APPLICATION DEVELOPMENT AND DEPLOYMENT ARCHITECTURE

CSI3025

DIGITAL ASSIGNMENT – 03

NAME: Brian E Shilo **DATE: 21-02-24**

REG NO: 21MIC0131 **SLOT: L23+L24**

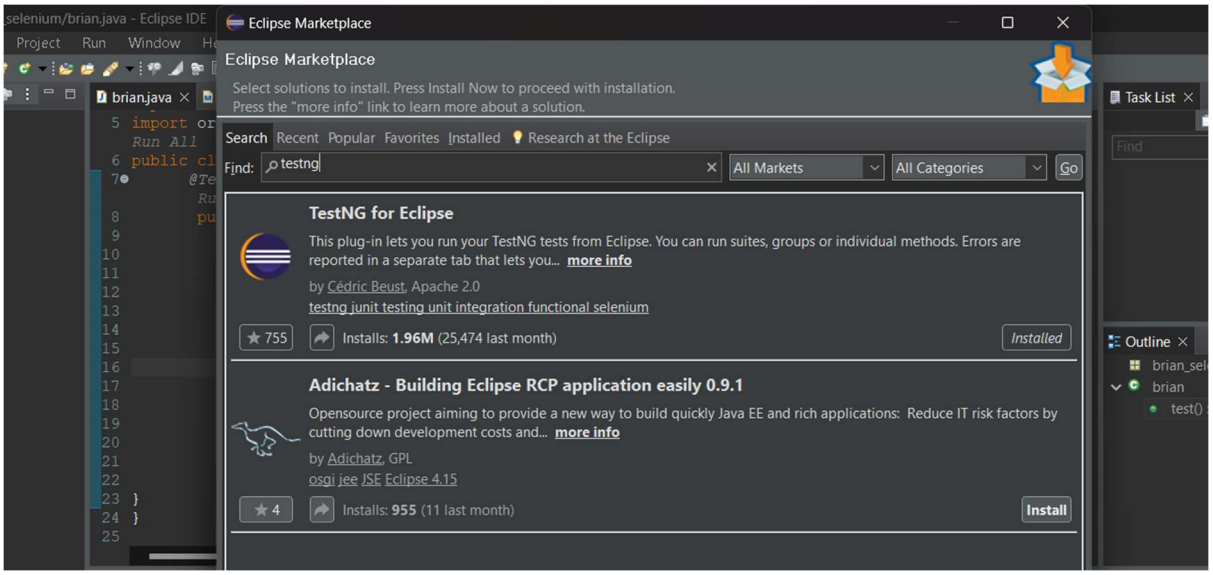
Aim:

Create a Maven project for Selenium automation to perform the login functionality on a web application. Choose a simple web application. (Here I have chosen VTOP)

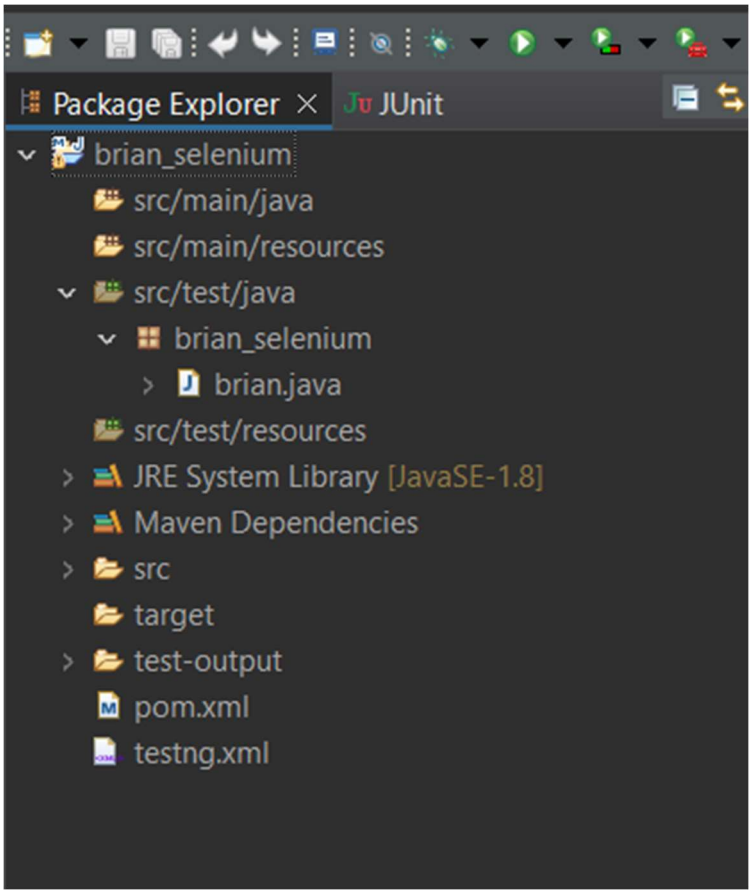
Procedure:

