

FORM AUTOMATION

PROBLEM STATEMENT

Need to automate the following activities using Selenium + Java.

Key Activities to implement:

1. Navigate to <https://demo.automationtesting.in/Register.html>.
2. Retrieve the title of the page. The title of the page should be "Register".
3. Click SwitchTo à Alert Link.
4. Validate if control is navigated to new page
5. Click on button to display the alert box
6. Validate if alert popup is shown.
7. Close the popup.
8. Click on Register menu option to navigate to Register page.
9. Fill the form with data provided in json file (expected_data.json).
10. Click on the country dropdown.(Second One)
11. Select each country option one by one.
12. Validate if each country option is selectable from the dropdown.
13. Check and uncheck each hobby checkbox.
14. Validate that the checkboxes are responding correctly to user interaction, allowing selection and deselection.
15. Select each radio button option for gender.
16. Validate that only one radio button option should be selectable at a time.
17. Enter passwords in the "Password" and "Confirm Password" fields according to different scenarios (matching, non-matching, pattern validation).
18. Select different dates from the Date Of Birth fields.
19. Validate that dates are selectable, and the selected values should be captured accurately.
20. Click on the image upload button.
21. Choose an image file from the file system.
22. Validate that the selected image should be displayed on the page after upload.
23. Leave the phone number empty.
24. Attempt to submit the form.
25. Verify that error messages for incomplete field displayed as appropriate.
26. Click on "SwitchTo" Tab, Click on Window and then Click on Tab.
27. Check of a new Tab is opened.
28. Click on "SwitchTo" Tab, Click on Window and then Click on Open New Separate Window
29. Check if a new Window is opened
30. Click on "SwitchTo" Tab, Click on Window and then Click on Open Separate Multiple Window.
31. Check if multiple windows are open.
32. Click on "SwitchTo" Tab, Click on Frame and then click on Single Frame and Pass the Value In Textbox
33. Click on "SwitchTo" Tab, Click on Frame and then click on Frame within Frame and Pass the Value In Textbox
34. Click on "Widgets" Tab, Click on Accordion, Select Any Groups And Fetch Data
35. Click on "Widgets" Tab, Click on Auto Complete and Pass Value
36. Click on "Widgets" Tab, Click on Slider and Perform Maximize Minimize activity
37. Click on "Interactions" Tab, Click on DragAndDrop, Click on Static and Perform Drag And Drop Action
38. Click on "Interactions" Tab, Click on DragAndDrop, Click on Dynamic and Perform Drag And Drop Action

Template Code Structure: Packages and files you will be required to work upon. You can avoid other files and packages

Package	Class/File	Description
pages	StartupPage.java	Initializing the WebDriver and navigating to URL (take use of TestBase.java)
	DemoRegisterPages.java	All core activities(mentioned in list above) to be performed here
resources	config.json	URL to navigate to
	expected_data.json	Data to fill in form
coreUtilities.testbase	TestBase.java	All code related to initializing the web driver
coreUtilities.utils	CommonEvents.java	Contains all common activities
	FileOperations.java	Contains methods to read from json 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.

