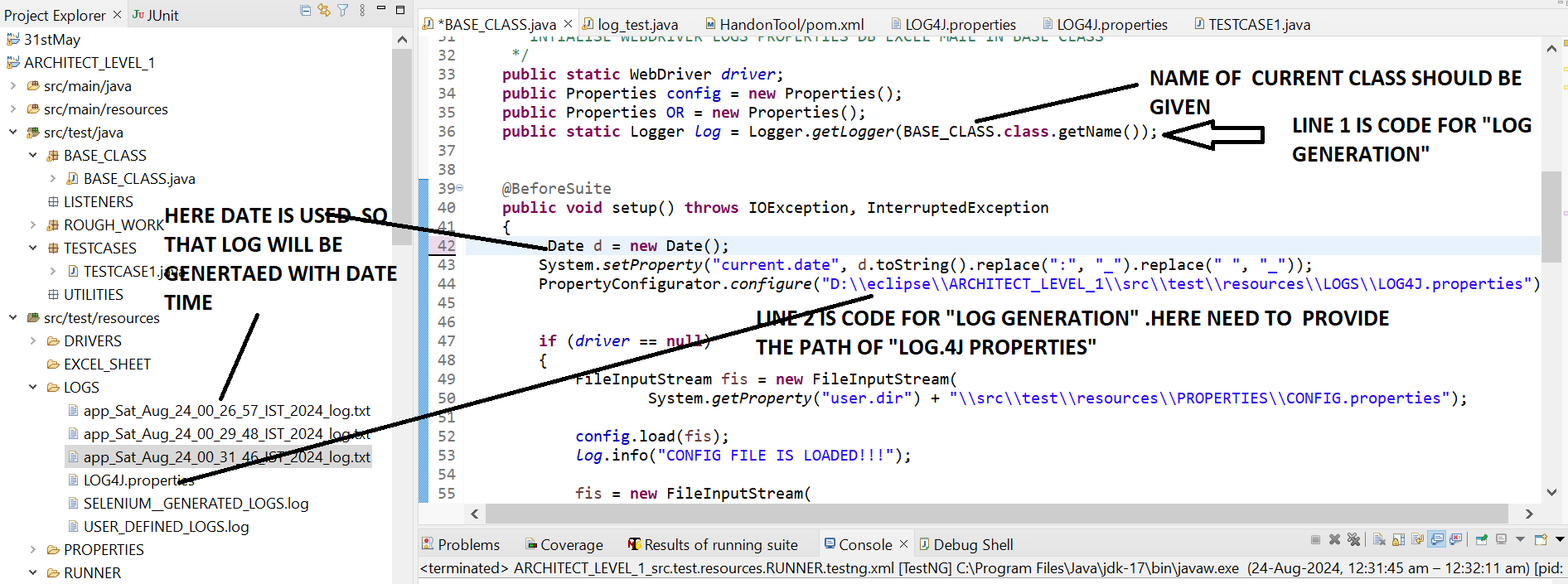
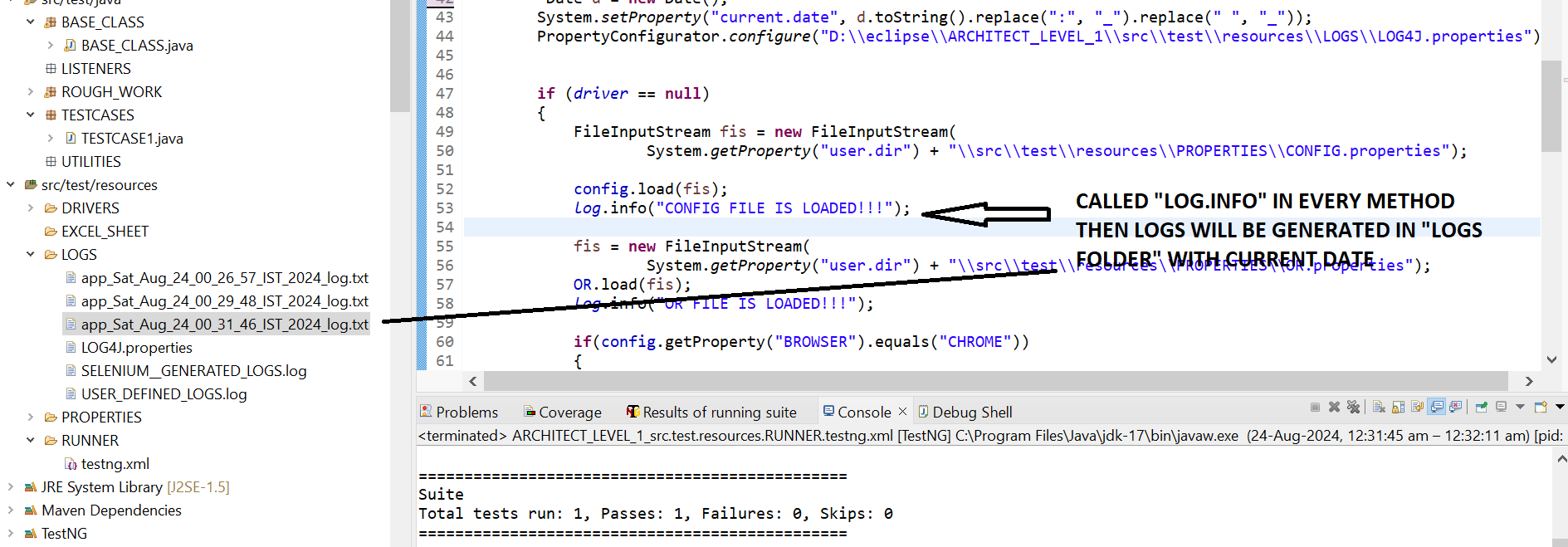
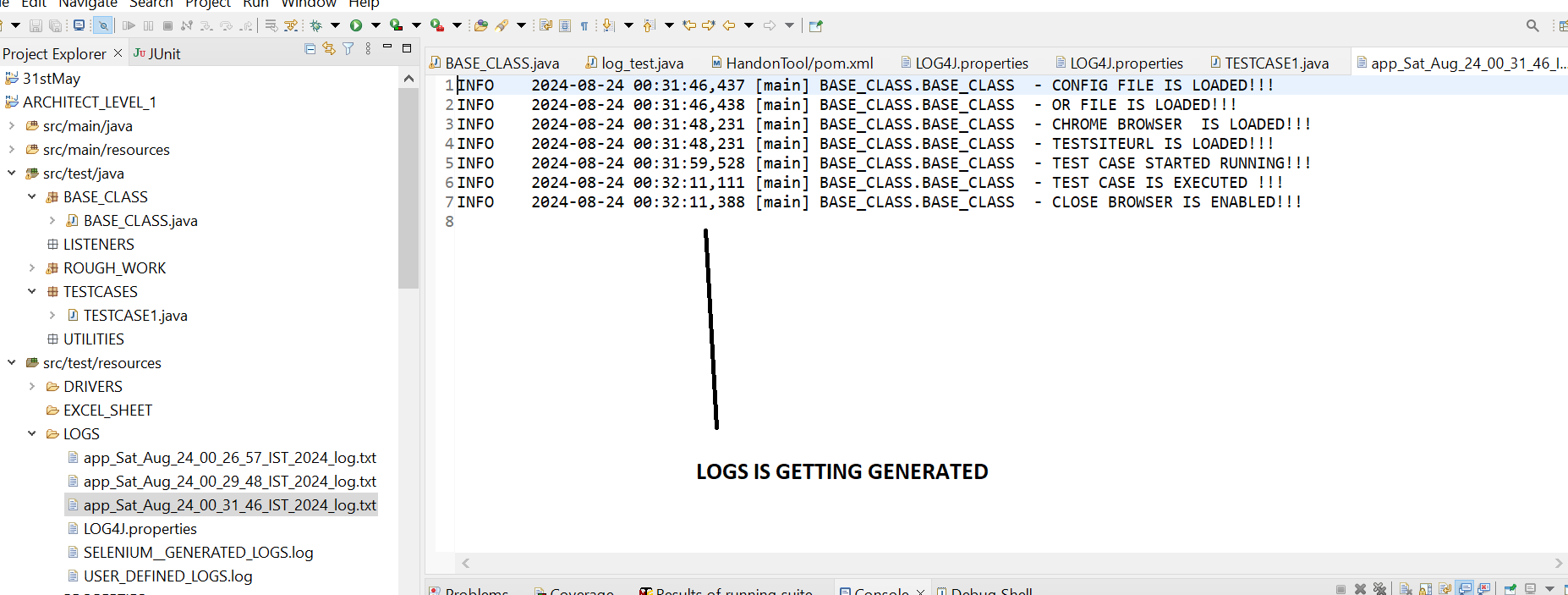
**DATA DRIVEN FRAMEWORK**

**IN PROPERTY CONFIGURATOR ,NEED TO PROVIDE THE LOG 4J.PROPERTIES PATH AND THEN CALLED “*LOG*.INFO("CONFIG FILE IS LOADED!!!");” IN EVERY METHOD &RUN THE PROJECTS AND REFRESH THE PROJECTS.LOG WILL AUTOMATICALLY GET GENERATED**

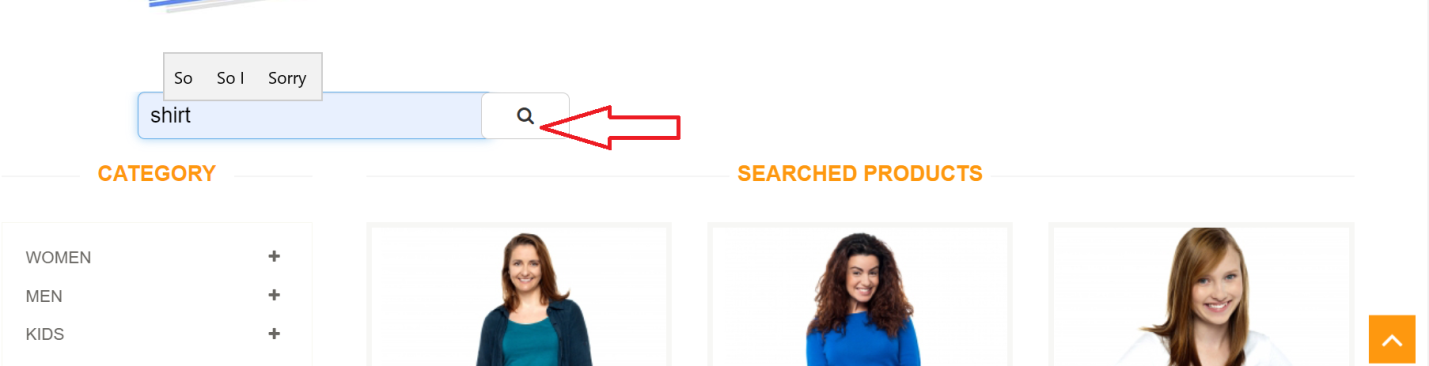
**IN “LOGS FOLDER “ WITH CURRENT DATE TIME AS SAME MENTIONED IN “LOG4J.PROPERTIES”**