1) Install the gecko driver zip file and extract the same.

2) Now in Eclipse we need to ensure that TestNG is installed or not if not, it must be installed. You can check the same by clicking on help-Eclipse Marketplace Now you can check if installed or not. If not, it must be installed.



3) Now create a new Maven project and provide group id and artifact id of your choice. Project Created:



4) Now after the project is created, create a new package followed by a new class file in src/test/java.

5) Now in the java file created add in the code given below.

brian.java :

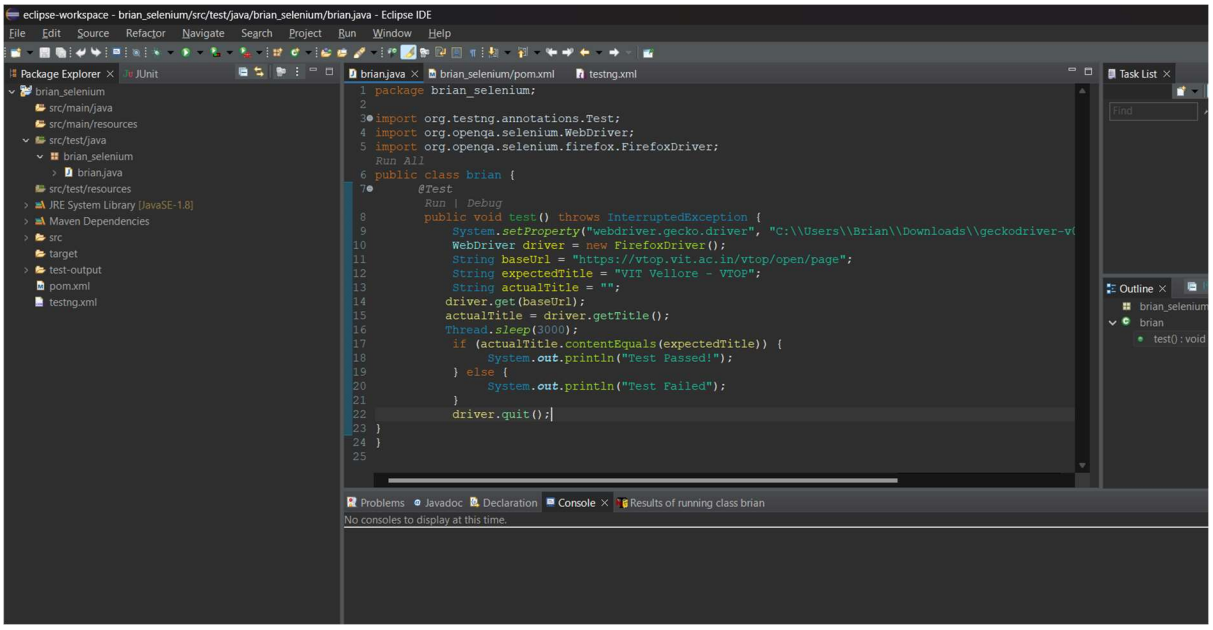
```
package brian_selenium;

import org.testng.annotations.Test;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;

public class brian {
      @Test

       public void test() throws InterruptedException {
           System.setProperty("webdriver.gecko.driver",
"C:\\Users\\Brian\\Downloads\\geckodriver-v0.34.0-win32\\geckodriver.exe");
```

```
            WebDriver driver = new FirefoxDriver();
            String baseUrl = "https://vtop.vit.ac.in/vtop/open/page";
            String expectedTitle = "VIT Vellore - VTOP";
            String actualTitle = "";
           driver.get(baseUrl);
           actualTitle = driver.getTitle();
           Thread.sleep(3000);
            if (actualTitle.contentEquals(expectedTitle)) {
                 System.out.println("Test Passed!");
            } else {
                 System.out.println("Test Failed");
            }
            driver.quit();
}
}
```