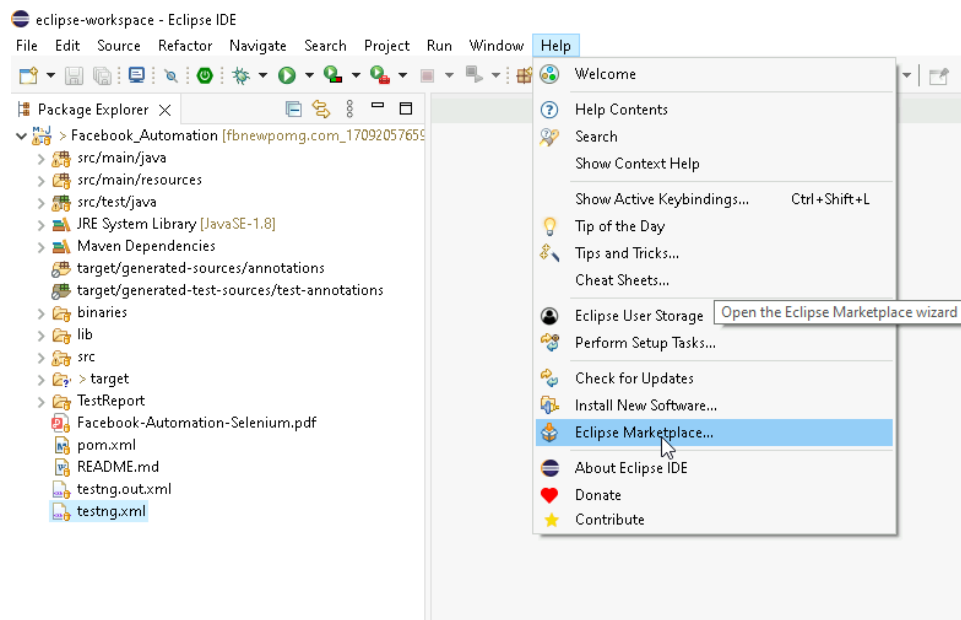
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