****

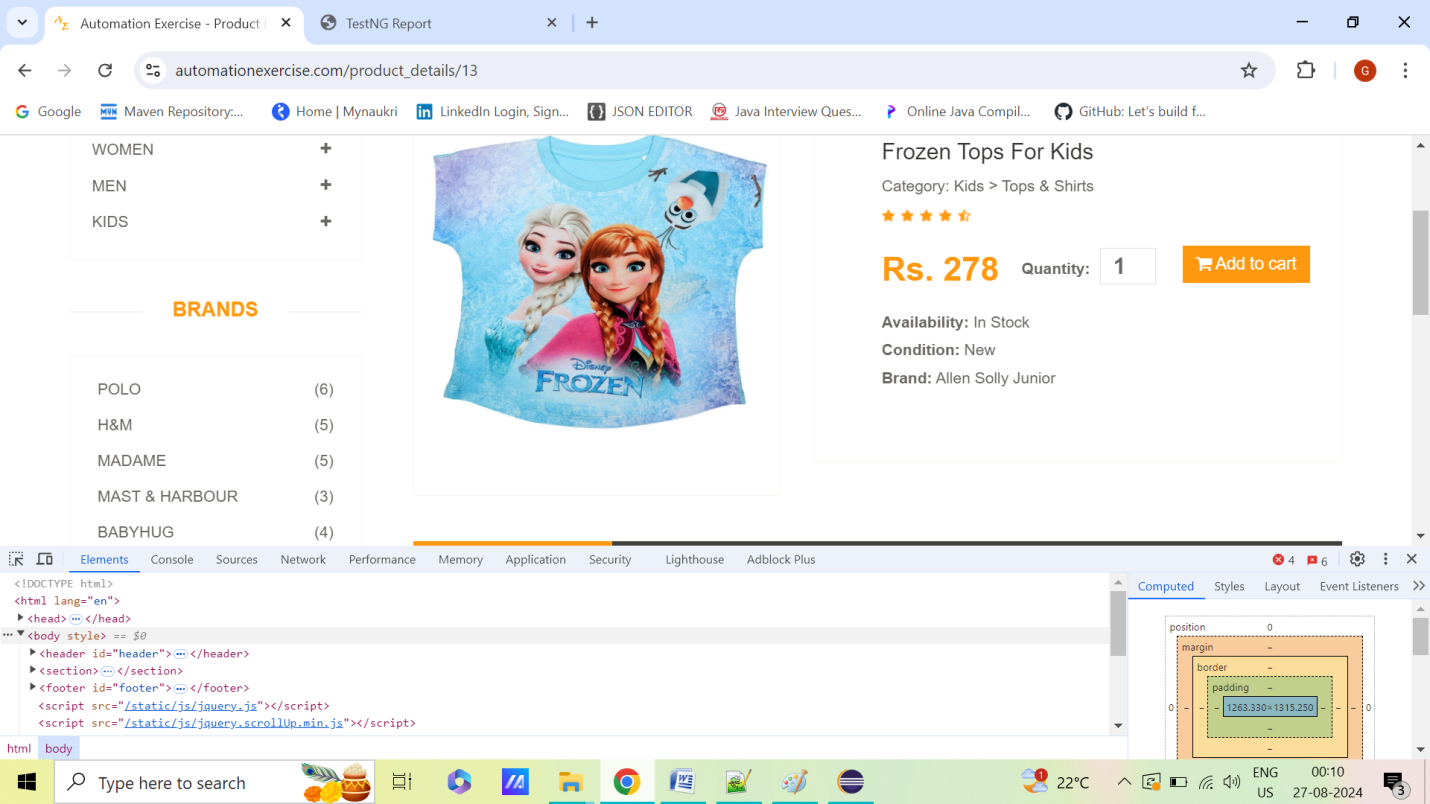
****

****

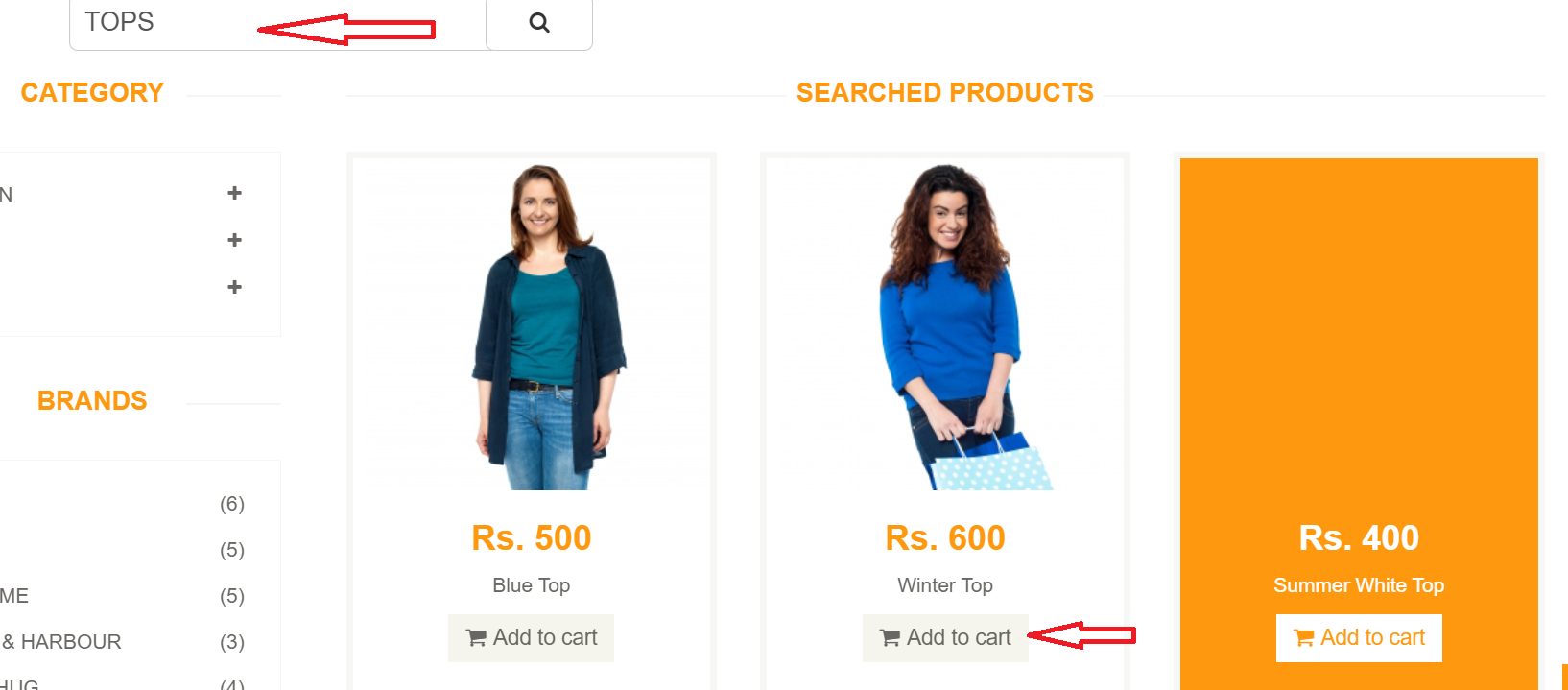
**SINCE WE ARE USING @BEFORE SUITE AND “AFTERSUITE” in BASE CLASS SO TESTCASE 2 WILL RUN WHERE TESTCASE 1 STOPS**

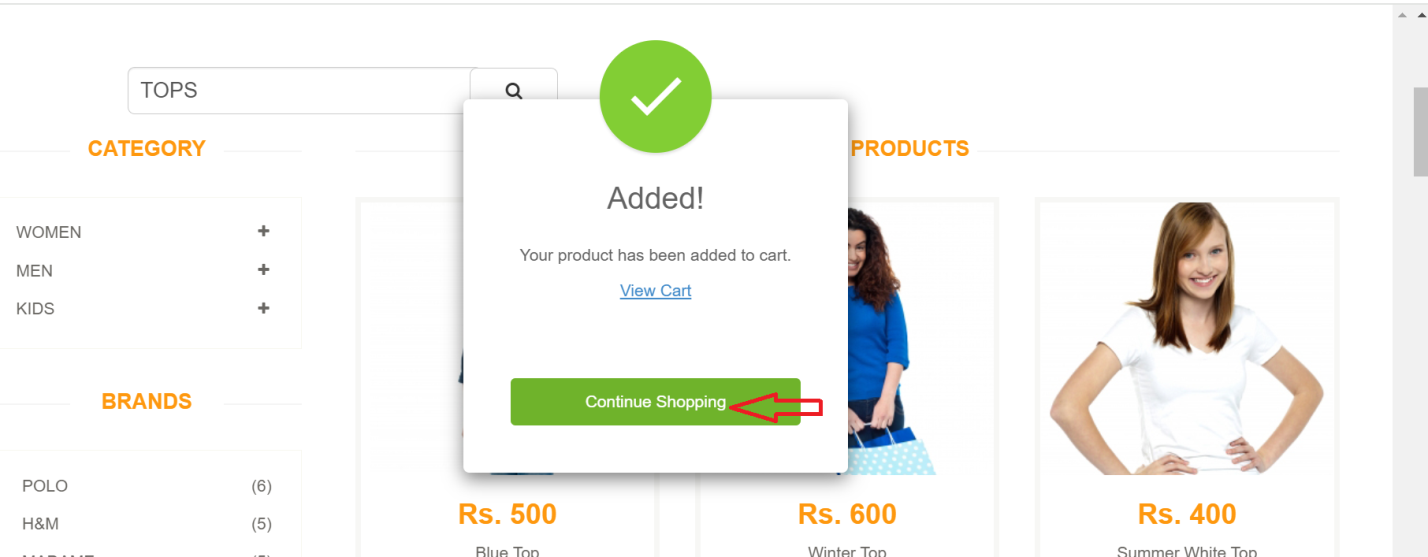
**TEST CASE 1=>IT ENTER THE VALUE AS “SHIRTS” AND CLICK SEARCH AND CHOOSE BELOW VALUES SUCH AS “FROZEN TSHIRT” THEN IT NAVIGATE BACK TO SEARCH BOX AS TEST CASE 2 SHOULD BE CONTINUATION OF TEST CASE 1 SO TEST CASE 2 WILL RUN WHERE TEST CASE1 HAS STOPS**

