

SHOPPING CART AUTOMATION

Table of Contents

Key Activities to implement.	Error! Bookmark not defined.
Helper activities to implement.	Error! Bookmark not defined.
Locator to implement.	Error! Bookmark not defined.

PROBLEM STATEMENT

Need to automate the following activities using Selenium + Java.

Key Activities to implement:

1. Navigate to: <http://practice.automationtesting.in/>
2. Check if "Shop" Menu Option is available.
3. Click on "Shop" Menu Option.
4. Check if the control is successfully navigated to "Shop" page.
5. Check if "Home" Menu Option is available on navigated page.
6. Click on "Home" Menu Option.
7. Check if the control is successfully navigated to "Home" page.
8. Check if "Home" page has a section of new arrivals.
9. Check of "Arrivals" section has exactly 3 products enlisted.
10. Check if product image is clickable.
11. Click on Product Image.
12. Check if the control is successfully navigated to Product Details page.
13. Check if product details page has an option "ADD TO BASKET".
14. Click on "ADD TO BASKET" option.
15. Check if product added message is shown when product is added to basket. Check for message
16. Go to cart(Click Cart icon)
17. Check how many items are there in cart.
18. Check if we have "View your shopping cart" anchor tag title or not.
19. Click on it.
20. Check if we have "APPLY COUPON" button or not.
21. Check if we have price defined after product name.
22. Check if we have price defined in correct format or not.
23. Check if we have quantity defined after product price.
24. Check if we have quantity defined in correct format or not.
25. Check if we have total defined after quantity.
26. Check if we have total defined in correct format or not.

Template Code Structure: Packages and files you will be required to work upon. Other Files and packages you can ignore

Package	Class/File	Description
pages	Practice_Automation_L1_Pages.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.
resources	Config.json	URL to navigate to. Already URL is defined here
	expected_data.json	Contains data to fill in form
coreUtilities.utils	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
	FileOperations.java	<ol style="list-style-type: none"> 1. Contains methods to read from json file. 2. Method is in templated form. 3. You will be required to implement these methods as very first activity, because even URL to navigate to, is read using these methods.

--	--	--

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.

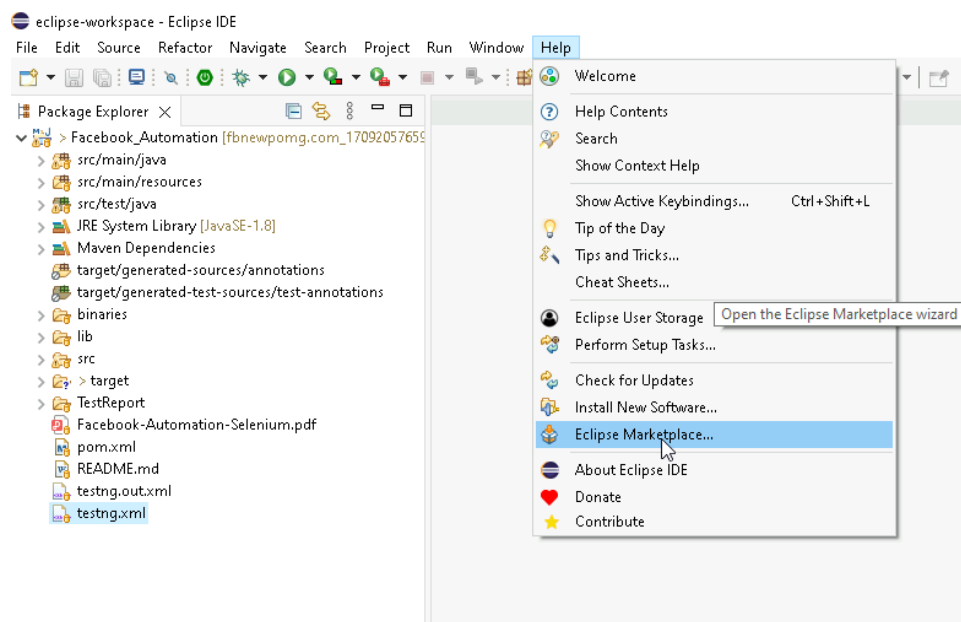
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

