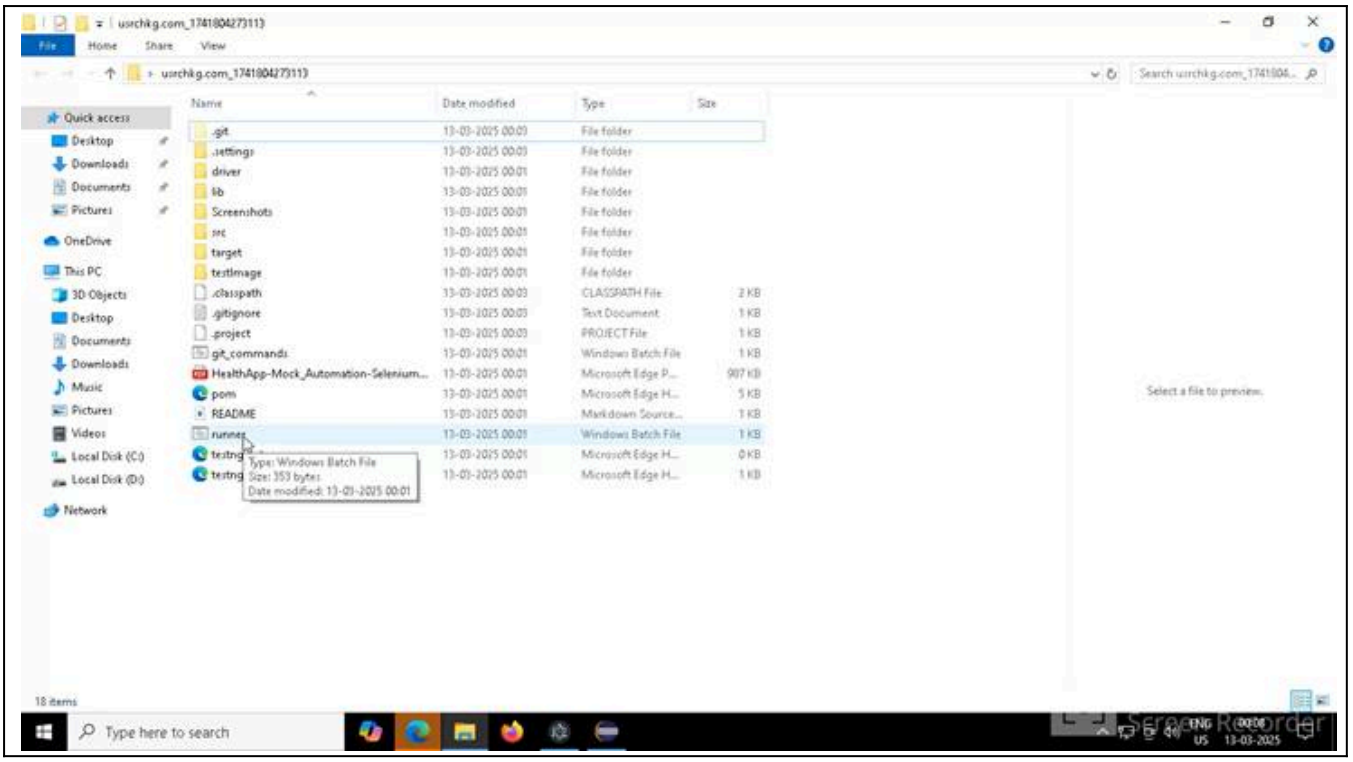
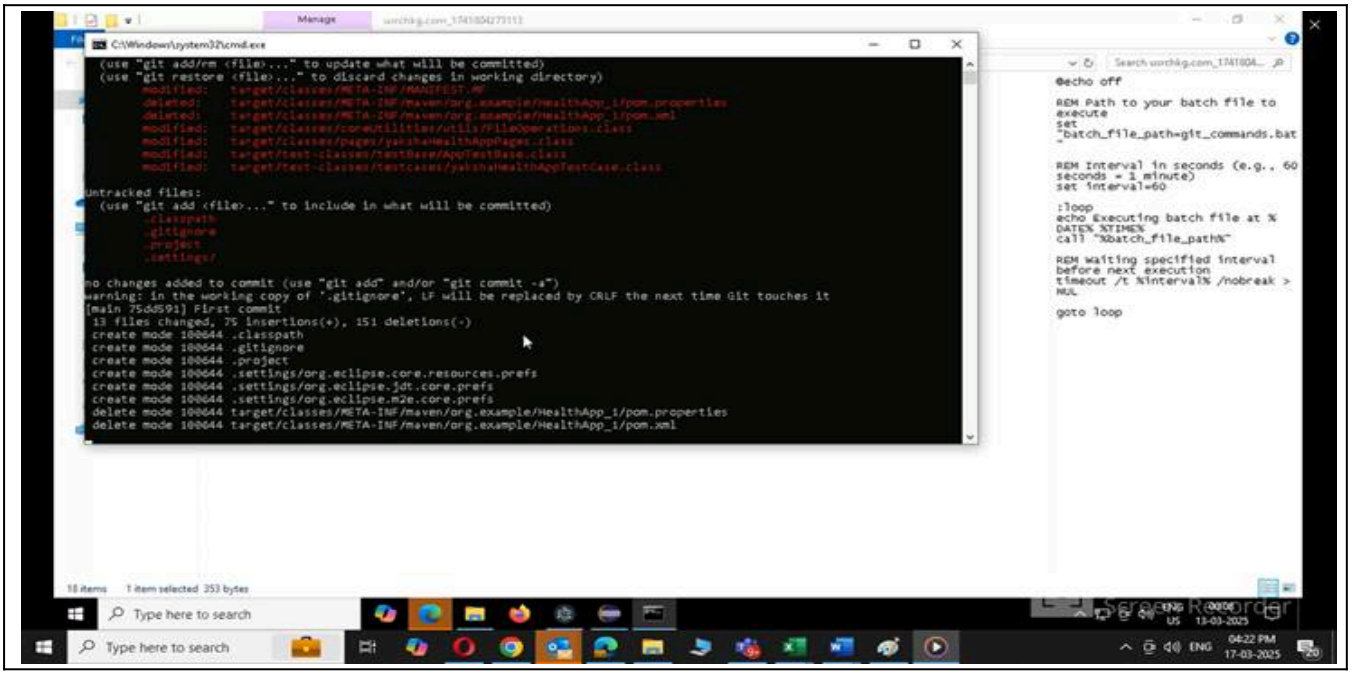
HealthApp Automation
MedicalRecord - PL2(9TCs)

