

Dieses Tutorial basiert tw. auf einem Tutorial von Michael Kofler <https://kofler.info/java-11-javafx-intellij-idea-und-linux/>. Es wurde für unsere Veranstaltung angepasst.

Schritt 1: Notwendige Software installieren:

Java 11 und IntelliJ installieren

Diesen Schritt sollten Sie bereits in Aufgabenblatt erledigt haben. Zur Vorgehensweise wird daher auf Aufgabenblatt A verwiesen.

JavaFX installieren

JavaFX kann [hier](#) kostenlos heruntergeladen werden. (JavaFX wird von der Community gepflegt. Die Downloads stellt die Firma Gluon zur Verfügung.) Sie können die Dateien nun in ein beliebiges Verzeichnis installieren.

JavaFX Scenbuilder installieren

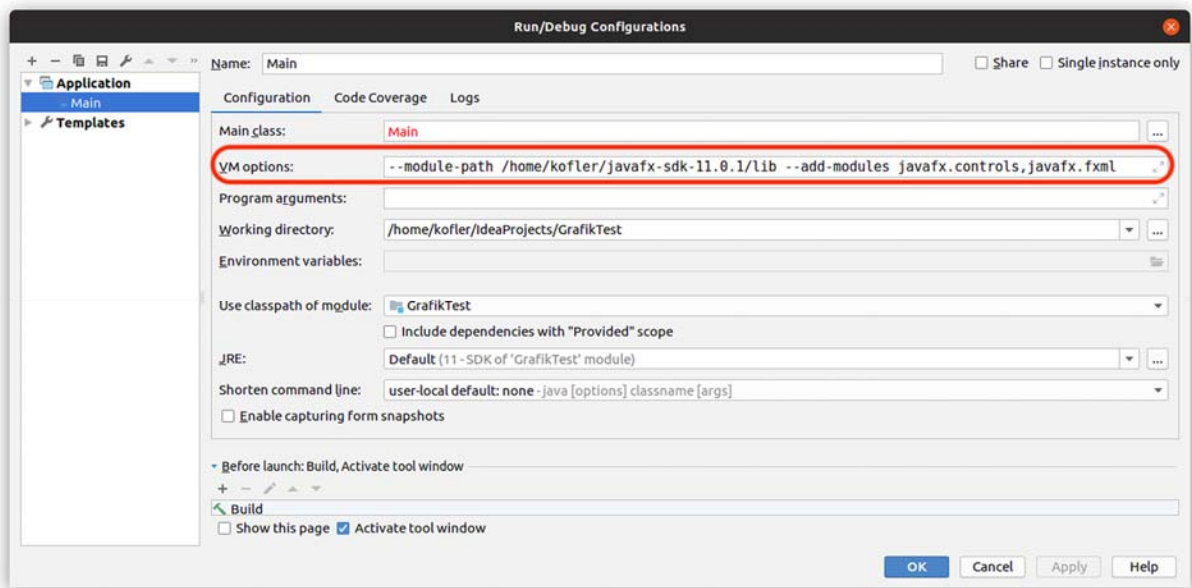
Der Scenbuilder, ein Werkzeug zur Gestaltung von Oberflächen, kann unter <https://www.oracle.com/java/technologies/javafxscenbuilder-1x-archive-downloads.html> heruntergeladen werden. Installieren Sie die Software auf Ihrem System. In IntelliJ müssen Sie im Anschluss den Pfad zu dieser Installation hinterlegen:

1. Im Dialog Settings/Preferences dialog (Ctrl+Alt+S), wählen Sie JavaFX unter Languages and Frameworks.
2. Tragen Sie den Pfad zum Scenbuilder ein und bestätigen Sie mit ok

JavaFX-Projekte einrichten

Damit IntelliJ die JavaFX-Bibliotheken findet und richtig nutzt, müssen zwei Projekteinstellungen verändert werden:

- Im Dialog *File/Project Structure*, Dialogblatt *Modules/Dependencies* wählen Sie mit dem Plus-Button *JARs or directories* aus und geben dann den Pfad zu Ihren lokalen JavaFX-Dateien an. Damit sollte IntelliJ im Quellcode alle javafx-Importe verstehen und in der Lage sein, das Programm zu kompilieren.
- Im Dialog *Run/Edit Configuration* fügen Sie bei den *VM options* die folgenden zwei Optionen hinzu. (Dabei müssen Sie natürlich den Pfad durch den Ort auf Ihrem Rechner ersetzen.)



JavaFX-Projekte erstellen

Erzeugen Sie über den entsprechenden Wizard ein neues JavaFX Projekt und achten Sie dabei darauf, dass auf Ihrem System installierte und für das Projekt verwendet SDK einzustellen.

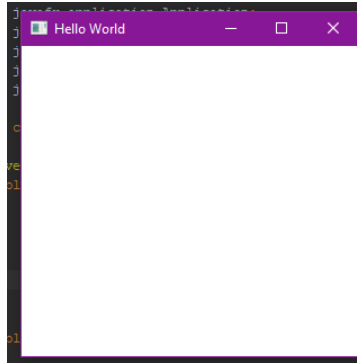


Im src Verzeichnis des neuen Projekts finden Sie drei Dateien:

- Main.java Enthält die Hauptapplikation und dient zum Start des Programms.
- sample.fxml Dies ist die FXML Datei die die GUI Ihrer Applikation beschreibt.