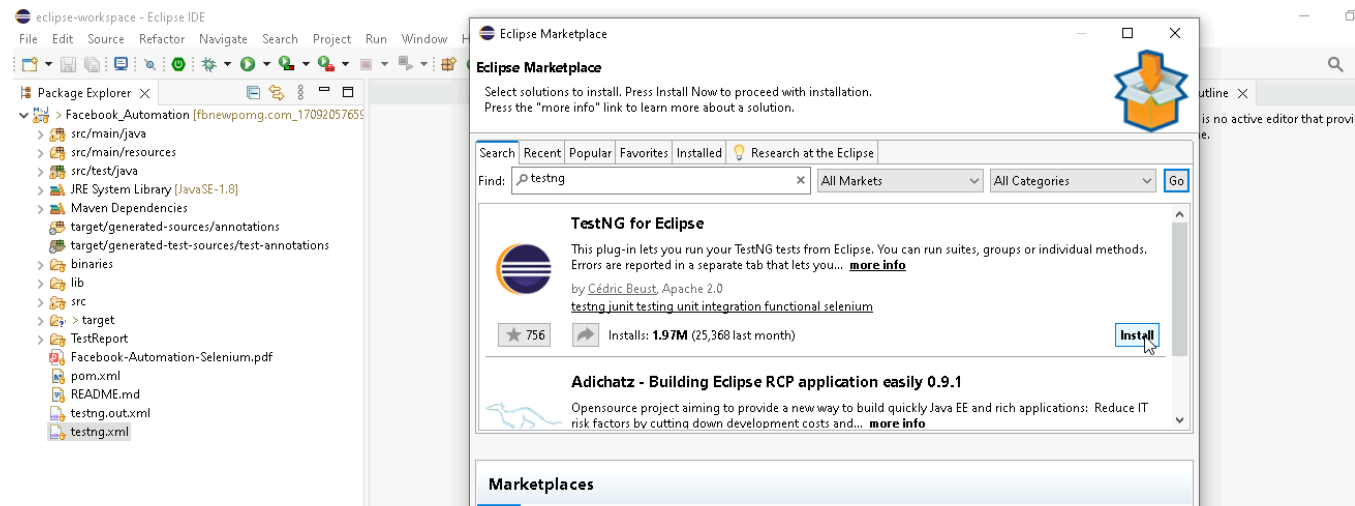
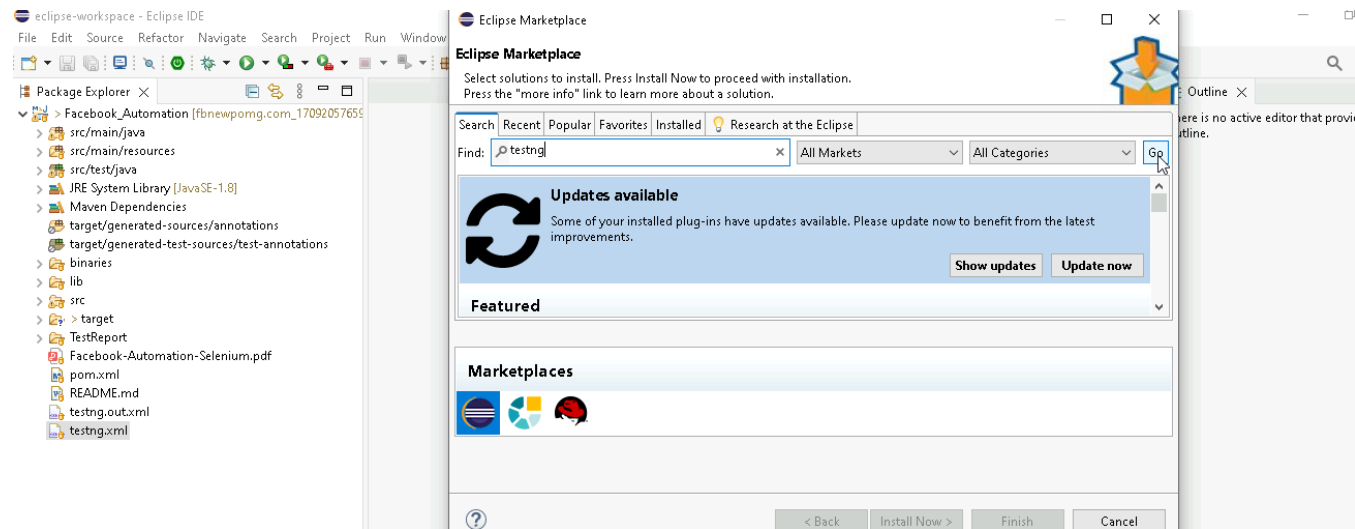
1.1 CODE QUALITY/OPTIMIZATIONS