Pre-requisite:

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

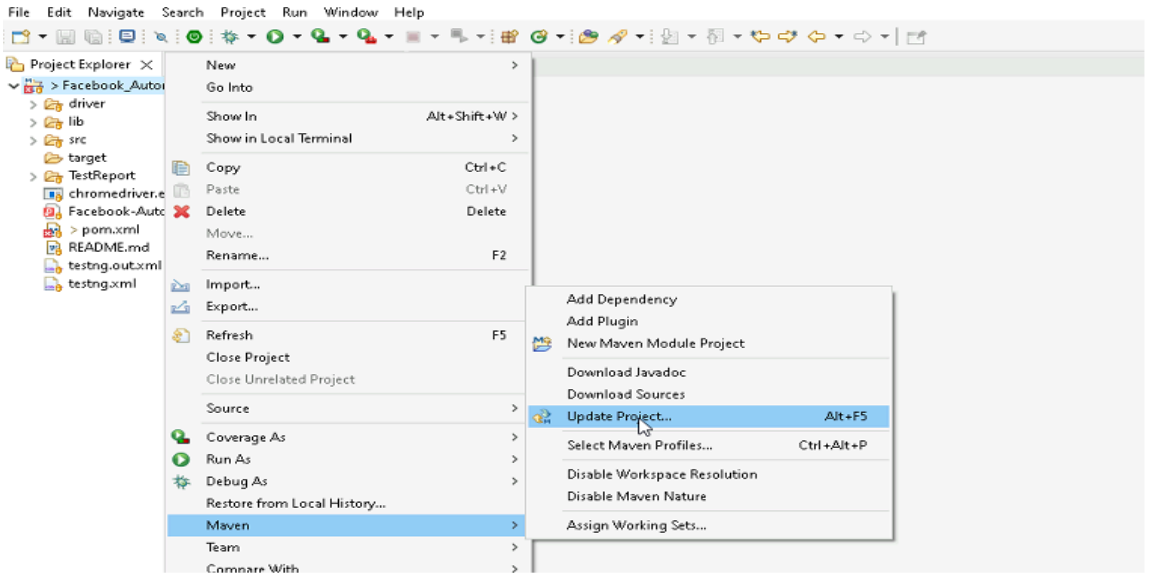


This will launch a command terminal for you where it will keep on pushing your updated code to GIT on regular intervals. Keep that command terminal open at backend and you can continue working on your project.

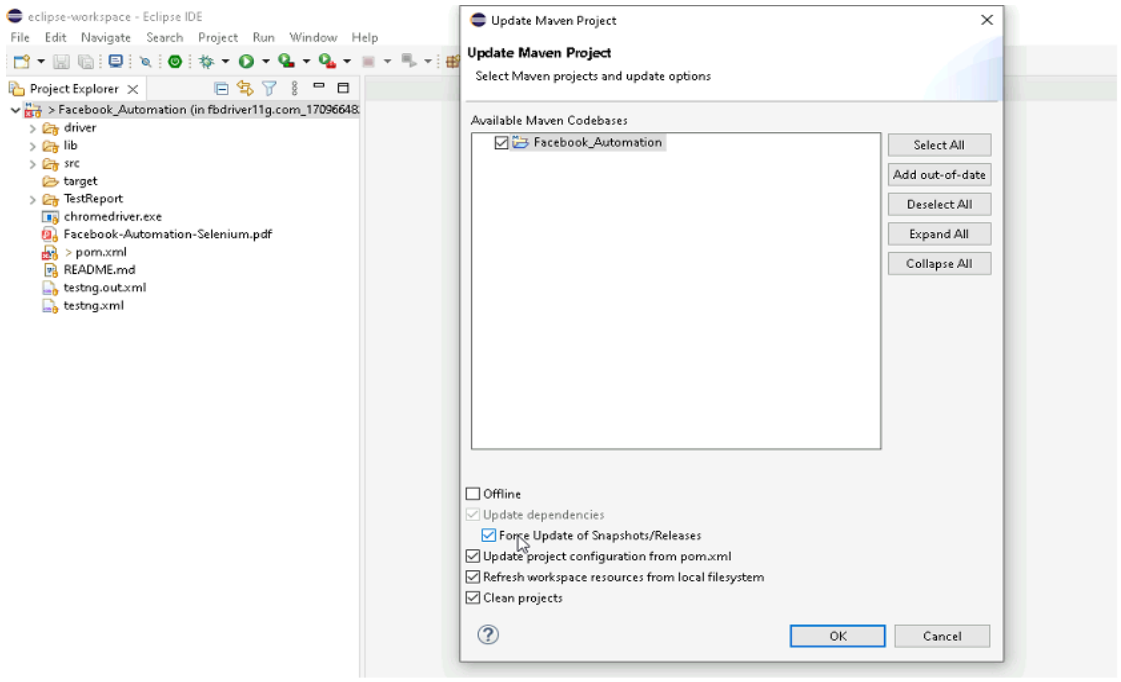


As soon as you import the project in Eclipse, update the project using the maven update option as below. This is to resolve the issue if any Maven dependency is not downloaded properly:

1. Right-click on the project: Go to “Maven” and select “Update Project”



2. In the 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 on.

b. Other Files and packages you can ignore.

c. In other Files and packages do not make any changes. It would affect your evaluation.

d. You are not required to work in the “Test” Folder. Files there are non-editable. Editing those files and trying to save them will throw errors and would affect your evaluation.

