

Exercise 10 – UI – JavaFX

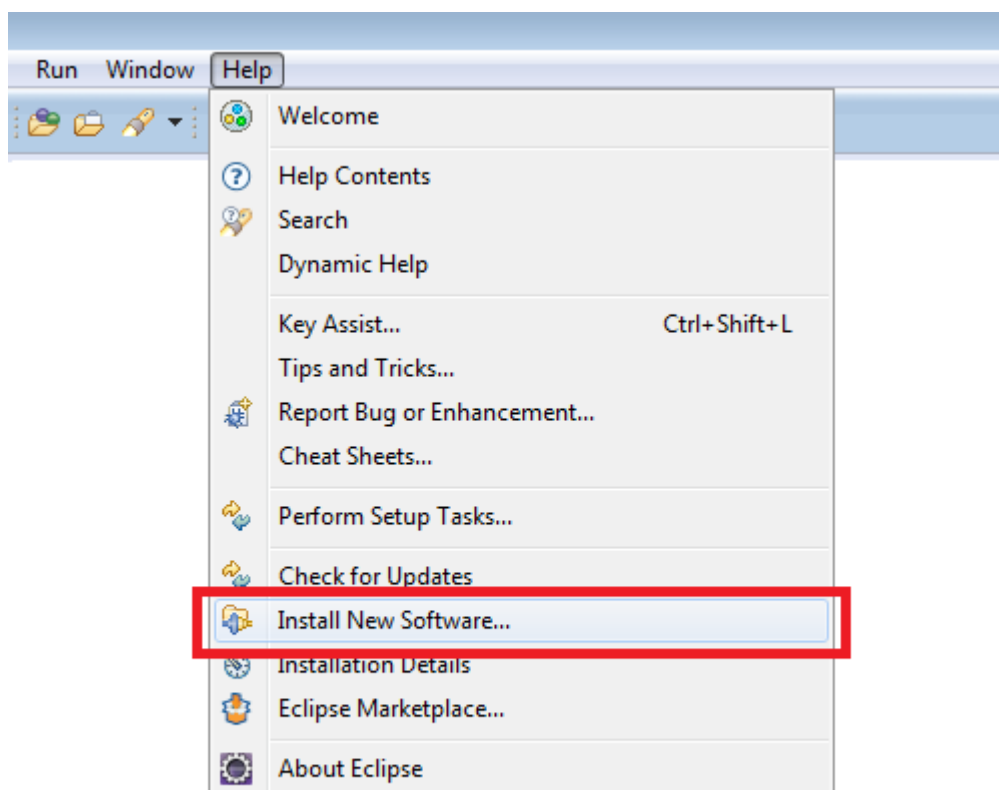
Objective

By the end of this exercise you should be able to understand the installation procedure of JavaFX plugin and comfortable with designing and building UIs using the JavaFX libraries. These exercises are a lot more free form and leave scope for you to explore the UI elements yourself and understand how to include them in your UIs. Not everything mentioned will have been covered in the slides!

