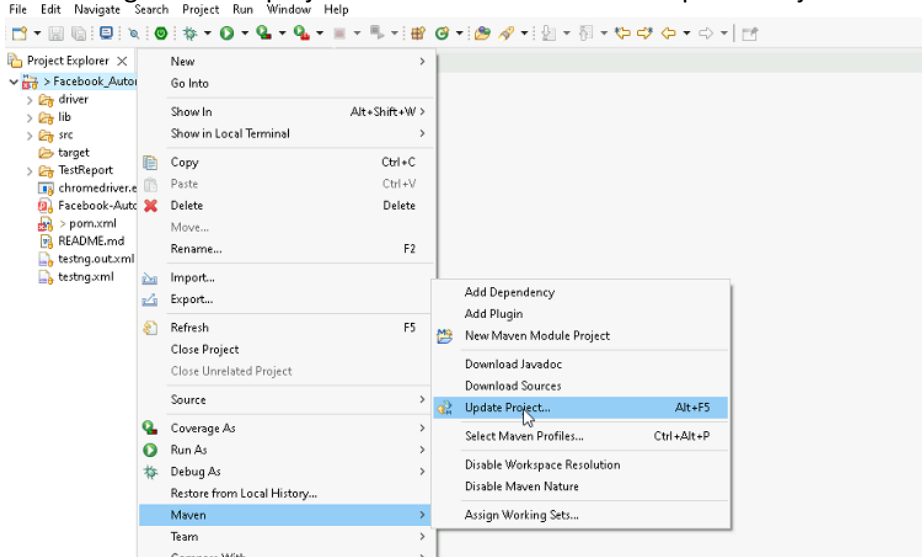


FACEBOOK AUTOMATION

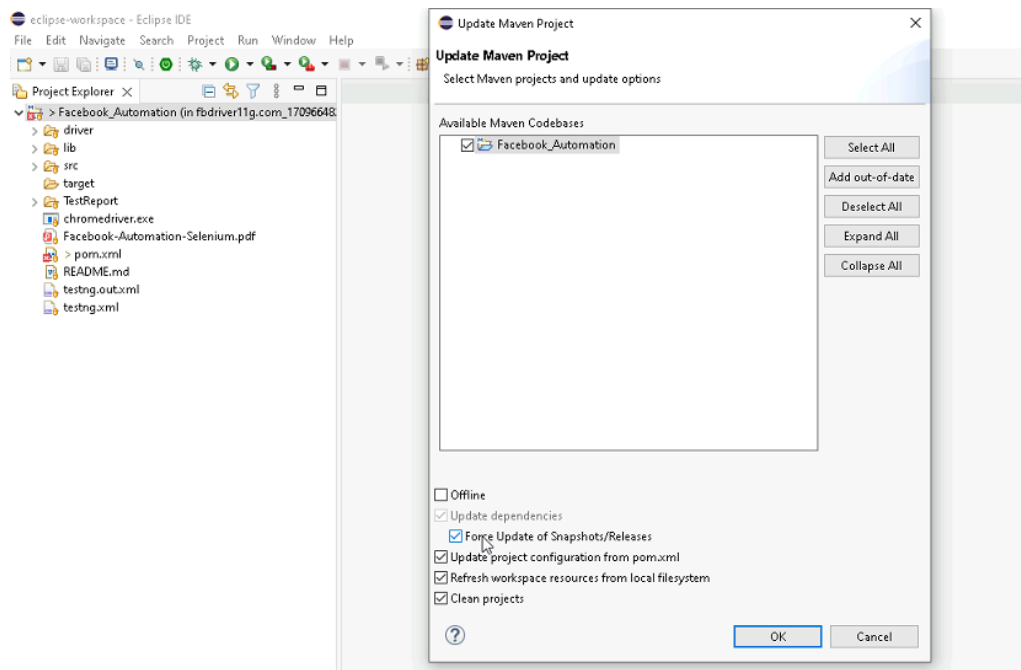
Pre-requisite:

