFX

- Two options:
 - Install java 8 (both JDK and JRE)
 - Install e(fx)clipse 3.6.x from Eclipse marketplace
 - This is the easiest option
 - Install Java 11 and JavaFX11 and apply appropriate steps in Eclipse!

Java 8



*optional: You may need to set restrictions:



The screenshot shows the Eclipse IDE interface. On the left is the 'Properties for Testfx8' sidebar with a tree view containing categories like Resource, Builders, Coverage, and Java Build Path. The main window displays the 'Java Build Path' dialog, with the 'Libraries' tab selected. Under 'JRE System Library [JavaSE-1.8]', the 'Access rules' section is expanded. A 'Type Access Rules' dialog box is overlaid on top, containing the following text:

Specify access rules for the library 'JRE System Library [JavaSE-1.8]'.
When accessing a type in a library child entry, these rules are processed top down until a rule pattern matches. When no pattern matches, the rules defined for library child entry are taken.

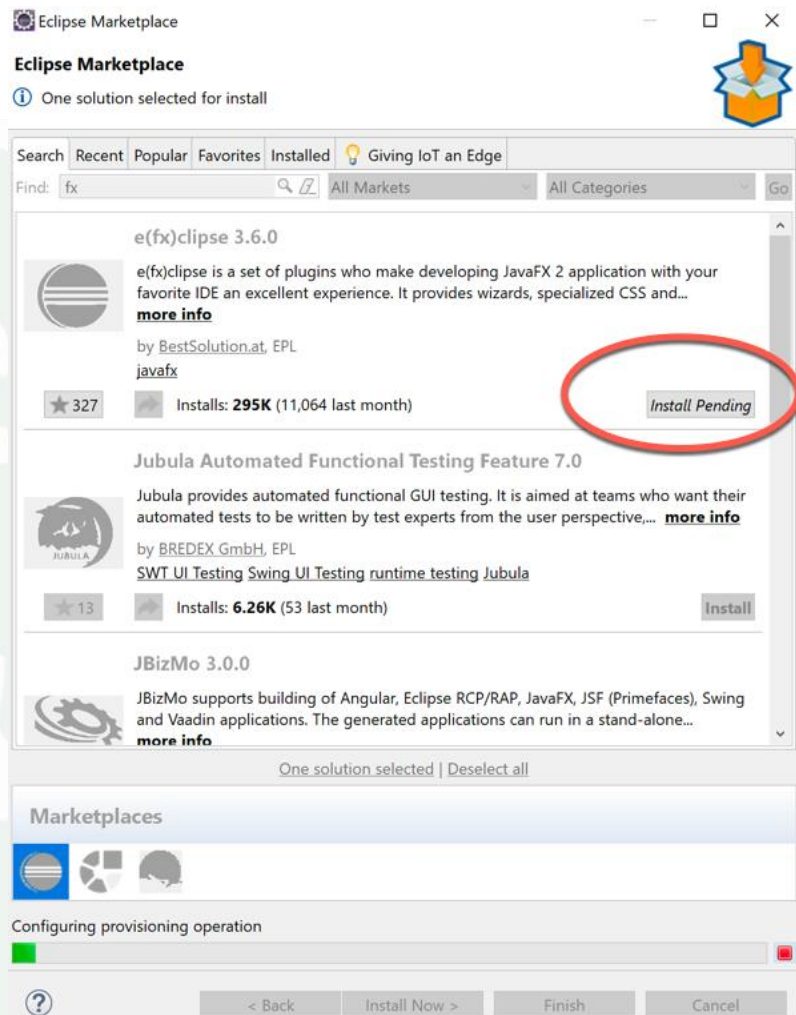
Access rules:

Access rule	Pattern	Buttons
<input checked="" type="checkbox"/> Accessible	javafx/**	Add... Edit... Up Down Remove

