# <u>Lab – Java - JavaFX - TableView</u>

#### Overview

Write a JavaFX application that uses a TableView. The TableView will display TV show data.

## **Create Project**

- 1. Create a new JavaFX project.
  - Choose JavaFX on the left.
  - Choose Maven for the Build system.

Vew Project		Х
Q		
New Project	Name:	demo1
Empty Project	Location:	~\IdeaProjects
Generators		Project will be created in: ~\IdeaProjects\demo1
Maven Archetype		Create Git repository
🐂 JavaFX	Language:	Java Kotlin Groovy
Kotlin Multiplatform		
Compose Multiplatform	Build system:	Maven Gradle
IDE Plugin	Group	com example
🐱 Android	oroupi	control on pre
	Artifact:	demo1
	JDK:	The Oracle OpenJDK version 16
(?)		Next Cancel

## Create TVShow Class

Create a class named TVShow. It should have the following member variables: name (String), id (int). Write get/set methods for all member variables.

## Add TableView to GUI

Update the primary.fxml GUI file.

1. Remove the Label and Button controls that are given by default.

- 2. Add a TableView to the VBox.
- 3. Set the fx:id of the TableView to tableViewShows.
- 4. Setup TableColumns in the TableView to display TV show data (columns should match the variables on the TVShow class).
  - a. Make sure to set the fx:id for each TableColumn.

Here is a screenshot of what the window should look like:

1			-	×
	Name	ld		
		No content	in table	

#### Populate TableView with Data

Update the PrimaryController class.

- 1. Add the necessary member variables to the PrimaryController class to enable using the TableView and TableColumns in code. Make sure they all have the @FXML annotation.
- 2. Add an initialize method with the following header: public void initialize()
- 3. Add code to associate TableColumns with member variables on the TVShow class. Add this code to the initialize method.
- 4. Add code to the initialize method that will create two instances of TVShow and add them to the TableView. Here is the data to add (name followed by id): ("What We Do in the Shadows", 1) and ("Superstore", 2).

Here is a screenshot of what the window should look like:

-		
la	ł	
1		
2		
		- 0

## Allow Users to Add Shows

Update the GUI so that the user can add new shows to the list. You will need to add two Labels, two TextFields, and a Button to the GUI. When the user presses the Add Show button it should add the show data they typed in to the TableView. The TextFields should be cleared automatically after adding the new show to the TableView. Following are screenshots of what it should look like after adding "Silicon Valley":

- 🗆 X		• –	I
ld	Name	Name	Name Id
Shadows 1	What We Do in the Sh	What We Do in the Shadows	What We Do in the Shadows 1
2	Superstore	Superstore 2	Superstore 2
	Silicon Valley	Silicon Valley 3	Silicon Valley 3
V Show Name	New TV SI	New TV Show Name	New TV Show Name
TV Show Id	New TV	New TV Show Id	New TV Show Id
dd Show	Add	Add Show	Add Show

#### Part 3

Update the GUI so that the user can clear the TableView. Add a Clear button below the Add TV Show button. Following are screenshots before and after the Clear button is pressed:

- 1		×			-		
Name	ld			Name		ld	
What We Do in the Shadows	1						
Superstore	2			No conten	t in table		
New TV Show Nam	ıe			New TV Sho	ow Name		
New TV Show Id				New TV S	how Id		
Add Show				Add S	how		
Clear				Clea	ar		

#### Part 4

Add code to handle a TableView item selection. When a TableView item is selected it should print a message in the console window. Here is what should be displayed in the console window when Superstore is clicked:

Selected Item: TVShow{name='Superstore', id=2}