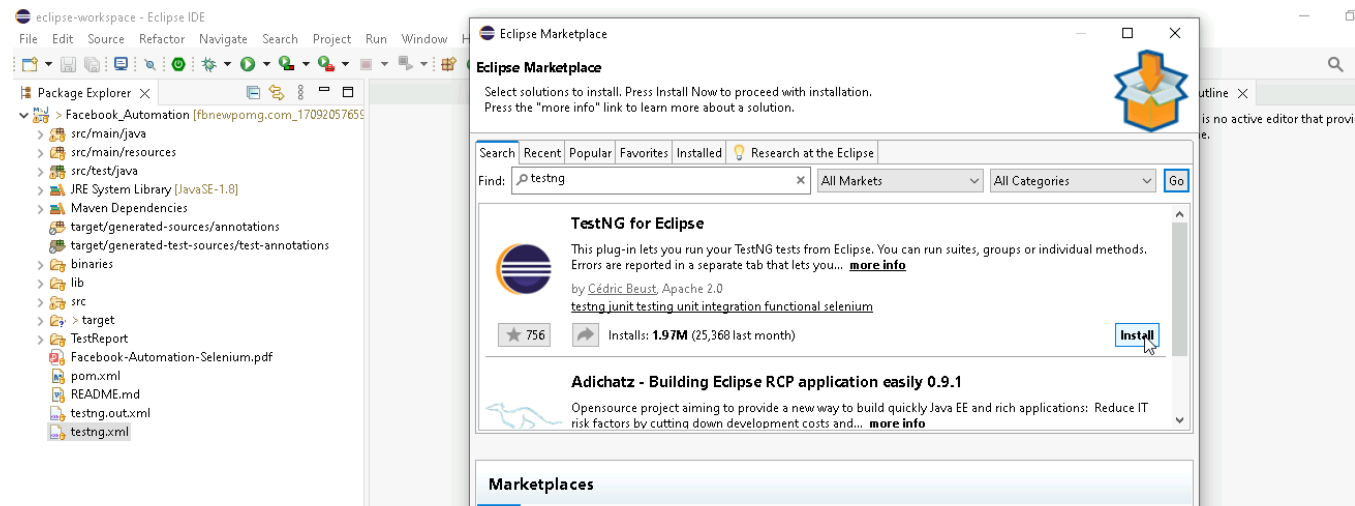
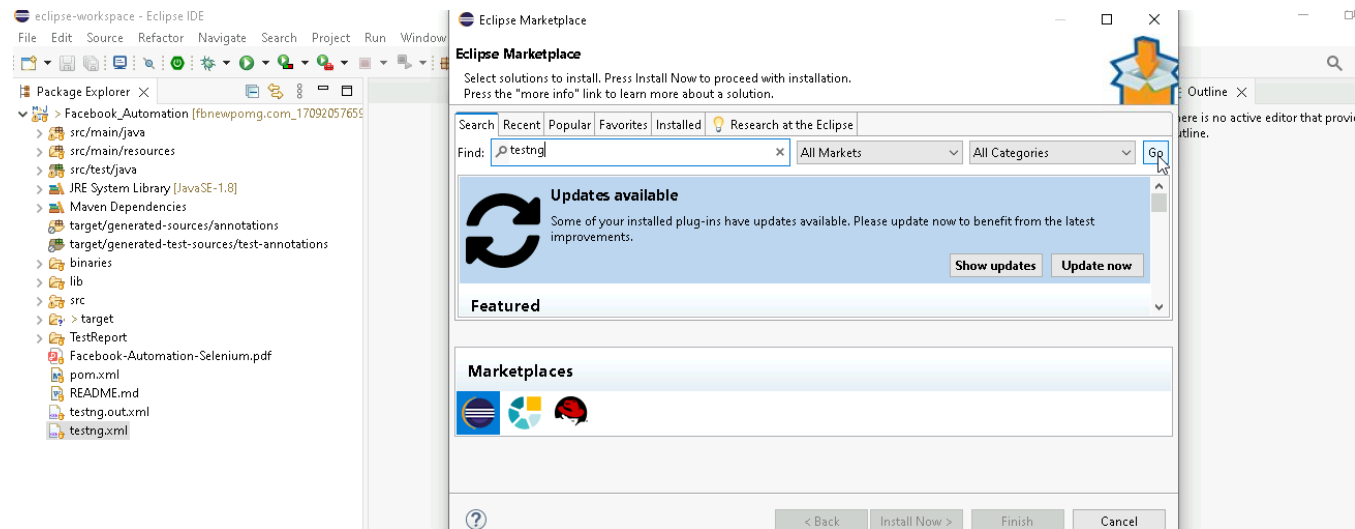
1.1 CODE QUALITY/OPTIMIZATIONS