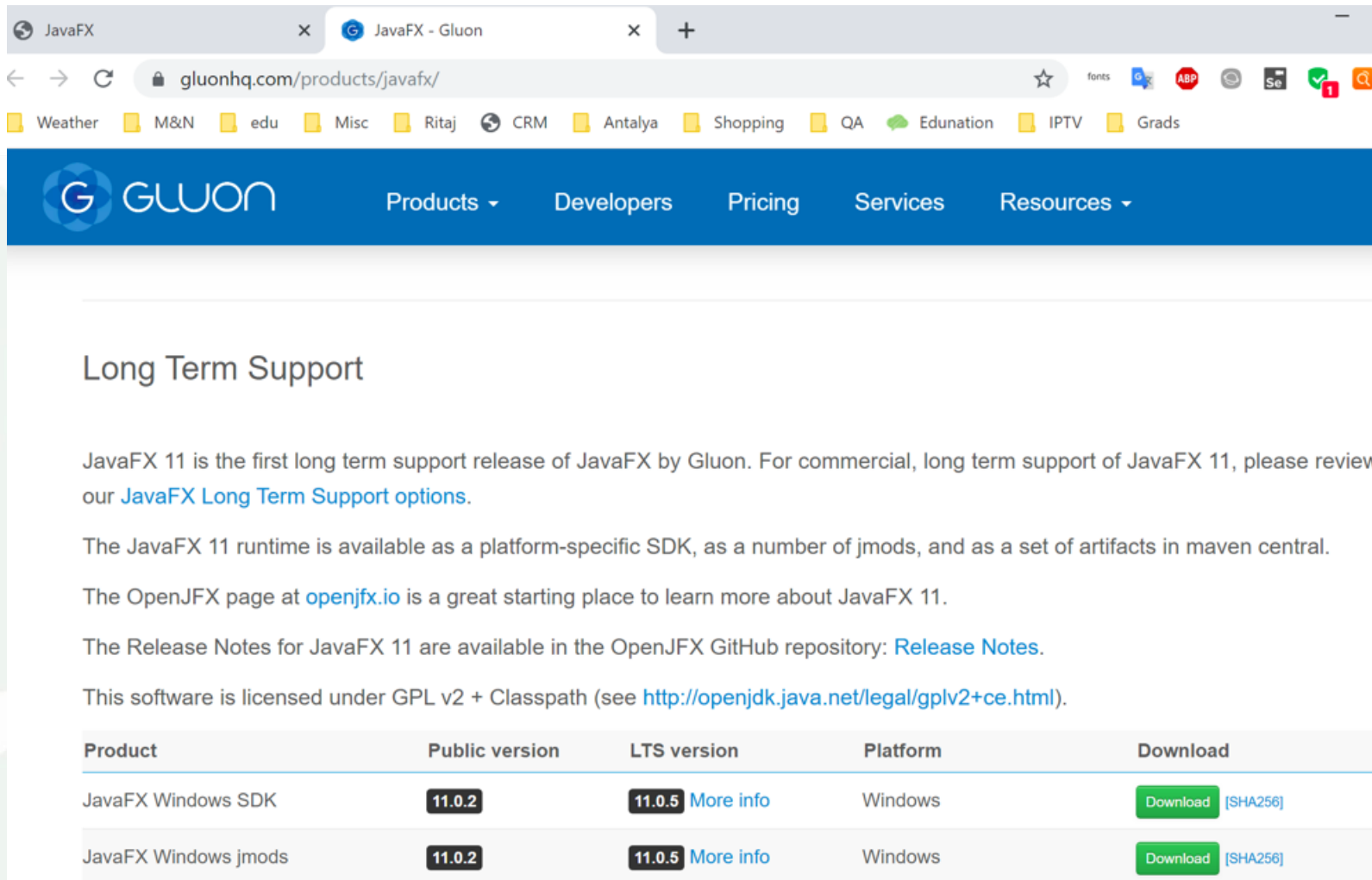
The problem severities as configured on the 'Error/Warning' page currently are:
Discouraged: **Warning**, Forbidden: **Error**

Buttons at the bottom of the dialog include OK and Cancel.

Optional: Java 8: Install e(fx)clipse inside eclipse



Java 11: Get JavaFX



The screenshot shows a web browser window with the URL `gluonhq.com/products/javafx/`. The page features a blue navigation bar with the Gluon logo and menu items: Products, Developers, Pricing, Services, and Resources. The main content area is titled "Long Term Support" and contains several paragraphs of text. At the bottom, there is a table with columns for Product, Public version, LTS version, Platform, and Download.