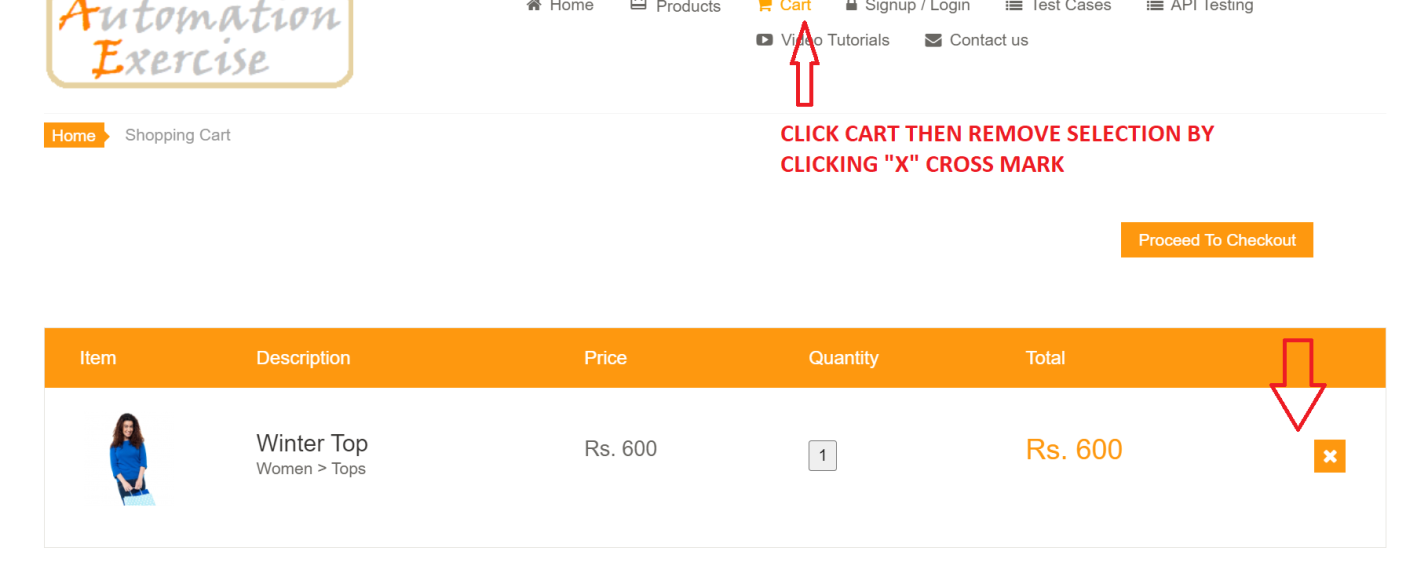
**IT SELECT BELOW SEARCH**

****

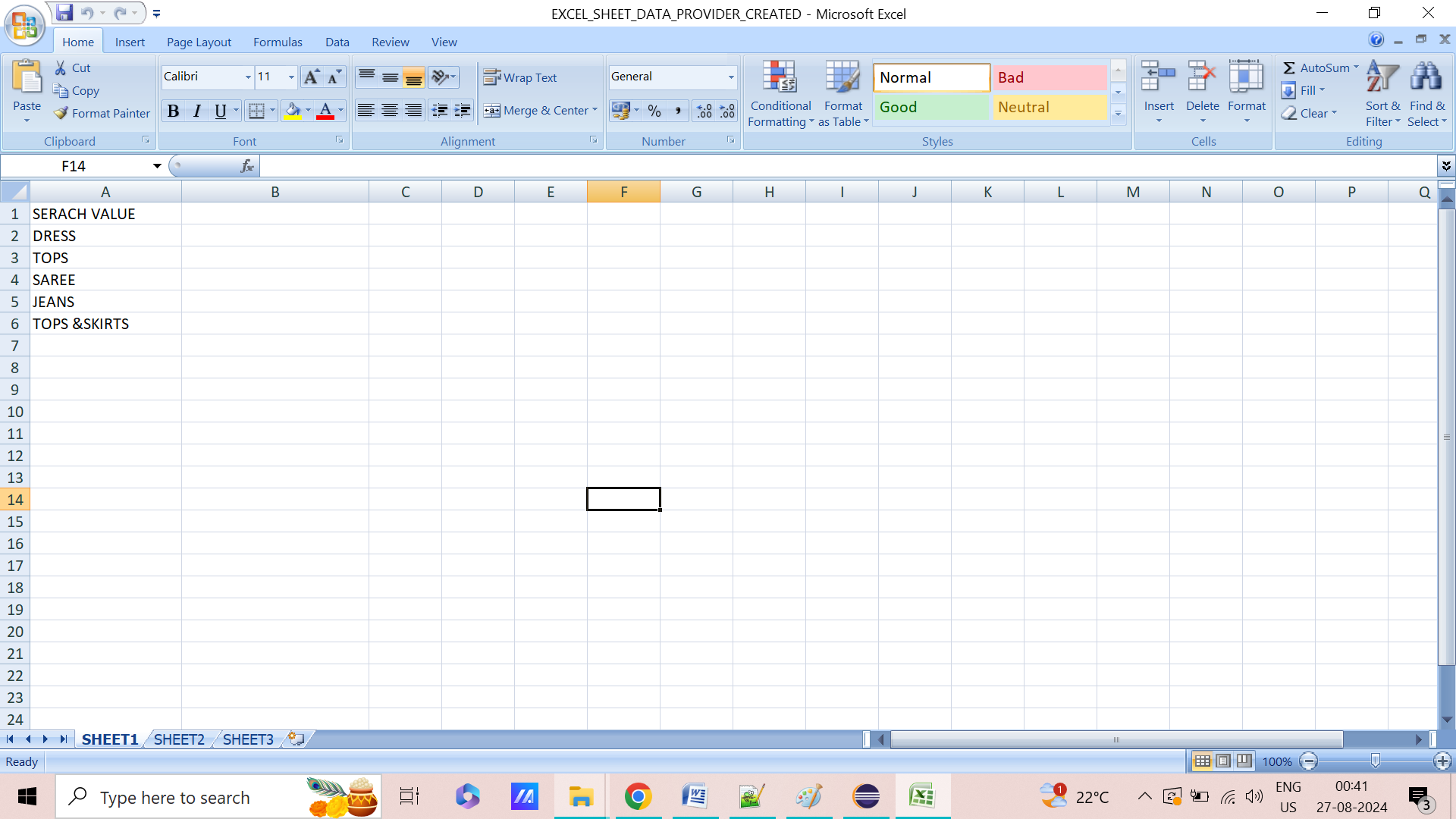
**TEST CASE 2 SHOULD BE CONTINUATION OF TEST CASE 1 SO TEST CASE 2 WILL RUN WHERE TEST CASE1 HAS STOPS.IT WILL TAKE ALL VALUE FROM EXCEL SHEET AND SEARCH THAT ITEM AND IT WILL CLICK “ADD TO CART” BUTTON AND THEN CLICK “CONTINUE SHOPPING” BUTTON AND THEN REMOVE THE “ADD TO CART” OPTIONS THEN IT WILL NAVIGATE BACK TO SEARCH BOX THEN AGAIN SEARCH VALUE FROM VALUE PRESENT IN EXCEL SHEET .THEN IT WILL AGAIN CLICK “ADD TO CART “ BUTTON AND THEN REMOVE SELECTIONS AGAIN.REMEMBER XPATH SHOULD BE WRITTEN IN SUCH A WAY THAT SAME XPATH SHOULD SHOWS UP AFTER SEARCHING DIFFERENT ITEMS SUCH AS “SHIRTS”,”PANT” ETC .WE SHOULD USE DYNAMIC XPATH WHICH WILL BE COMMON AFTER SEARCHING DIFFERENT ITEMS**

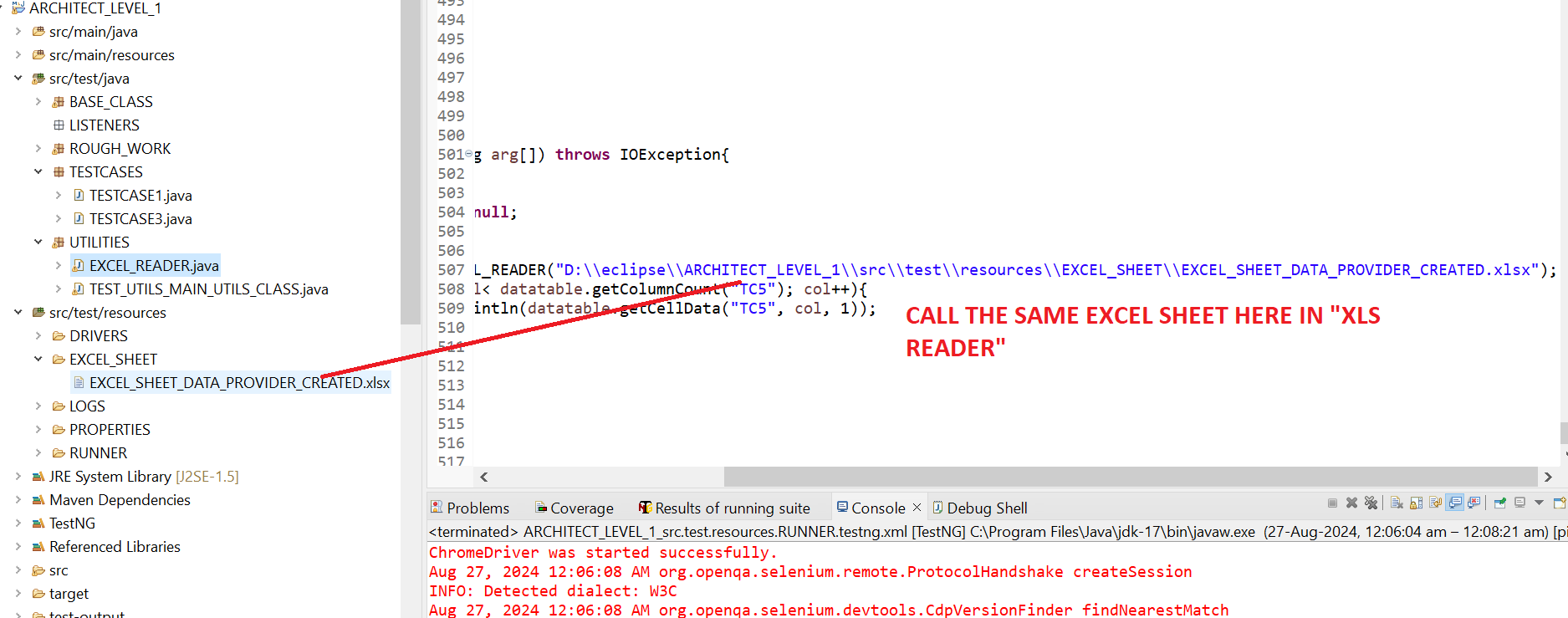
****

****

****

**STEPS 1=> CREATE “EXCEL READER CLASS” AND MENTION THE PATH OF EXCEL SHEET .CALL THE SAME EXCEL SHEET HERE IN "XLS READER"&CREATE BELOW EXCEL SHEET IN “EXCEL FOLDER” AS IT WILL BE XLS READER” TO FETCH ROW AND COLUMN VALUES**

****

****

**STEPS 2=>MAKE “STATIC” TO BOTH PROPERTIES VARAIBLE IN BASE CLASS**

**STEPS 3=> CREATE “XLS\_READER” IN BASE CLASS**

****

**STEPS 4=>WRITE BELOW CODE IN TESTCASE1 .REMEMEBER TEST CASE 2 WILL START WHERE TEST CASE 1 WILL STOP AS WE HAVE METHOD SUCH AS “BEFORE SUITE” AND “AFTERSUITE” IN BASE CLASS FOR LAUNCHING DRIVER AND QUITING THE DRIVER SO TEST CASE WILL PICK THE APPLICATION WHERE TESTCASE 1 WILL STOPS**

**package** TESTCASES;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.JavascriptExecutor;

**import** org.testng.Assert;

**import** org.testng.annotations.Test;

**import** BASE\_CLASS.BASE\_CLASS;

**public** **class** TESTCASE1 **extends** BASE\_CLASS

{

@Test

**public** **void** method1() **throws** InterruptedException

{

*log*.info("TEST CASE1 STARTED RUNNING!!!");

String Title=*driver*.getTitle();

System.***out***.println("Page Title is " +Title);

*driver*.findElement(By.*cssSelector*(*OR*.getProperty("SEARCH\_PRODUCT"))).clear();

*driver*.findElement(By.*cssSelector*(*OR*.getProperty("SEARCH\_PRODUCT"))).sendKeys("Shirts");

Thread.*sleep*(3000);

*driver*.findElement(By.*cssSelector*(*OR*.getProperty("SUBMIT\_SEARCH"))).click();

Assert.*assertTrue*(IsElementPresent(By.*cssSelector*(*OR*.getProperty("SUBMIT\_SEARCH"))),"Login not successful");

Thread.*sleep*(4000);

JavascriptExecutor js = (JavascriptExecutor) *driver*;

js.executeScript("window.scrollBy(0,800)", "");

Thread.*sleep*(4000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("SUBMIT\_SEARCHs"))).click();