As soon as you import project in eclipse, update the project using maven update option as below. This is to resolve issue if any maven dependency not downloaded properly:

1. Right click on project : Go to “Maven” : Select “Update Project”



2. In Update Maven Project Box Select “Force Update of Snapshots/Releases” and click OK



Template Code Structure:

- a. Below are the packages and files you will be required to work upon.
- b. Other Files and packages you can ignore.
- c. In other Files and packages do not do any changes. It would affect your evaluation.
- d. You are not required to work in “Test” Folder. Files there are non-editable. Editing those files and trying to save them will throw error and would affect your evaluation.

Package	Class/File	Description
src/main/java/coreUtilities/utis/	FileOperations.java	<ol style="list-style-type: none">1. Contains methods to read from json file.2. Method is in templated form.
/src/main/java/pages	FaceBookPage.java	<ol style="list-style-type: none">1. All core activities (mentioned in list above) to be performed here.2. The comments associated with each templated method here describe the expectation.3. You can define locators and xpath here.4. Declare any variable/object you need to share data/status between different methods.5. Do not modify the signature of methods declared here.6. You can create additional supportive common methods in CommonEvents class.
/src/main/resources/	Config.json	URL to navigate to. Already URL is defined here
	expected_data.json	Contains data to fill in form
/src/main/java/coreUtilities/utis	CommonEvents.java	<ol style="list-style-type: none">1. Contains all common activities.2. Certain templated

		common method declared here. 3. You implement them as per your need. 4. You can add any additional method for common activity here
	Testng.xml	Execution needs to kick started from TestNG xml

PROBLEM STATEMENT

Need to automate the following activities using Selenium + Java.

Key Activities to implement:

SN	Test Steps
1	Navigate to the URL (https://www.facebook.com) and Validate the facebook LogIn Page Title
2	Verify Presence Of All Fields in The Login Page
3	Validate the error message, when user does not provide any credentials
4	Enter Email or Phone No in Email or Phone No Text field and click on Login and validate Error Message
5	Enter Password in password Text field and click on Login and validate Error Message
6	Go Back To LogIn Page And Validate CreateNewAccount button Is Present or not
7	Click on Create Account Button and Verify navigate to the signUp page.
8	Verify first name,surname,Mobile Number or email address,New password field are present in the signup page or not
9	Enter data(Read the data from Json File) in the first name field
10	Enter data (Read the data from Json File)in the surname,Mobile Number email address,New password field.
11	Select any date from date dropdown field and and verify the date dropdown is present or not
12	Select any Month from month dropdown field and and verify the Month dropdown is present or not
13	Select any Year from the Year dropdown field and and verify the Year dropdown is present or not
14	Close the "signup" page and verify the All Fields prsence in The Login Page or not
15	Enter valid email or phone no in email or phone no text field and enter password in password textfield and validate "login" button is present or not

NOTE: "Please do not delete any file in the src folder. But you are free to add any other file".

Expectations:

- 1) Learners should write automation script using Java and selenium to automate all the steps in the above question. In other words, automation script should perform all mentioned steps.

- 2) Learners should not use any tools to create the xpath. They should develop the xpath/cssselector on their own.
-