Long Term Support

JavaFX 11 is the first long term support release of JavaFX by Gluon. For commercial, long term support of JavaFX 11, please review our [JavaFX Long Term Support options](#).

The JavaFX 11 runtime is available as a platform-specific SDK, as a number of jmods, and as a set of artifacts in maven central.

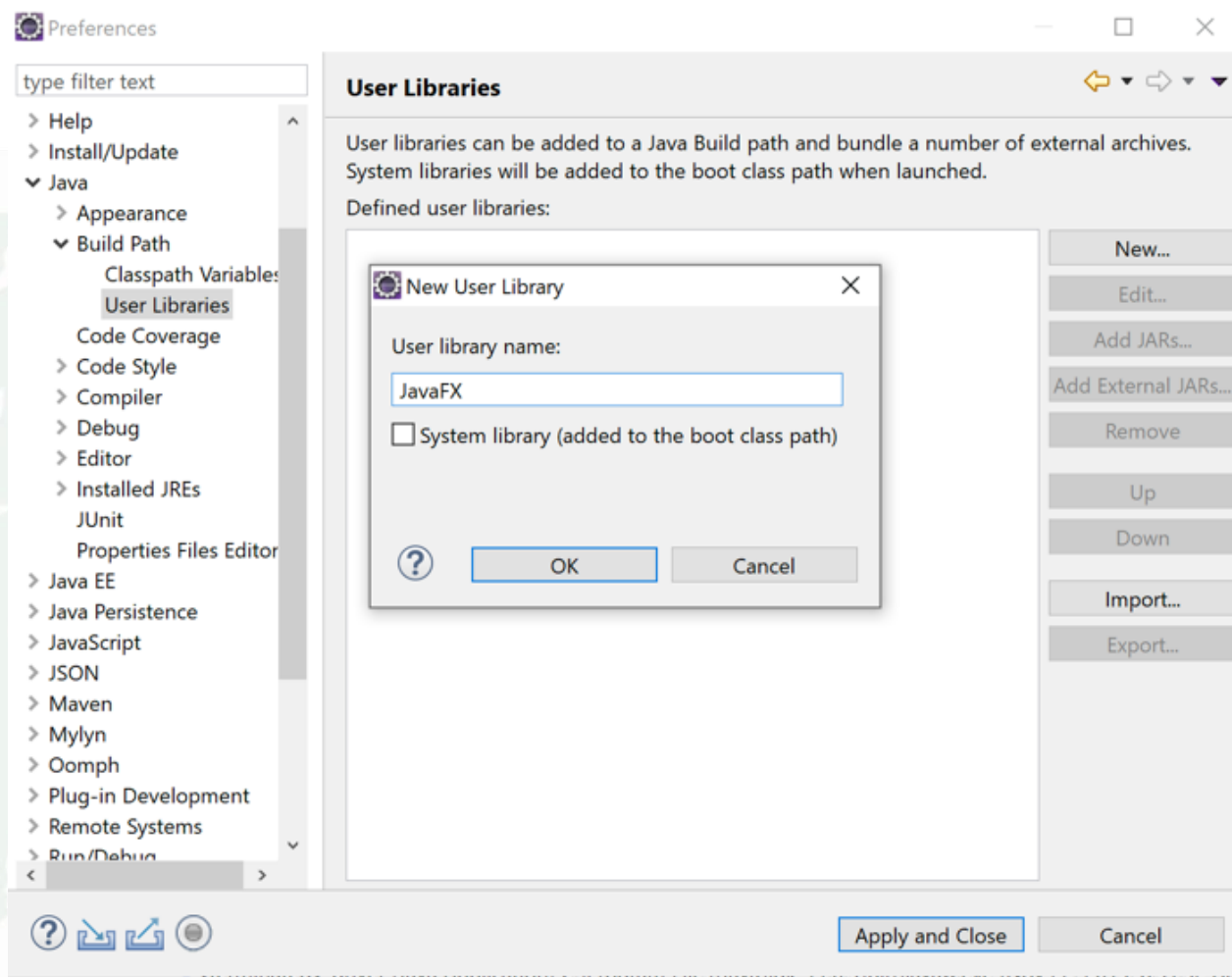
The OpenJFX page at openjfx.io is a great starting place to learn more about JavaFX 11.

