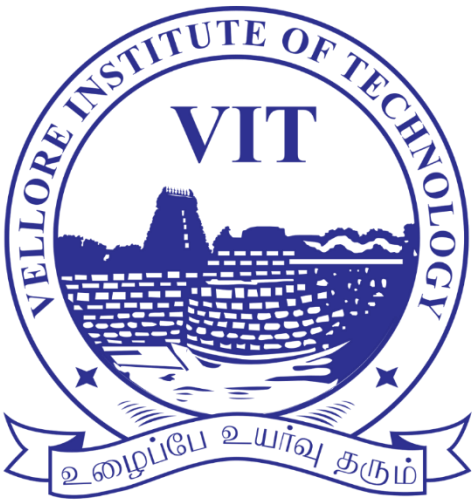


CSI3025

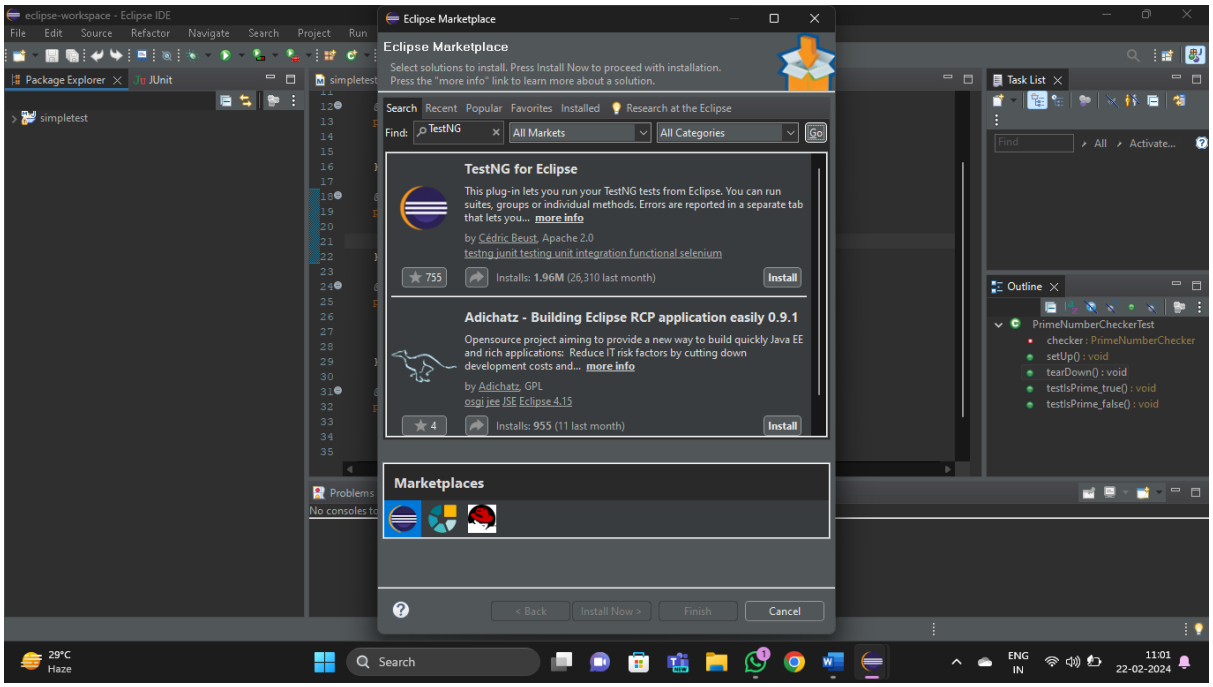
Application Development and Deployment Architecture

