SIT707 Software Quality and Testing

Pass Task: Selenium drives chromium browser!

Overview

Now that you have everything setup, we can start using Selenium to automate browser using Java programming. In this task, you will use Selenium API to do several actions including starting the chromium browser, opening a URL, resizing it and finally closing it.

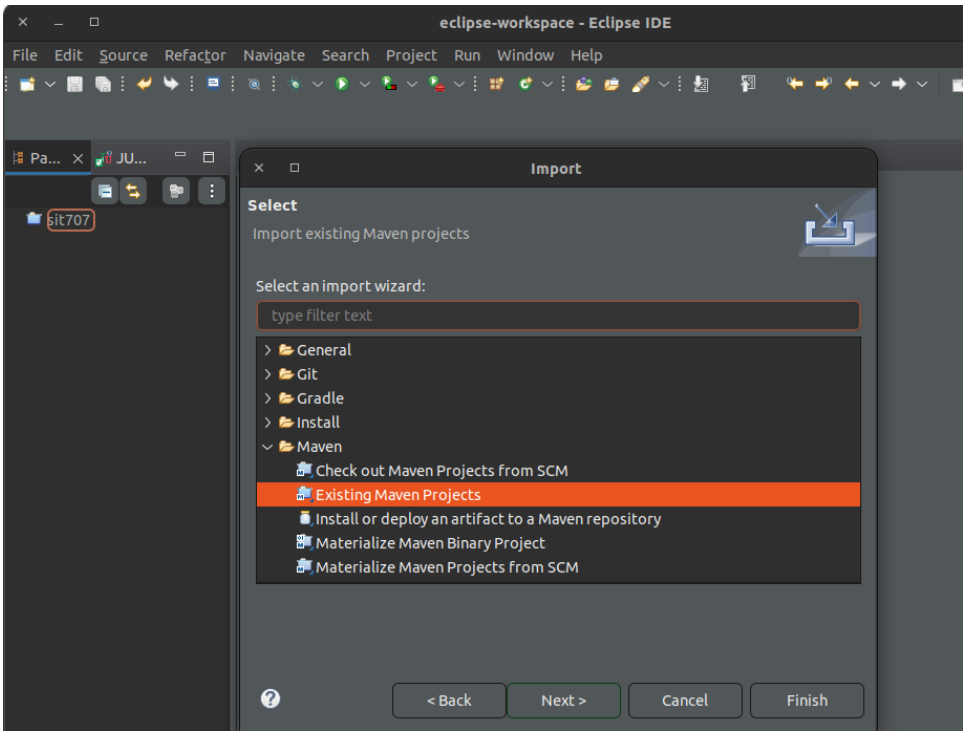
Task objective

For this task you will need to study the existing project task1_2P.zip which has SeleniumOperations.java file where you need to fill in the missing lines in the function *open_chrome_loadpage_resize_close()*. You must run Main.java file in Eclipse and capture the output.