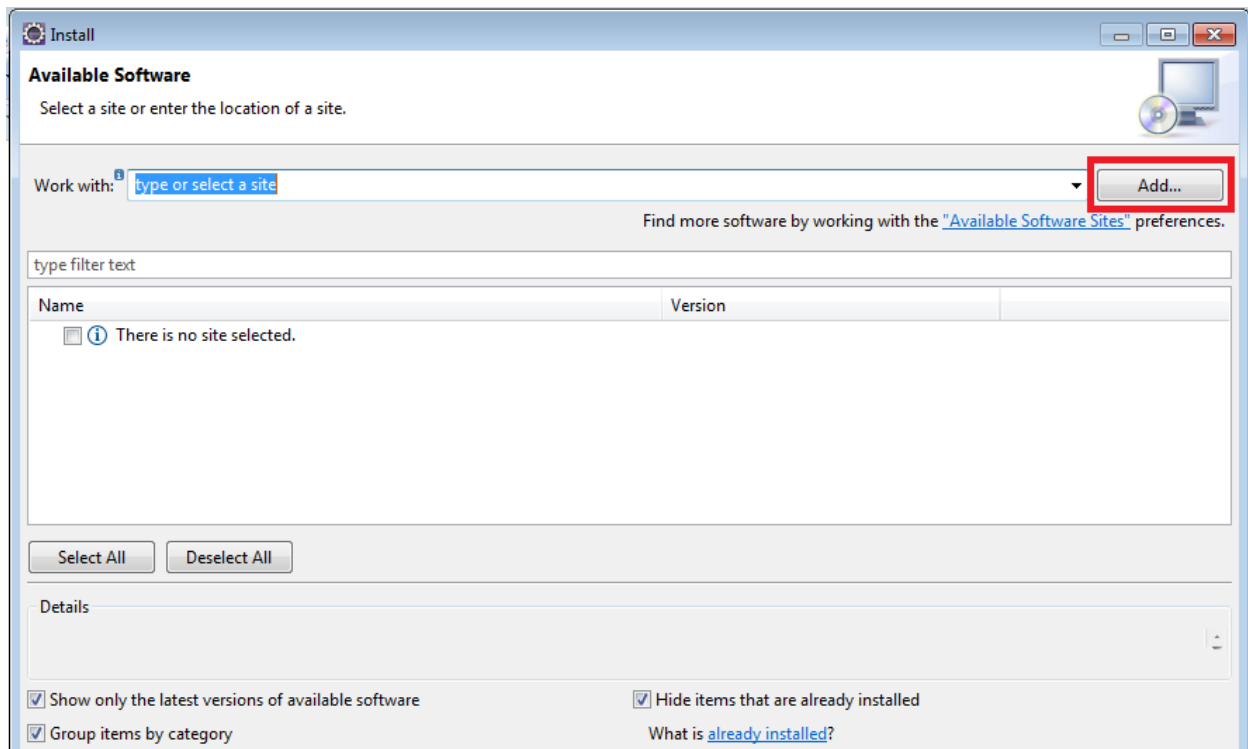
Overview

Part A: Installing JavaFX plugin

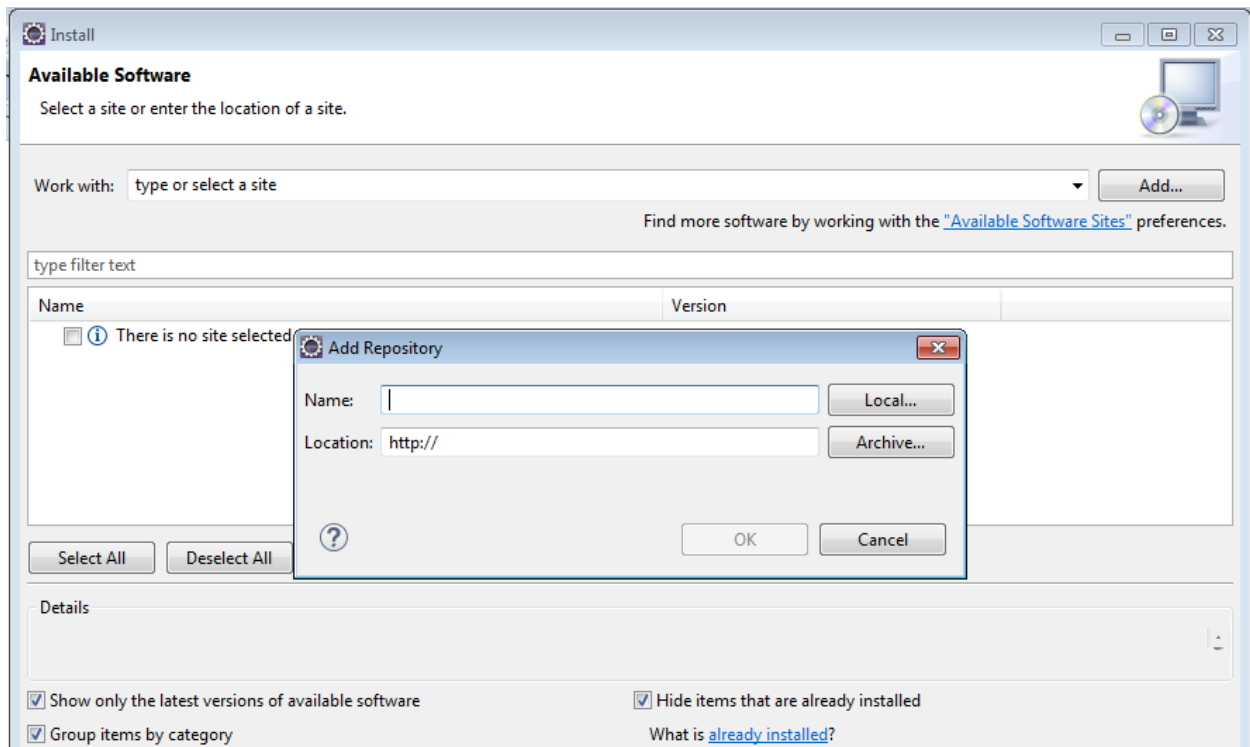
JavaFX plugin can be installed using the “Install New Software” wizard and to launch this wizard you should use help menu.



