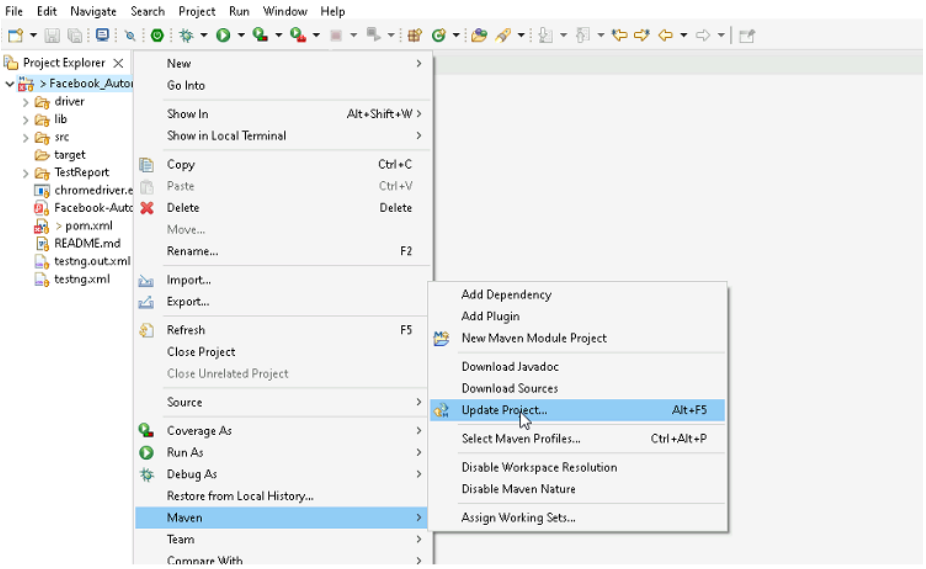
HealthApp Automation Mock-PL2

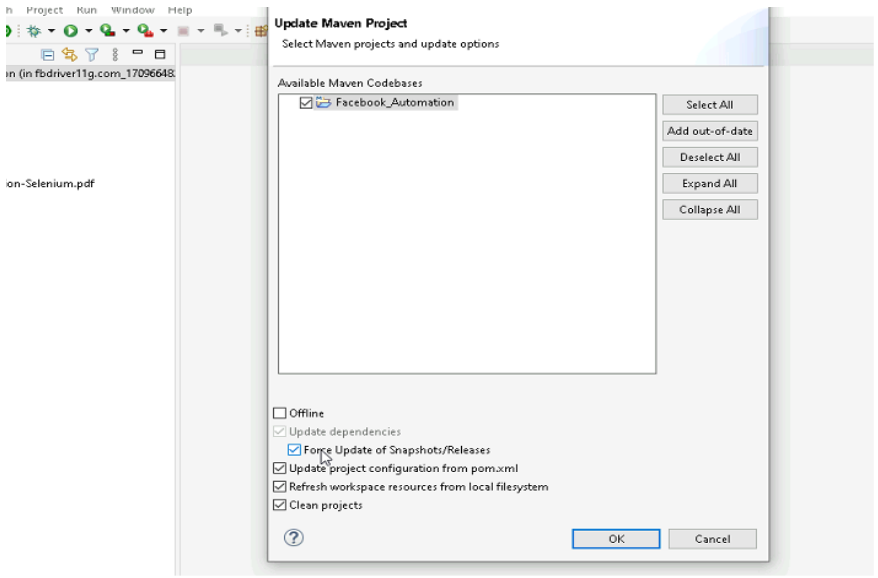
Pre-requisite:

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 affect your evaluation.