The Release Notes for JavaFX 11 are available in the OpenJFX GitHub repository: [Release Notes](#).

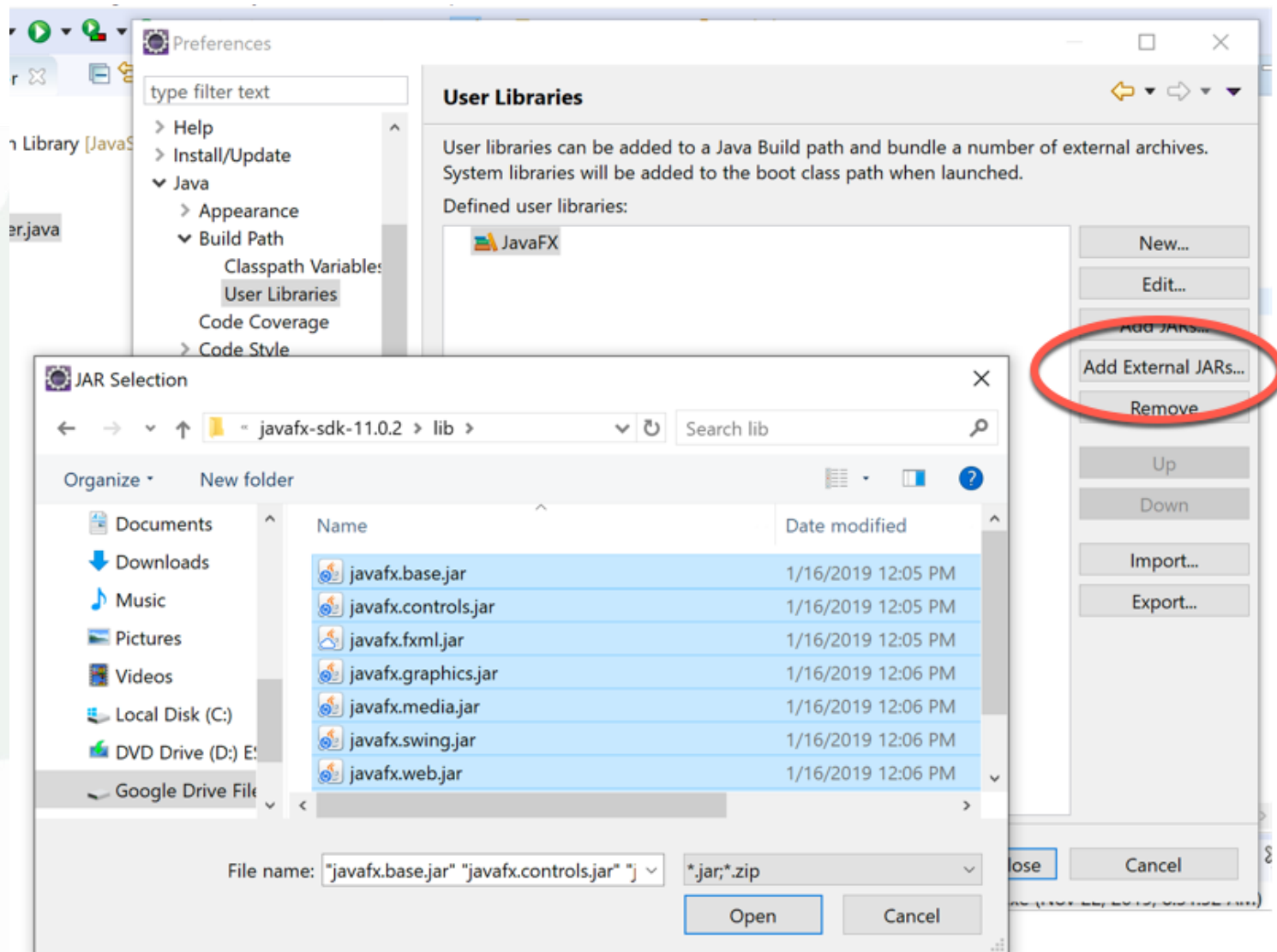
This software is licensed under GPL v2 + Classpath (see <http://openjdk.java.net/legal/gplv2+ce.html>).