Package	Class/File	Description
src/main/java/coreUtilities/utils/	FileOperations.java	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	medicalRecord_page.java	1. All core activities (mentioned in the” Key activities to implement” list below) will 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: URL is already defined here
	expected_data.xlsx	Contains data to fill in form\
/src/main/java/coreUtilities/utils	CommonEvents.java	1. Contains all common activities. 2. Certain templated common methods are declared here. 3. You implement them as per your needs. 4. You can add any additional method for common activity here
	Testng.xml	Execution needs to be kick-started from TestNG.xml

PROBLEM STATEMENT

Need to automate the following activities using Selenium + Java.

Key Activities to implement:

#	Summary	Action	Expected Result
1	Verifying the Presence of the Medical Records Module	go to URL:**https://healthapp.yaksha.com/** 1. log in with a valid credential **(username: admin, password: pass123)** and click on the "**Sign In**" Button. 2. Scroll down the menu to Medical Records. 3. Click on the Medical Records	Verify the user is logged into the health app. Verify the URL as follows: https://healthapp.yaksha.com/Home/Index#/Medical-records/InpatientList
2	Ensure that all designated sub-modules within the Medical Records module are correctly displayed and accessible.	**Preconditions**: The user is logged in and currently on the Medical Records module. Ensure the User is on the "MR Outpatient" module.	Verify that each sub-module is displayed. The expected sub-modules are "**MR Outpatient List", "MR Inpatient List", "Birth List", "Death List", "Reports" and "Emergency Patient List"**

#	Summary	Action	Expected Result
3	Ensure the presence of the Elements on MR Outpatient.	**Preconditions**: The user is logged in and currently on the Medical Records module. Ensure the User is on the “MR Outpatient” module.	Verify all listed components should be properly displayed: **Buttons**: 'First', 'Previous', 'Next', 'Last', 'Add Final Diagnosis', 'Edit Final Diagnosis'.
4	Ensure that all the Sub-modules are accessible.	**Precondition:** The user is located within the Medical Records module, specifically in the MR Outpatient section. **Steps**: 1. Click on the "Reports". 2. Click on the "Birth List" 3. **Click on the** "Death List" section. 4. **Click on the** "Emergency Patient List" section. 5. **Click on** "MR Inpatient List"	Verify the user is able to access all the listed sub-modules.
5	Ensure that setting a date range in the MR Outpatient section filters and displays only the records with 'Appointment Dates' falling within the specified range.	**Preconditions**: The user is logged into the health system. The user is located in the MR Outpatient section, ready to apply a date filter. **Steps**: 1. Click on the "From" date picker and select the starting date as "Jan 2020". 2. Click on the "To" date picker and select the ending date as "March 2025". 3. After setting the desired date range, click the "OK" button to apply the filter.	Verify that the 'Appointment Date' column date must fall within the selected date.
6	Ensure that selecting "Cardiology" from the "Department Filter" dropdown effectively filters and displays outpatient records.	**Preconditions**: The user must be logged into the health system. The user is located in the MR Outpatient section, ready to utilize the filtering features. **Steps**: 1. From the dropdown options, select "Cardiology" to set the filter for this specific department.	Upon selection of "Cardiology," the outpatient records displayed should only include those related to the Cardiology department.
7	Ensure the functionality of the “Date Range” option.	**Preconditions**: The user must be logged into the health system. The user is located in the MR Outpatient section. **Steps**: 1. From the date range option (the ‘-’ symbol between To date picker and OK button, select "Last 1 Week".	Verify that after choosing “Last 1 Week", the outpatient records displayed should only include those which fall in this criteria.

#	Summary	Action	Expected Result
8	Ensure the functionality of the Pagination navigator at the bottom of the page.	**Preconditions**: The user must be logged into the health system. The user is located in the MR Outpatient section. **Steps**: **1.** Apply date filter, choose From Date = "01-01-2023" and To Date as the current date. 2. Scroll to the bottom of the page. 3. Click the “Next” button. 4. Click the “Previous” button.	• Verify after the test step 2 that the current navigation page is “1”. • Verify after the test step 3 that the current navigation page is “2”.
9	Ensure the functionality of the “Select Diagnosis” filter.	**Preconditions**: The user must be logged into the health system. The user is located in the MR Outpatient section. **Steps**: 1. Click on "MR Inpatient List". 2. Click on "MR Outpatient List". 3. Click the diagnosis dropdown. 4. Inputs the provided diagnosis Code. 5. Click the TAB key on the keyboard 6. Click on the OK button.	Verify that the data in the table is filtered as per the input diagnosis code.