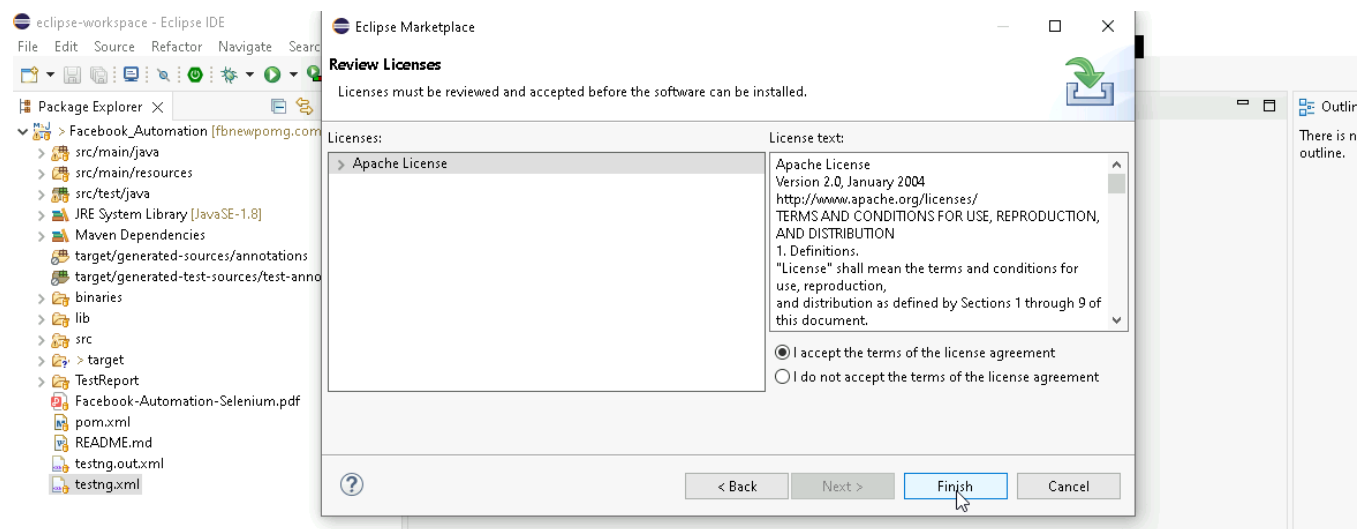
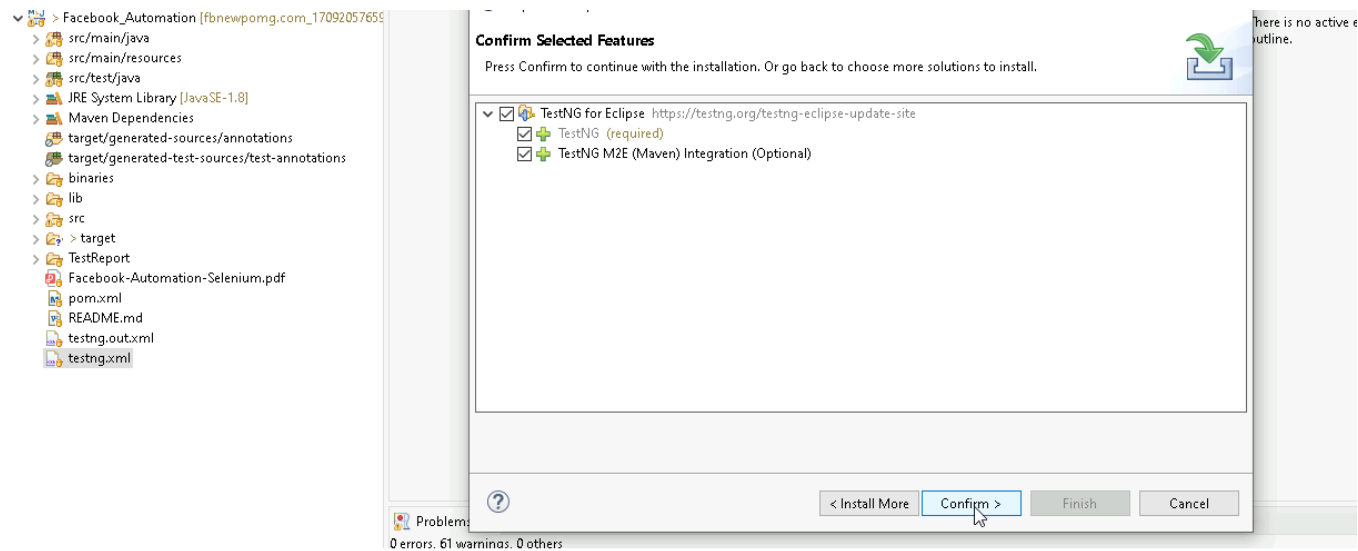
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