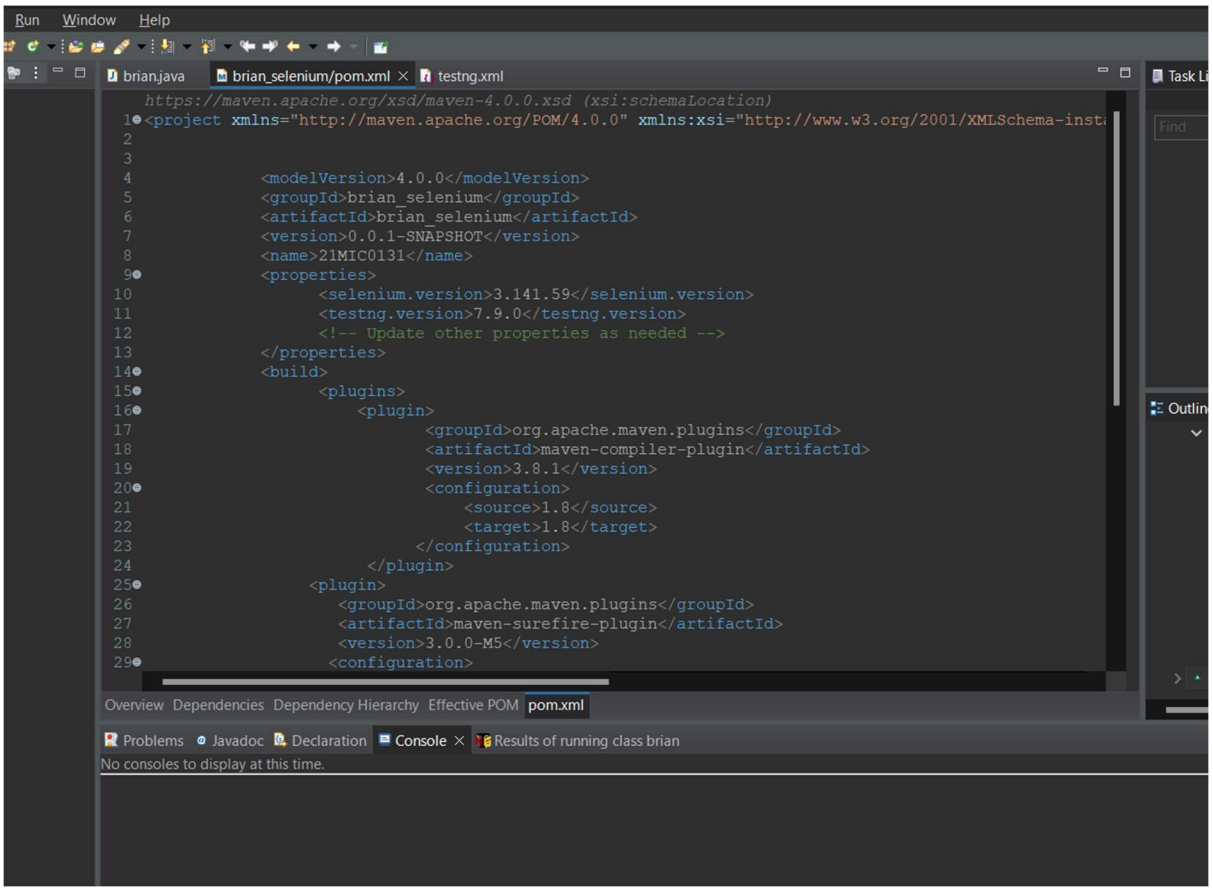
6) Now in the pom.xml the required properties, plugins and dependencies are to be added.