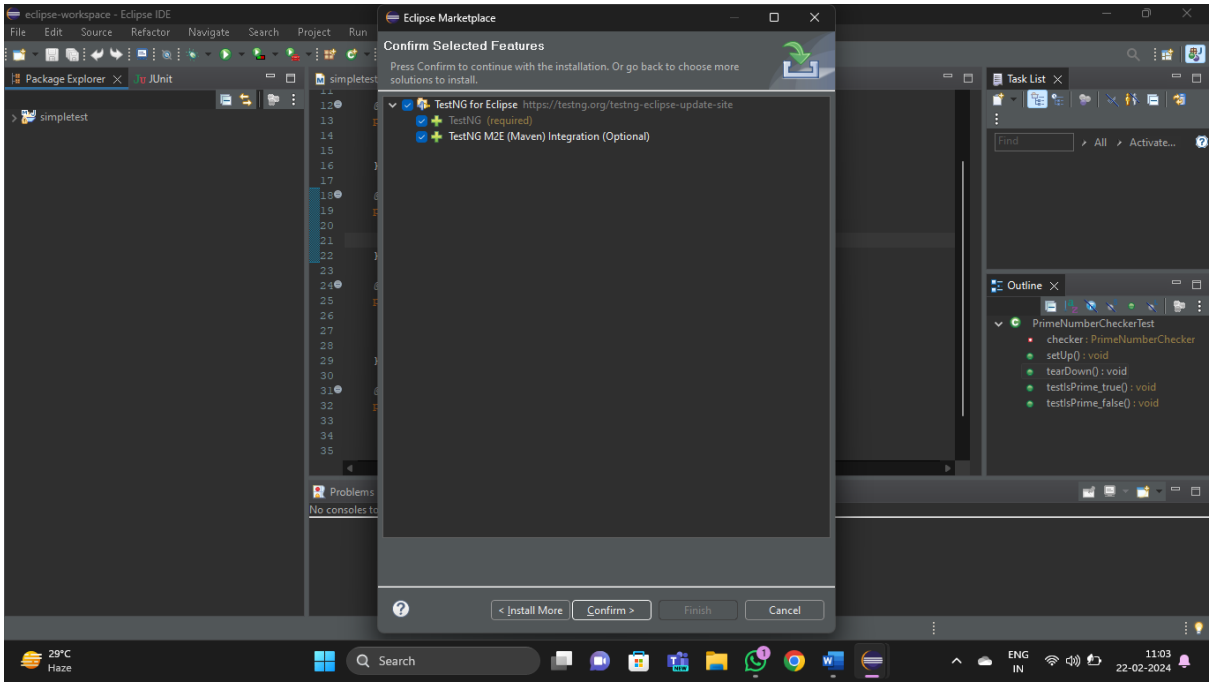
Winter Semester 2023 – ‘24

Digital Assignment - 3

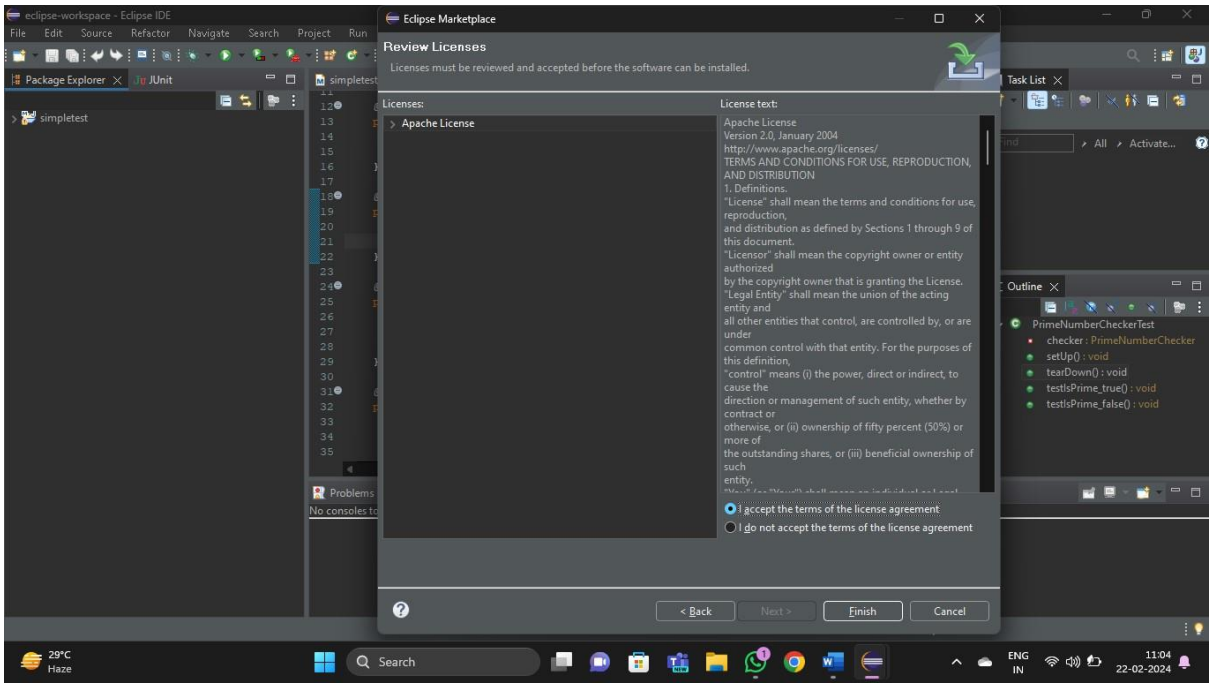
SARAGA S
20MIC0081

Step – 1: Open Eclipse -> Help -> Eclipse Marketplace -> Search TestNG and install.

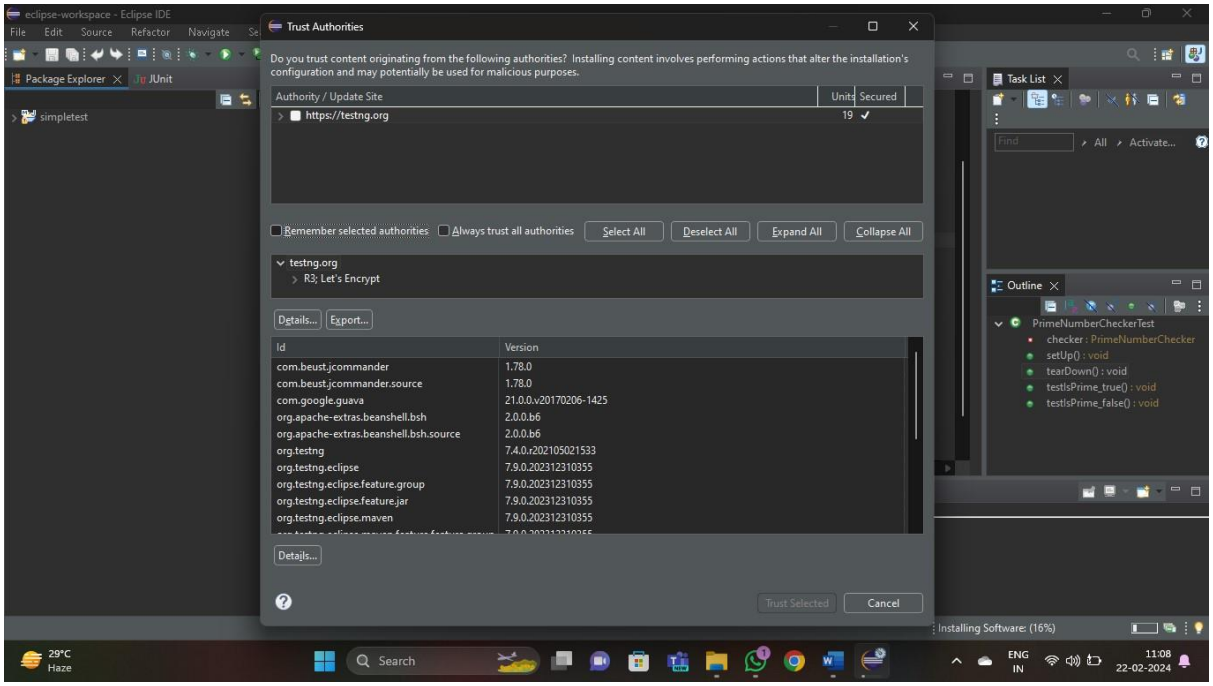


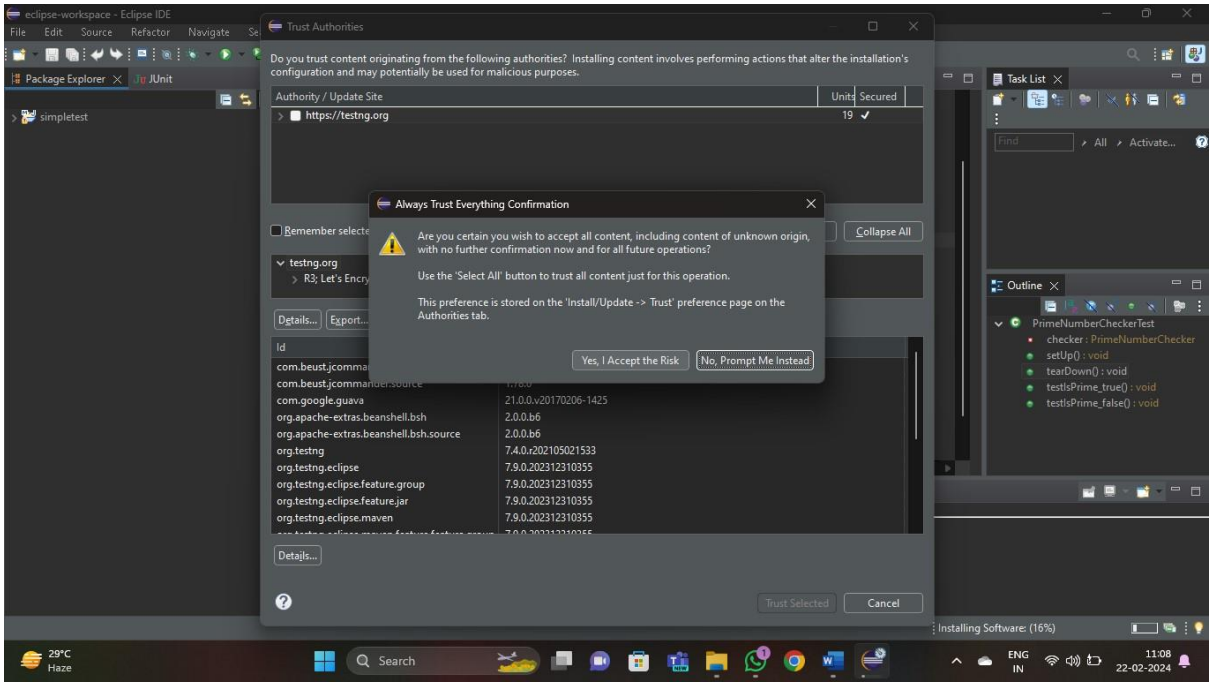


Step – 2: Accept all terms -> Finish.

