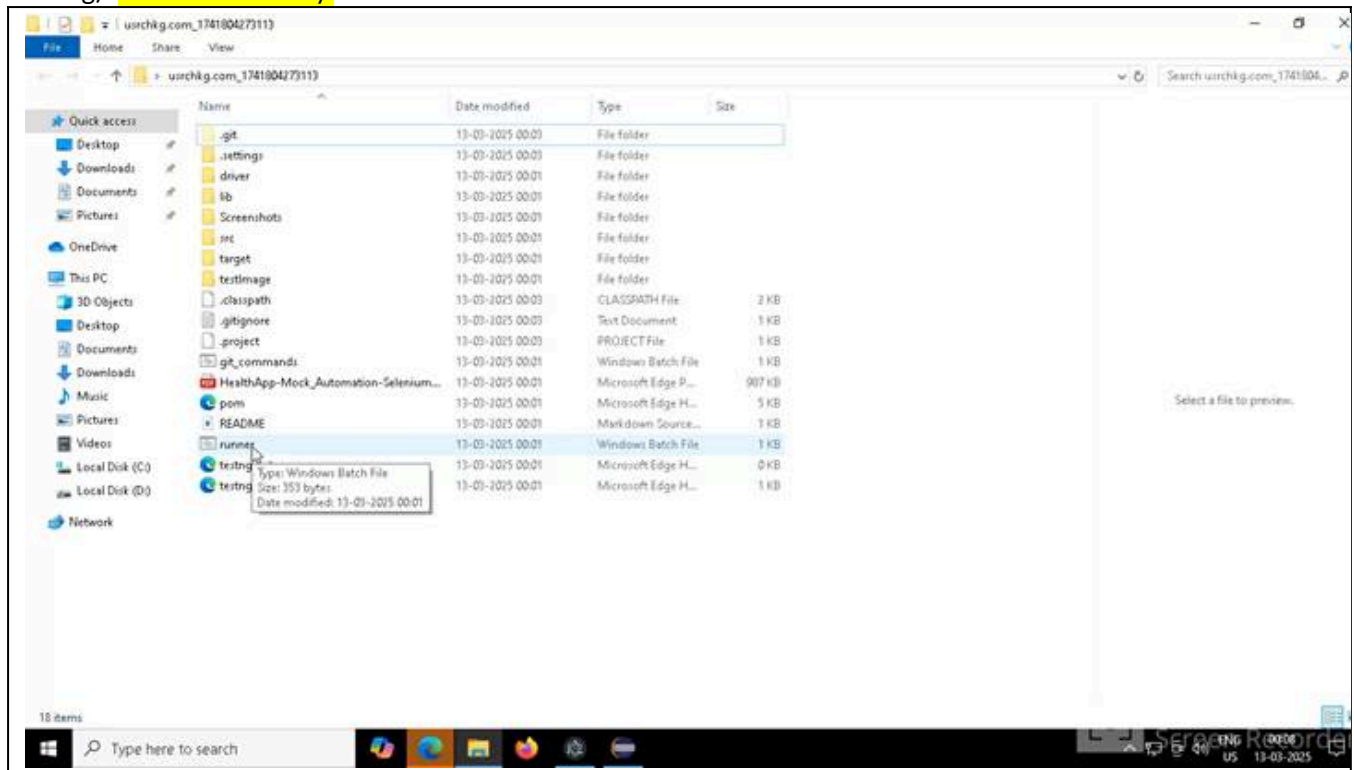


SHOPPING CART AUTOMATION

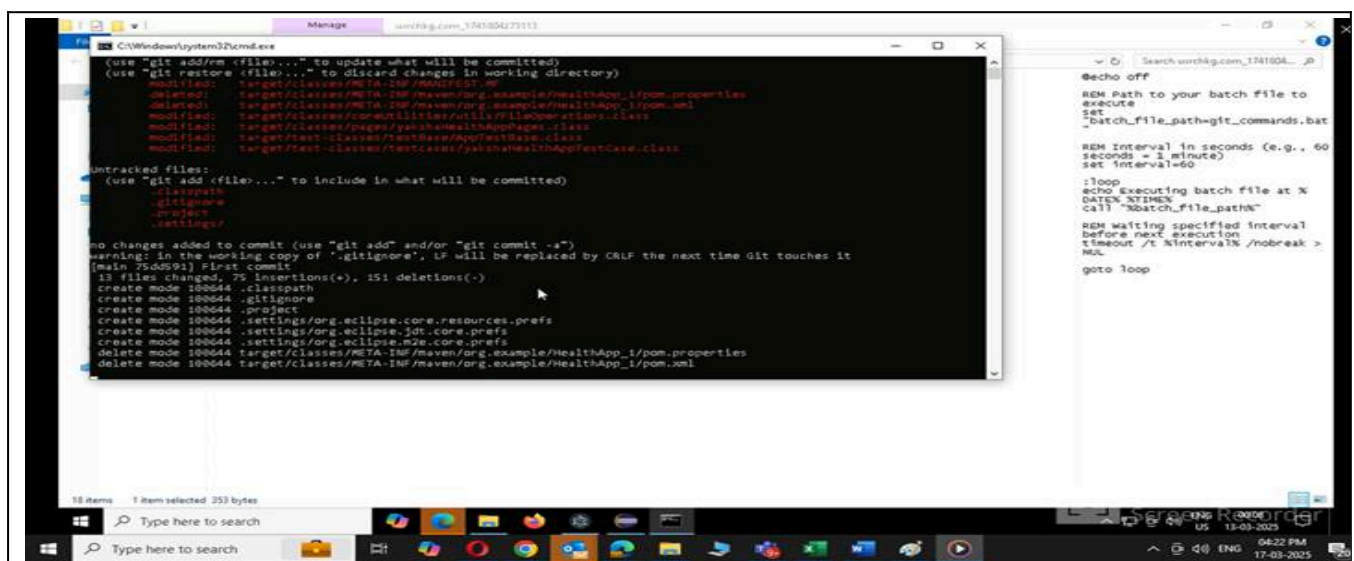
PL2-(9 TCs)

Pre-requisite:

Before you start working on your project, execute the runner file present in your project folder (Simply by double clicking). **This is mandatory.**