Pom.xml :

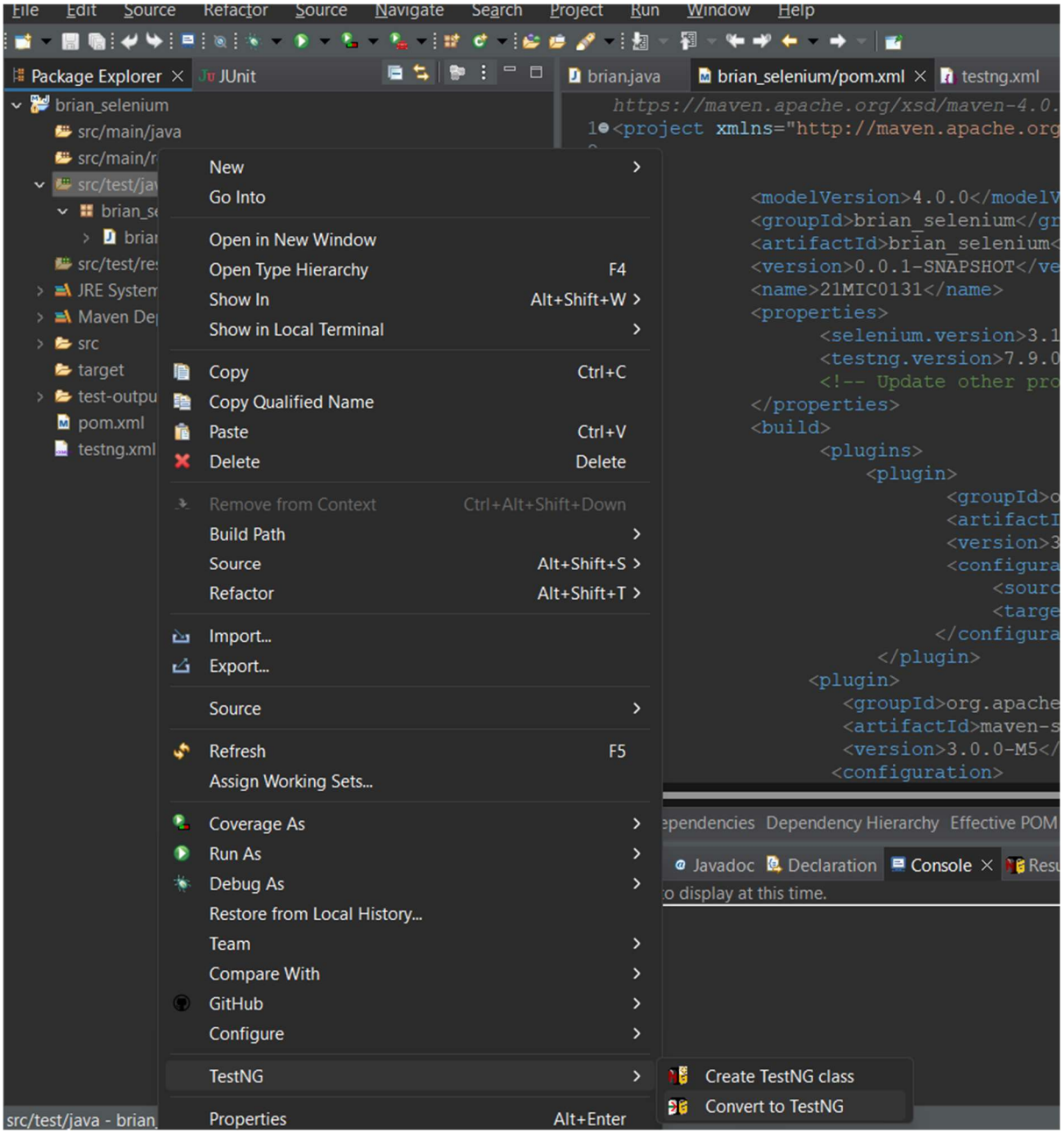
```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">


            <modelVersion>4.0.0</modelVersion>
            <groupId>brian_selenium</groupId>
            <artifactId>brian_selenium</artifactId>
            <version>0.0.1-SNAPSHOT</version>
            <name>21MIC0131</name>
            <properties>
```