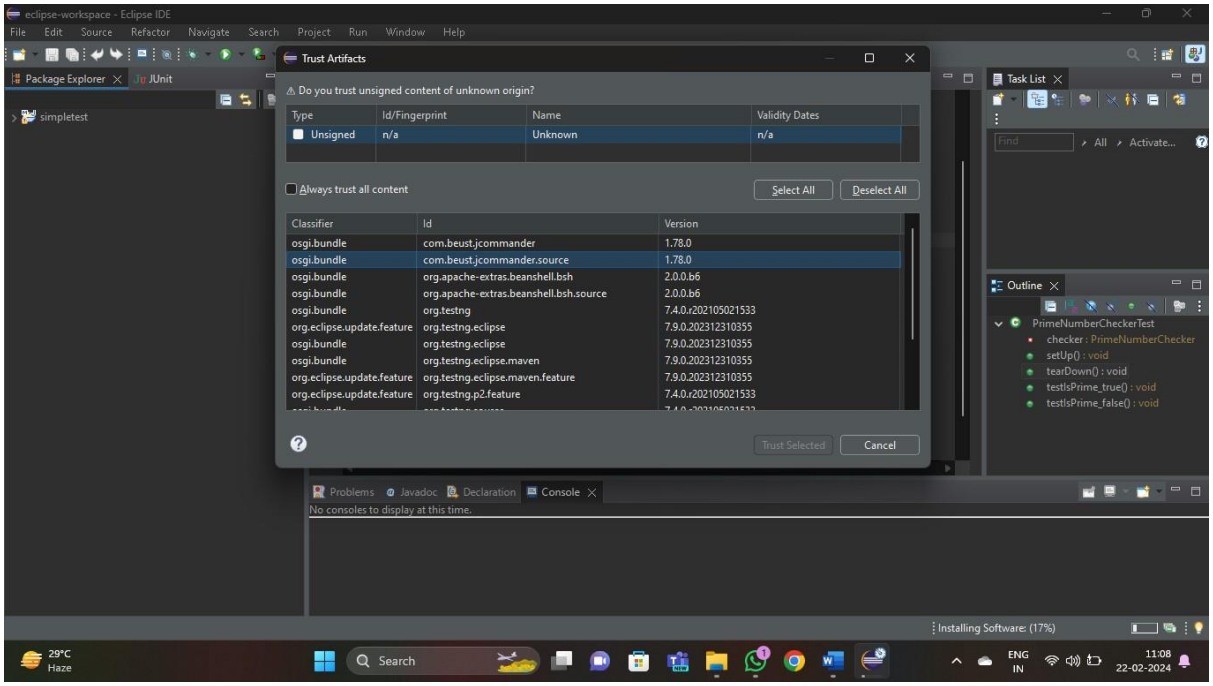


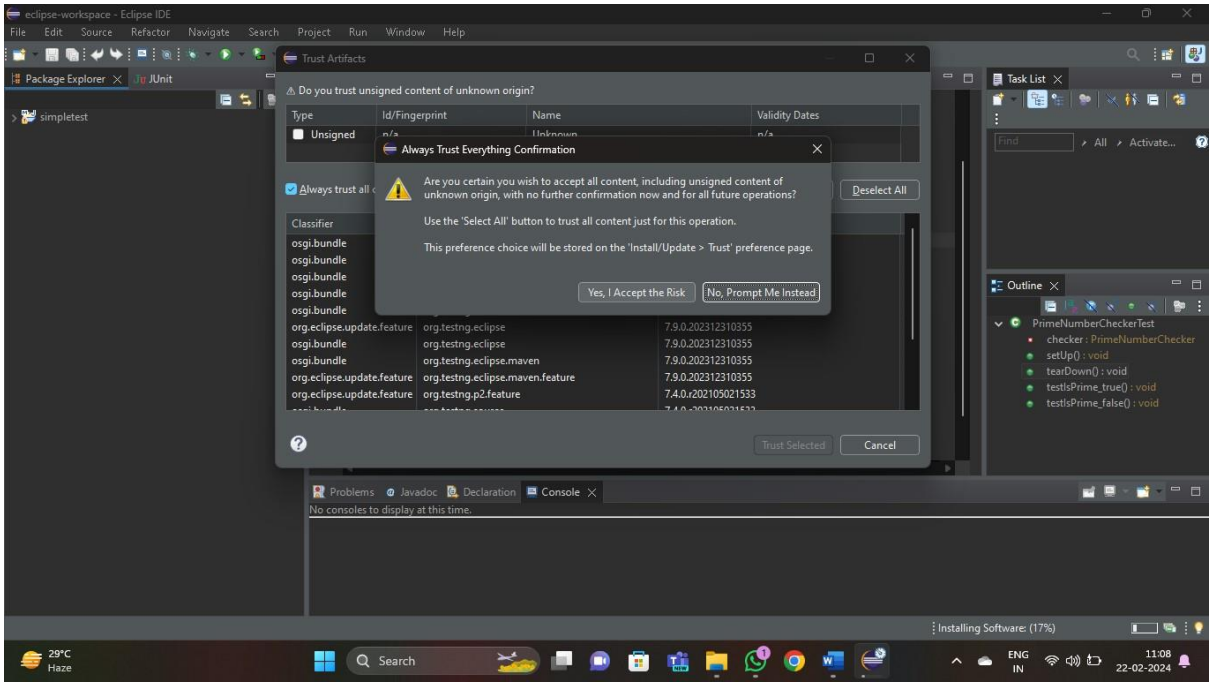
Step – 3: Always trust authorities -> Accept risk.