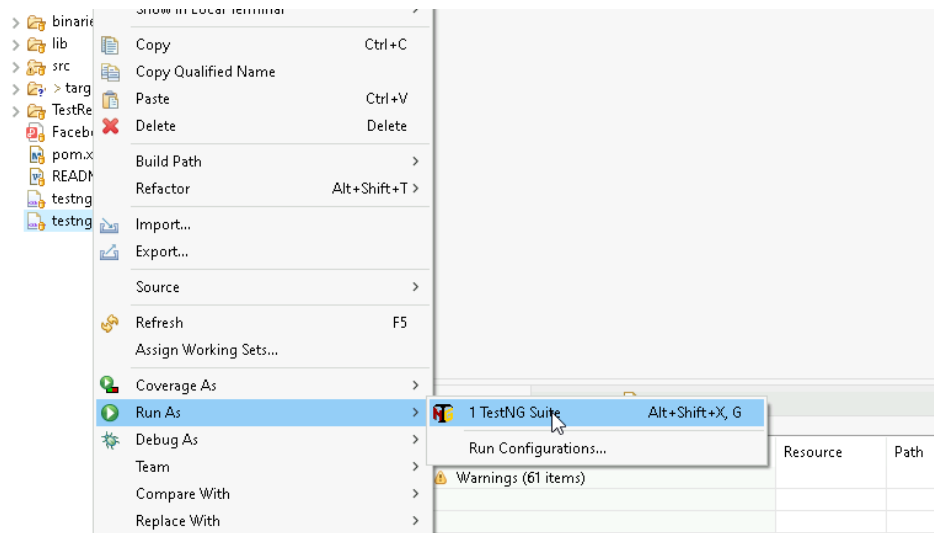
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS

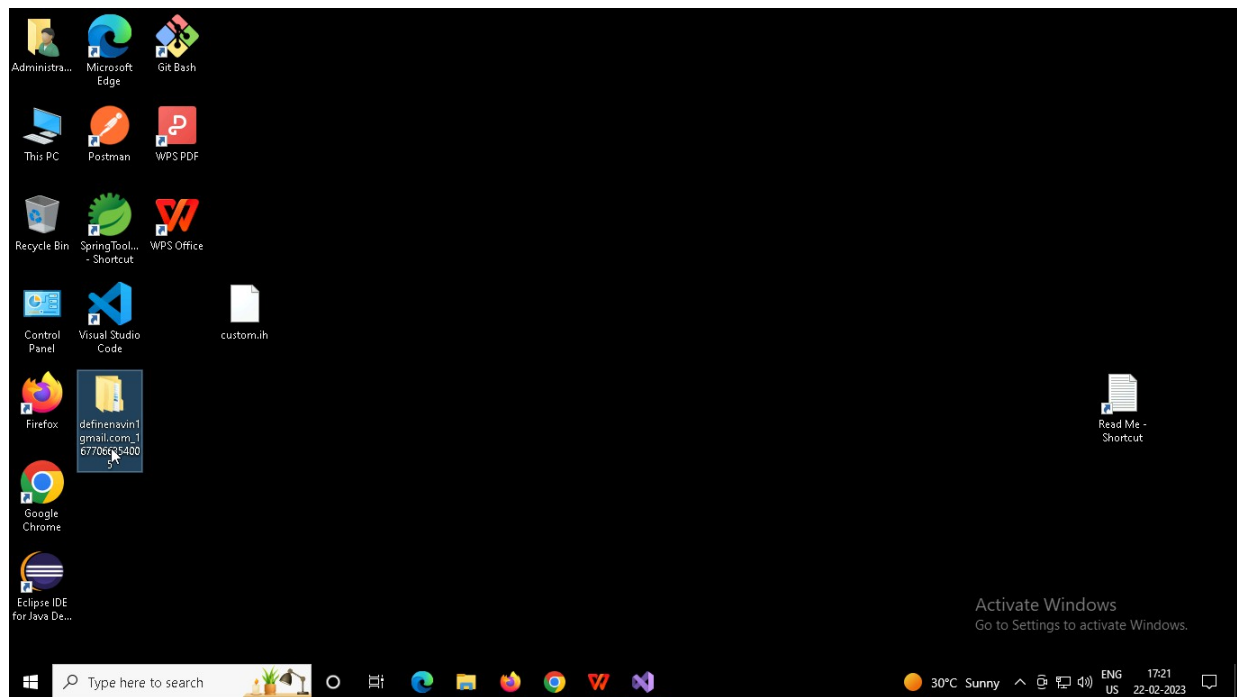
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