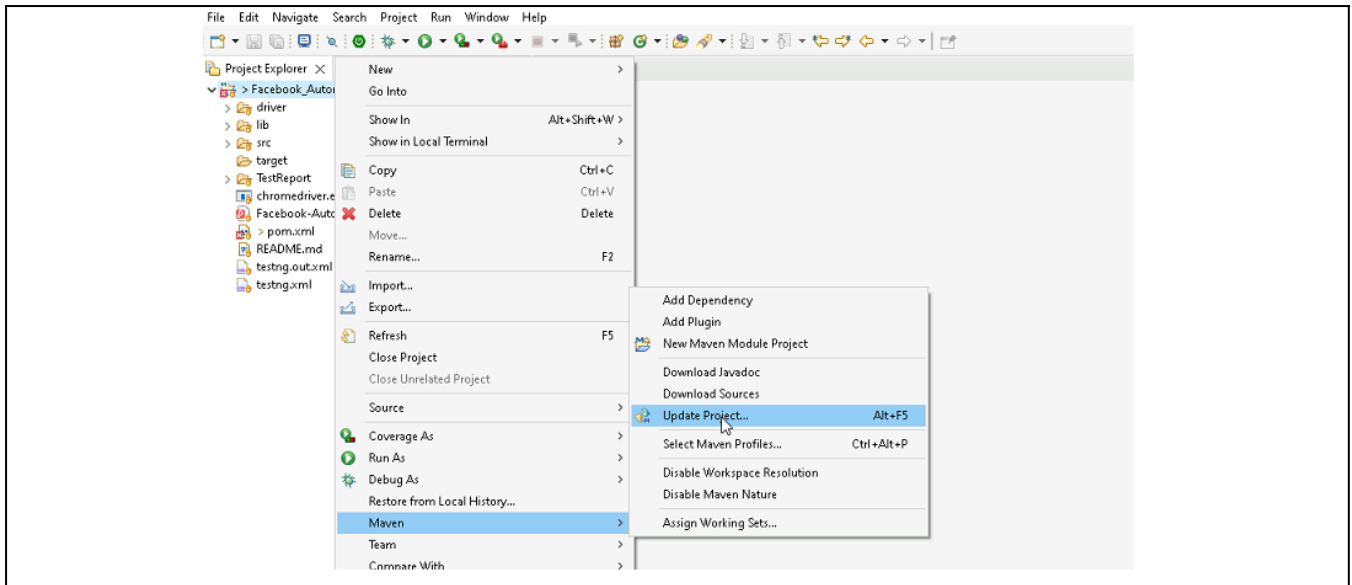


This will launch a command terminal for you, pushing your updated code to GIT at regular intervals. Keep that command terminal open at the backend, and you can continue working on your project.