Product	Public version	LTS version	Platform	Download
JavaFX Windows SDK	11.0.2	11.0.5 More info	Windows	Download [SHA256]
JavaFX Windows jmods	11.0.2	11.0.5 More info	Windows	Download [SHA256]

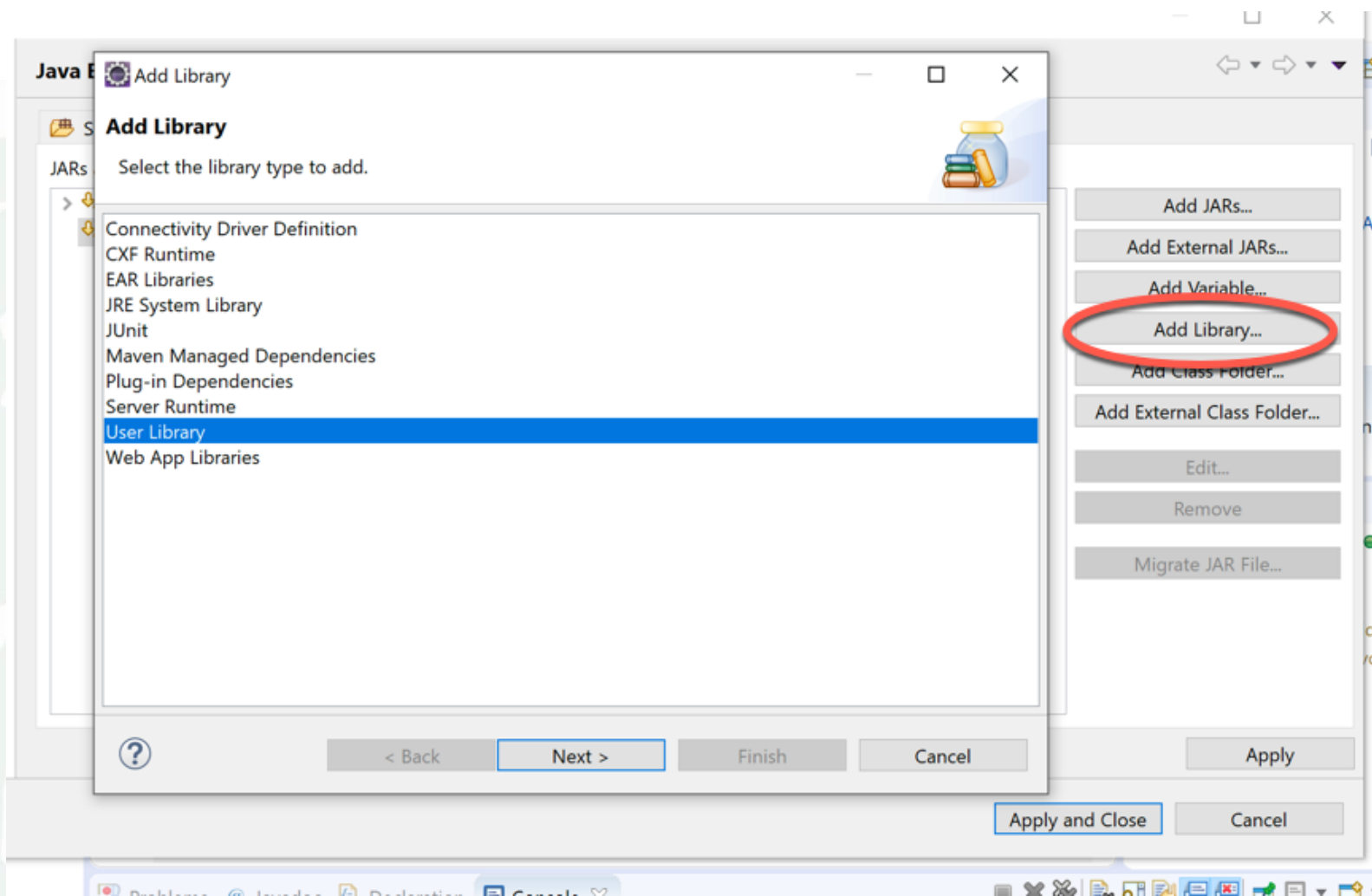
Create user library JavaFX



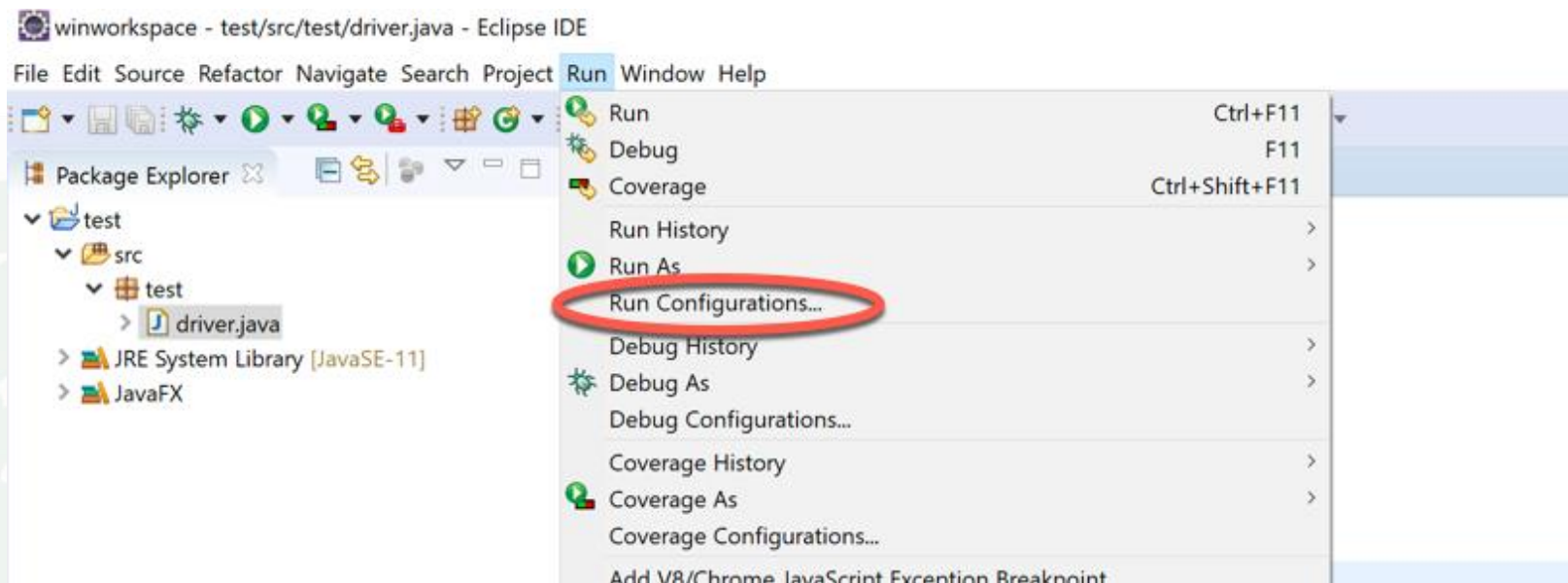
Add javafx files to library