Once you select “Install New Software”, you should see a new window as the following



Click on Add, and you should see a screen like the following.



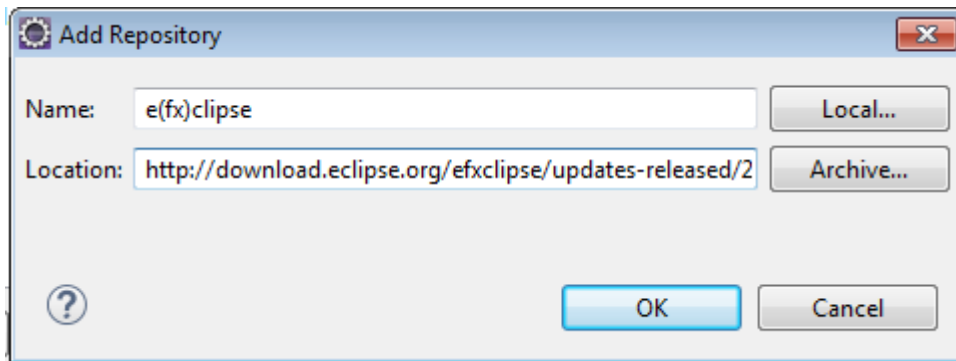
Enter values for name and location fields

Name : e(fx)clipse

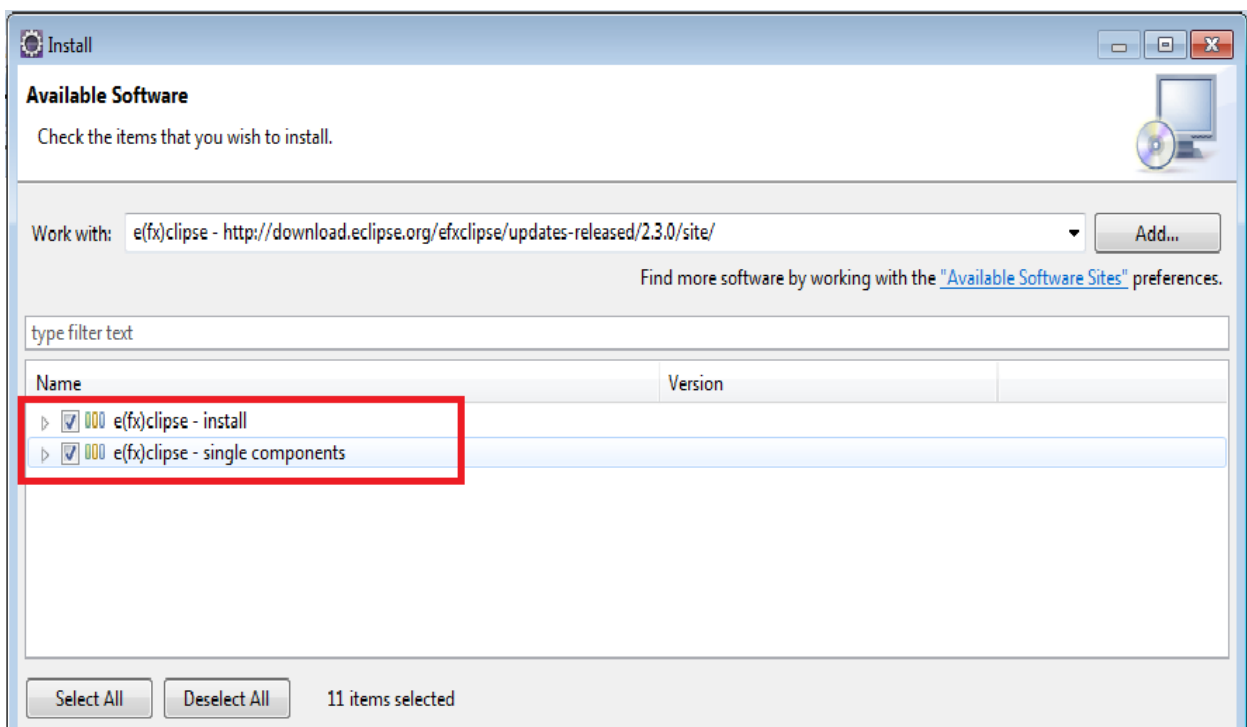
Copy the given name and paste it in name field