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

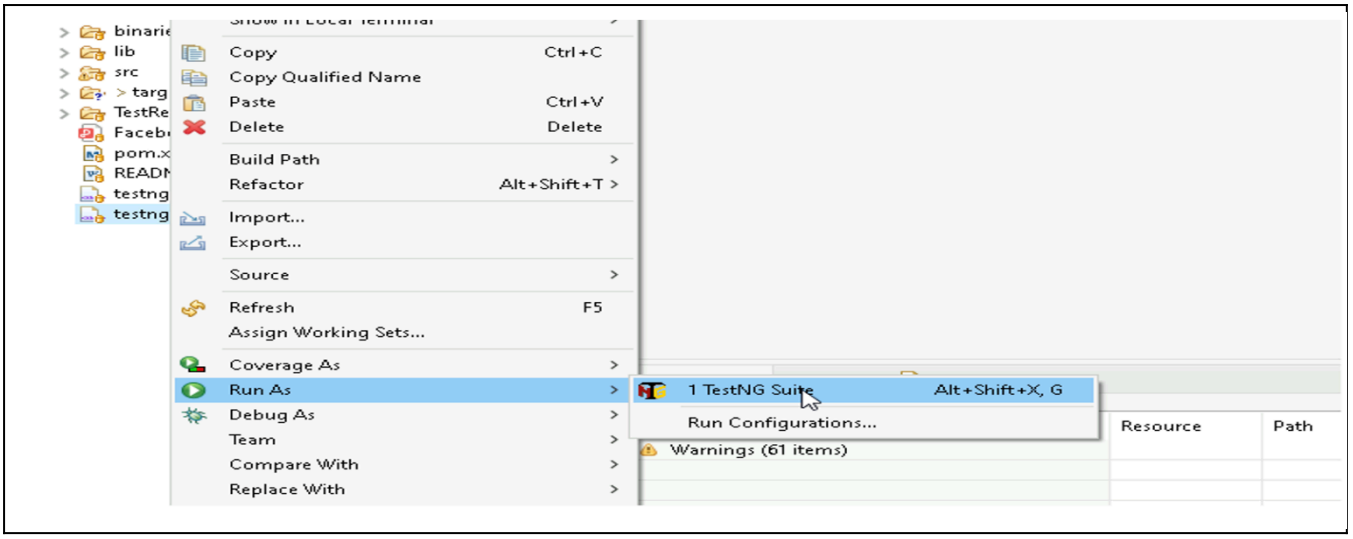
1.1 CODE QUALITY/OPTIMIZATIONS

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