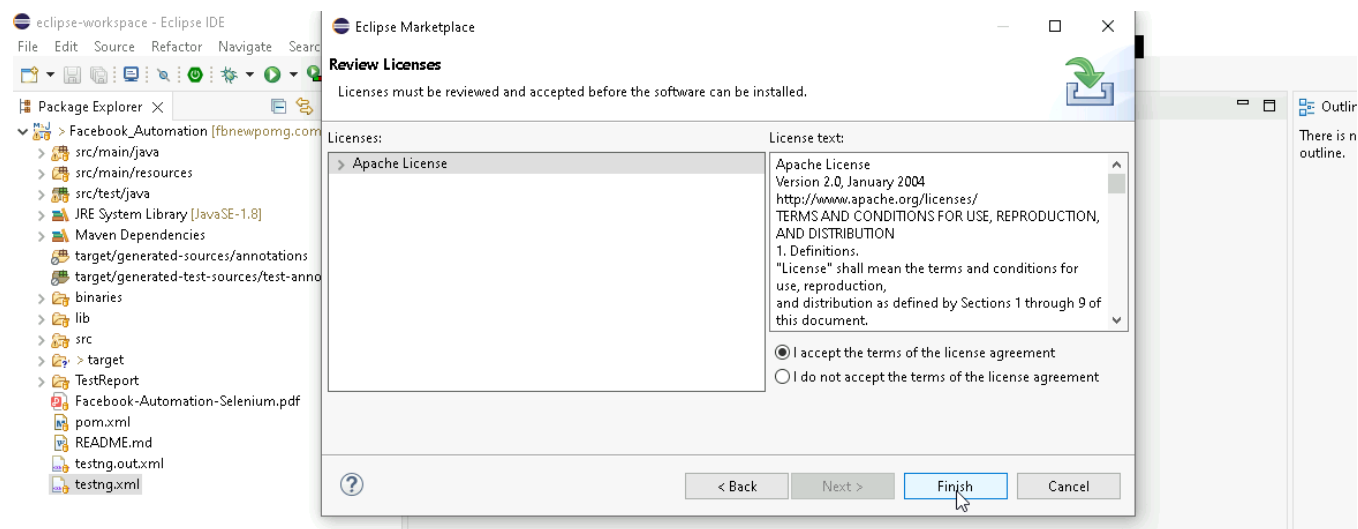
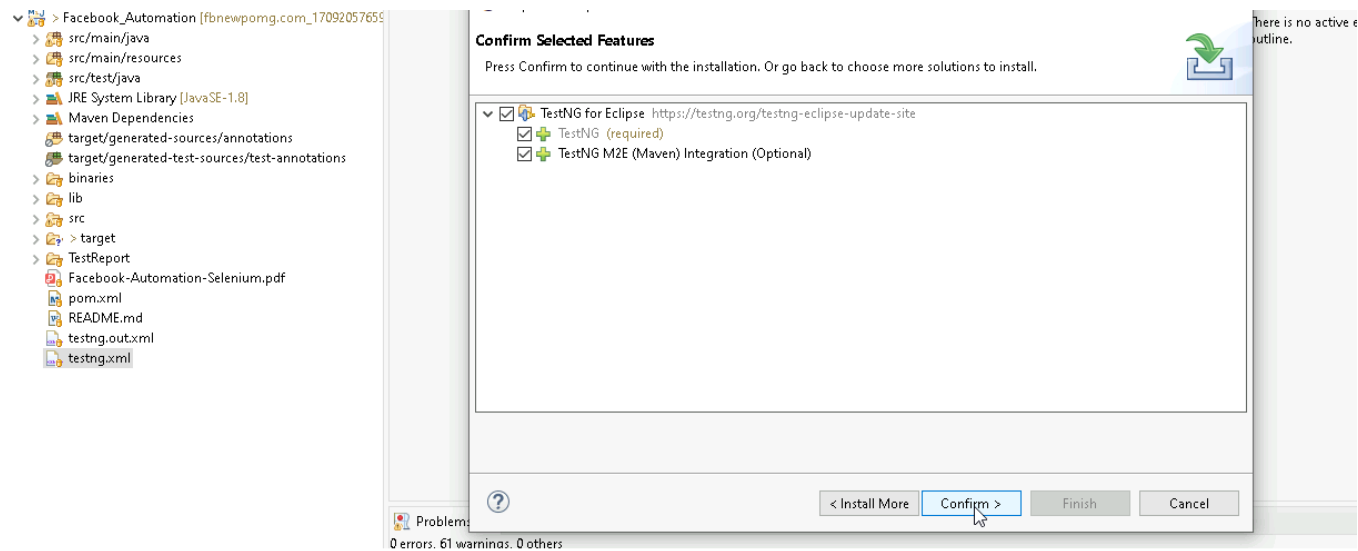
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