Location : <http://download.eclipse.org/efxclipse/updates-released/2.3.0/site/>

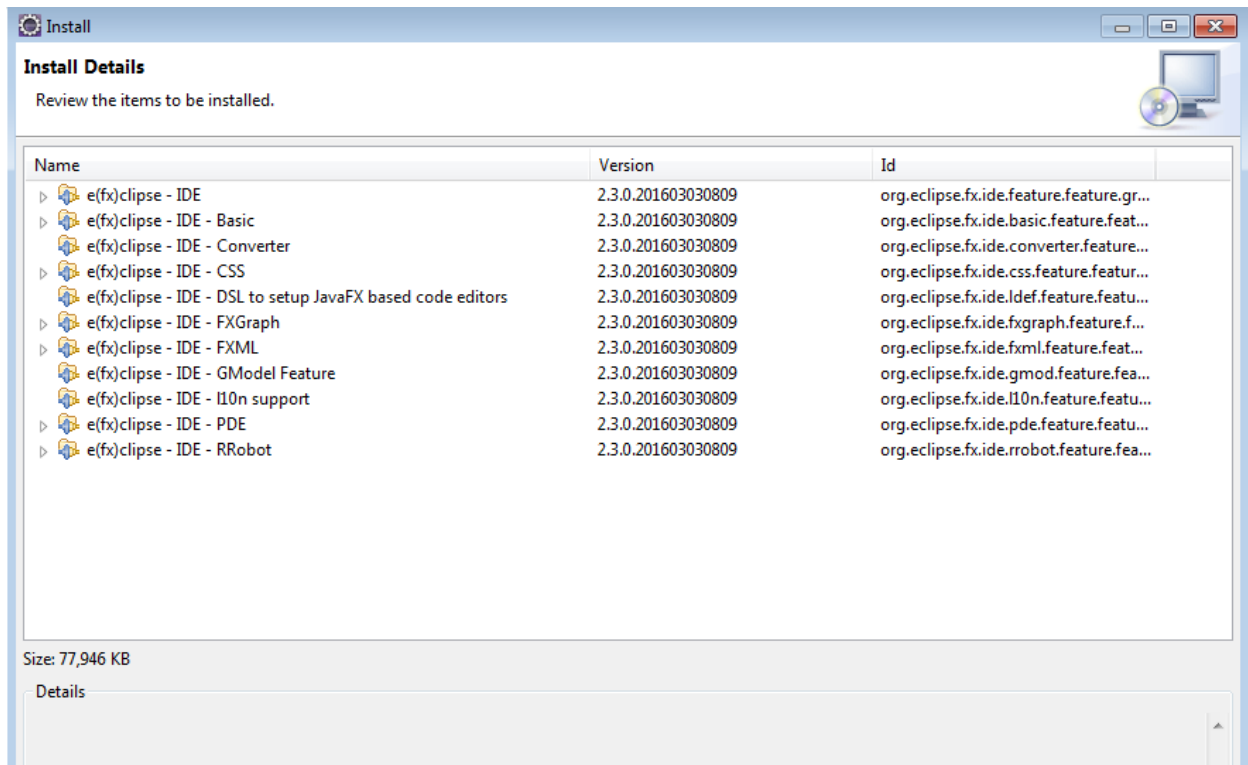
Copy the given link and paste it in location field