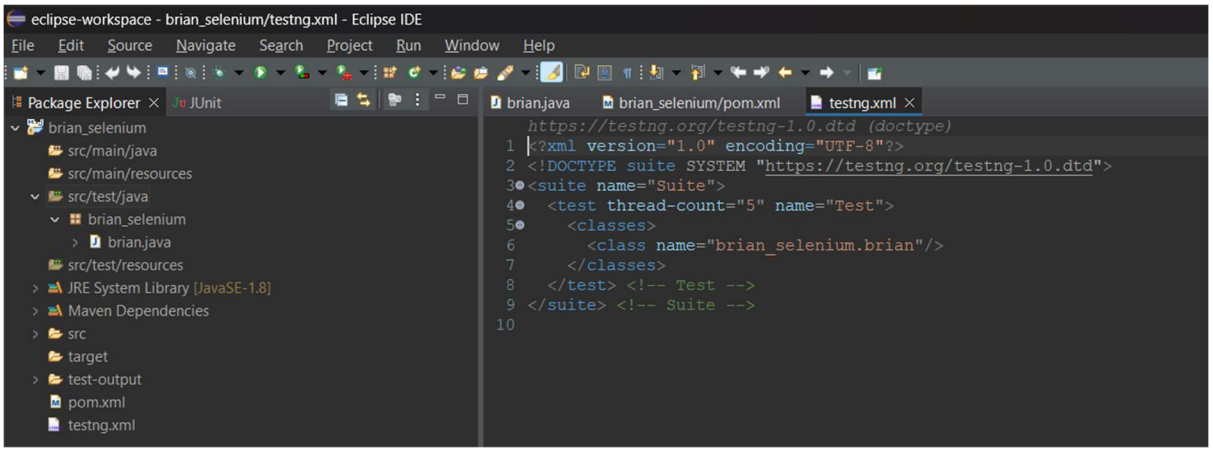
```
                <selenium.version>3.141.59</selenium.version>
                <testng.version>7.9.0</testng.version>
                <!-- Update other properties as needed -->
            </properties>
            <build>
                  <plugins>
                      <plugin>
                            <groupId>org.apache.maven.plugins</groupId>
                            <artifactId>maven-compiler-plugin</artifactId>
                            <version>3.8.1</version>
                            <configuration>
                                <source>1.8</source>
                                <target>1.8</target>
                           </configuration>
                       </plugin>
                   <plugin>
                      <groupId>org.apache.maven.plugins</groupId>
                      <artifactId>maven-surefire-plugin</artifactId>
                      <version>3.0.0-M5</version>
                     <configuration>
                         <suiteXmlFiles>
                               <suiteXmlFile>testng.xml</suiteXmlFile>
                         </suiteXmlFiles>
                     </configuration>
              </plugin>
              <!-- Add other plugins as needed -->
          </plugins>
      </build>
      <dependencies>
             <dependency>
                  <groupId>org.seleniumhq.selenium</groupId>
                  <artifactId>selenium-java</artifactId>
                  <version>${selenium.version}</version>
            </dependency>
            <!-- https://mvnrepository.com/artifact/org.testng/testng -->
                    <dependency>
                            <groupId>org.testng</groupId>
                            <artifactId>testng</artifactId>
                            <version>7.9.0</version>
                            <scope>test</scope>
                    </dependency>
                    <!-- Add other dependencies as needed -->
             </dependencies>
</project>
```


Run
Window
Help
brian.java
brian_selenium/pom.xml
testng.xml
https://maven.apache.org/xsd/maven-4.0.0.xsd (xsi:schemaLocation)
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-insta
<modelVersion>4.0.0</modelVersion>
<groupId>brian_selenium</groupId>
<artifactId>brian_selenium</artifactId>
<version>0.0.1-SNAPSHOT</version>
<name>21MIC0131</name>
<properties>
<selenium.version>3.141.59</selenium.version>
<testng.version>7.9.0</testng.version>
<!-- Update other properties as needed -->
</properties>
<build>
<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-compiler-plugin</artifactId>
<version>3.8.1</version>
<configuration>
<source>1.8</source>
<target>1.8</target>
</configuration>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-surefire-plugin</artifactId>
<version>3.0.0-M5</version>
<configuration>
Overview
Dependencies
Dependency Hierarchy
Effective POM
pom.xml
Problems
Javadoc
Declaration
Console
Results of running class brian
No consoles to display at this time.