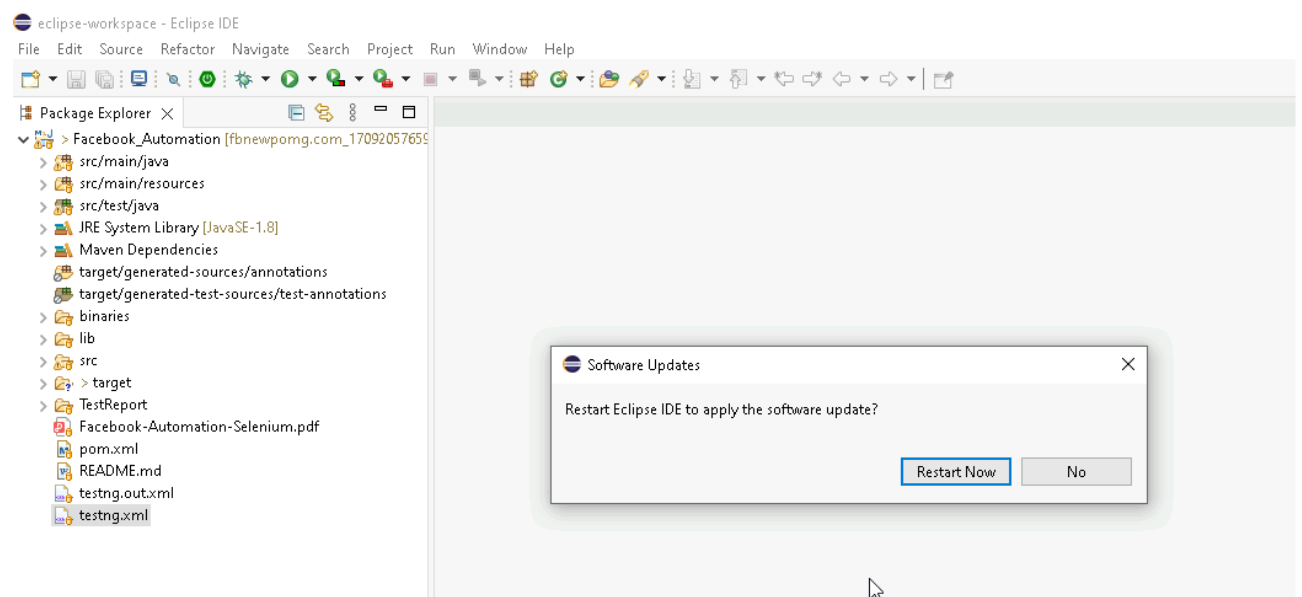
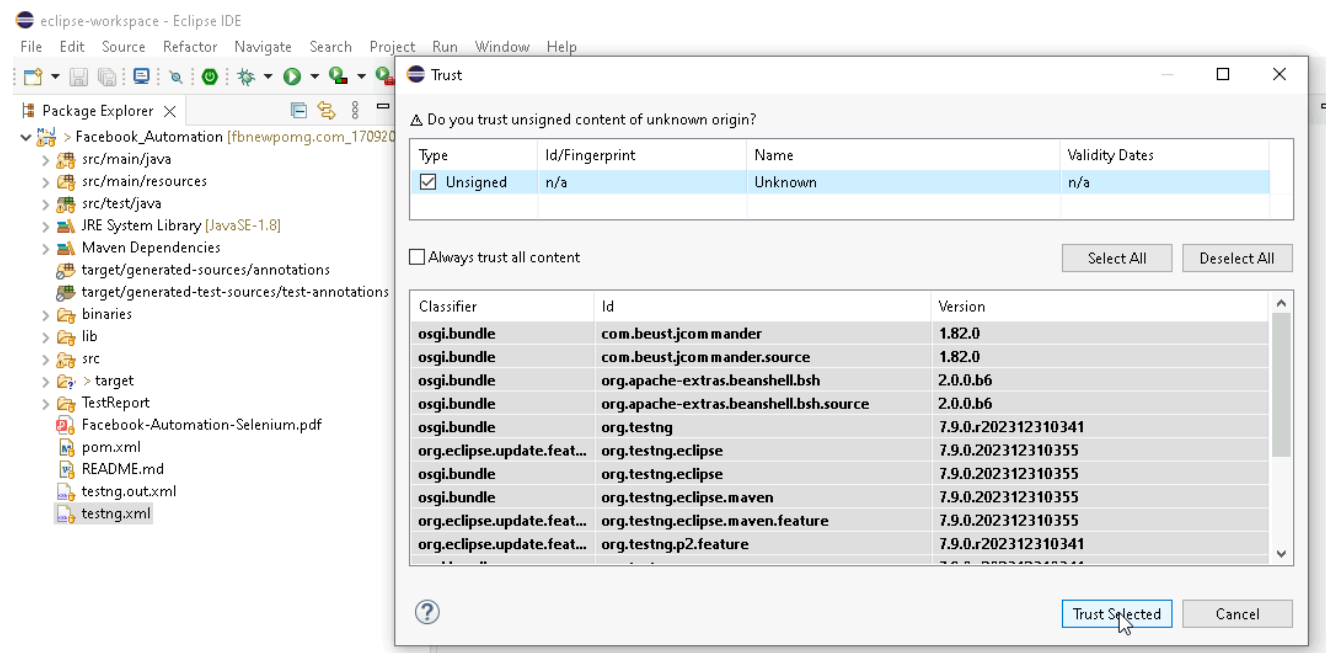
EXECUTION STEPS TO FOLLOW