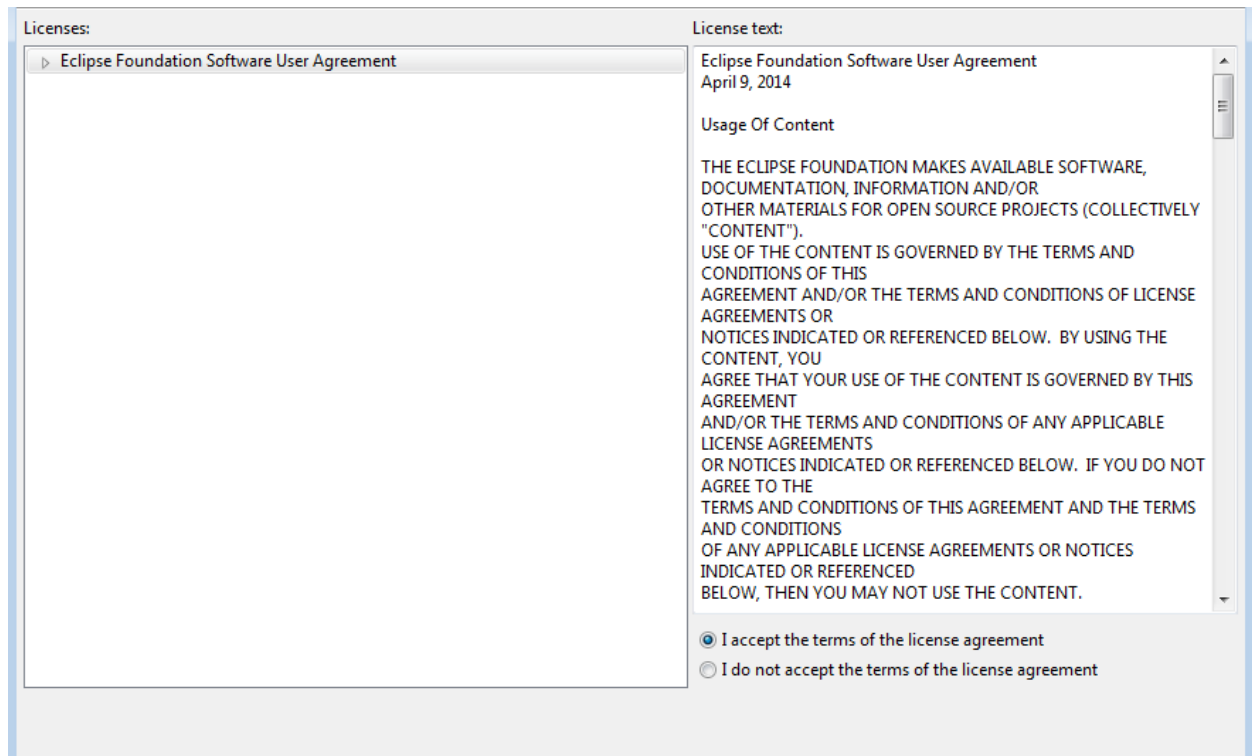
Click on OK button



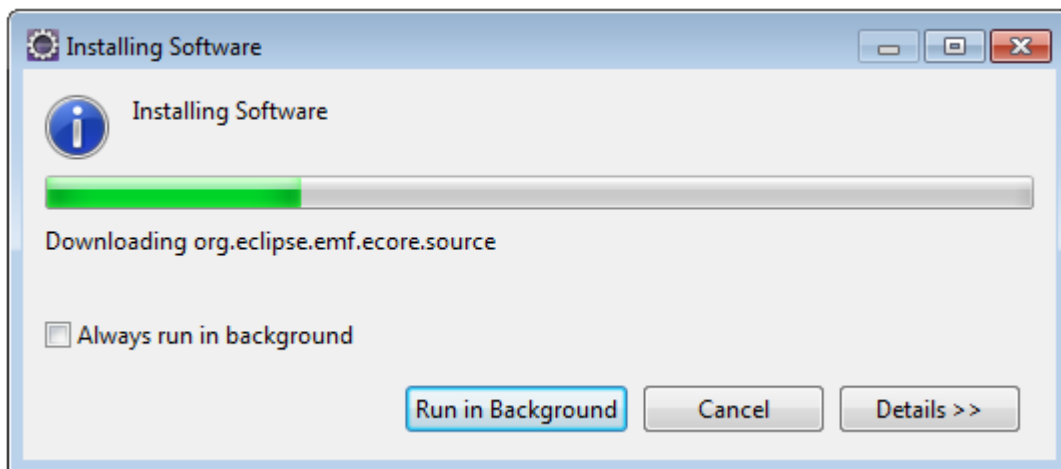
Select all installable features as shown above and click and **Next** button