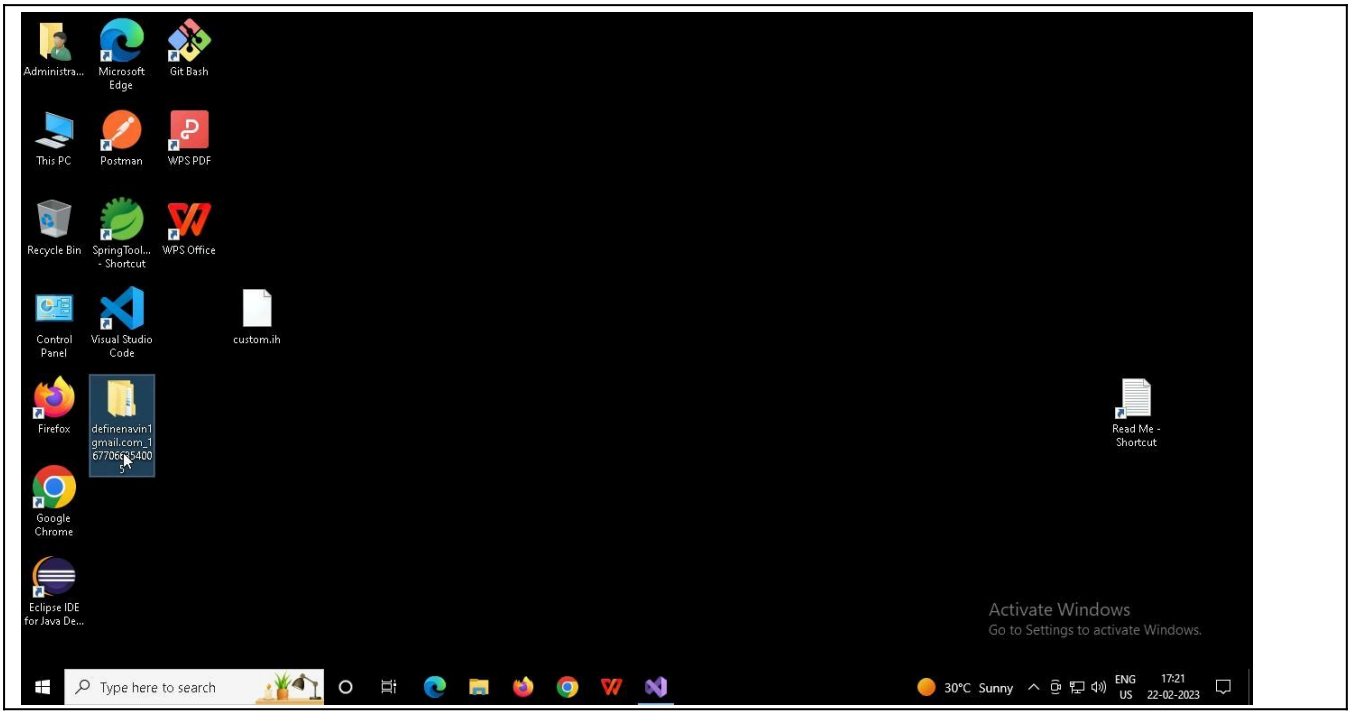
EXECUTION STEPS TO FOLLOW

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**

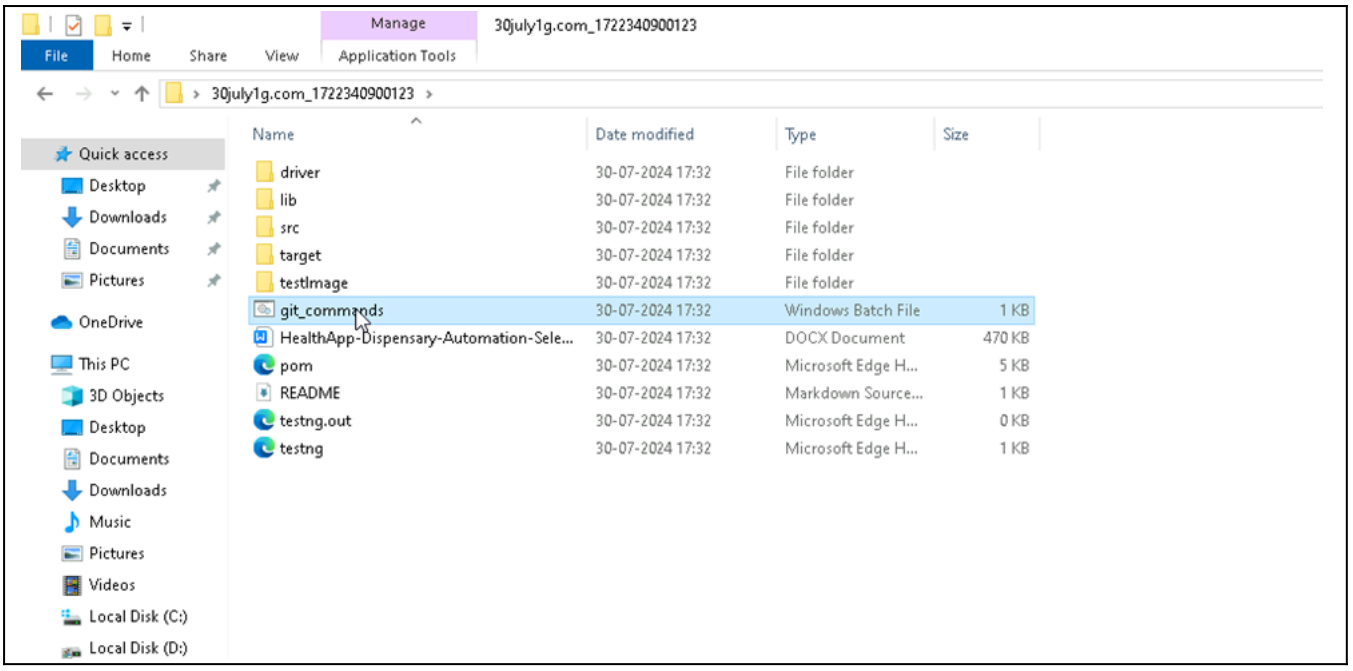
TestNGSuite.