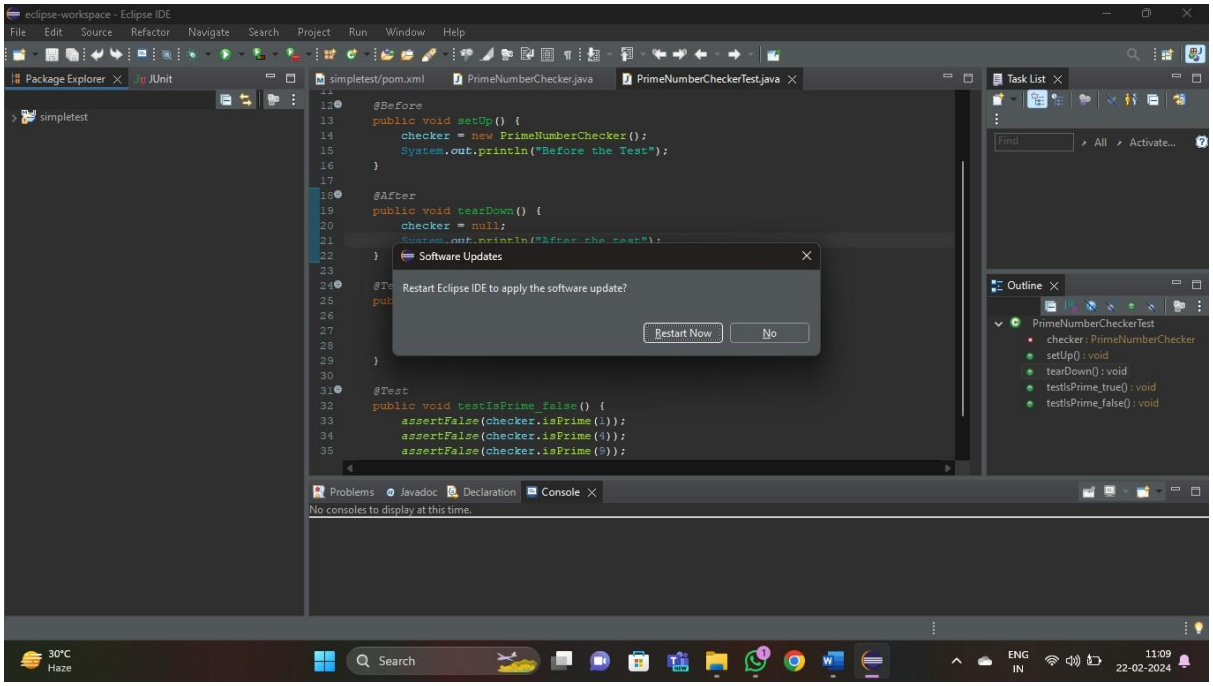


Step – 4: Always trust all content -> Accept risk.

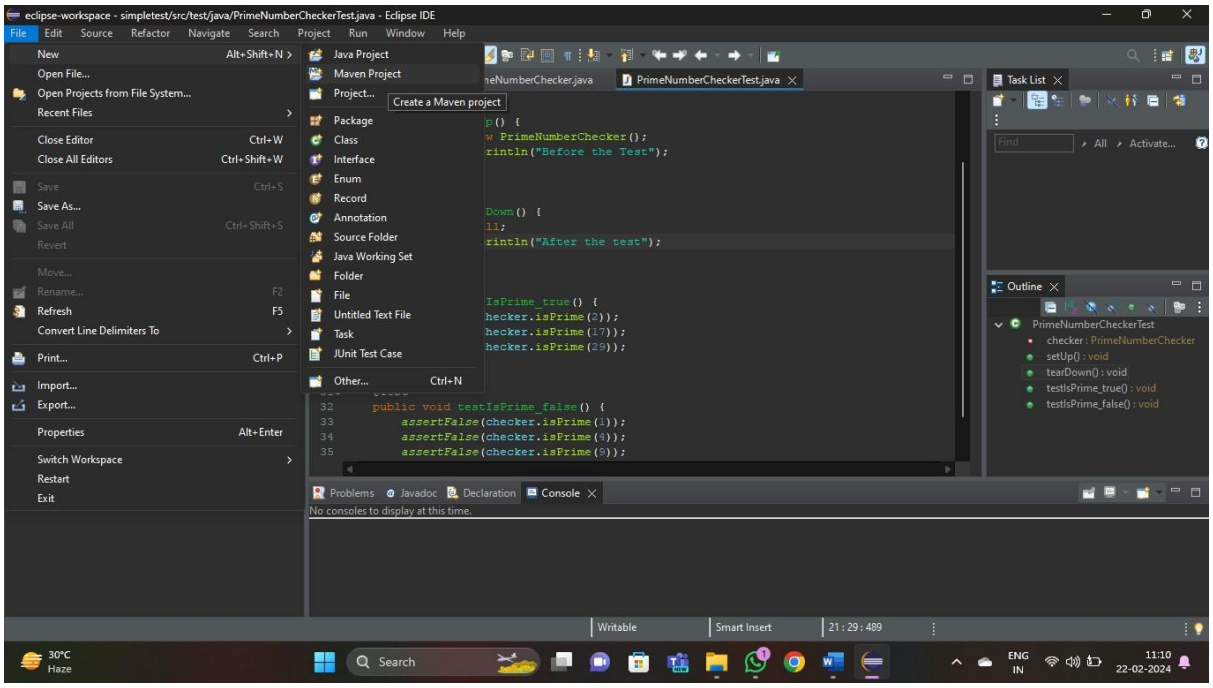


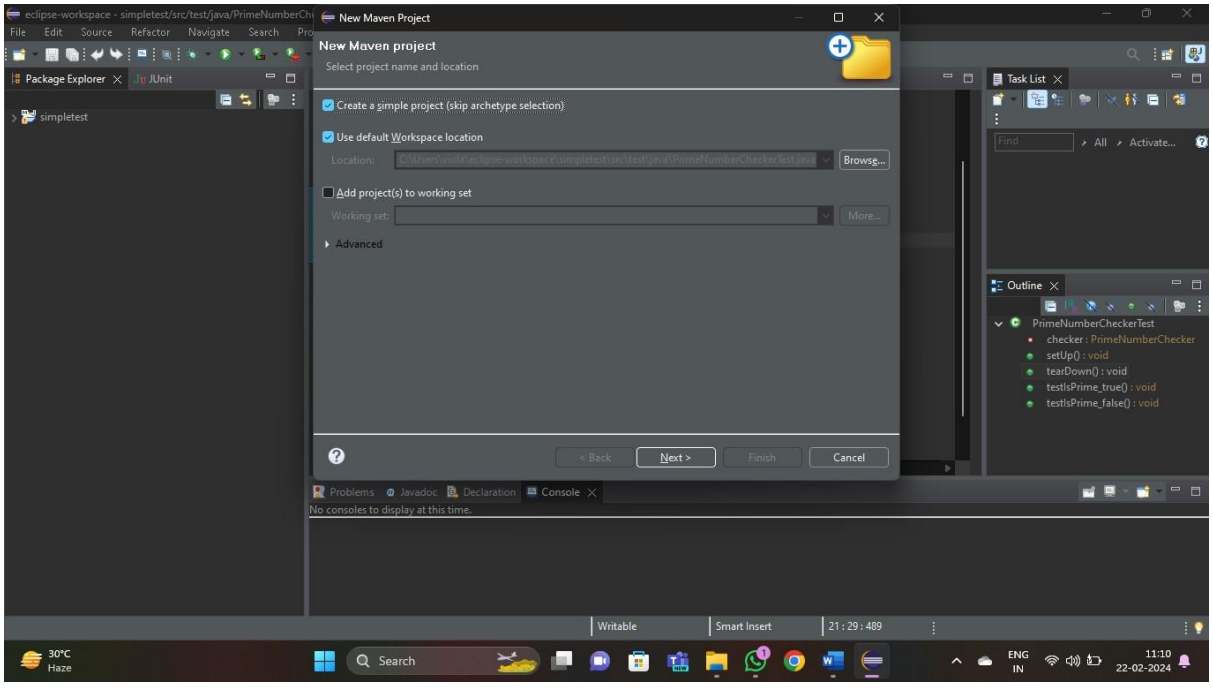


Step – 5: Restart now.