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



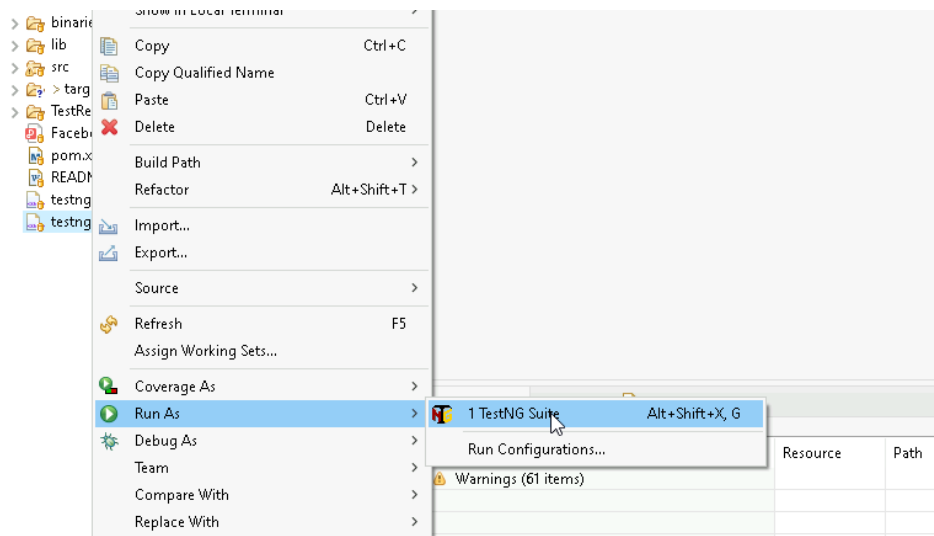




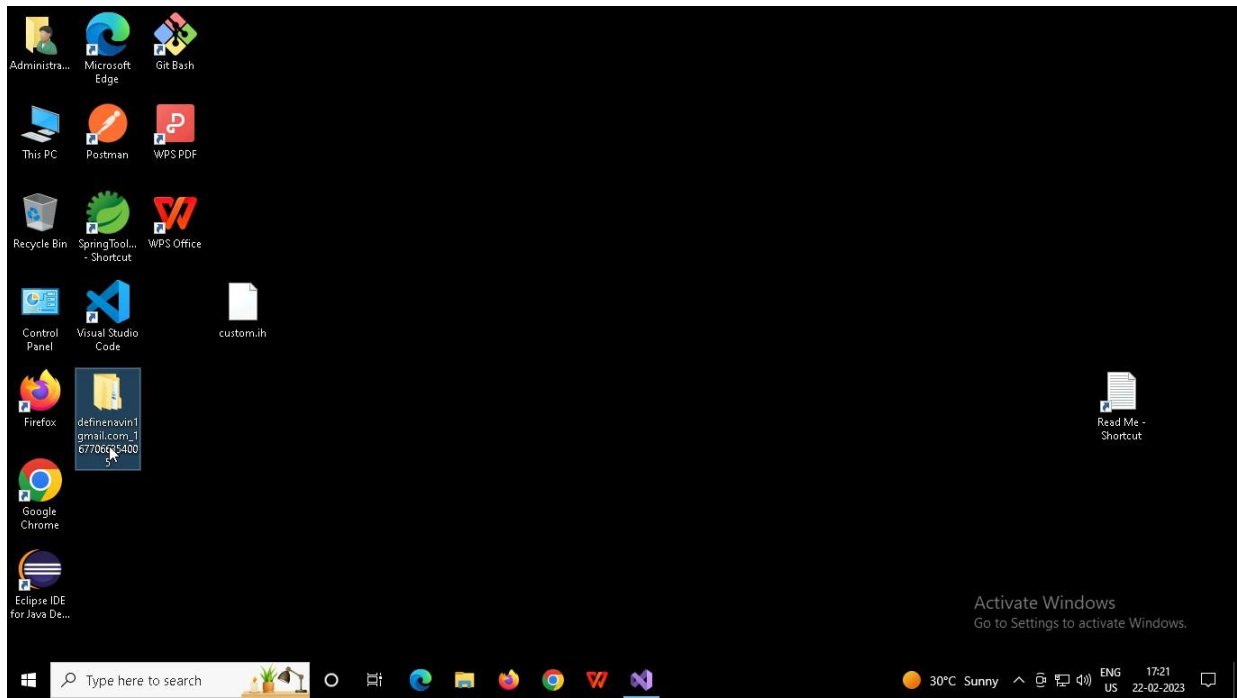


C. All the above steps will help install TestNG plugin.