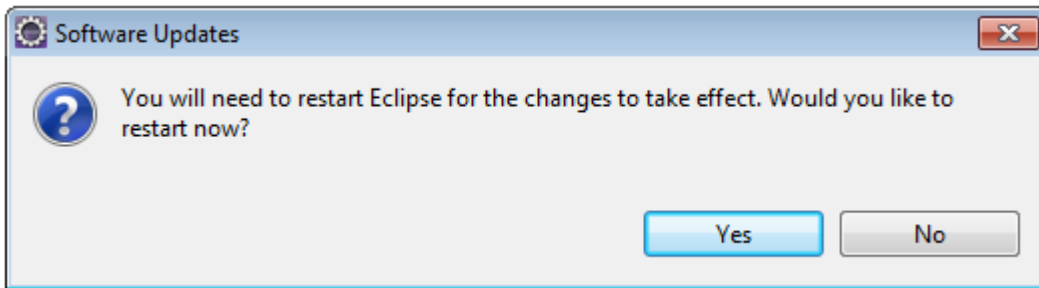


You must accept the licence agreement and click on **Install** button



Let Eclipse IDE do its job



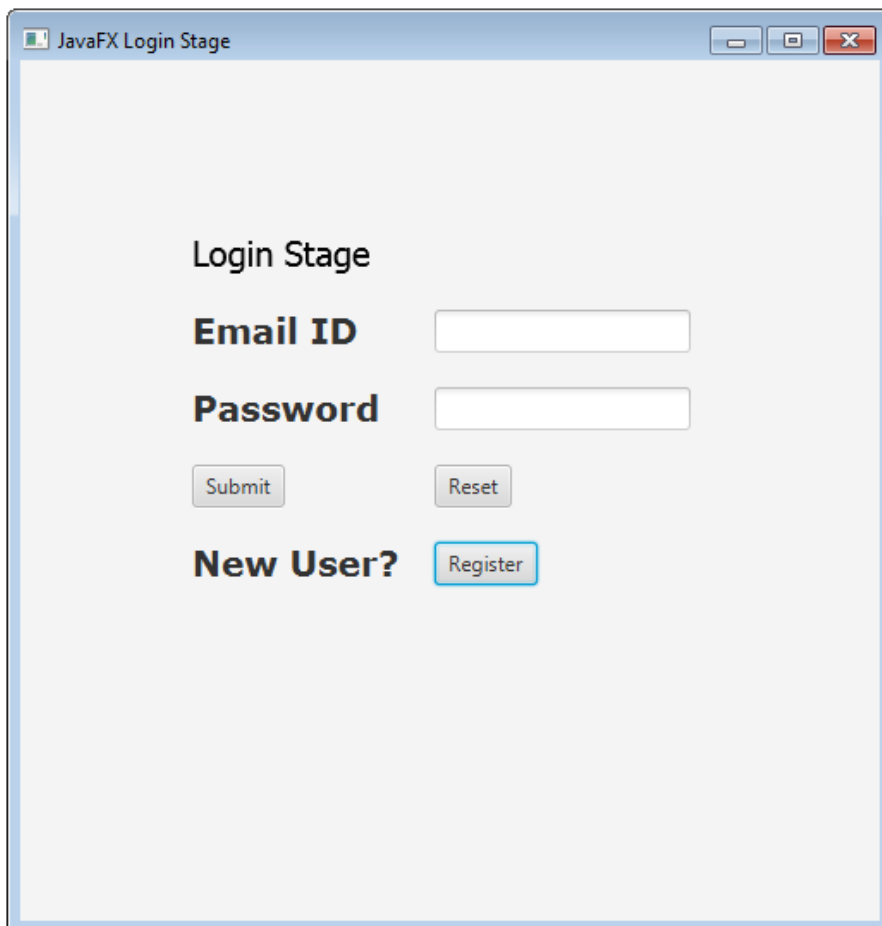


When requested, restart Eclipse.