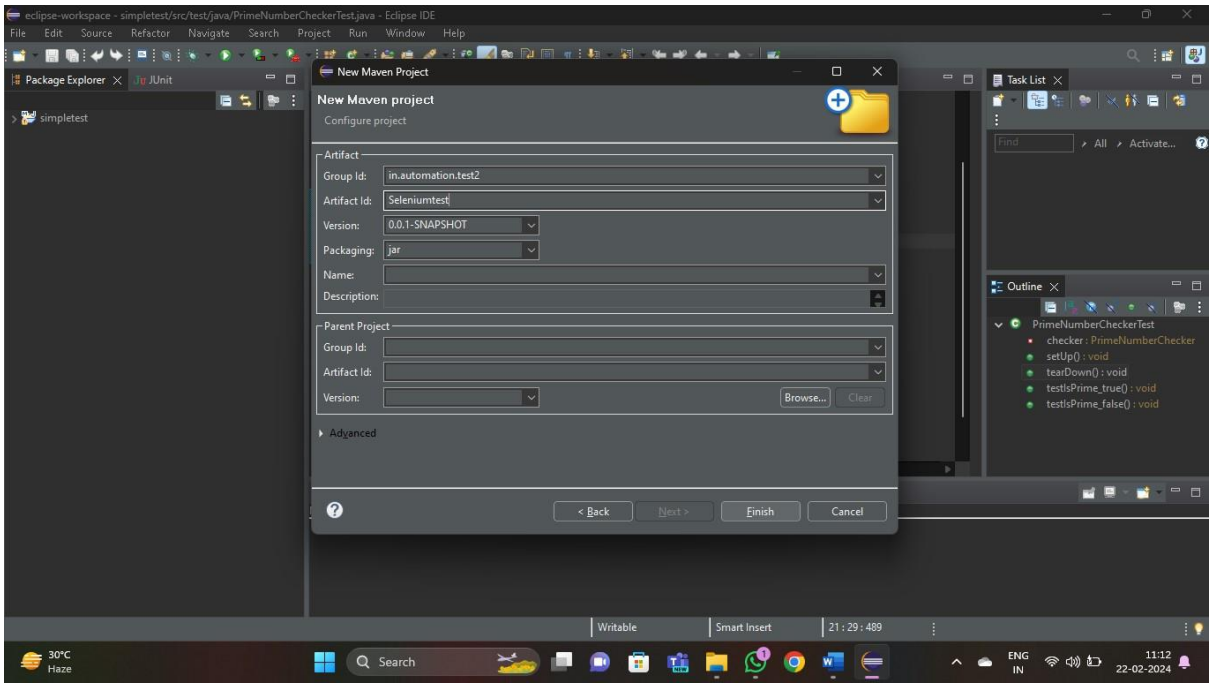


Step – 6: New Maven Project.





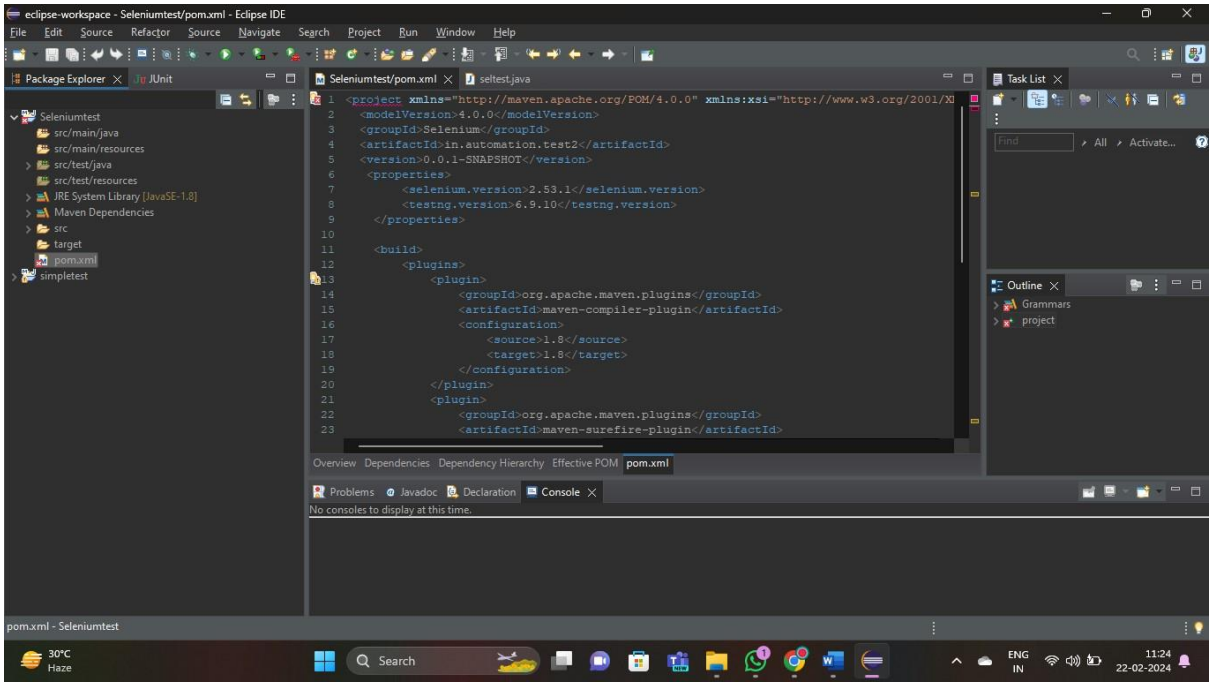
Step – 7: Group Id : in.automation.test2 -> Artifact Id: Seleniumtest.