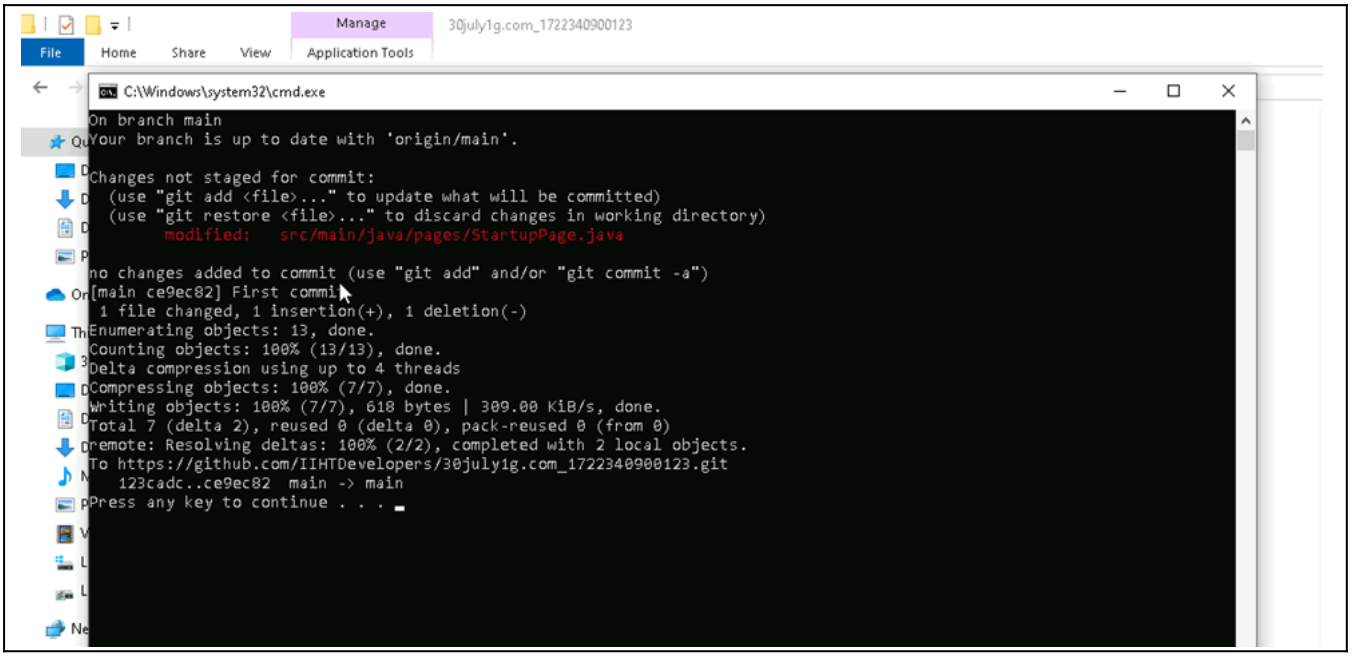
3. **Before final submission, you are also required to push your code to GIT. Following are the steps to follow:**