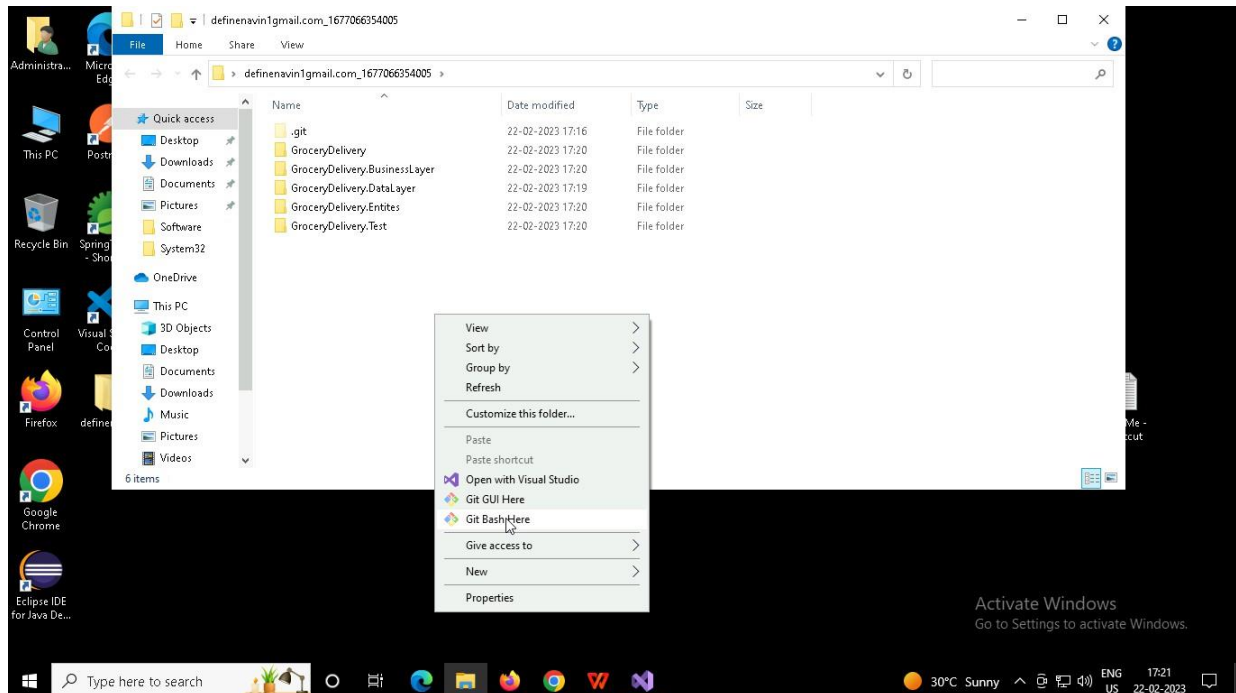
- d. Post the installation and restart of Eclipse, you can launch test cases any time as follows: Right click on testng.xml and run TestNGSuite