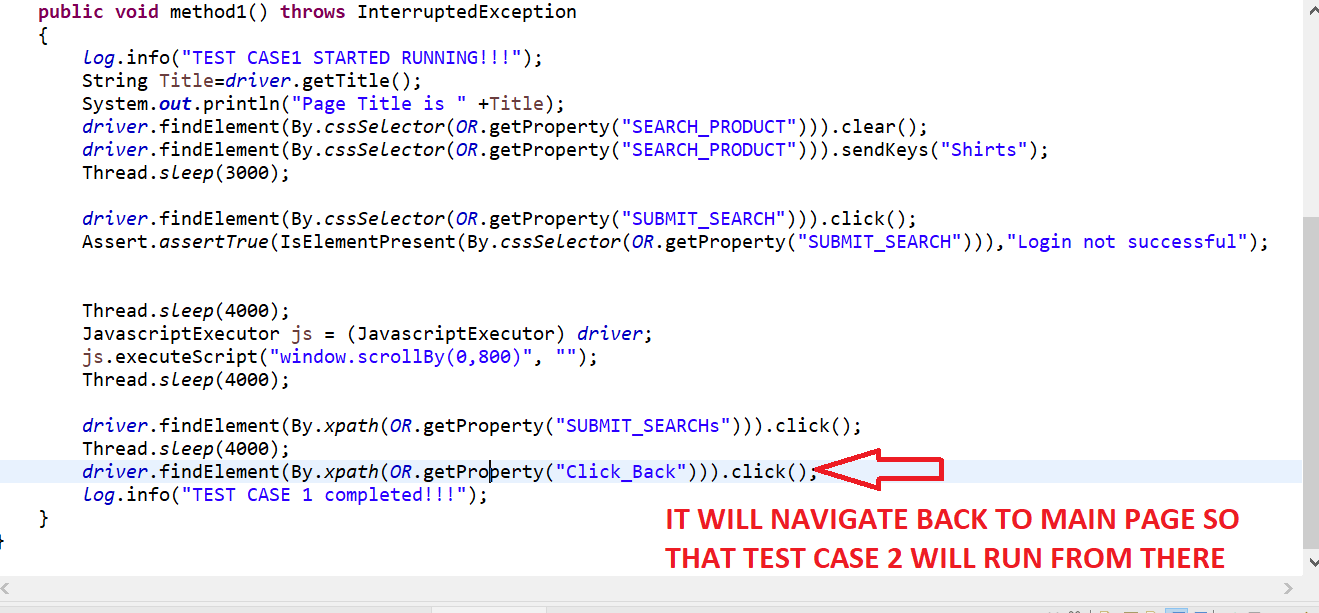
Thread.*sleep*(4000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("Click\_Back"))).click();

*log*.info("TEST CASE 1 completed!!!");

}

}

****

**STEPS 5=> CALL THE SAME “SHEETNAME” IN “DATA PROVIDER” OTHERWISE PROGRAM WILL NOT RUN.CREATE BELOW TEST CASE WITH “DATAPROVIDER” IN “TESTCASE2”. REMEMEBER TEST CASE 2 WILL START WHERE TEST CASE 1 WILL STOP AS WE HAVE METHOD SUCH AS “BEFORE SUITE” AND “AFTERSUITE” IN BASE CLASS FOR LAUNCHING DRIVER AND QUITING THE DRIVER**

**package** TESTCASES;

**import** org.openqa.selenium.By;

**import** org.testng.annotations.DataProvider;

**import** org.testng.annotations.Test;

**import** BASE\_CLASS.BASE\_CLASS;

**import** UTILITIES.EXCEL\_READER;

**public** **class** TESTCASE3 **extends** BASE\_CLASS

{

@Test(dataProvider="DataProvide\_1")

**public** **void** method3(String value) **throws** InterruptedException

{

*log*.info("TEST CASE3 STARTED RUNNING!!!");

*driver*.findElement(By.*cssSelector*(*OR*.getProperty("SEARCH\_PRODUCT"))).clear();

*driver*.findElement(By.*cssSelector*(*OR*.getProperty("SEARCH\_PRODUCT"))).sendKeys(value);

Thread.*sleep*(3000);

*driver*.findElement(By.*cssSelector*(*OR*.getProperty("SUBMIT\_SEARCH"))).click();

Thread.*sleep*(4000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("dress2"))).click();

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("button\_select"))).click();

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("Add\_tocart"))).click();

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("close\_button"))).click();

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("Click\_Back"))).click();

*log*.info("TEST CASE 3 completed!!!");

}

@DataProvider(name="DataProvide\_1")

**public** Object[][] getData() {

**if** (*xlsreadER* == **null**) {

*xlsreadER* = **new** EXCEL\_READER("D:\\eclipse\\ARCHITECT\_LEVEL\_1\\src\\test\\resources\\EXCEL\_SHEET\\EXCEL\_SHEET\_DATA\_PROVIDER\_CREATED.xlsx");

}

String sheetName = "SHEET1";

**int** rows = *xlsreadER*.getRowCount(sheetName);

**int** cols = *xlsreadER*.getColumnCount(sheetName);

Object[][] data = **new** Object[rows - 1][cols];

**for** (**int** rowNum = 2; rowNum <= rows; rowNum++) {

**for** (**int** colNum = 0; colNum < cols; colNum++) {

data[rowNum - 2][colNum] = *xlsreadER*.getCellData(sheetName, colNum, rowNum);