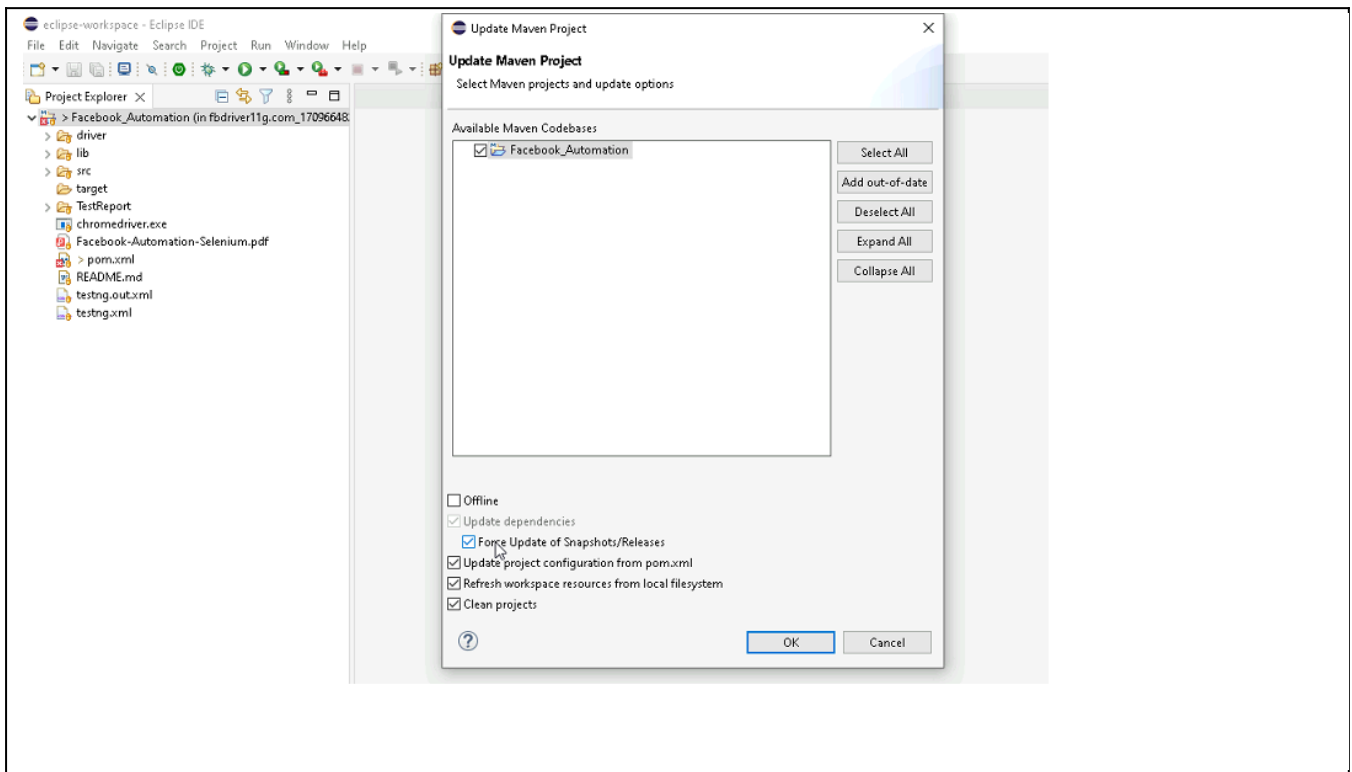


As soon as you import the project in Eclipse, update the project using maven update option as below. This is to resolve issue if any maven dependency not downloaded properly:

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:

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

Package	Class/File	Description
src/main/java/coreUtilities/utis /	FileOperations.java	<ol style="list-style-type: none"> 1. It contains methods to read data from Excel files. 2. The method is in templated form. 3. You will be required to implement these methods as the very first activity, because even the URL to navigate to is read using these methods.
/src/main/java/pages	ShoppingCartAutomationPages.java	<ol style="list-style-type: none"> 1. All core activities (mentioned in the list below “Key Activities to implement”) are 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.xlsx	URL to navigate to: The URL is defined here.
	expected_data.xlsx	Contains data to fill in the form.
/src/main/java/coreUtilities/utis	CommonEvents.java	<ol style="list-style-type: none"> 1. Contains all common activities.

		2. A certain templated common method is declared here. 3. You implement them as per your needs. 4. You can add any additional methods for a common activity here
	Testng.xml	Execution needs to kick-start from TestNG xml

PROBLEM STATEMENT

Need to automate the following activities using Selenium + Java.

Key Activities to implement:

SN	TC Summary	Action	Expected Result
1	Check the tile of the home page.	Steps: 1. Navigate to: http://practice.automationtesting.in/ .	Verify the title of home page.
2	Check "Shop" Menu Option is present or not.	Precondition: User should be on the home page of the website.	Verify that the "Shop" menu option should be present on the page.
3	Check the title of the shop page to ensure it is navigable.	Precondition: User should be on the home page of the website. Steps: 1. Click on "Shop" Menu Option.	Verify the title of the Shop page.
4	Click on the Automation Testing logo.	Precondition: User should be on the "Shop" page of the website. Steps: 1. Click on the logo of the "Automation Testing"	Verify if navigate to the home page by clicking on the Logo.
5	Check if Home page has a section of new arrivals.	Precondition: User should be on the home page of the website.	Verify that home page has "new arrivals" section.
6	Check of "Arrivals" section has three products enlisted.	Precondition: User should be on the home page of the website.	Verify the following items are displayed under the "New Arrival" Setion: "Selenium Ruby", "Thinking in HTML" and "Mastering JavaScript".

7	Navigate to Selenium Ruby Product Details page.	Precondition: User should be on the home page of the website. Steps: 1. Click on Selenium Ruby Product.	Verify the Title of the "Selenium Ruby" page to ensure that it is navigated.
8	Check if product details page has an option "ADD TO BASKET".	Precondition: User should be on the "Selenium Ruby" product page. Steps: 1. Click the "ADD TO BASKET" button.	Verify that "ADD TO BASKET" button is present.
9	Ensure after click on add to basket, "VIEW Basket" button appears.	Precondition: User should be on the "Selenium Ruby" product page. Steps: 1. Click the "VIEW BASKET" button.	Verify that "View Basket" button is present.

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.

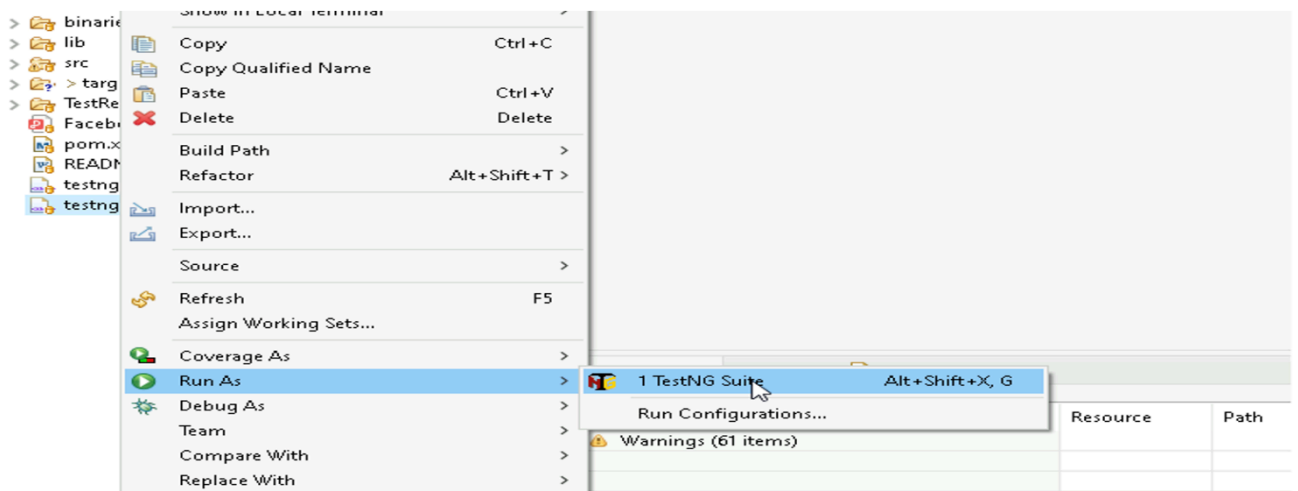
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 CODE QUALITY/OPTIMIZATIONS