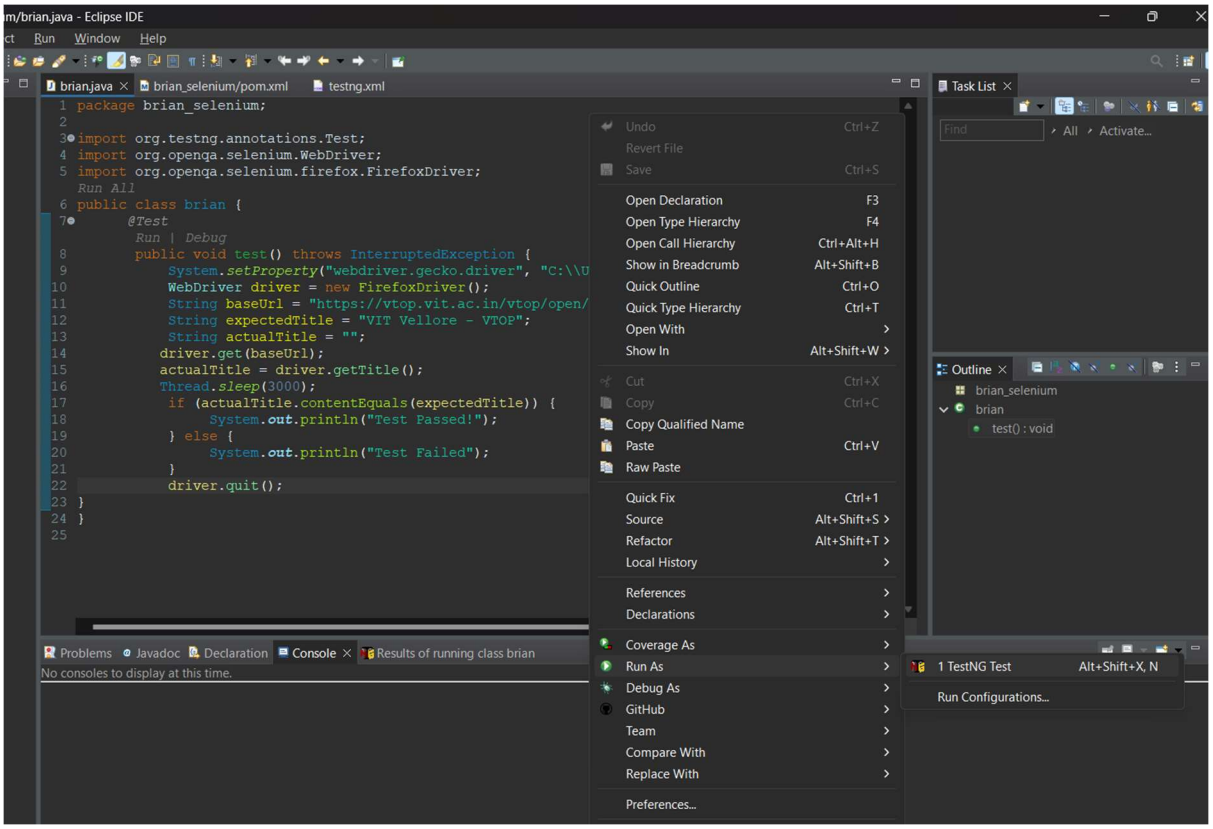
7) Now need to create the testing.xml file. In src/test/java on right click TestNG option is available followed by which we need to click on “convert to TestNG”