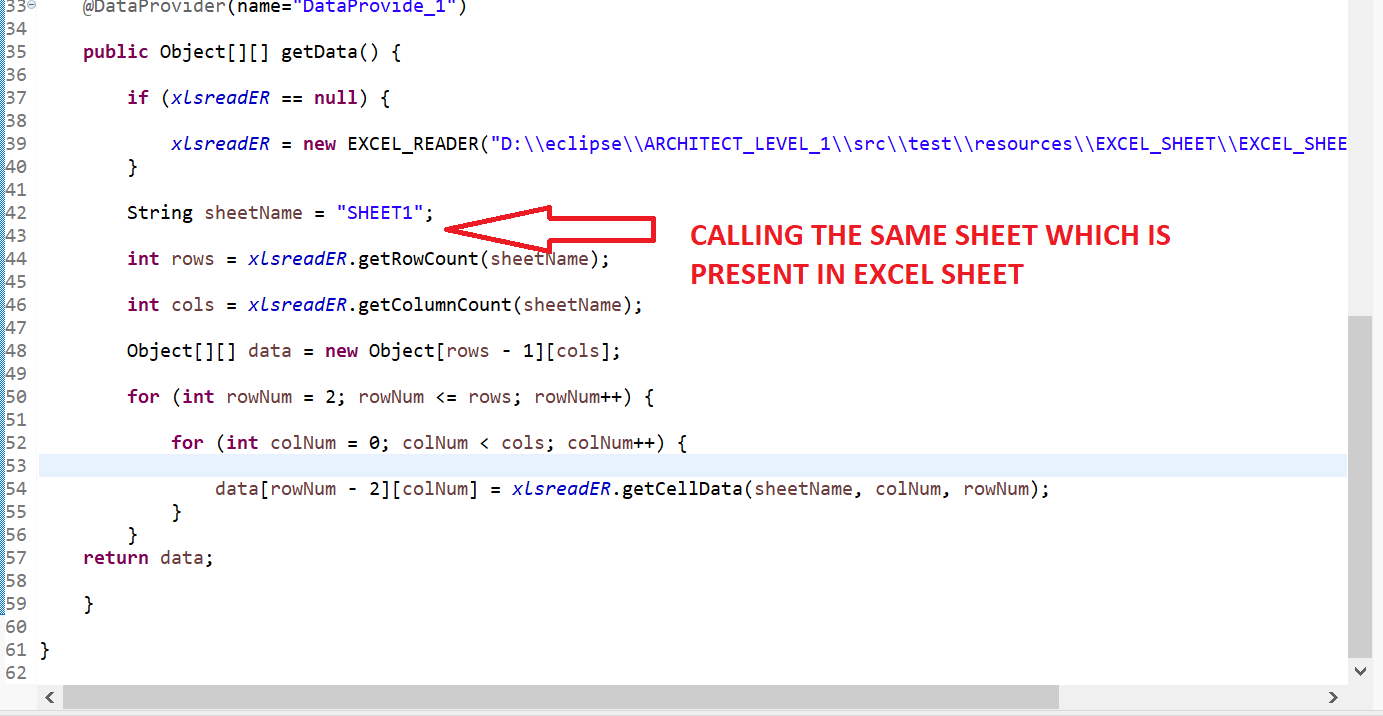
}

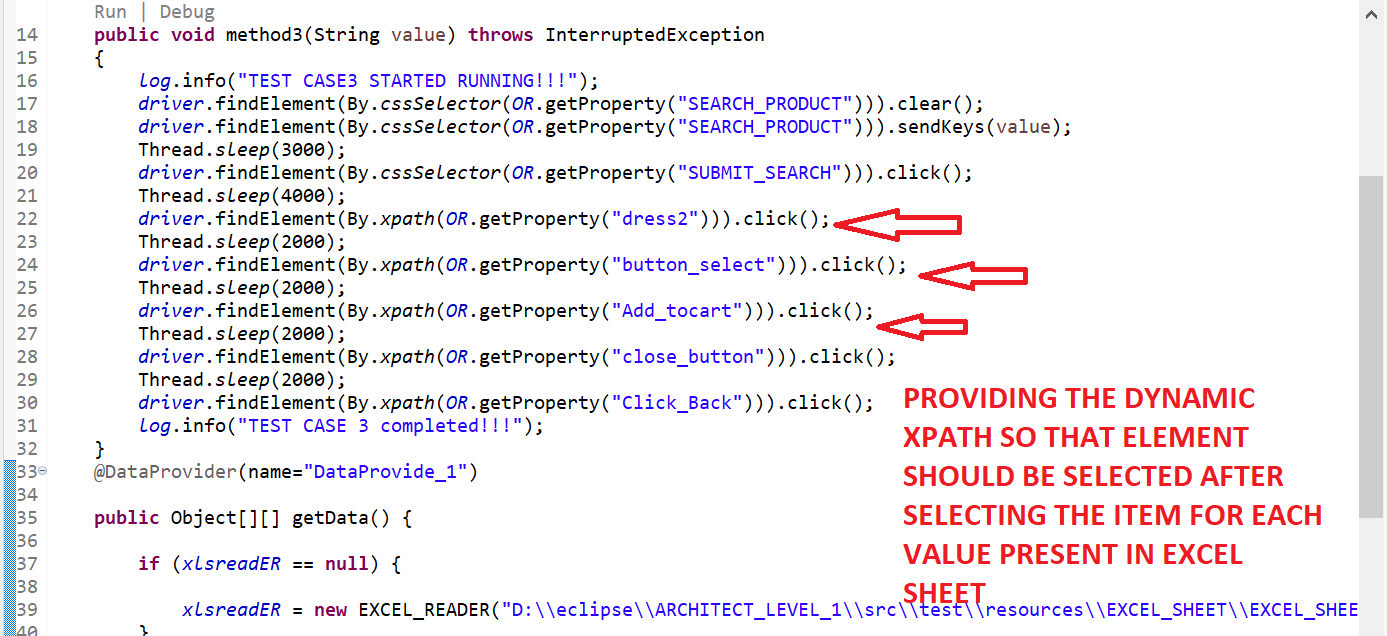
}

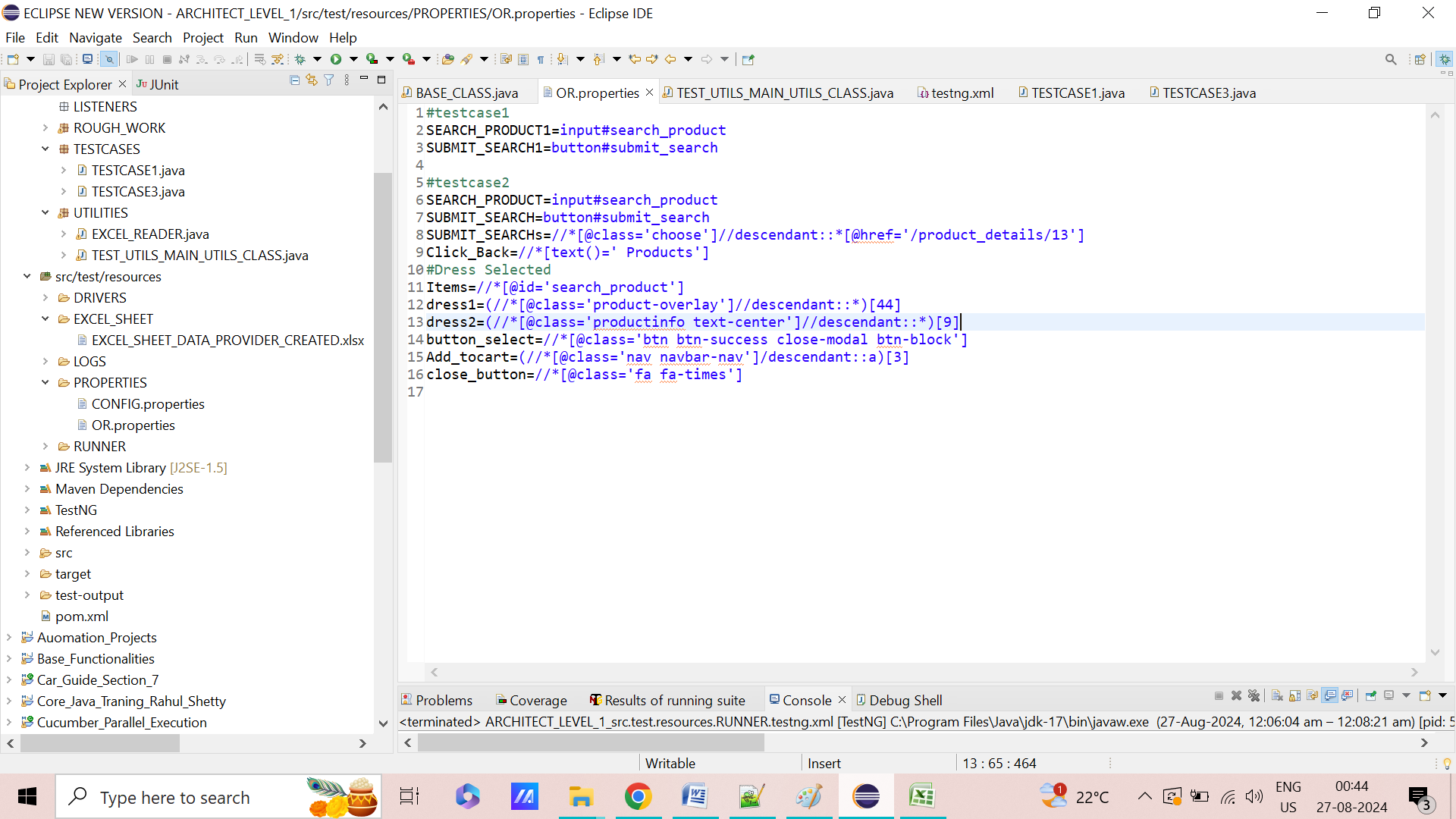
**return** data;