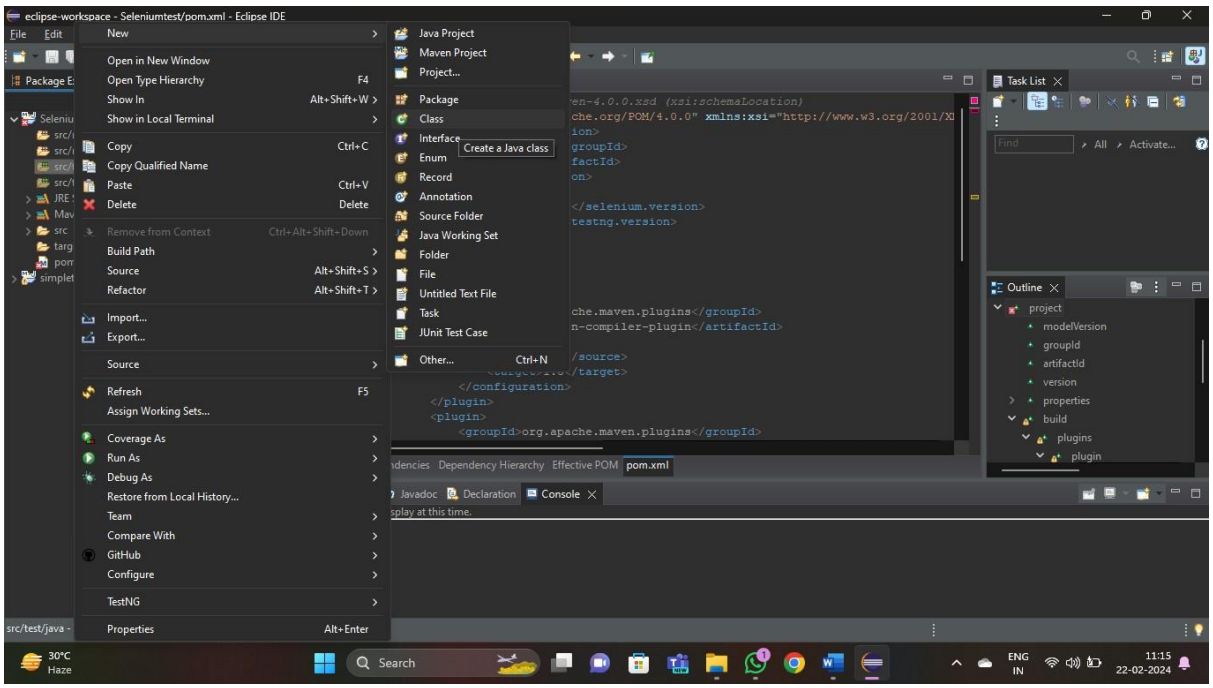
Step – 8: Copy this code to pom.xml file.

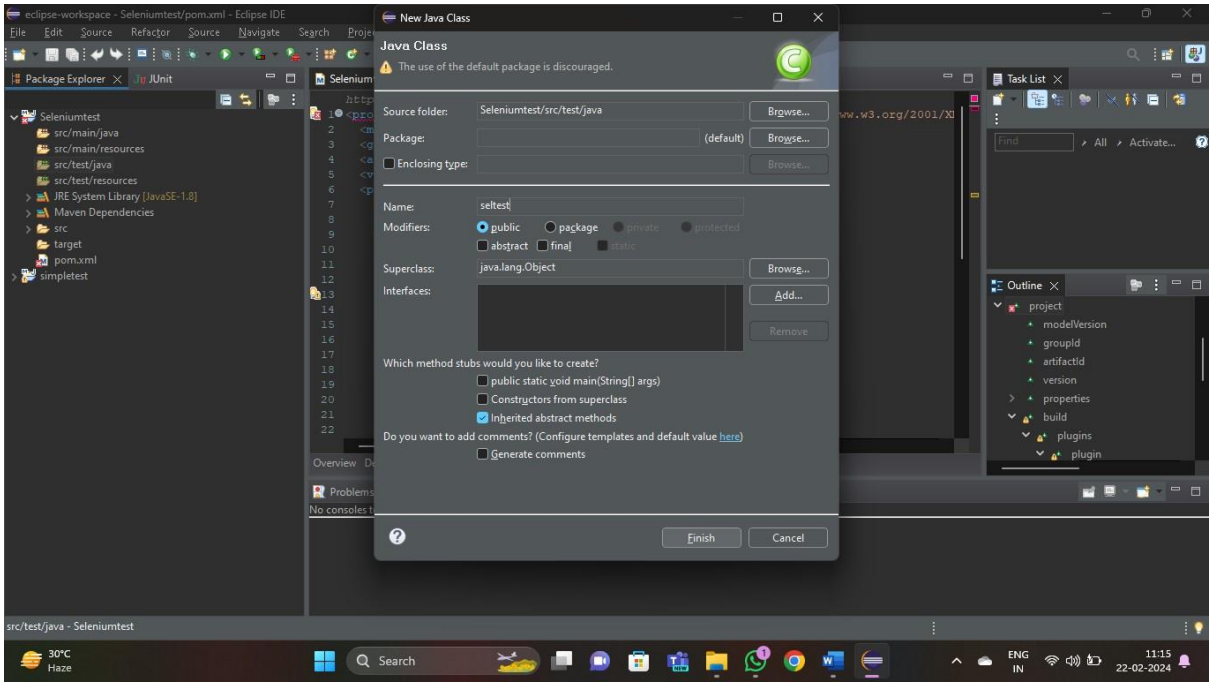
```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>Selenium</groupId>
  <artifactId>in.automation.test2</artifactId>
  <version>0.0.1-SNAPSHOT</version>
   <properties>
        <selenium.version>2.53.1</selenium.version>
        <testng.version>6.9.10</testng.version>
    </properties>

    <build>
        <plugins>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-compiler-plugin</artifactId>
                <configuration>
                    <source>1.8</source>
                    <target>1.8</target>
                </configuration>
```

```
            </plugin>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-surefire-plugin</artifactId>
                <version>2.18.1</version>
                <configuration>
                    <suiteXmlFiles>
                        <suiteXmlFile>testng.xml</suiteXmlFile>
                    </suiteXmlFiles>
                </configuration>
            </plugin>
        </plugins>
    </build>

    <dependencies>
        <dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>4.18.1</version>
</dependency>
        <dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>7.9.0</version>
    <scope>test</scope>
</dependency>
    </dependencies>
</project>
```

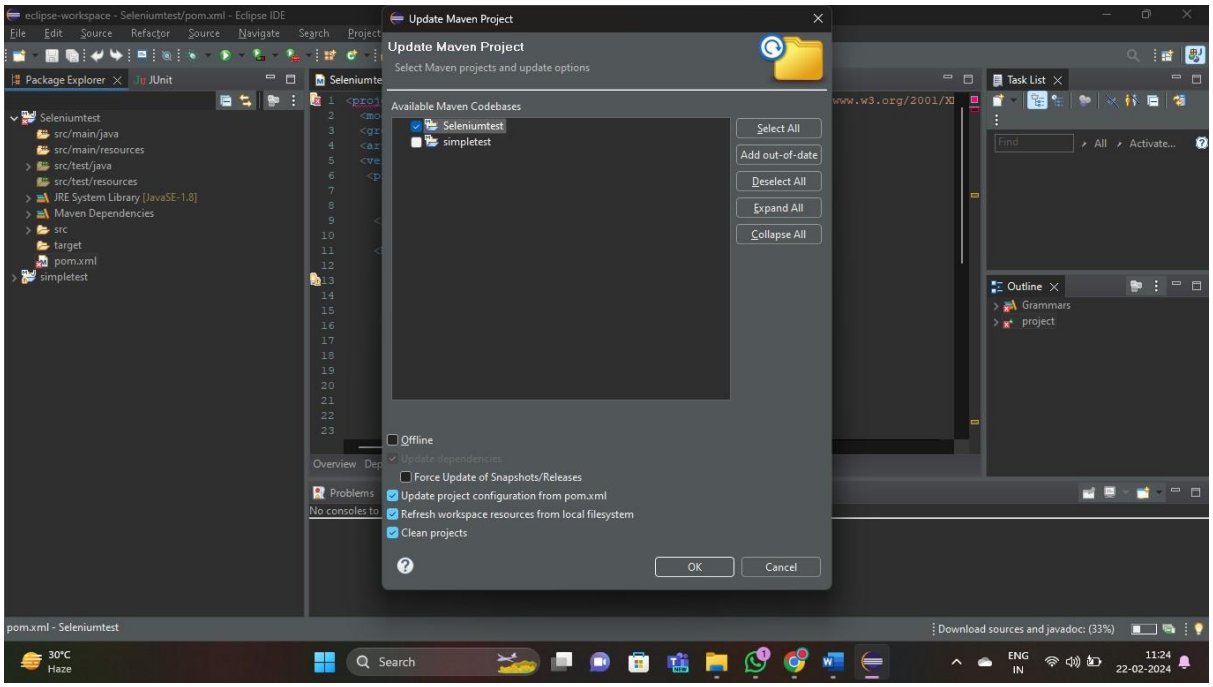


Step – 9: src/test/java -> new -> class -> name it as seltest.