EXECUTION STEPS TO FOLLOW

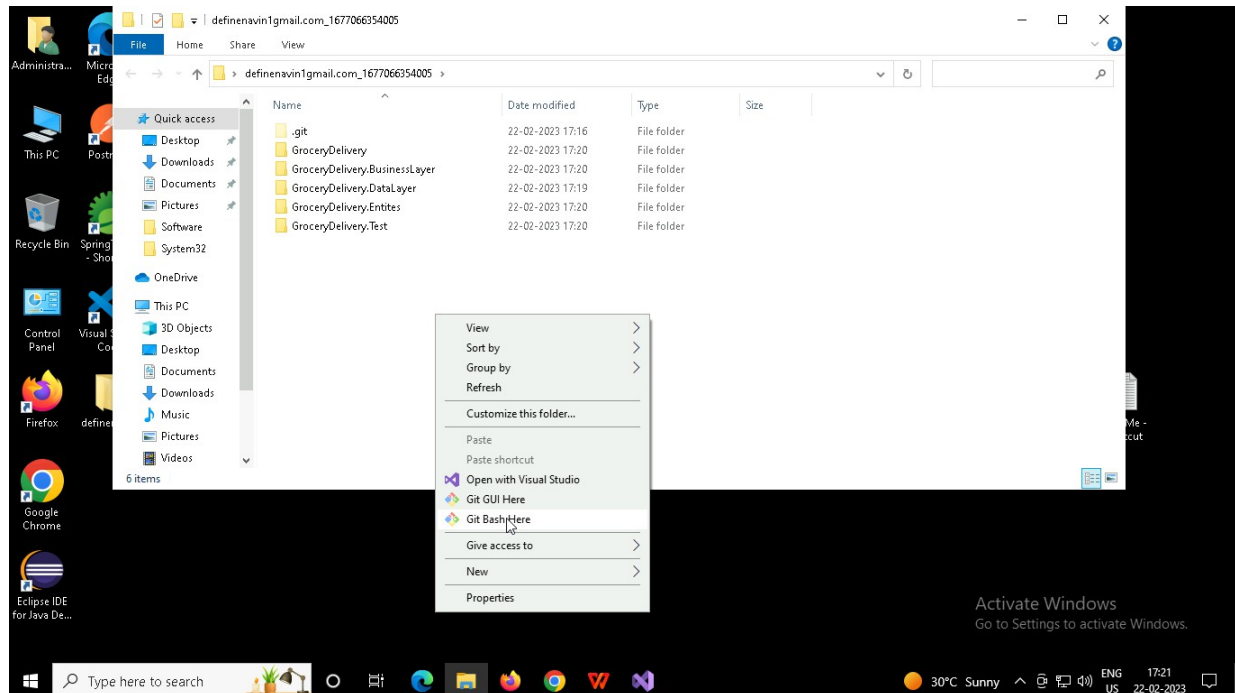
1. You are mandatory required to run test cases for applications before final submission. Without which project evaluation will not happen.
2. You can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite



3. Before final submission, you are also required to push your code to GIT. Following are the steps to follow:

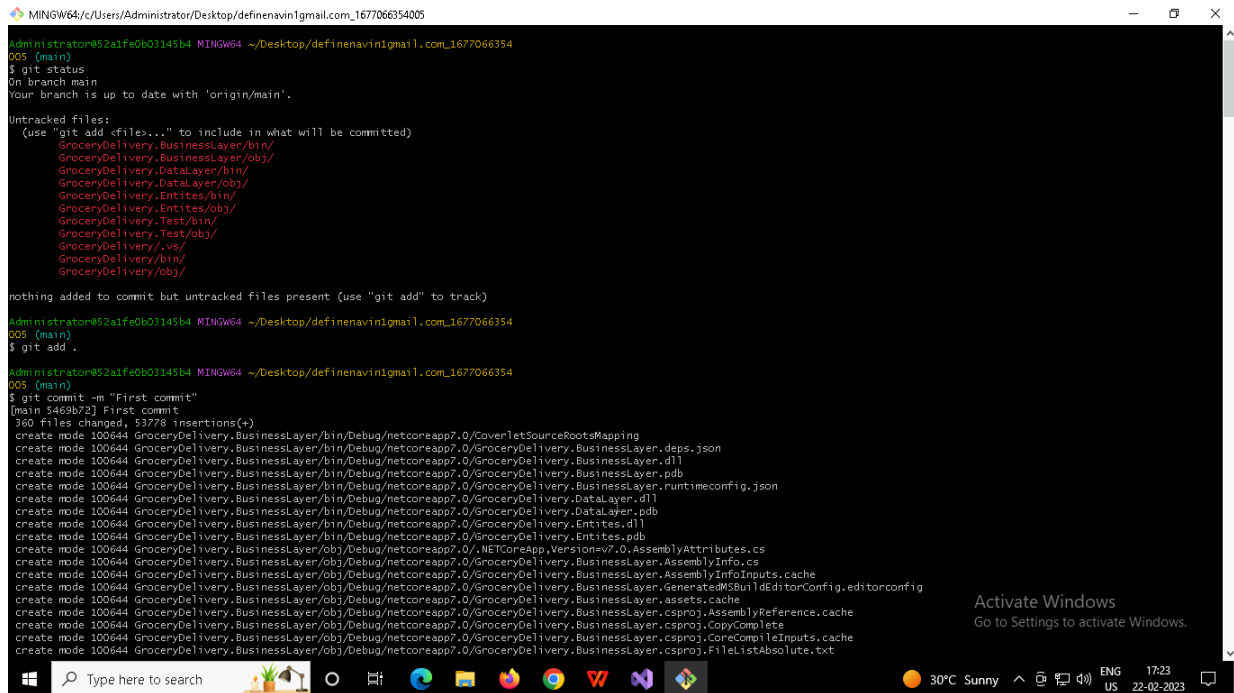


Right click in folder and open Git Bash



In Git bash terminal, run following commands

- git status
- git add .
- git commit -m "First commit"
(You can provide any message every time you commit)
- git push



```
Administrator@52a1fe0b03145b4 MINGW64 ~/Desktop/definavin@gmail.com_1677066354
005 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    GroceryDelivery.BusinessLayer/bin/
    GroceryDelivery.BusinessLayer/obj/
    GroceryDelivery.DataLayer/bin/
    GroceryDelivery.DataLayer/obj/
    GroceryDelivery.Entities/bin/
    GroceryDelivery.Entities/obj/
    GroceryDelivery.Test/bin/
    GroceryDelivery.Test/obj/
    GroceryDelivery/.vs/
    GroceryDelivery/bin/
    GroceryDelivery/obj/

nothing added to commit but untracked files present (use "git add" to track)
Administrator@52a1fe0b03145b4 MINGW64 ~/Desktop/definavin@gmail.com_1677066354
005 (main)
$ git add .
Administrator@52a1fe0b03145b4 MINGW64 ~/Desktop/definavin@gmail.com_1677066354
005 (main)
$ git commit -m "First commit"
[main 5469b72] First commit
360 files changed, 53778 insertions(+)
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/CoverletSourceRootsMapping
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.deps.json
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.dll
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.pdb
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.runtimeconfig.json
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.DataLayer.dll
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.DataLayer.pdb
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.Entities.dll
create mode 100644 GroceryDelivery.BusinessLayer/bin/Debug/netcoreapp7.0/GroceryDelivery.Entities.pdb
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/.NETCoreApp,Version=v7.0.AssemblyAttributes.cs
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.AssemblyInfo.cs
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.AssemblyInfoInputs.cache
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.GeneratedMSBuildEditorConfig.editorconfig
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.assets.cache
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.csproj.AssemblyReference.cache
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.csproj.CopyComplete
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.csproj.CoreCompileInputs.cache
create mode 100644 GroceryDelivery.BusinessLayer/obj/Debug/netcoreapp7.0/GroceryDelivery.BusinessLayer.csproj.FileListAbsolute.txt
```