}

}

****

****

****

**STEPS 6=>CALL TWO TESTCASES IN “TESTNG.XML” FILE.**

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

<suite name="Suite">

<test name="TESTCASE1">

<classes>

<class name="TESTCASES.TESTCASE1"/>

</classes>

</test> <!-- Test -->

<test name="TESTCASE3">

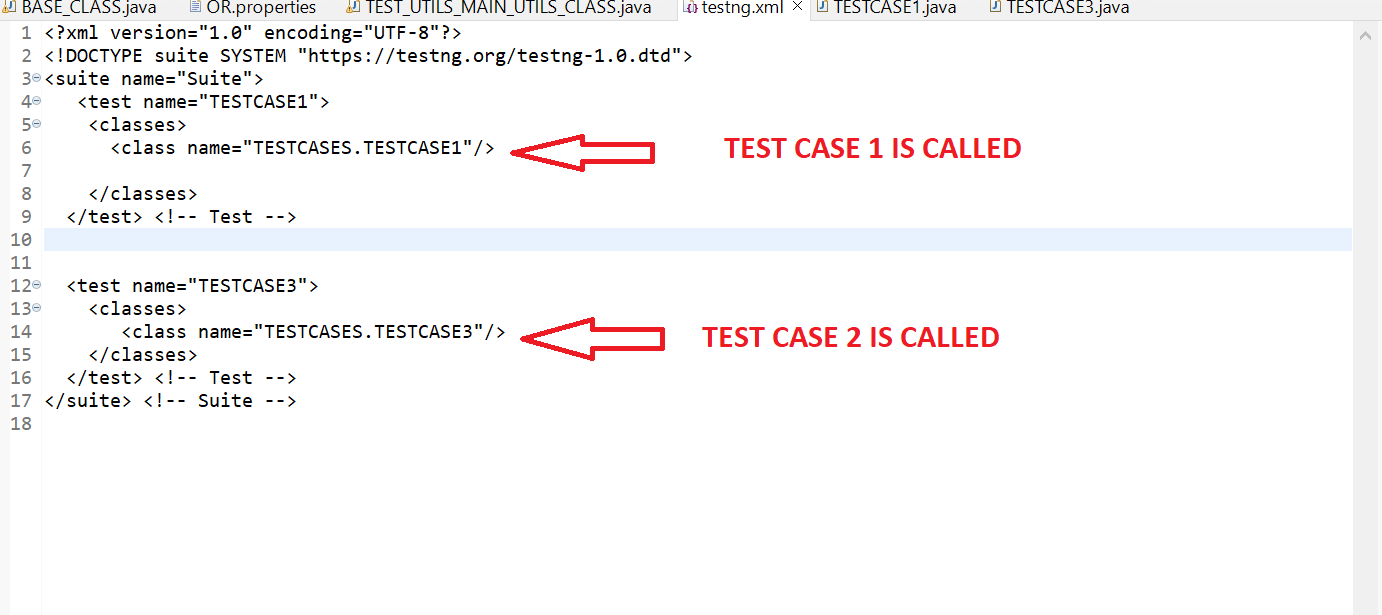
<classes>

<class name="TESTCASES.TESTCASE3"/>

</classes>

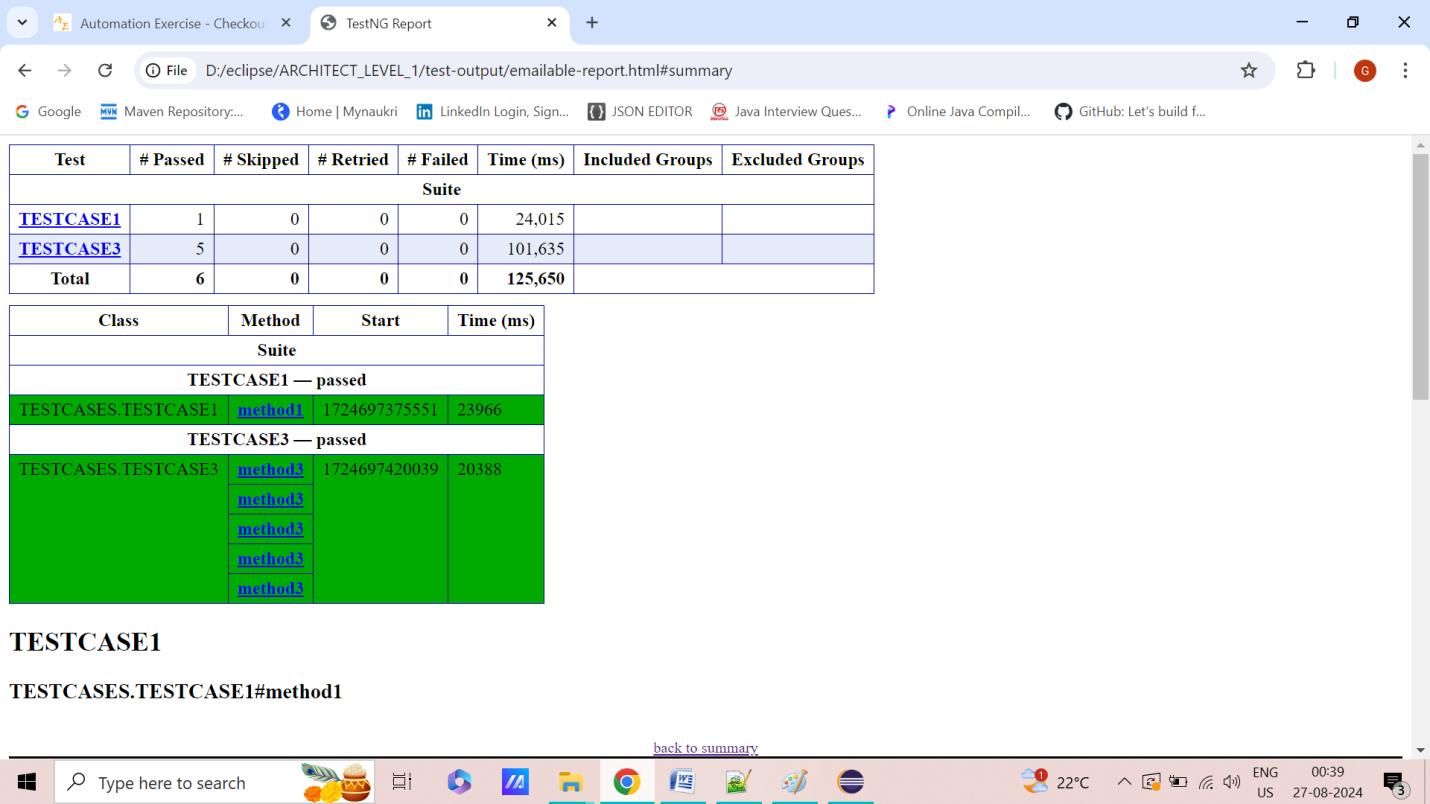
</test> <!-- Test -->

</suite> <!-- Suite -->

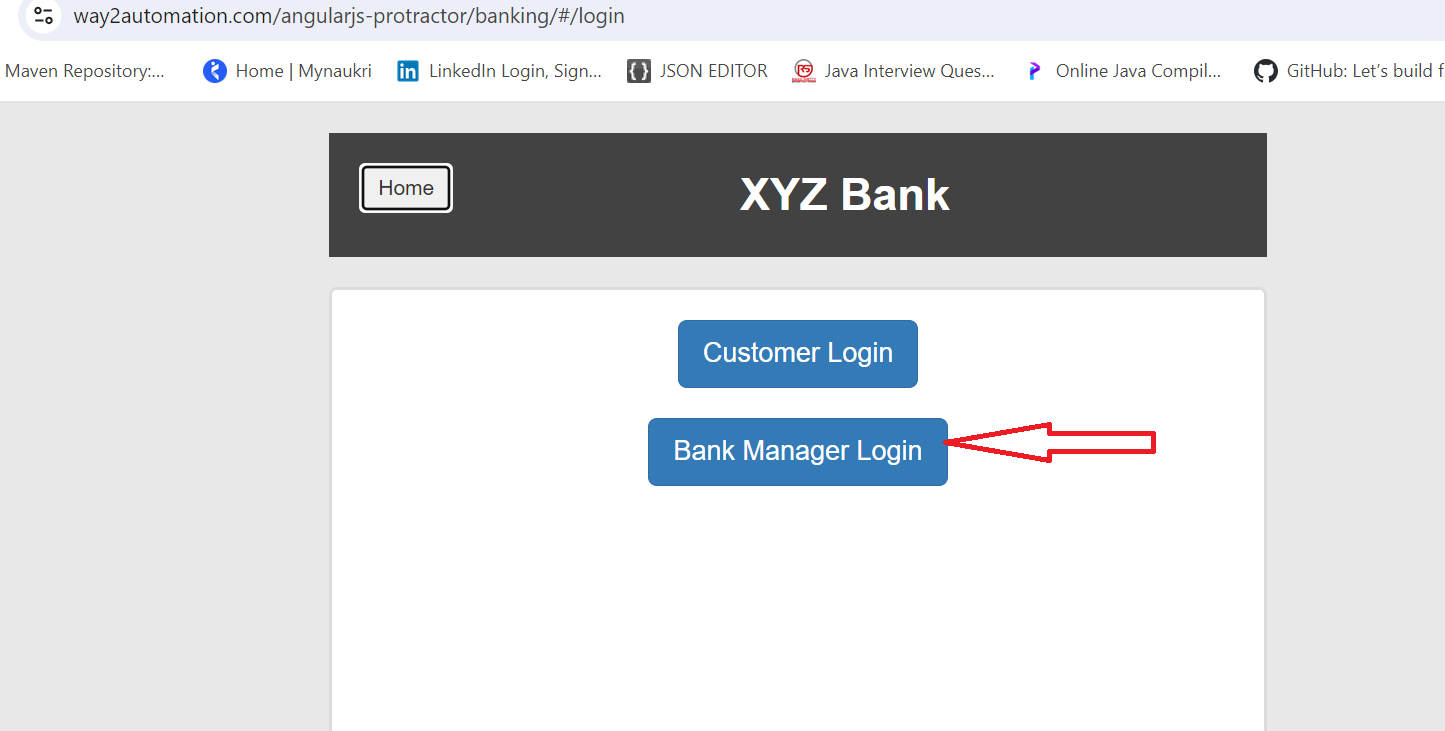
****

**STEPS 7=>THE UPDATED CODE IN “BASE CLASS ARE AS FOLLOW**

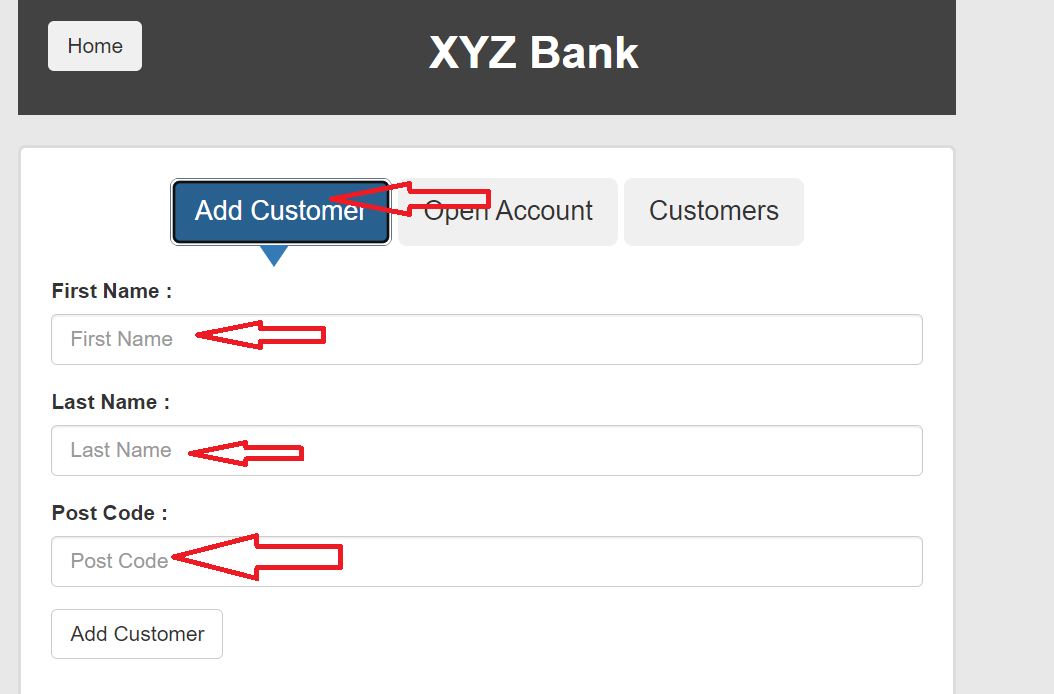
**STEPS 8=> RIGHT CLICK ON PROJECT AND RUN AS “TESTNG”.IT SHOULD WORK AS EXPECTED**

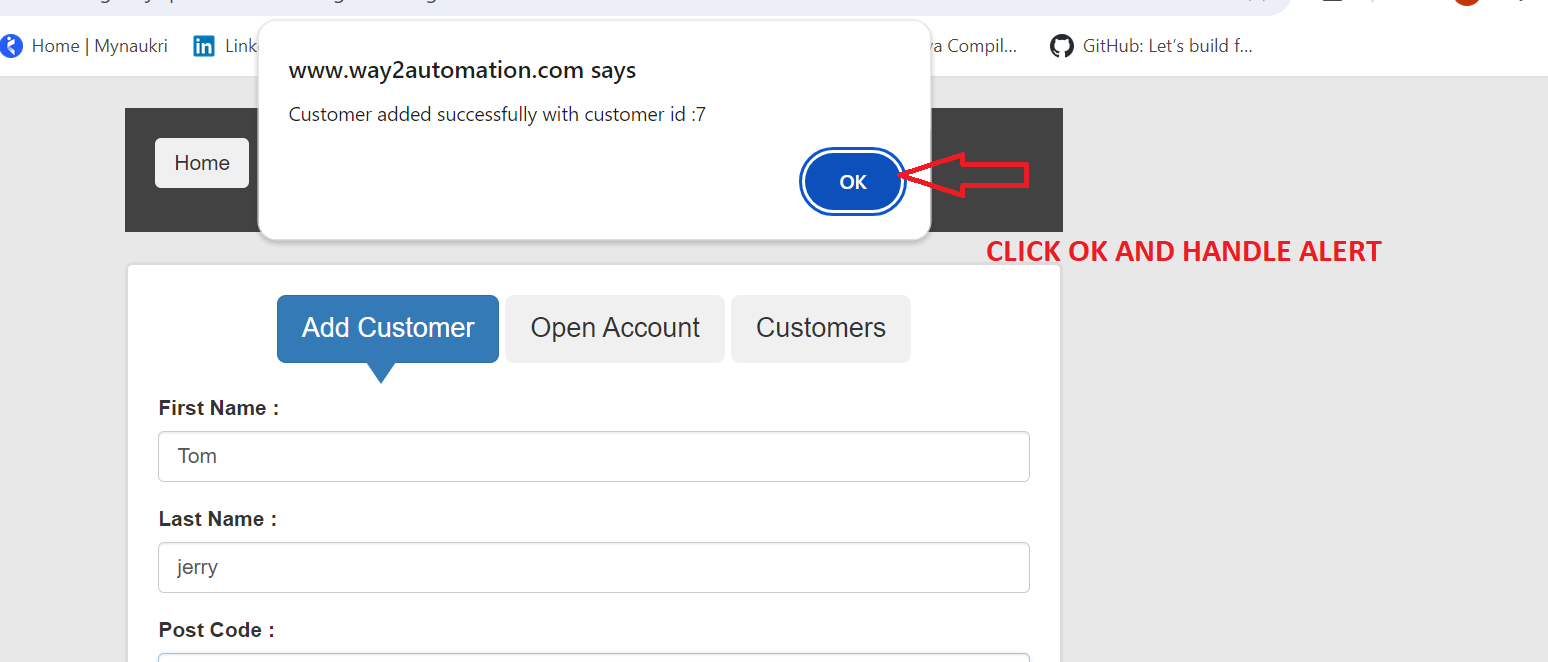
****

**PROJECT 2=>ADD “BANK CUSTOMER LOGINS” PAGE TO EXISTING PROJECTS**

****

**ENTER THE FIRST NAME,LAST NAME AND POSTAL CODE USING THE EXCEL SHEET**

****

****

**STEP 20=>WHAT IS DATA PROVIDER AND HOW TO USE THEM ?**

**STEPS 1=> CREATE NEW BASE CLASS WHERE LAUNCHING BROWSER DIFFERENT URL SHOULD BE PROVIDED AND “PUBLIC STATIC WEBDRIVER WAIT” IS DEFINED**

**package** BASE\_CLASS;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.IOException;

**import** java.time.Duration;

**import** java.util.Date;

**import** java.util.NoSuchElementException;

**import** java.util.Properties;

**import** java.util.concurrent.TimeUnit;

**import** org.apache.log4j.Logger;

**import** org.apache.log4j.PropertyConfigurator;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.chrome.ChromeDriver;

**import** org.openqa.selenium.chrome.ChromeOptions;

**import** org.openqa.selenium.edge.EdgeDriver;

**import** org.openqa.selenium.edge.EdgeOptions;

**import** org.openqa.selenium.firefox.FirefoxDriver;

**import** org.openqa.selenium.firefox.FirefoxOptions;

**import** org.openqa.selenium.support.ui.WebDriverWait;

**import** org.testng.annotations.AfterSuite;

**import** org.testng.annotations.BeforeSuite;

**import** UTILITIES.EXCEL\_READER;

**import** io.github.bonigarcia.wdm.WebDriverManager;

**public** **class** BASE\_CLASS2 {

/\*

\* INTIALISE WEBDRIVER LOGS PROPERTIES DB EXCEL MAIL IN BASE CLASS

\*/

**public** **static** WebDriver *driver*;

**public** **static** Properties *config* = **new** Properties();

**public** **static** Properties *OR* = **new** Properties();

**public** **static** Logger *log* = Logger.*getLogger*(BASE\_CLASS2.**class**.getName());

**public** **static** EXCEL\_READER *xlsreadER* = **new** EXCEL\_READER(

System.*getProperty*("user.dir") + "\\src\\test\\resources\\EXCEL\_SHEET\\EXCEL\_SHEET\_DATA\_PROVIDER\_CREATED.xlsx");

**public** **static** WebDriverWait *wait* ;

@BeforeSuite

**public** **void** setup() **throws** IOException, InterruptedException

{

Date d = **new** Date();

System.*setProperty*("current.date", d.toString().replace(":", "\_").replace(" ", "\_"));

PropertyConfigurator.*configure*("D:\\eclipse\\ARCHITECT\_LEVEL\_1\\src\\test\\resources\\LOGS\\LOG4J.properties");

**if** (*driver* == **null**)

{

FileInputStream fis = **new** FileInputStream(

System.*getProperty*("user.dir") + "\\src\\test\\resources\\PROPERTIES\\CONFIG.properties");

*config*.load(fis);

*log*.info("CONFIG FILE IS LOADED!!!");

fis = **new** FileInputStream(

System.*getProperty*("user.dir") + "\\src\\test\\resources\\PROPERTIES\\OR.properties");

*OR*.load(fis);

*log*.info("OR FILE IS LOADED!!!");

**if**(*config*.getProperty("BROWSER").equals("CHROME"))

{

System.*setProperty*("Webdriver.chrome.driver",*config*.getProperty("chromepath"));

ChromeOptions option = **new** ChromeOptions();

option.addArguments("--remote-allow-origins=\*");

WebDriverManager.*chromedriver*().setup();

*driver* = **new** ChromeDriver(option);

*log*.info("CHROME BROWSER IS LOADED!!!");

}

**else** **if**(*config*.getProperty("BROWSER").equals("firefox"))

{

System.*setProperty*("Webdriver.gecko.driver",*config*.getProperty("geckodriverpath"));

FirefoxOptions option = **new** FirefoxOptions();

//option.addArguments("--remote-allow-origins=\*");

WebDriverManager.*firefoxdriver*().setup();

*driver* = **new** FirefoxDriver(option);

*log*.info("FIREFOX BROWSER IS LOADED!!!");

}

**else** **if**(*config*.getProperty("BROWSER").equals("ie"))

{

System.*setProperty*("Webdriver.edge.driver",*config*.getProperty("ie\_driverpath"));

EdgeOptions option = **new** EdgeOptions ();

option.addArguments("--remote-allow-origins=\*");

WebDriverManager.*edgedriver*().setup();

*driver* =**new** EdgeDriver(option);

*log*.info("EDGE BROWSER IS LOADED!!!");

}

*log*.info("TESTSITEURL2 IS LOADED!!!");

*driver*.get(*config*.getProperty("TESTSITEURL2"));

Thread.*sleep*(3000);

*driver*.manage().window().maximize();

*driver*.manage().deleteAllCookies();

*driver*.manage().timeouts().implicitlyWait(Duration.*ofHours*(2));

*wait* = **new** WebDriverWait(*driver*, Duration.*ofSeconds*(10));

}

}

**public** **boolean** IsElementPresent(By by)

{

**try**

{

*driver*.findElement(by);