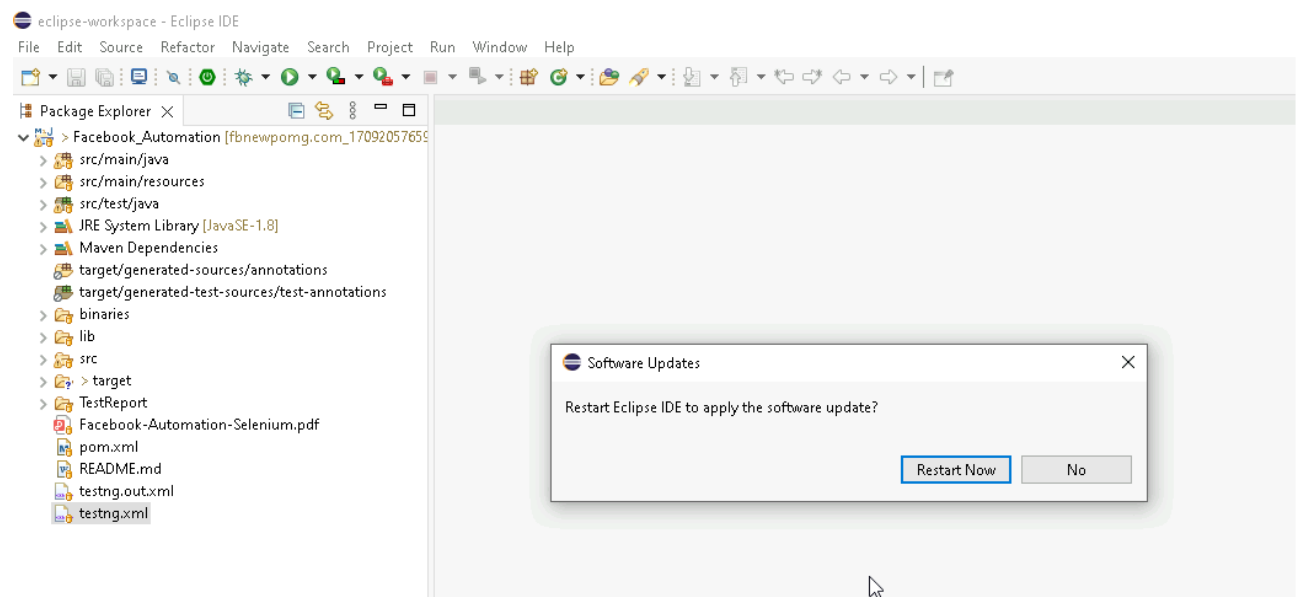
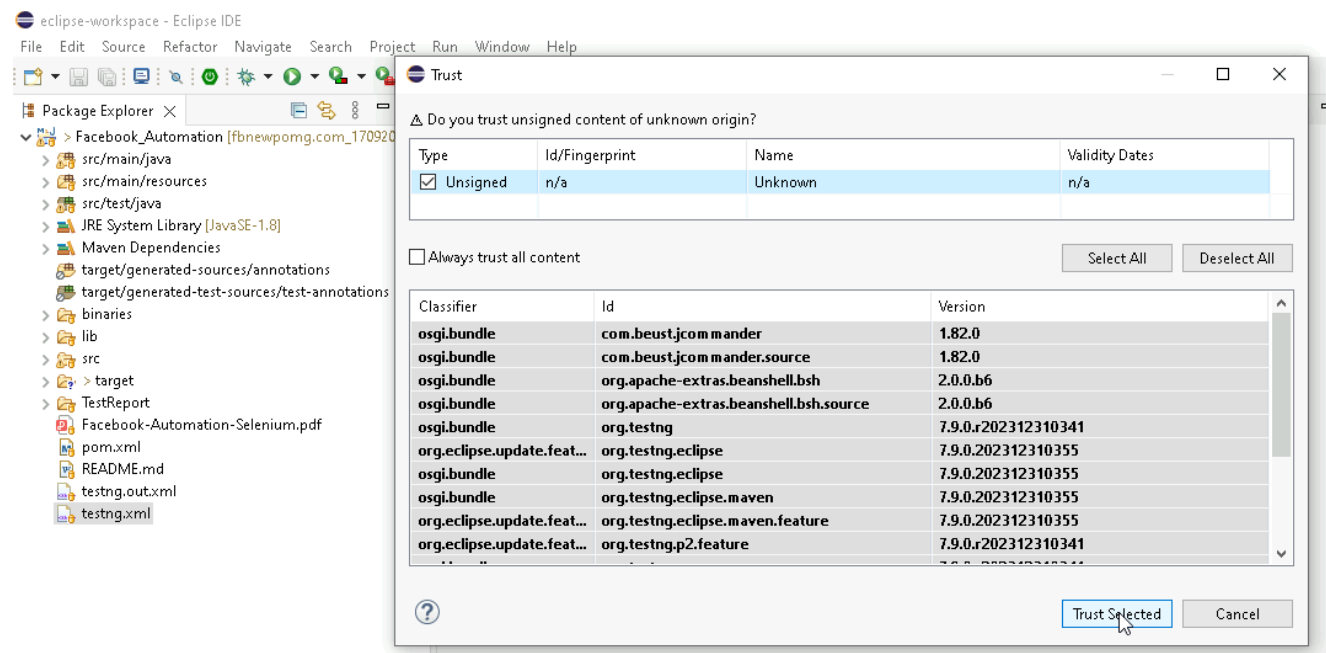
EXECUTION STEPS TO FOLLOW