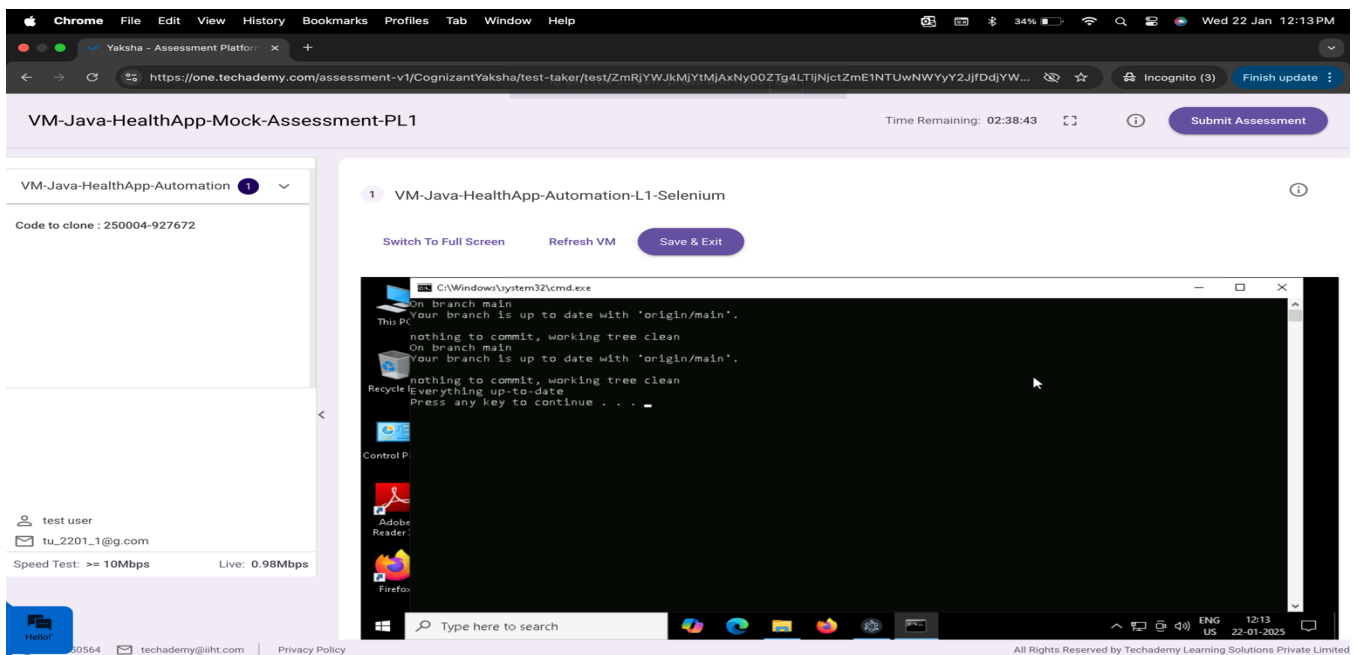
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

EXECUTION STEPS TO FOLLOW

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



0. To do the final submission of the assessment :
 - Press escape to come out of Fullscreen mode.
 - Submit the assessment.



After the successful submission of the assessment, you will get a confirmation message displayed on your screen.

=====

All the Best