Step – 10: Update project.

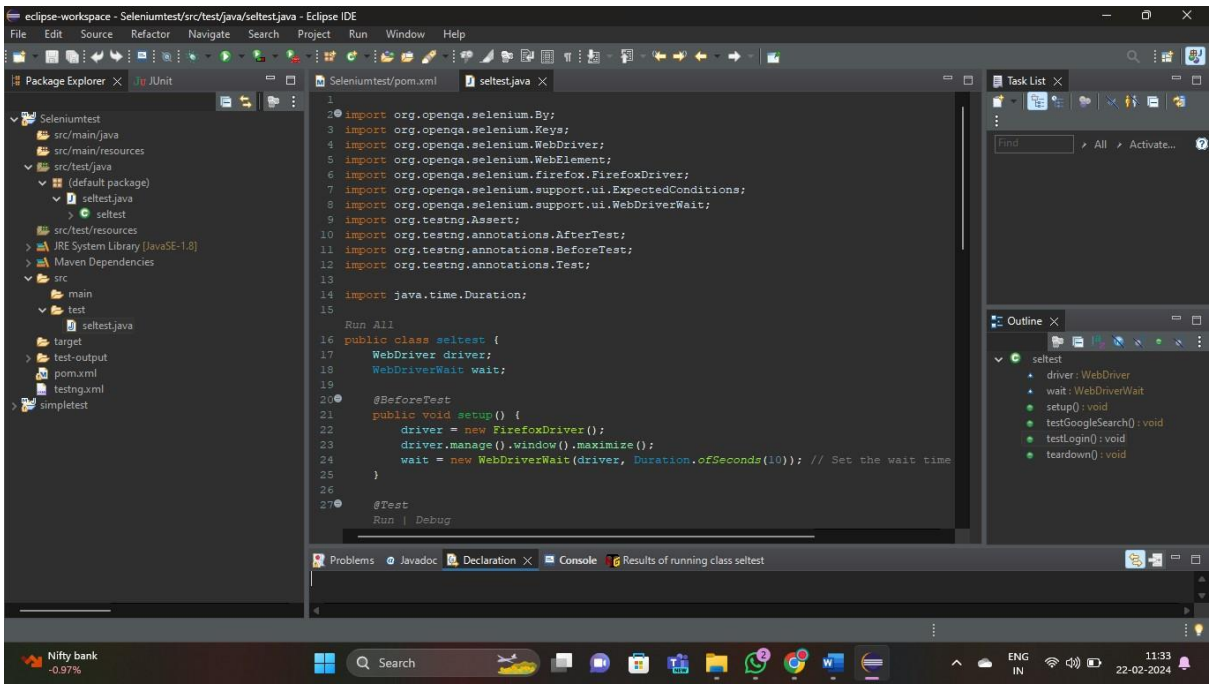


Step – 11: Copy the following code to seltest.java .

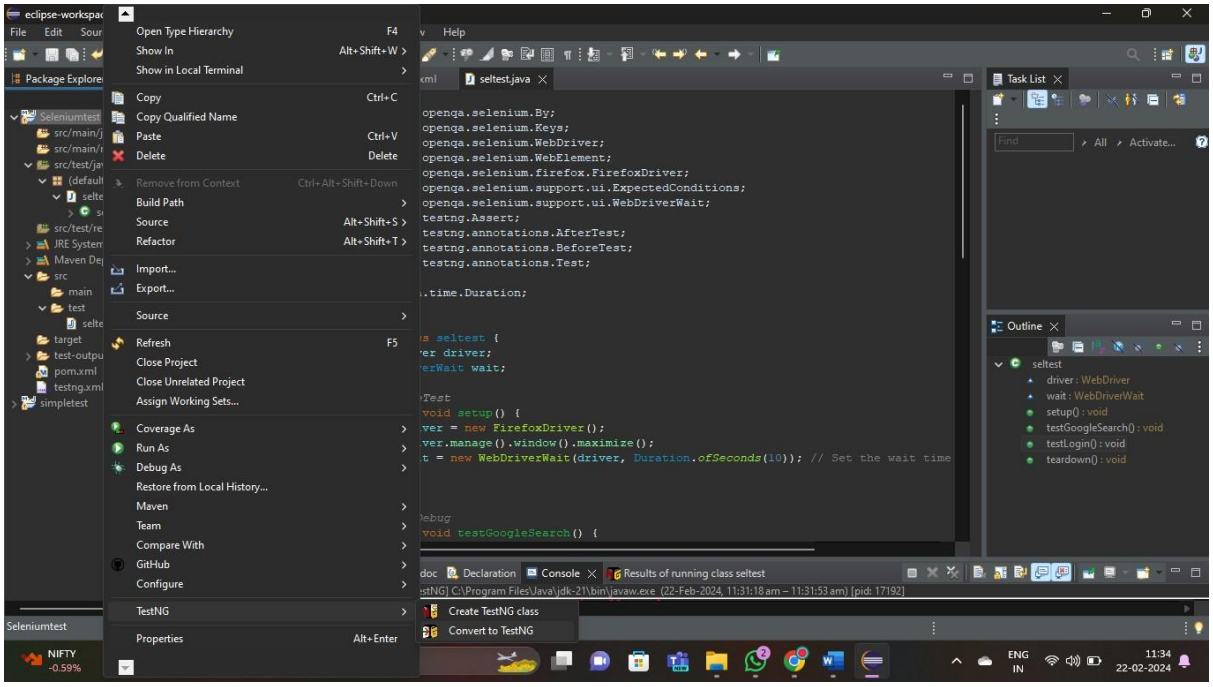
```
import org.openqa.selenium.By;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.Assert;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

import java.time.Duration;


public class seltest {
    WebDriver driver;
    WebDriverWait wait;

    @BeforeTest
    public void setup() {
        driver = new FirefoxDriver();
        driver.manage().window().maximize();
```