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

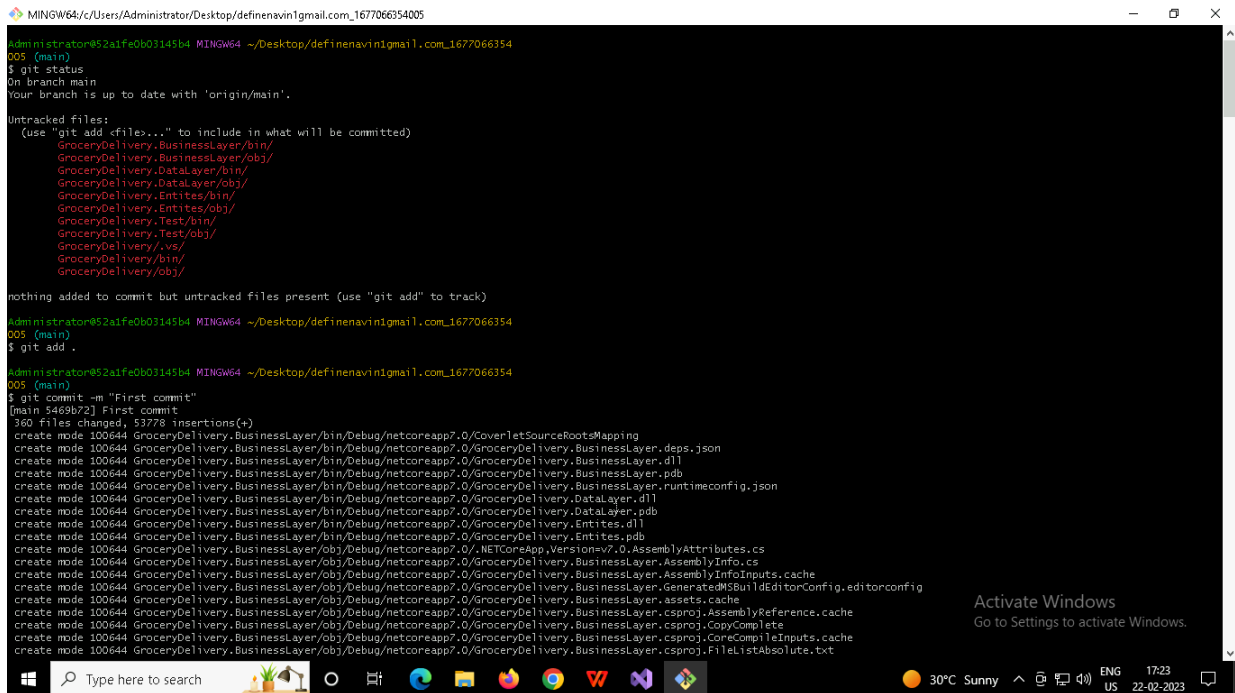


Right click in folder and open Git Bash



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 <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
```

