

### 3.You are given a project to implement soft and hard assertions on your test cases

#### Step 1: Creating a simple Java project

- Open Eclipse.
- Go to the **File** menu. Select **New->Java Project**.
- Enter the project name as **Test Assertions**. Click on **Next**.
- This will create the project files in the Project Explorer.

#### Step 2: Downloading Selenium WebDriver jar, chromedriver.exe, and firefoxdriver.exe

- Selenium WebDriver is already installed in your Practice lab. Refer QA to QE lab guide for Phase 2 for more information.
- Go to <https://selenium.dev/thirdparty/>
- To download chromedriver.exe, click on Google ChromeDriver and select the appropriate version as per your Chrome version.
- To download firefoxdriver.exe, click on Microsoft GeckoDriver.

#### Step 3: Adding the WebDriver dependency in the project

- In the Project Explorer, right-click on **Test Assertions**.
- Select **Properties**. Select **Java Build Path** from the list. Go to **Libraries**.
- Click on **Add External JARs** and browse the location where you have downloaded the JAR files.
- Select JARs from the **root** folder and the **libs** folder.
- Click on **Apply and Close**.
- Copy the chromedriver.exe and geckodriver.exe, and paste it to your project creating a resource folder.

#### Step 4: Installing TestNG

- TestNG is already installed in your Practice labs. To learn about its directory details you can refer to the lab guide for Phase 2.

#### Step 5: Adding TestNG libraries to the Class Path

- In the Project Explorer, right-click on **Test Assertions**.
- Select **Properties**. Select **Java Build Path** from the list. Go to **Libraries**.

- Click on **Add Library**. Select **TestNG**. Click on **Next**. Click on **Finish**.
- Click on **Apply and Close**.

#### Step 6: Creating a Java class named ParallelTest.java

- In the Project Explorer, expand **Test Assertions->Java Resources**
- Right-click on **src** and select **New->Class**
- In **Class Name**, enter **Assertions** and click on **Finish**. In **Package Name**, enter **com.assert** and click on **Finish**.
- Enter the following code:

```
package com.asserts;
```

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.Test;
import org.testng.asserts.SoftAssert;
```

```
public class Assertions {
```

```
    SoftAssert soft = new SoftAssert();
    WebDriver driver;
    @Test
    public void Launch() {
        System.setProperty("webdriver.chrome.driver", "./Resources/chromedriver.exe");
        driver = new ChromeDriver();
        try {
            Thread.sleep(3000);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
```

```
    @Test(dependsOnMethods = { "Launch" })
    public void Facebook() {
        driver.get("https://www.facebook.com");
        soft.assertEquals("FB Title", driver.getTitle());
        try {
            Thread.sleep(2000);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
```

```
    @Test(dependsOnMethods = { "Facebook" })
    public void Login() {
        driver.findElement(By.id("email")).sendKeys("ravi10thstudent@gmail.com");
        driver.findElement(By.id("pass")).sendKeys("12345");
        driver.findElement(By.id("loginbutton")).click();
        soft.assertAll();
    }
```

```

        try {
            Thread.sleep(3000);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

#### Step 7: Running the project

- Right-click on **Assertions** class. Click on **TestNG->Convert to TestNG**.
- Click on **Finish**. It will create a **TestNG.xml** file. Open that file.
- Right-click on the screen. Select **Run As ->TestNG Suite**.

#### Step 8: Pushing the code to your GitHub repositories

Open your command prompt and navigate to the folder where you have created your files.

cd <folder path>

Initialize your repository using the following command:

git init

Add all the files to your git repository using the following command:

git add .

Commit the changes using the following command:

git commit . -m "Changes have been committed."

Push the files to the folder you initially created using the following command:

git push -u origin master