Package	Class/File	Description
src/main/java/coreUtilities/utils/	FileOperations.java	1. Contains methods to read 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	yakshaHealthAppPages.java	1. All core activities (mentioned in the list below “Key activates 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 is defined here
	expected_data.xlsx	Contains data to fill in the 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 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	Verify the title and URL of the current page.	**Test Steps:** 1. Go to URL: **https://healthapp.yaksha.com/** 2. Login as a valid credential **(username: admin, password: pass123)** **3.** click on "**SignIn**" Button	Verify the title and URL of the current page. The title should be: DanpheHealth The URL should be: https://healthapp.yaksha.com/Home/Index#/
2	Verify the pop-up and proceed to the next page in the new 1 counter page.	**Precondition:** The user must be logged in. **Test Steps:** 1. Click on the Billing tab 2. Let the "Select Counter" pop up. 3. Click on the "New 1" Counter.	Verify "Search patient" page is present.
3	Perform the keyboard operation to open the popup and verify that the popup is displayed or not.	**Precondition:** The user must be logged in and should be on "Search Patient". **Test Steps:** 1. Click on the blank space below the "Search OPD Patients" section 2. Hit "Alt + N" through the keyboard to open the add new patient popup.	Verify the form should pop up with **"Add New Patient"** as the main title of the form.
4	First Name, Middle Name, Last Name, Age and Contact Number text fields are present inside the Add New Patient form.	**Precondition:** The user must be logged in and should be on the "Search Patient" Module. The "Add New Patient" form should be open.	Validate Whether the following fields are displayed in the add new form: "First Name", "Middle Name", "Last Name", "Age" and "Contact number"
5	Enter the values from the Expected_Data.xlsx and validate the values are getting populated correctly.	**Precondition:** User must be logged in and should be on "Search Patient". "Add New Patient" form should be open. **Test Steps (for data entry only): Fetch the following values from expected_data.xlsx** **Test** 1. Fetch and enter the value in the First Name Field. 2. Fetch and enter the value in the Middle Name Field. 3. Fetch and enter the value in the Last Name Field. 4. Fetch and enter the value in the Age Field. 5. Fetch and enter the value in the Contact Number Field.	Validate the Values entered in First Name, Middle Name, Last name, Age and Contact number text field as per Expected_Data.xlsx.
6	Simply select India from the **country** dropdown menu and validate that "India" is selected.	**Precondition:** User must be logged in and should be on "Search Patient". "Add New Patient" form should be open. **Test Steps:** 1. Select the country as "India" in the country dropdown.	Verify that the User should be able to select India as a country from the country dropdown menu.
7	Open a new instance of "Add New Patient" and press the Register and Billing button without filling in any data to display the error message and verify the message.	**Precondition:** User must be logged in and should be on "Search Patient". "Add New Patient" form should be open. **Test Steps:** 1. Close the previously filled "Add New Patient" form. 2. Open a new instance of the "Add New Patient" form by clicking on the Alt+N button. 3.do not fill in any data and Click on the "Register and Billing" button. 4. Click on the Close button.	Verify the error message "Some of the inputs are invalid. Please check and try again. !" popup after the 3rd test step.
8	Verify that the dispensary arrow is expanding, and submodules are being displayed.	**Precondition:** The user must be logged in. **Test Steps:** 1. Click on the down arrow(Toggle key) of the dispensary module.	Dispensary arrow should expand, and submodules should be displayed. Expected subModules are: **Prescription, Sale, Stock,**

			Counter, Reports, Patient Consumption
9	Verify that the dispensary arrow is minimised properly.	**Precondition:** The user must be logged in. **Test Steps:** 1. Click on the up arrow (Toggle key) of the dispensary module.	Dispensary arrow should minimize to hind the displayed submodules.
10	Verify that the left navigation menu scroll bar is scrolling from the Dispensary module to the settings module.	**Precondition:** The user must be logged in. **Test Steps:** 1. Select the dispensary module as the first module. 2. Scroll up to the settings module.	Scrolling should happen from the Dispensary module to the settings module.
11	Search for a patient name and view the details of the selected patient.	**Precondition:** The user must be logged in. **Test Steps:** 1. Go to Ip Billing tab 2. Search "**Akshit**" in the **Search OPD Patients** search bar 3. Select the patient name from drop-down. 4. Verify if the details are visible on the page.	Particular patient details should be displayed.
12	Take a screenshot for the Doctor page.	**Precondition:** The user must be logged in. **Test Steps:** 1. Click on the Doctor module. 2. Take a screenshot of the Doctor's home page.	The user should be able to take the screenshot.
13	Find and Click the Deactivate button to manage a user or employee.	**Precondition:** The user must be logged in. **Test Steps:** 1. Go to the settings module 2. Click on security 3. Select any employee and click on the Deactivate button. 5. Click on Cancel in the alert popup	Users should be able to Click deactivate button for a user or employee by managing the alert popup.
14	Verify the tooltip of an element which is present inside the Appointment page.	**Precondition:** The user must be logged in. **Test Steps:** 1. Click on the Appointment module. 2. Mouse hover to the Keyboard button. 3. Verify the tooltip message.	The tooltip message should be " ALT + N = New Patient".

