# GUI with JavaFX – A Movie GUI

## About:

You can reuse a lot of your code from Lab 8 and/or 10. We will build a GUI to enter movies and display some of the information about each movie below. You will use Scene Builder to create an FXML file that will specify the layout of GUI. You should also be sure to catch any errors so your program does not crash. When catching an error be sure to display a message to let the user know it failed and try again.

## Things to Consider:

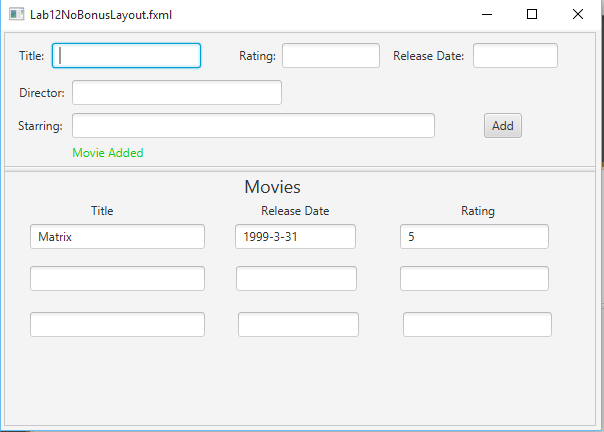
* User must enter all information about the movie in order for it to be accepted. Modify your exception to account for this. Only minor modification needed to lab 10 Movie class
* Look at your control options choose controls that make things easier
* Do your layout in scene builder first and set your fx:id’s so you can use the preview controller skeleton to get a good start on your controller
* I was able to do this lab, with bonuses, writing about 80 lines of code – doesn’t include auto generated skeleton and minor adjustments to main application class copied from KilometerConverter.java
* Remember your controller class will do most of the work.

## The Challenge:

Design a GUI, ideally like the below image, that accepts input for each of the required movie elements. If the user does not enter all the fields with valid information then display an error in red on the GUI saying there was an error and leave all fields as the user entered them. If the user enters valid input for each then display movie added message in green and display the elements, like below image, for each movie only once and clear the input fields for the user to enter another movie. Design your layout so the user can enter 3 valid movies that get displayed. You must keep all your movie objects in an ArrayList with all the data even though we don’t display each field. The GUI should be 600 wide and 400 tall.

You will have at least 5 files to submit with this lab; the main application class – Lab12.java, your fxml file – Lab12.fxml, your Movie class – Movie.java, your application controller – Lab12Controller.java, and your PreFilmDate exception class – PreFilmDate.java. **DO NOT SUBMIT YOUR** .class files or anything else. When we check your code we will compile Lab12.java and Lab12Controller.java then to execute we will do java Lab12. Be sure yours runs like this.

## Samples:



## Submission:

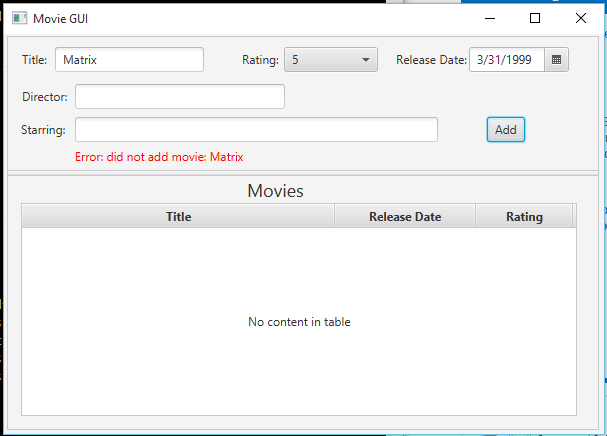
Zip the required source files, your .java files and .fxml file only, in a .zip folder. You must use a .zip extension when zipping your files. The zip folder should be called Lab12LastNameFI.zip. Do not submit additional files. Submission must be completed by 23:59:00 on the due date and uploaded to CourseWeb. No email submissions will be accepted.

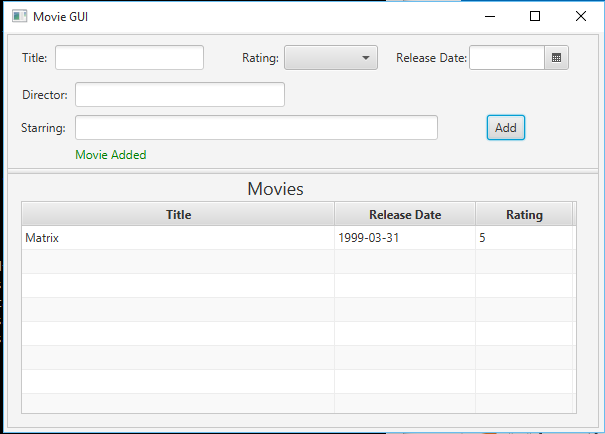
## Bonus Points(2): Must be your own work no TA help

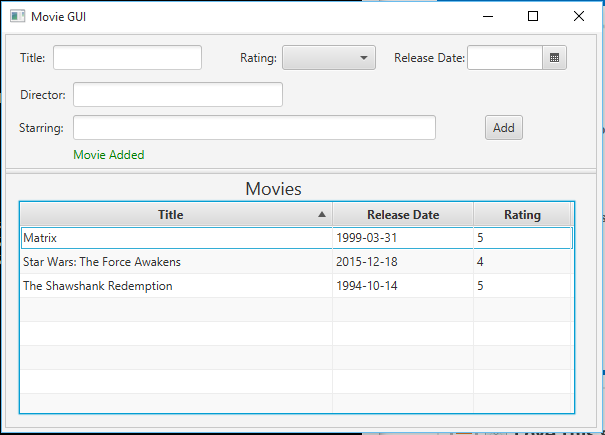
For the first bonus point make your input section a little better. For the rating make a drop down, aka combobox, with the values 1-5 and the release date a date picker. You can see this below.

For bonus point 2 make it so an unlimited number of movies can be added and displayed by using a tableview on the bottom section of the GUI. This table view can sort by any of the columns by clicking on it. See below

Hint – You will likely need to do some stuff in your initialize to make this work.







## Grading:

4 – Does it compile and execute and show you are close

4 – Does it provide the correct results accounting for any input checking formatting etc.

2 – Proper labeling, comments, followed submission instructions, and is it readable.

1 – Uses drop down for rating and date picker for release date \*bonus

1 – Uses TableView to display movies allowing for sorting on each row \*bonus