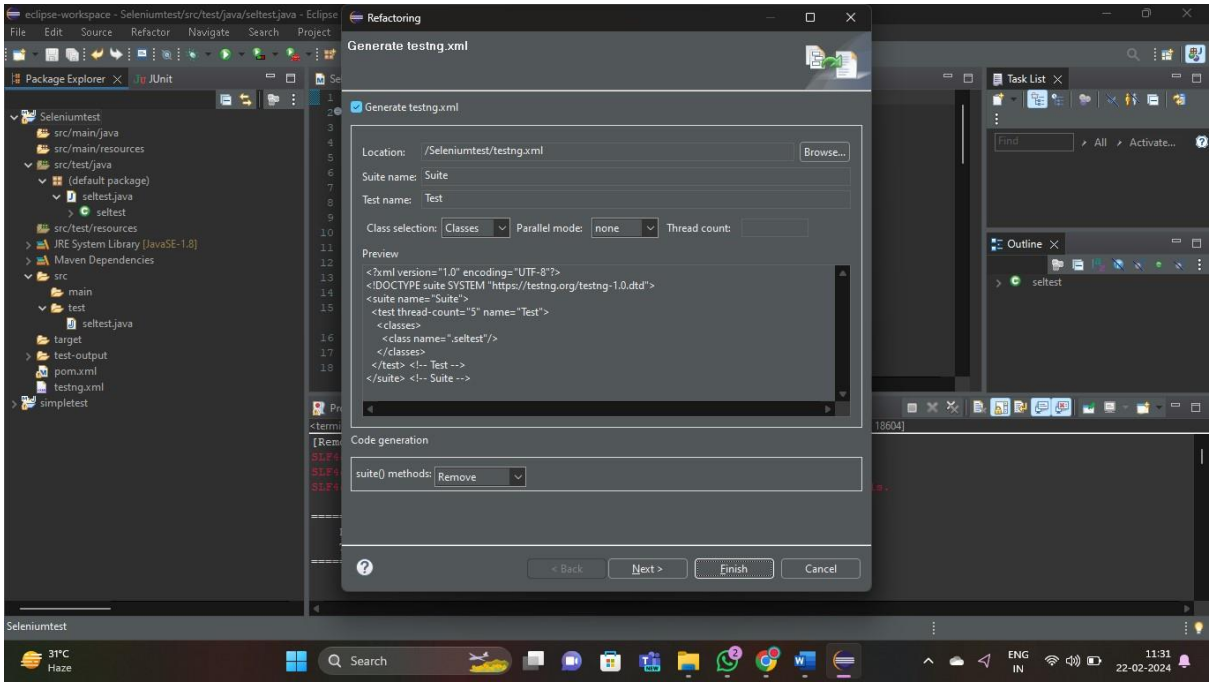
```
        wait = new WebDriverWait(driver, Duration.ofSeconds(10)); // Set
the wait time to 10 seconds
    }

    @Test

    public void testGoogleSearch() {
        // Open Google
        driver.get("https://duckduckgo.com/");

//        // Verify title
//        Assert.assertEquals(driver.getTitle(), "Google");

        // Enter search term
        WebElement searchBox =
wait.until(ExpectedConditions.presenceOfElementLocated(By.className("search
box_input__bEGm3")));
        searchBox.sendKeys("Google Maps");
        searchBox.sendKeys(Keys.RETURN);

        wait.until(ExpectedConditions.titleContains("Google Maps at
DuckDuckGo"));

        // Verify search results page title
        Assert.assertTrue(driver.getTitle().contains("Google Maps at
DuckDuckGo"));


    }

    @Test

    public void testLogin() {

        driver.get("https://practicetestautomation.com/practice-test-
login/");

        // Verify title

        // Use more specific locators
        WebElement emailInput = driver.findElement(By.id("username"));
        WebElement passwordInput = driver.findElement(By.id("password"));
        WebElement loginButton = driver.findElement(By.id("submit"));

        // Use descriptive variable names
        String invalidEmail = "invalid_email@example.com";
        String invalidPassword = "wrong_password";

        // Enter invalid credentials
        emailInput.sendKeys(invalidEmail);
        passwordInput.sendKeys(invalidPassword);
        loginButton.click();

        // Verify login error message
        WebElement errorMessage = driver.findElement(By.id("error"));
        String expectedErrorMessage = "Your username is invalid!";
        String actualErrorMessage = errorMessage.getText();
        Assert.assertEquals(actualErrorMessage, expectedErrorMessage,
"Login error message mismatch");
    }
```

```
    @AfterTest
    public void teardown() {
        if (driver != null) {
            driver.quit();
        }
    }
}
```

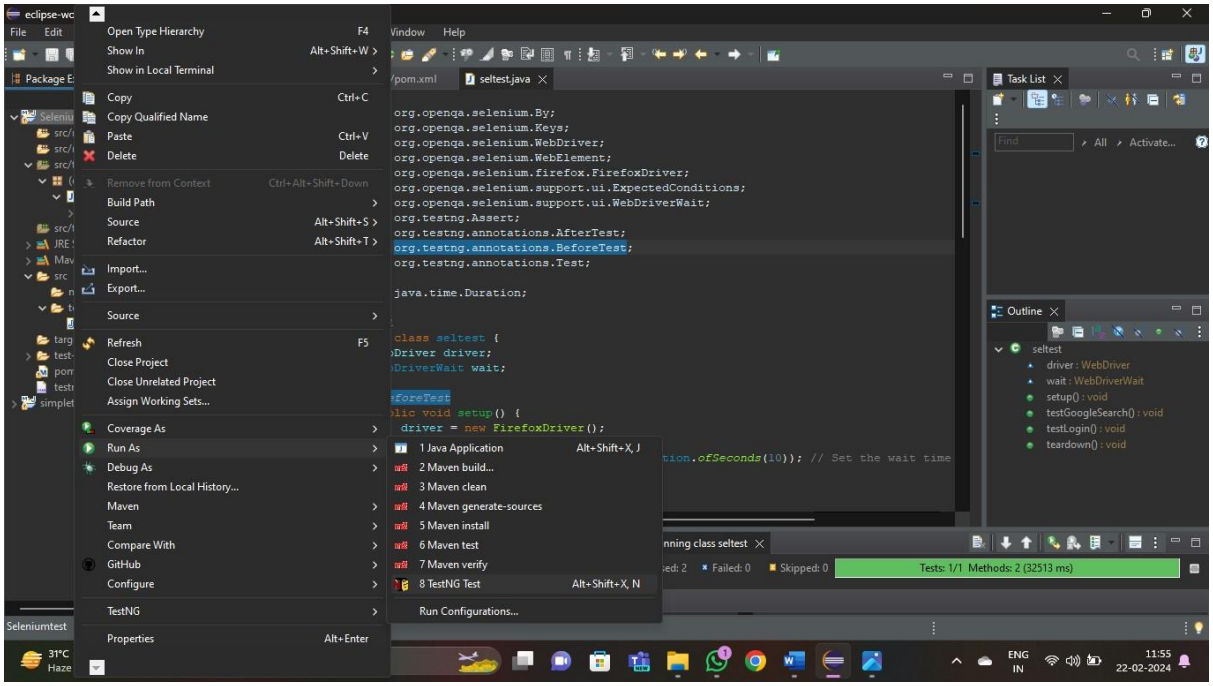


Step – 12: Selenium test -> TestNG -> Convert to TestNG .





Step – 13: Selenium test -> RunAs-> TestNG Test.



Step – 14: Output : Test Cases Passed.