**return** **true**;

}

**catch**(NoSuchElementException e)

{

**return** **false**;

}

}

@AfterSuite

**public** **void** Tearup() {

**if**(*driver*!=**null**)

{

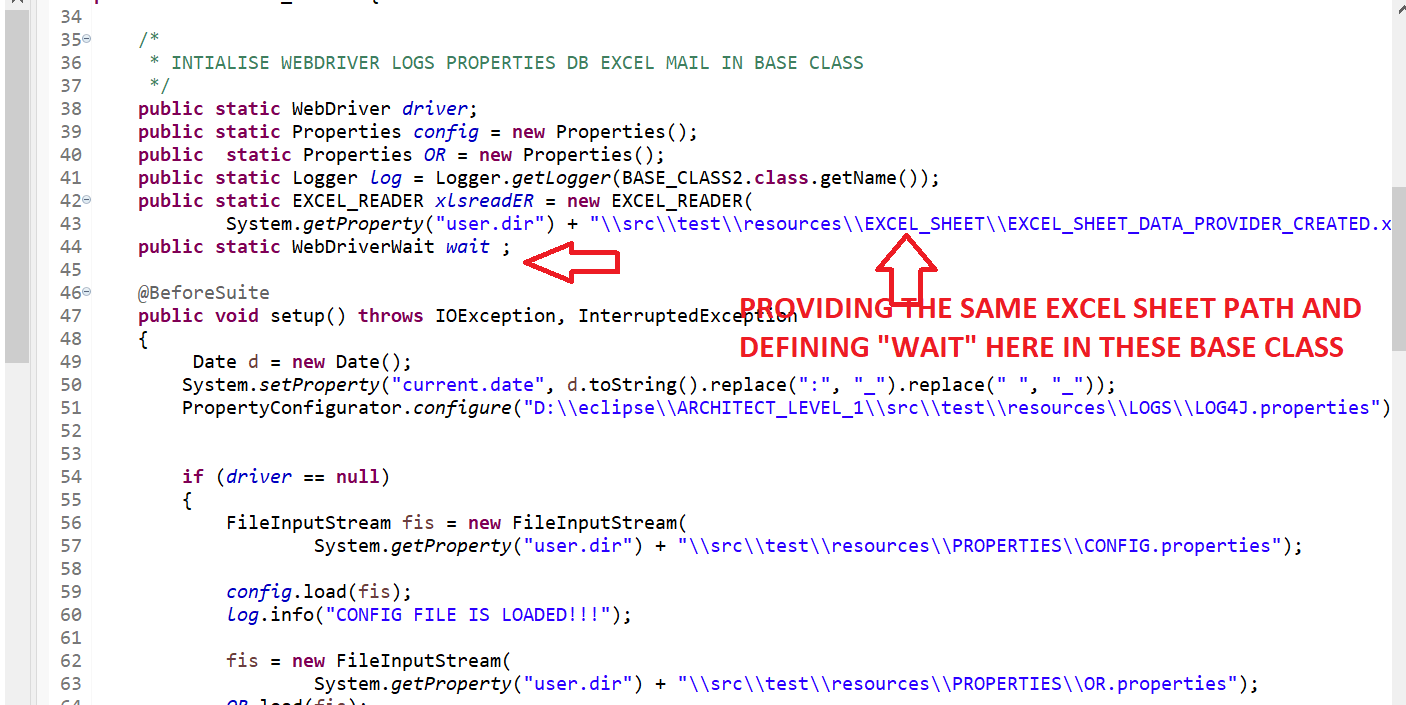
*driver*.quit();

}

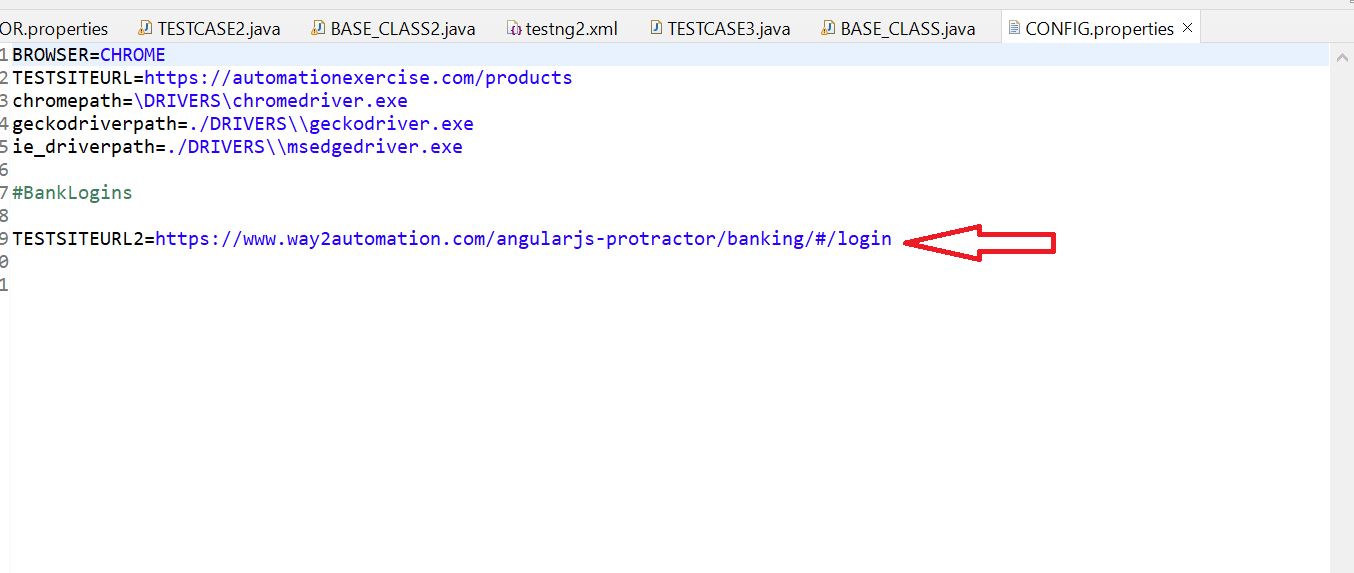
*log*.info("CLOSE BROWSER IS ENABLED!!!");

}

}

****

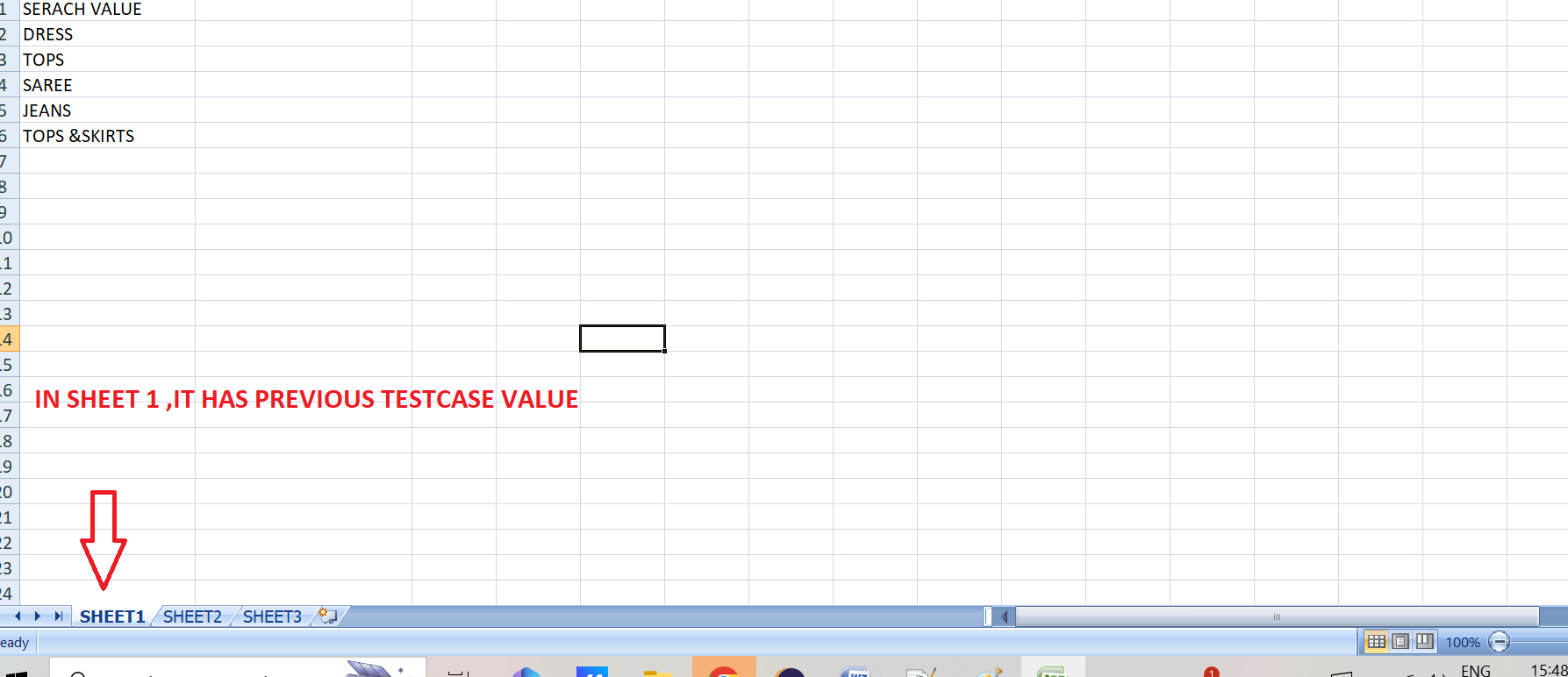
****

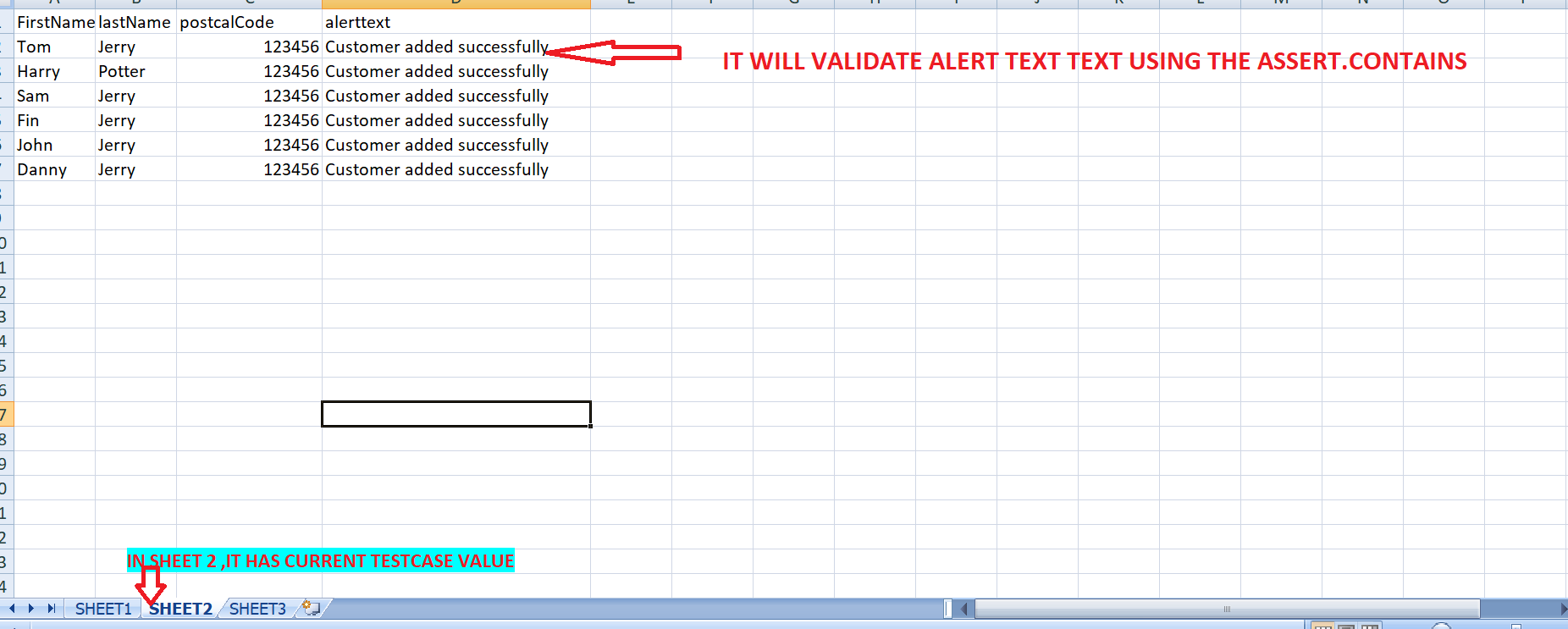
****

**STEPS 2=> CREATE NEW EXCEL SHEET WITH BELOW FIELD**

**IN SHEET 1 ,IT HAS PREVIOUS TESTCASE VALUE**

**IN SHEET 2 ,IT HAS CURRENT TESTCASE VALUE**

****

****

**STEPS 3=> BELOW IS TESTCASES THAT NEED TO BE CALLED .CREATE NEW TEST METHOD WITH “DATAPROVIDER” AND WRITING ONLY THOSE VALUE THAT NEED TO BE FETCHED FROM EXCEL SHEET SUCH AS “FIRST NAME”,”LASTNAME”,”POSTAL CODE”, “ALERTTEXT”**