- `Controller.java`. Diese Klasse dient der Kontrolle von Nutzerinteraktionen mit der GUI

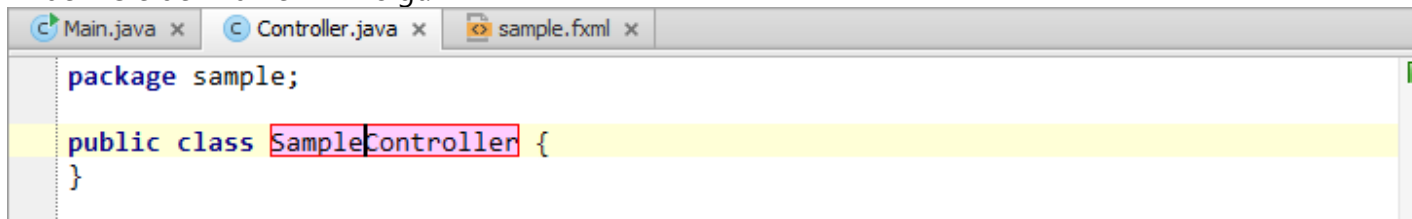
Starten Sie das Programm mittels (run):



Umbenennen der Controller-Klasse

Wir beginnen unser Projekt mit der Umbenennung der existierenden `Controller.Klasse` in „`SampleController`“. Dazu verwenden wir die Refactoring Funktion von IntelliJ:

1. Klicken Sie im Textfenster auf den Klassennamen und wählen Sie: **Refactor** | **Rename** oder nutzen Sie den Shortcut `Shift+F6`
2. Ändern Sie den namen wir folgt:



3. Bestätigen Sie mit `Enter` und wechseln Sie zur Datei `sample.fxml`. Hier sollten die Änderung des Klassennamens bereits vollständig sein.

Erstellen der HelloWorld Applikation

Wir erweitern das UserInterface mit einem Button. Wird dieser angeklickt wird der Text „Hello World“ ausgegeben. Dazu fügen wir zunächst zwei Elemente im Tag „`GridPane`“ ein:

```
<?import javafx.geometry.Insets?>
<?import javafx.scene.layout.GridPane?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<GridPane fx:controller="sample.SampleController"
  xmlns:fx="http://javafx.com/fxml" alignment="center" hgap="10" vgap="10">
  <Button text="Say 'Hello World'" onAction="#sayHelloWorld"/>
  <Label GridPane.rowIndex="1" fx:id="helloWorld"/>
</GridPane>
```

IntelliJ zeigt uns nun an, dass es „sayHelloWorld“ nicht kennt. Gehen Sie zunächst mit dem Cursor auf „hellWorld“ im Label Element, drücken Sie `Alt+Enter` und wählen Sie `CreateField`

aus

```
<?import javafx.geometry.Insets?>
<?import javafx.scene.layout.GridPane?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<GridPane fx:controller="sample.SampleController"
  xmlns:fx="http://javafx.com/fxml" alignment="center" hgap="10" vgap="10">
  <Button text="Say 'Hello World'" onAction="#sayHelloWorld"/>
  <Label GridPane.rowIndex="1" fx:id="helloWorld"/>
</GridPane>
```