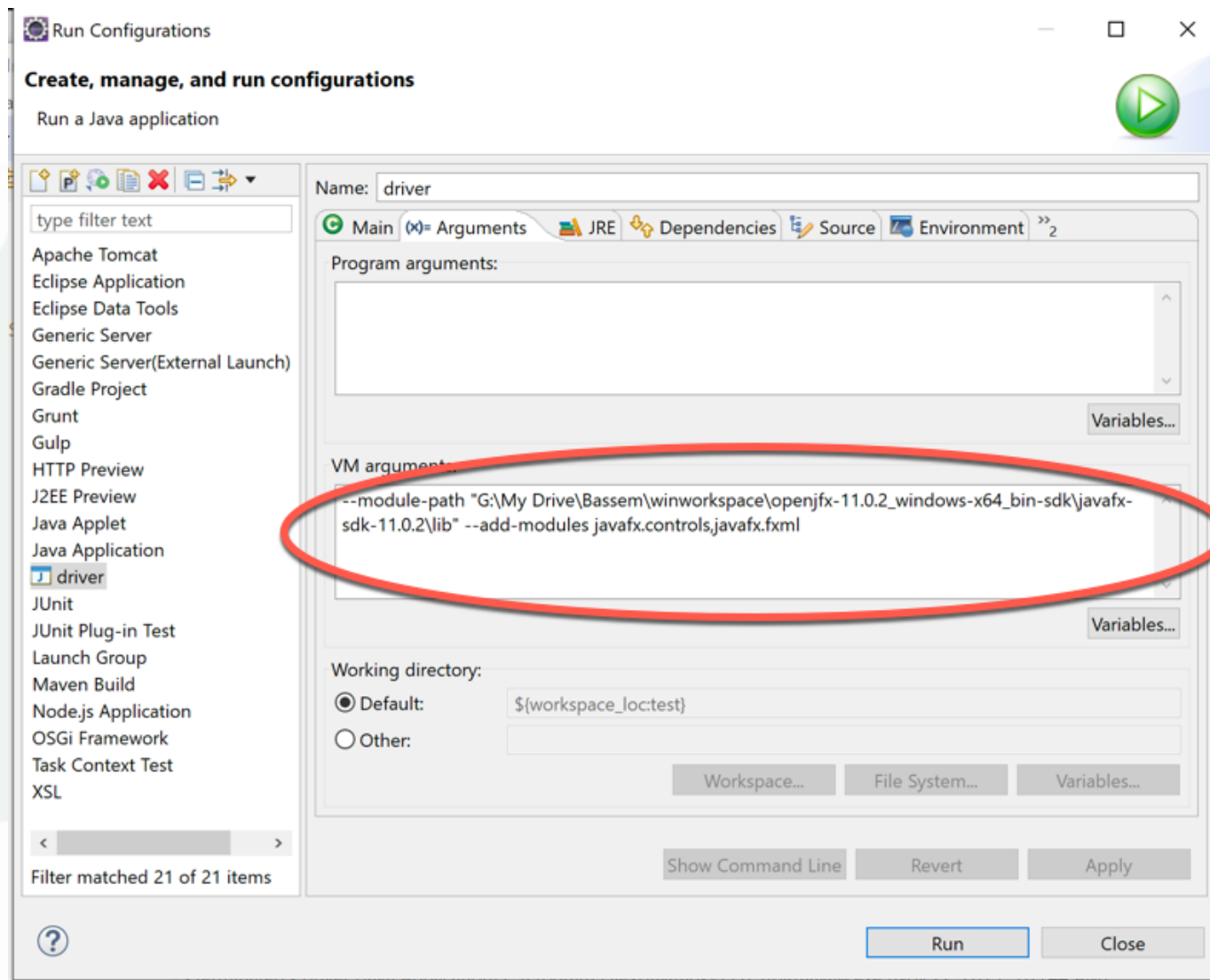
Project Build path



Run Configuration



Important: Add vm arguments



Run First application (Java 8 and 11)

```
package test;

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

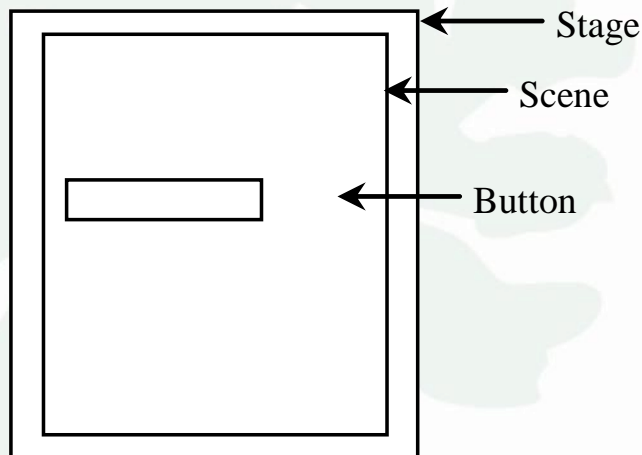
public class driver extends Application{

    public void start(Stage primaryStage) {
        // Create a scene and place a button 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) {
        Application.launch(args);
    }
}
```

Basic Structure of JavaFX

- Application
- Override the start(Stage) method
- Stage, Scene, and Nodes



MyJavaFX

Run

MultipleStageDemo

Run