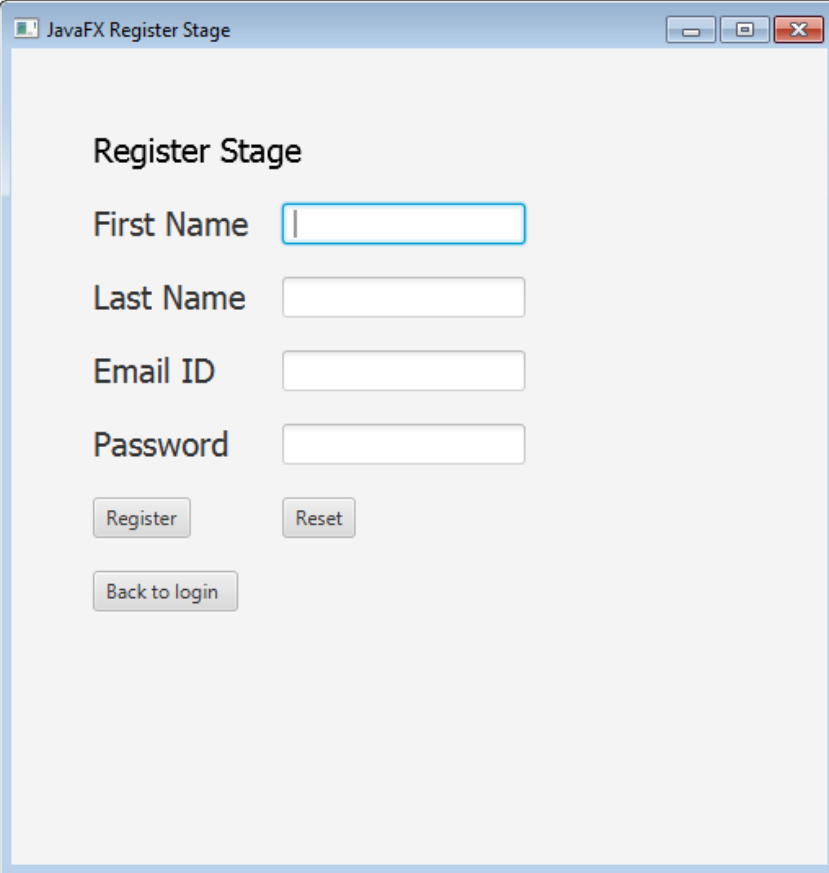
Part B: Building User Interfaces

You have been given the following image to turn into a java program. For now we are not concerned with the buttons or menu options doing anything when we click on them.

Login Screen



Register Screen



The screenshot shows a JavaFX window titled "JavaFX Register Stage". Inside the window, the text "Register Stage" is displayed at the top. Below it, there are four input fields: "First Name", "Last Name", "Email ID", and "Password". The "First Name" field is currently active, showing a cursor. Below the input fields, there are three buttons: "Register", "Reset", and "Back to login". The "Register" and "Reset" buttons are positioned side-by-side, and the "Back to login" button is centered below them.

Implement this UI in Java.

Useful components to look up are:

Label, TextField, PasswordField, Button, GridPane (Layout Manager)

Think about how to use JavaFX Application, Stage, Scene and layouts to control the flow of your UI.

Part C: Responding to events

The UI you've built now needs to have events added to it. The final program should be able to allow users to register and login.

Register Page/Stage Functionality:

Register Button

1. Should capture the input values(first name, last name, email and password) and format a string object with comma-separated values (CSV) as the following format

firstname,lastname,email,password

For Example

David,Beckham,david@gmail.com,1234

2. Validate the input values (Input values must not be empty)
3. Create a text file (**userdata.txt**) and write the above CSV into **userdata.txt** (You should recall the concepts of file handling)

Reset Button

4. Should reset/clear all the form fields (textfields)

Back to Login Button

5. Should re-direct the user to Login screen

Login Page/Stage Functionality:

Login Button

6. Should capture the users inputs (email and password) and store them in local variables
7. Read **userdata.txt** file content into StringBuffer object (use append method)
8. Split the string value based on delimiter – comma (,) – **Hint** use split method on string object
9. Validate the user – if email and password entered by the user match with **userdata.txt** values output the following statement

System.out.println("Valid User");

If input values mismatch, output the following statement

System.out.println("In-Valid User");

Reset Button

10. Should reset/clear all the form fields (textfields)

Register Button

11. Should re-direct the user to Register screen

Glossary of key terms**JavaFX**

A graphics framework for creating GUIs in Java applications

JavaFX Stage

It is the main window of JavaFX on which every component resides

JavaFX Scene

It is the placeholder for all JavaFX components (Button, Label, TextField etc)

JavaFX Layouts

JavaFX layout managers tell Java how to organize the components to Stage or Scene

GridPane

This layout represents a table structure. It can specify horizontal and vertical spaces

Events

Events occur when something happens in your program, actions like mouse is moved, button is clicked and text is changed

Event Listener

When events are triggered the appropriate listener will respond to the Action Command passed to it

ActionListener

It responds to button click events

JavaFX Event Handler

An event handler is an implementation class of a specific event type. JavaFx provides an interface named EventHandler which can handle all sorts of JavaFX events