1. You are mandatory required to run test cases for applications before final submission. Without which project evaluation will not happen.
2. Following are the steps to enable test case execution:
 - a. You will need to install TestNG from eclipse marketplace. Following are the steps to follow:
 - b. Go to eclipse marketplace



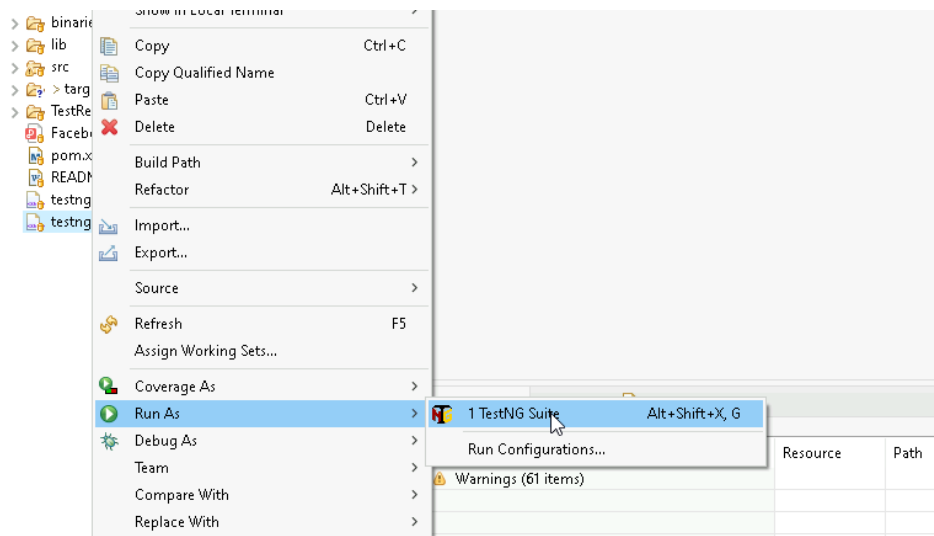




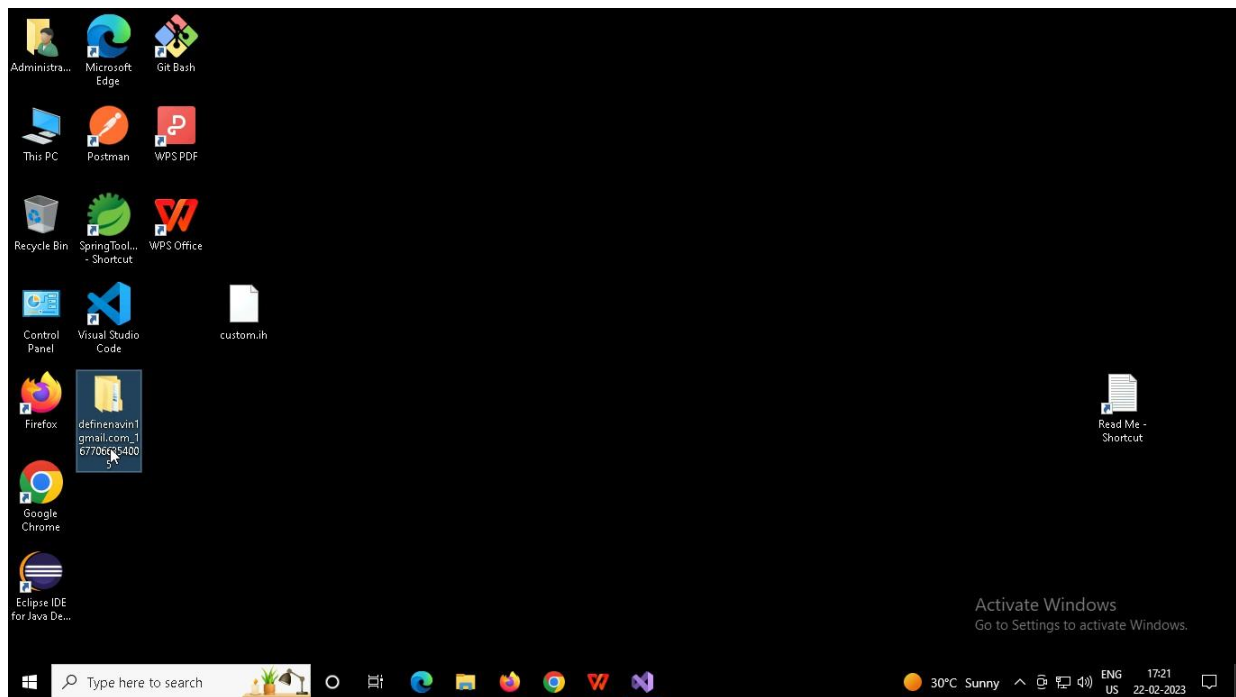


C. All the above steps will help install TestNG plugin.

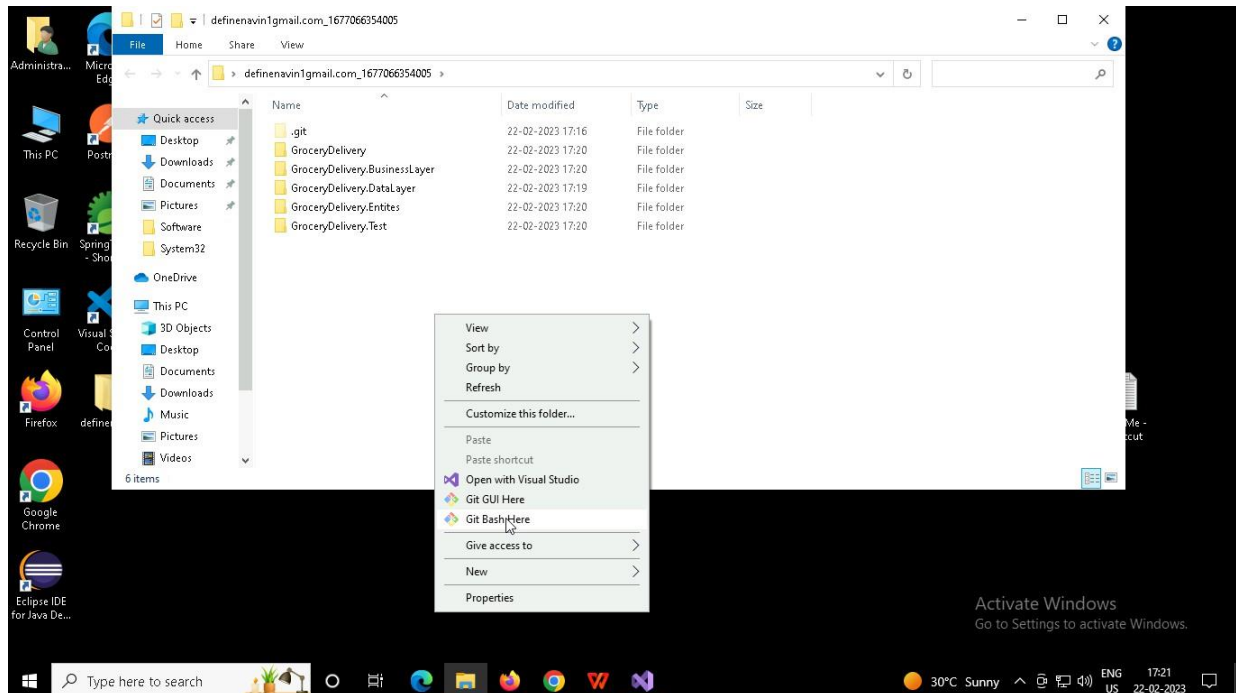
- d. Post the installation and restart of Eclipse, you can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite