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 scripts using Java and Selenium to automate all the steps in the above question. In other words, the automation script should perform all the 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

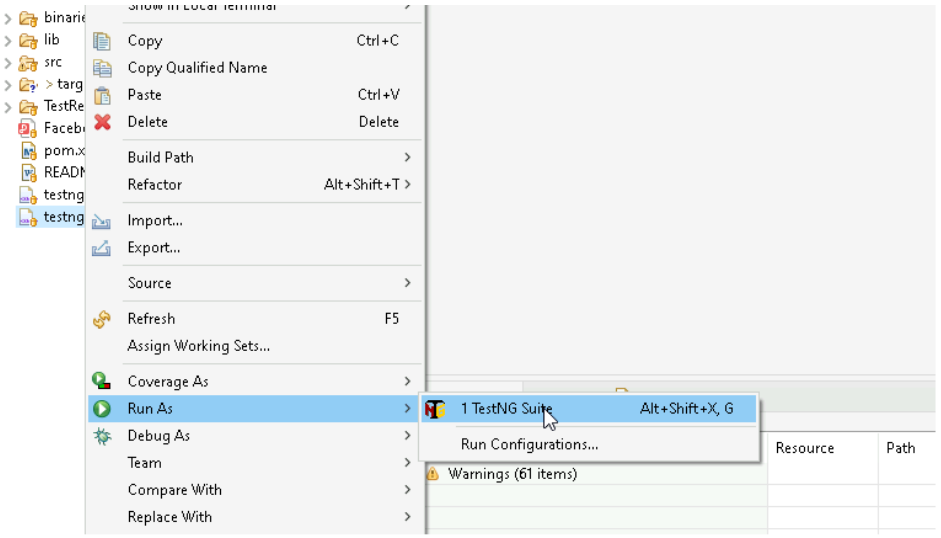
1.1 CODE QUALITY/OPTIMIZATIONS

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