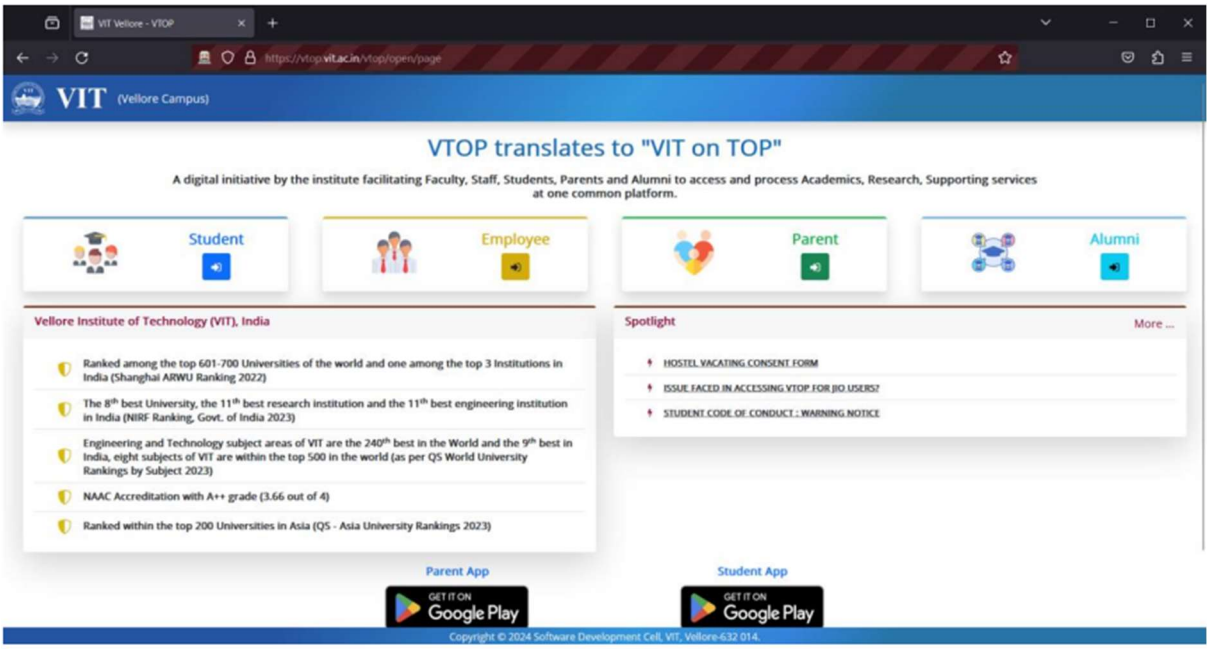
Now it can be noted that a new testing.xml file is created.



8) Now to run the test we need to go to the java file created and on right click and then run as TestNG option can be found.



9) Now when the TestNG is performed it can be noted that the page automatically opens to the browser and redirects to the page we mentioned in the code.



After which the result of the test can also be viewed in the console for crosschecking if the test was successful or not.

Result:

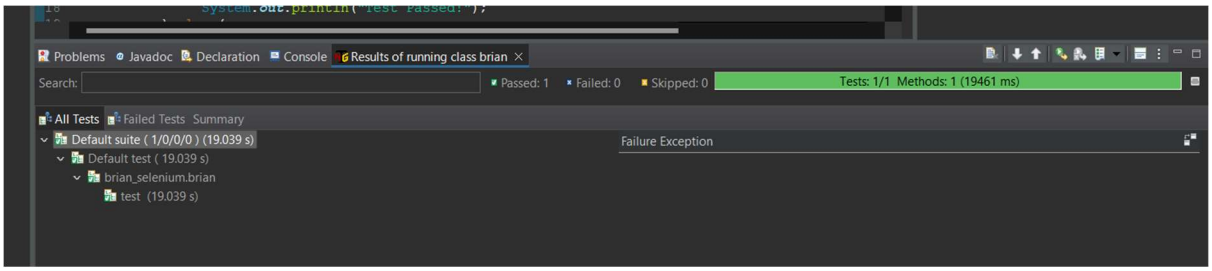
```
15          actualTitle = driver.getTitle();
16          Thread.sleep(3000);
17           if (actualTitle.contentEquals(expectedTitle)) {
18                System.out.println("Test Passed!");
```

Problems Javadoc Declaration Console × Results of running class brian

```
<terminated> brian [TestNG] C:\Program Files\Java\jdk-17\bin\javaw.exe  (22-Feb-2024, 12:52:23 pm – 12:52:43 pm) [pid: 17732]
INFO: Detected dialect: W3C
Test Passed!
1708586562677   Marionette      INFO    Stopped listening on port 51547
Dynamically enable window occlusion 1
PASSED: brian_selenium.brian.test

===============================================
    Default test
    Tests run: 1, Failures: 0, Skips: 0
===============================================


===============================================
Default suite
Total tests run: 1, Passes: 1, Failures: 0, Skips: 0
===============================================
```



.....