In your project folder, you will find a batch file named git_commands

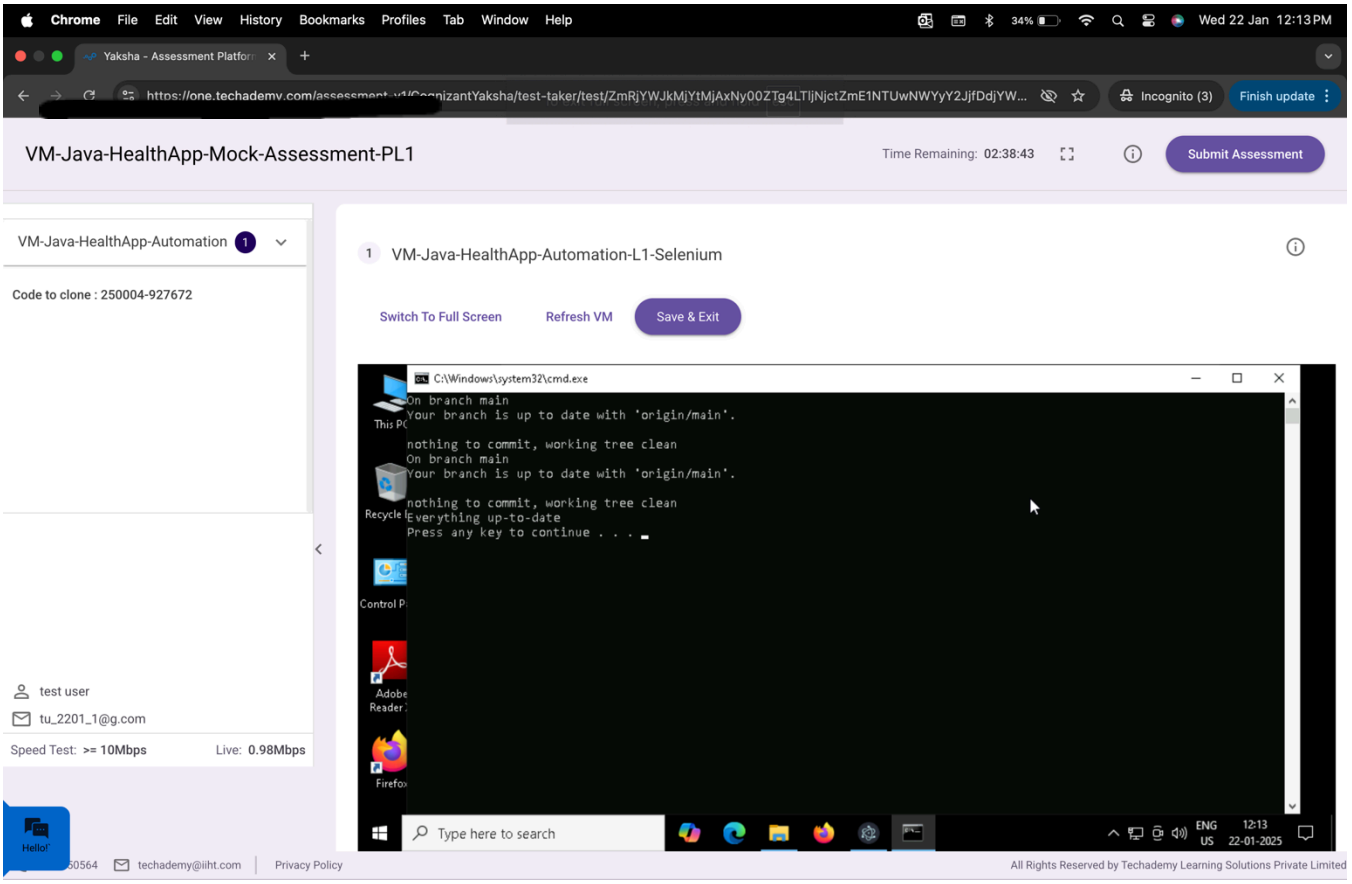


Double-click the batch file to run it. It will run the commands to push your code to GIT.



Once the code is pushed to git, you can go for 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