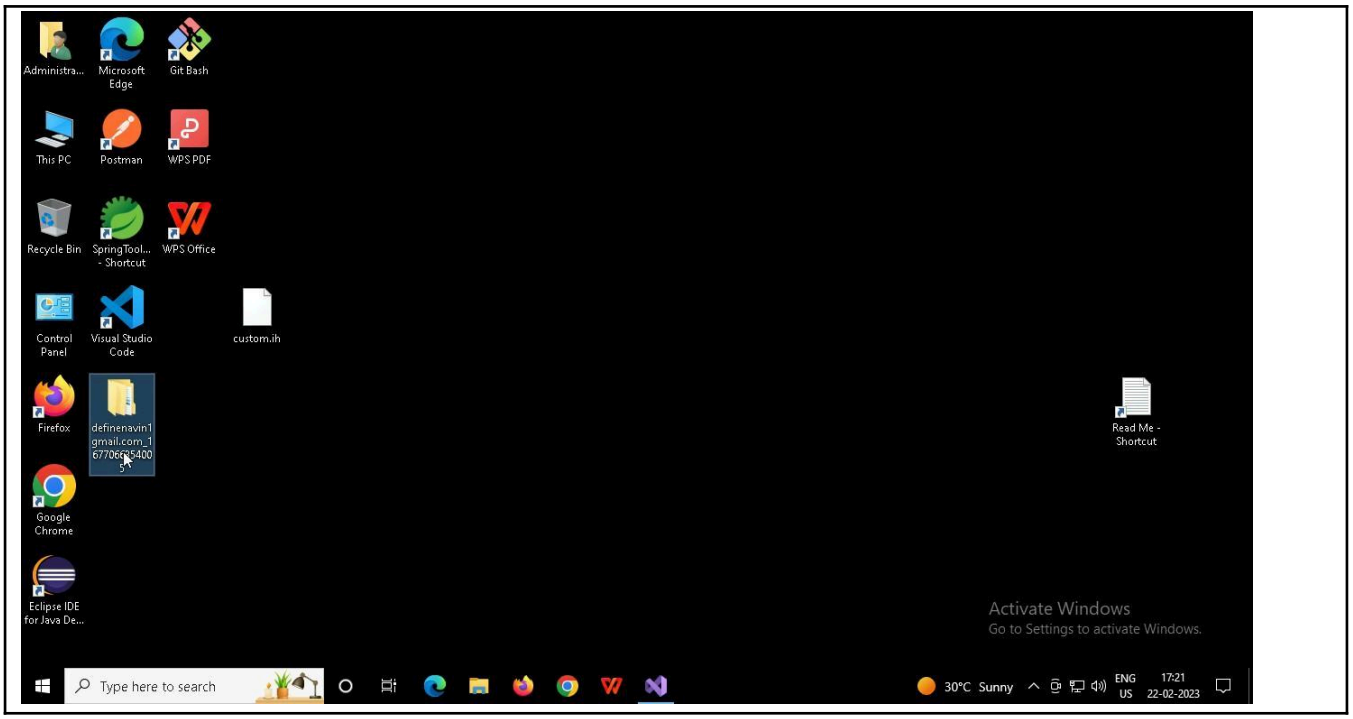
EXECUTION STEPS TO FOLLOW

1. **You are mandatory required to run test cases for applications before final submission. Without this project evaluation will not happen.**

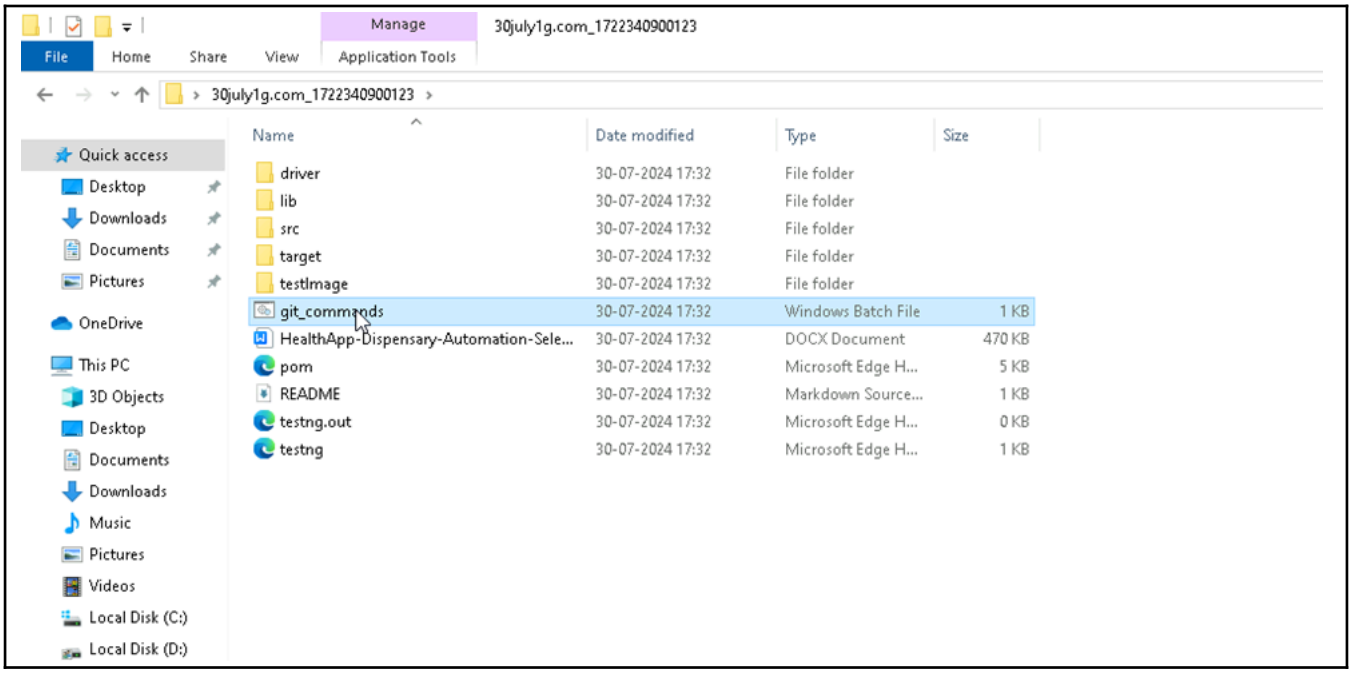
2. **You can launch test cases any time as follows: Right-click on testng.xml and run TestNGSuite**