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

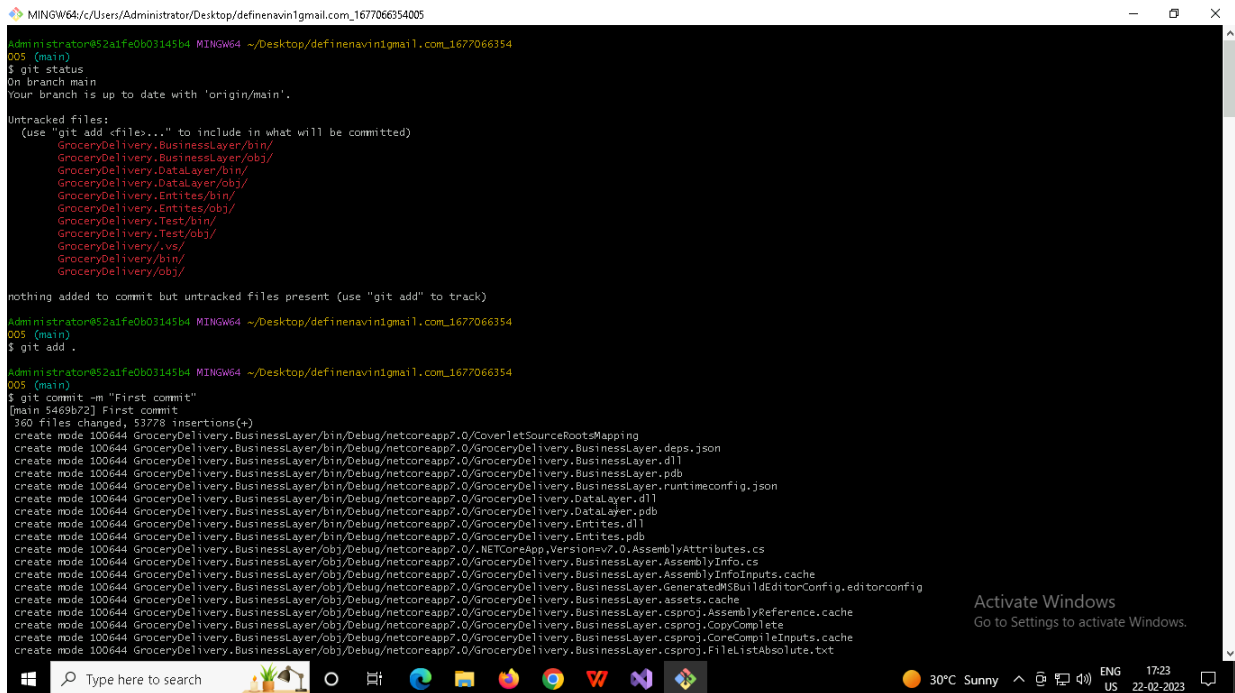


Right click in folder and open Git Bash



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 <files>..." 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
```