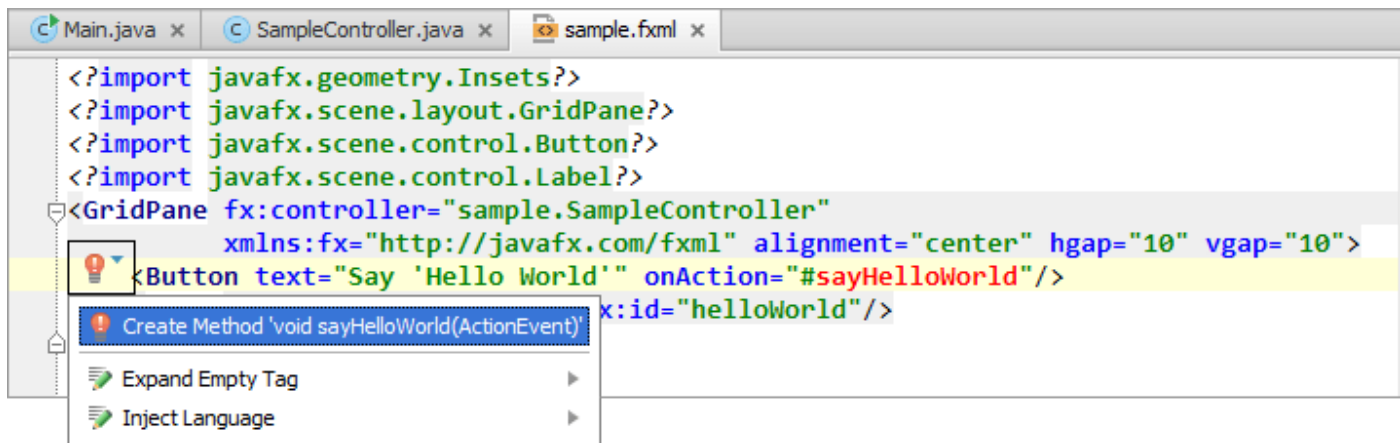
IntelliJ IDEA wechselt nun zu `SampleController.java` und zeigt die eingefügte Deklaration des Feldes.

```
package sample;

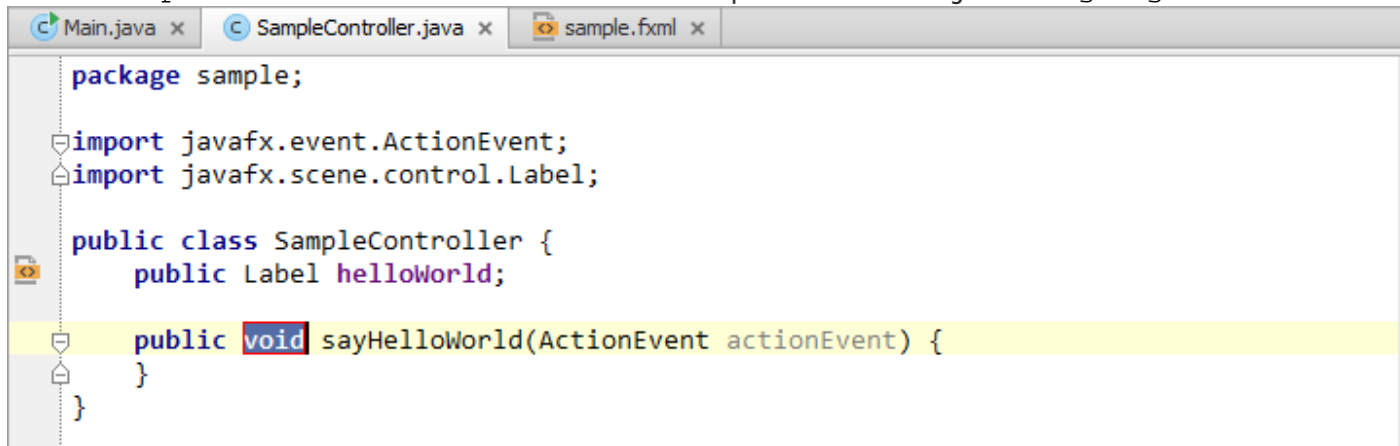
import javafx.scene.control.Label;

public class SampleController {
  public Label helloWorld;
}
```

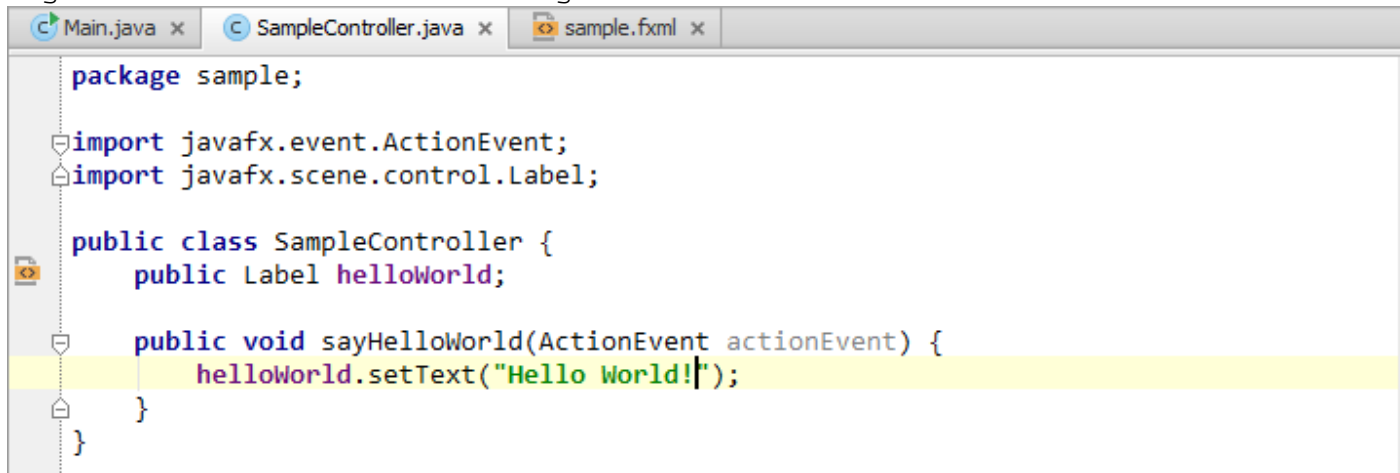
Verfahren Sie analog mit „sayHelloWorld“



Eine entsprechende Deklaration wird in SampleController.java eingefügt:



1. Ergänzen Sie die Methode wie folgt:



Sie können die Applikation nun ausführen und testen.