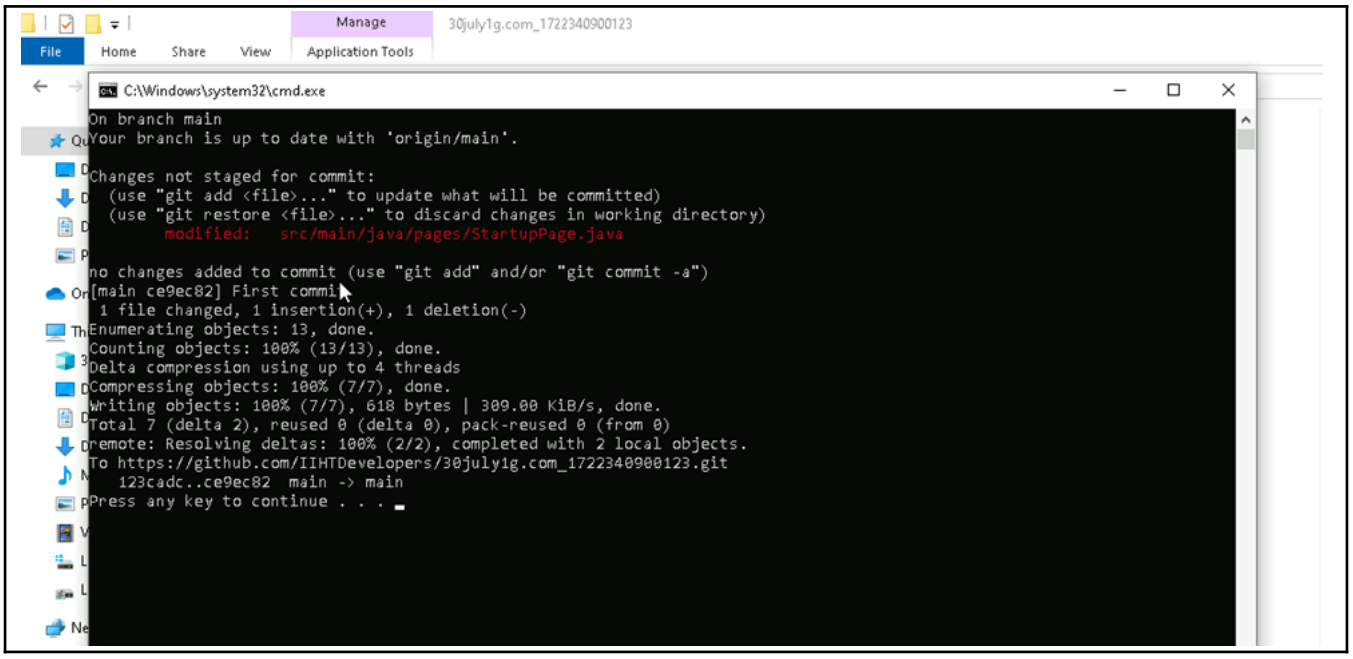
3. **Before final submission, you are also required to push your code to GIT. Following are the steps to follow:**
4.