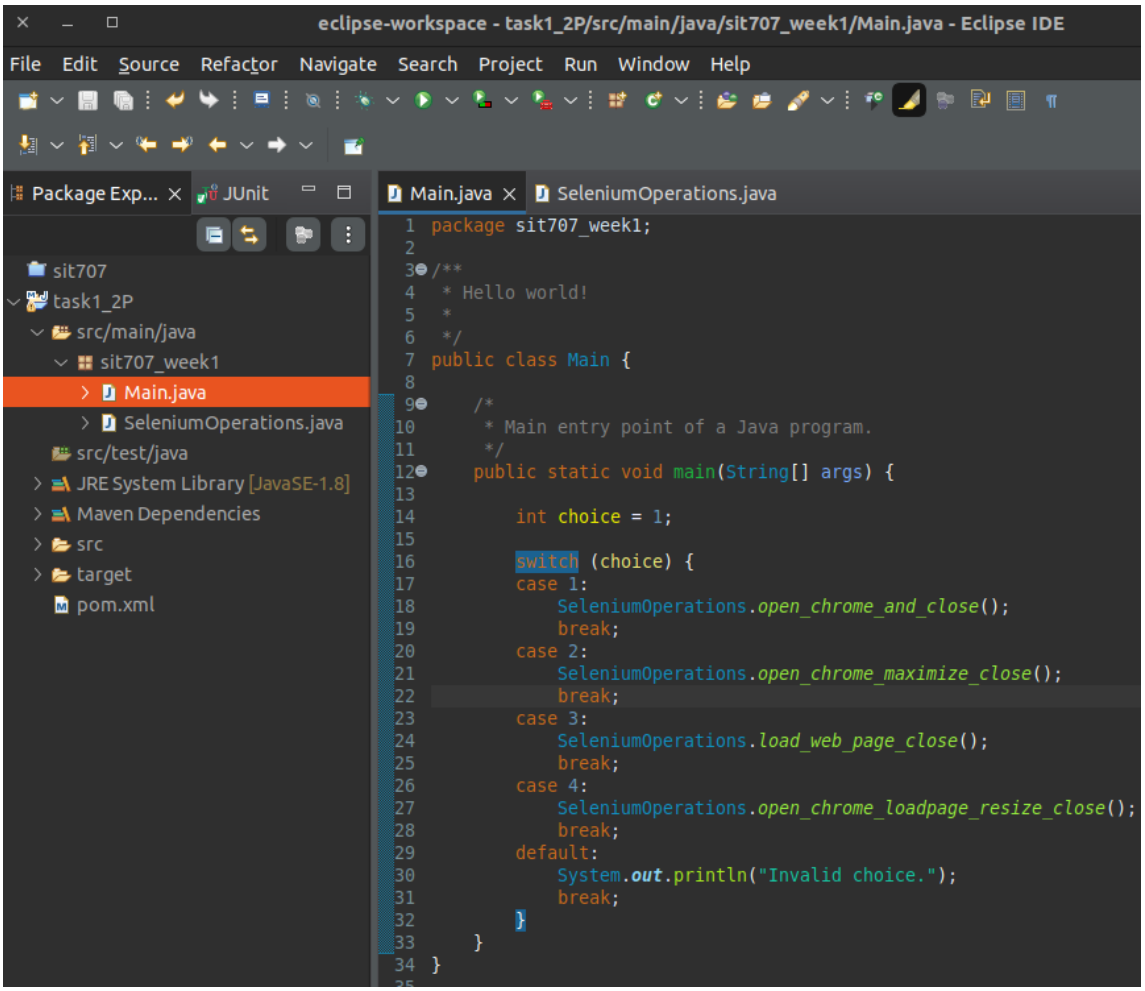
Instructions

Follow steps as below -

1. Download task1_2P.zip Java project and unzip it in a common folder (say, java_projects) which you will be using to store all the weekly projects.
2. Import the project (as a *maven* project) in Eclipse IDE



3. Observe different functions by altering the *choice* variable in Main.java and run Main.java file. For instance, *choice = 1* activates *case 1* which is ***SeleniumOperations.open_chorme_and_close()*** function defined in the SeleniumOperations.java file which opens chrome browser, waits a while and closes it.



4. You can observe the behaviors of case 1, 2 and 3. Your task is to complete the case 4 function **open_change_loadpage_resize_close()**, shown below in red color text which includes below steps:
    A. Modify the Hello message at the beginning of the function with your student ID and full name.
    B. Open URL https://www.google.com using Selenium API.
    C. Resize the chromium browser window size using Selenium API to 640x480 and sleep for 2 seconds.
    D. Resize the chromium browser window to double the previous step to 1280x960 and sleep for 2 seconds.
    E. Run the Main.java with choice=4 and test your program works.
    F. Once done, you have to run the Main.java file with choice set to 4, capture Eclipse IDE's console output and the Selenium controlled chromium browser.

```
public static void open_chrome_loadpage_resize_close() {
// Fill your personal information.
```

```
System.out.println("Hello from <Student ID>, <Full name>");

// Step 1: Locate chrome driver folder in the local drive.
System.setProperty("webdriver.chrome.driver",
"/home/mahabib/java_lib/chromedriver-linux64/chromedriver");

// Step 2: Use above chrome driver to open up a chromium browser.
System.out.println("Fire up chrome browser.");
WebDriver driver = new ChromeDriver();

System.out.println("Driver info: " + driver);

sleep(2);

/*
 * Load google page.
 */
// Write code here.

/*
 * Set window size manually to 640x480 and wait 2 second.
 * Hint: Explore Selenium window functions to select one which allows to pass
window size in WxH.
 */
// write code

/*
 * Double window size manually to 1280x960 and wait 2 second.
 */
// write code

// Sleep a while
sleep(2);

// close chrome driver
driver.close();
}
```

5. Upload your folder to your github account and take a screenshot. Find instructions in the unit site on how to create a new empty project (say, task1_2P_<studentID>) in github and upload your Eclipse Maven project with the same name in the newly created github empty project.
6. Finally, submit a pdf combining below items -

A. A screenshot of your screen showing Eclipse IDE console output
B. A screenshot of your screen showing chromium browser opened by Selenium from your code
C. Your program's source code (SeleniumOperations.java)
D. A screenshot of your github page where your latest project folder is pushed.

You should discuss the following key ideas while preparing the submission items:

- Java programming (main function, static keyword)
- Selenium automates browsers (Selenium API calls w.r.t. web-driver).

Consolidate outputs following the submission details above into a single PDF file.

Submit your work

When you are ready, login to OnTrack and submit your pdf which consolidates all the items mentioned in the submission detail section above. Remember to save and backup your work.

Complete your work

After your submission, your OnTrack reviewer (tutor) will review your submission and give you feedback in about 5 business days. Your reviewer may further ask you some questions on the weekly topics and/or about your submissions. You are required to address your OnTrack reviewer's questions as a form of task discussions. Please frequently login to OnTrack for the task ***Discuss/Demonstrate*** or ***Resubmit*** equivalent to fix your work (if needed) based on the feedback to get your task signed as ***Complete***.