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

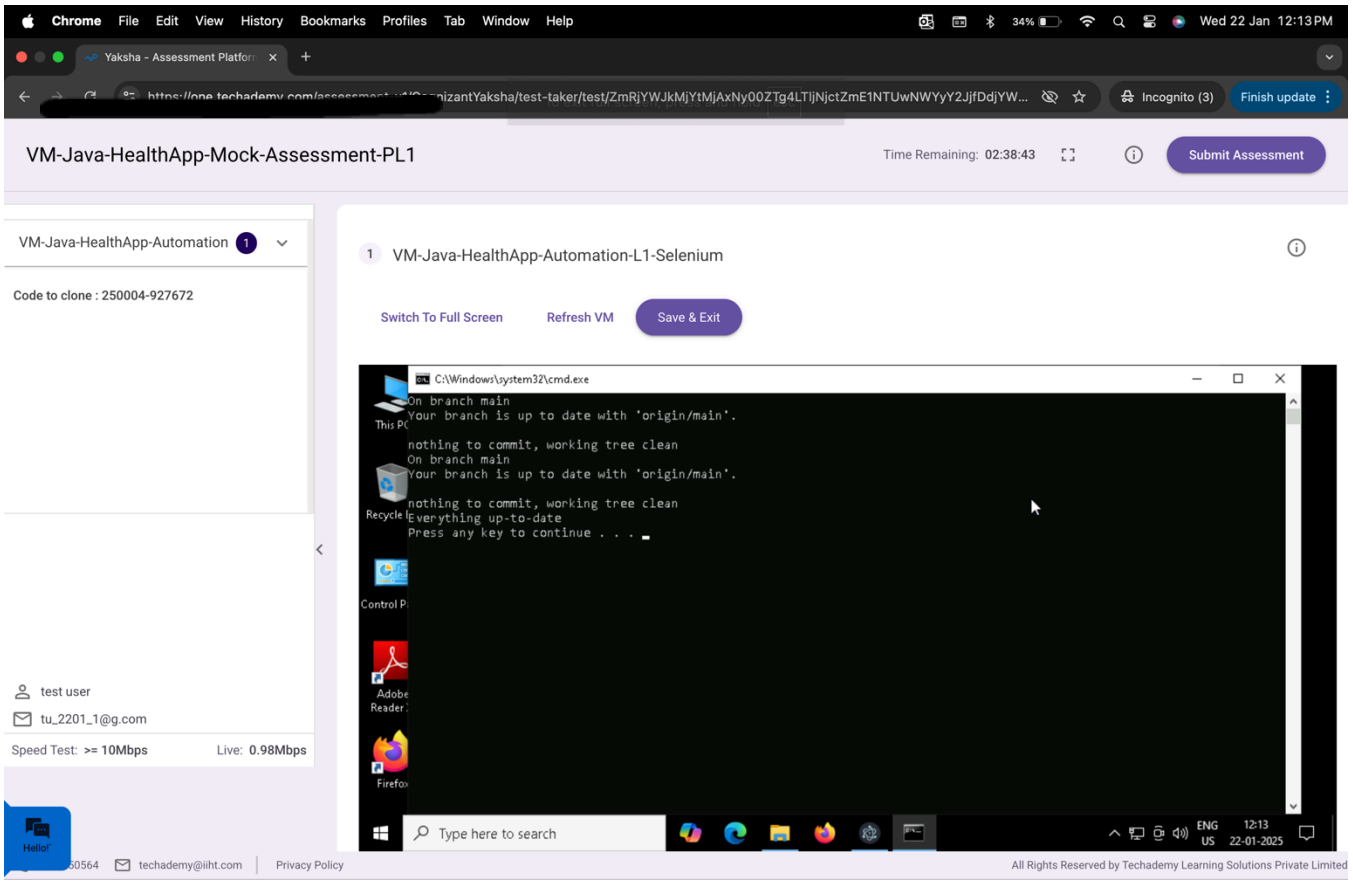


Double-click the batch file to run it. It will run the commands to push your code to GIT.



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