**package** TESTCASES;

**import** java.time.Duration;

**import** org.openqa.selenium.Alert;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.support.ui.ExpectedConditions;

**import** org.testng.Assert;

**import** org.testng.annotations.DataProvider;

**import** org.testng.annotations.Test;

**import** BASE\_CLASS.BASE\_CLASS2;

**import** UTILITIES.EXCEL\_READER;

**public** **class** TESTCASE2 **extends** BASE\_CLASS2

{

@Test()

**public** **void** method1() **throws** InterruptedException

{

*log*.info("TEST CASE2 STARTED RUNNING!!!");

Thread.*sleep*(3000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("customer\_login"))).click();

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("Addcusotmer"))).click();

}

@Test(dataProvider="DataProvide\_4")

**public** **void** method6(String FirstName,String lastName,String postcalCode,String alerttext) **throws** Throwable

{

*log*.info("First Name,last Name and postcalCode name is entered using the above method!!!");

Thread.*sleep*(3000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("firstname"))).sendKeys(FirstName);

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("lastname"))).sendKeys(lastName);

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("postalcode"))).sendKeys(postcalCode);

Thread.*sleep*(2000);

*driver*.findElement(By.*xpath*(*OR*.getProperty("add\_cusomter\_button"))).click();

Thread.*sleep*(2000);

Alert alert = *wait*.until(ExpectedConditions.*alertIsPresent*());

Thread.*sleep*(2000);

Assert.*assertTrue*(alert.getText().contains(alerttext));

alert.accept();

*log*.info("TEST CASE 2 completed!!!");

}

@DataProvider(name="DataProvide\_4")

**public** Object[][] getData() {

**if** (*xlsreadER* == **null**) {

*xlsreadER* = **new** EXCEL\_READER("D:\\eclipse\\ARCHITECT\_LEVEL\_1\\src\\test\\resources\\EXCEL\_SHEET\\EXCEL\_SHEET\_DATA\_PROVIDER\_CREATED.xlsx");

}

String sheetName = "SHEET2";

**int** rows = *xlsreadER*.getRowCount(sheetName);

**int** cols = *xlsreadER*.getColumnCount(sheetName);

Object[][] data = **new** Object[rows-1][cols];

**for** (**int** rowNum = 2; rowNum <= rows; rowNum++) {

**for** (**int** colNum = 0; colNum < cols; colNum++) {

data[rowNum - 2][colNum] = *xlsreadER*.getCellData(sheetName, colNum, rowNum);

}

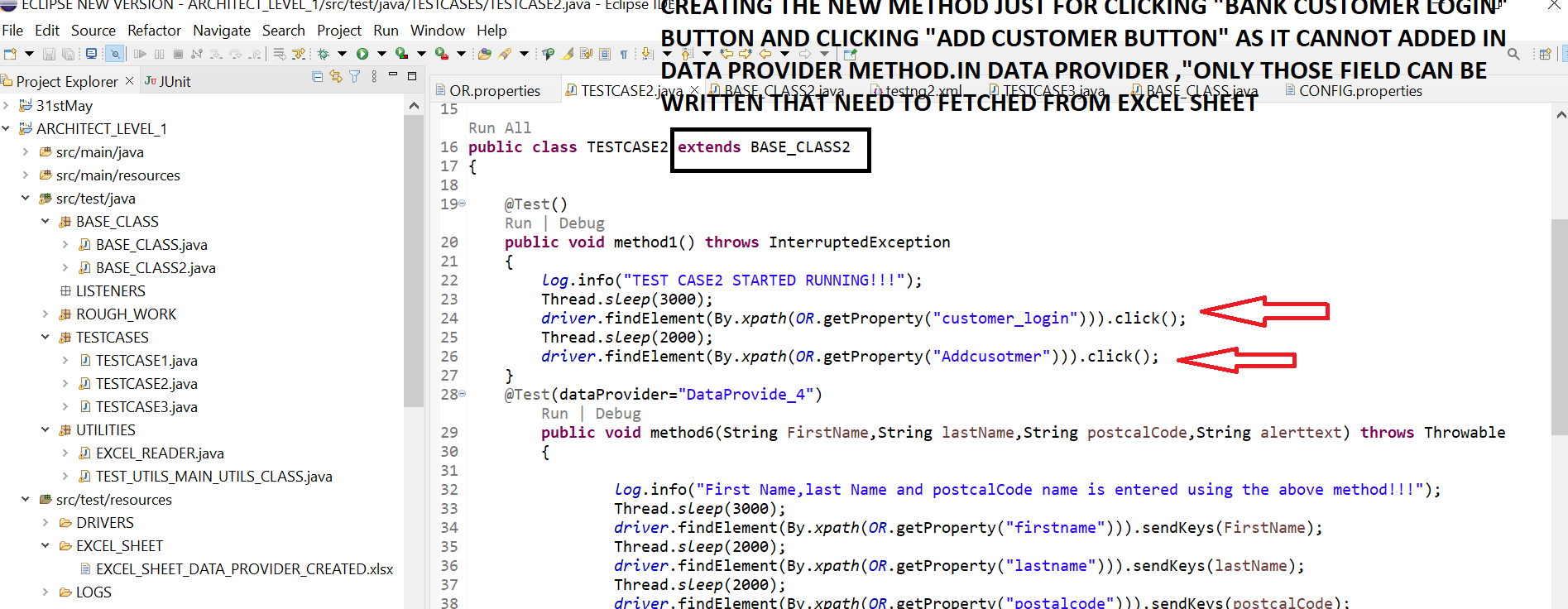
}

**return** data;

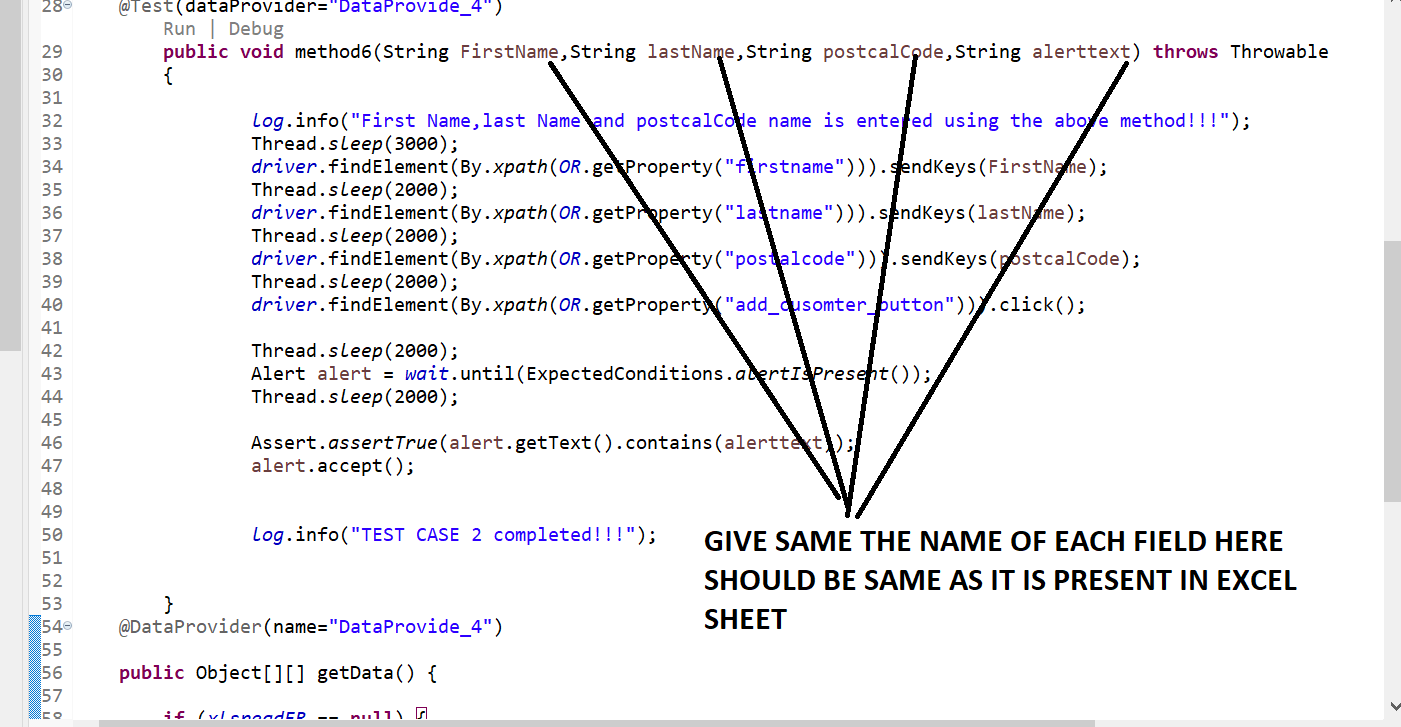
}

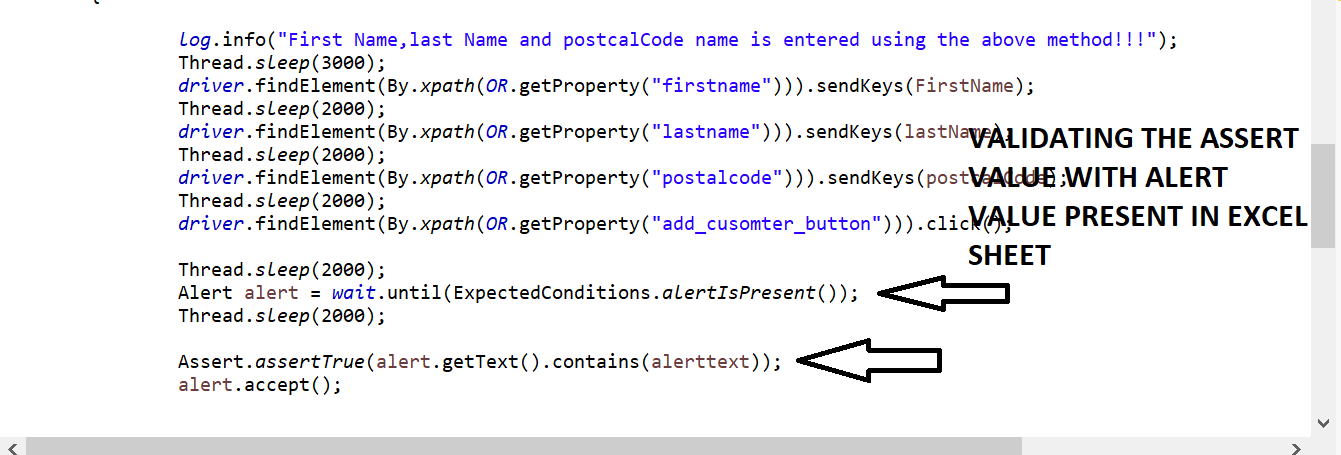
**EXPLANATION**

**CREATING THE NEW METHOD JUST FOR CLICKING "BANK CUSTOMER LOGIN" BUTTON AND CLICKING "ADD CUSTOMER BUTTON" AS IT CANNOT ADDED IN DATA PROVIDER METHOD.IN DATA PROVIDER ,"ONLY THOSE FIELD CAN BE WRITTEN THAT NEED TO FETCHED FROM EXCEL SHEET**

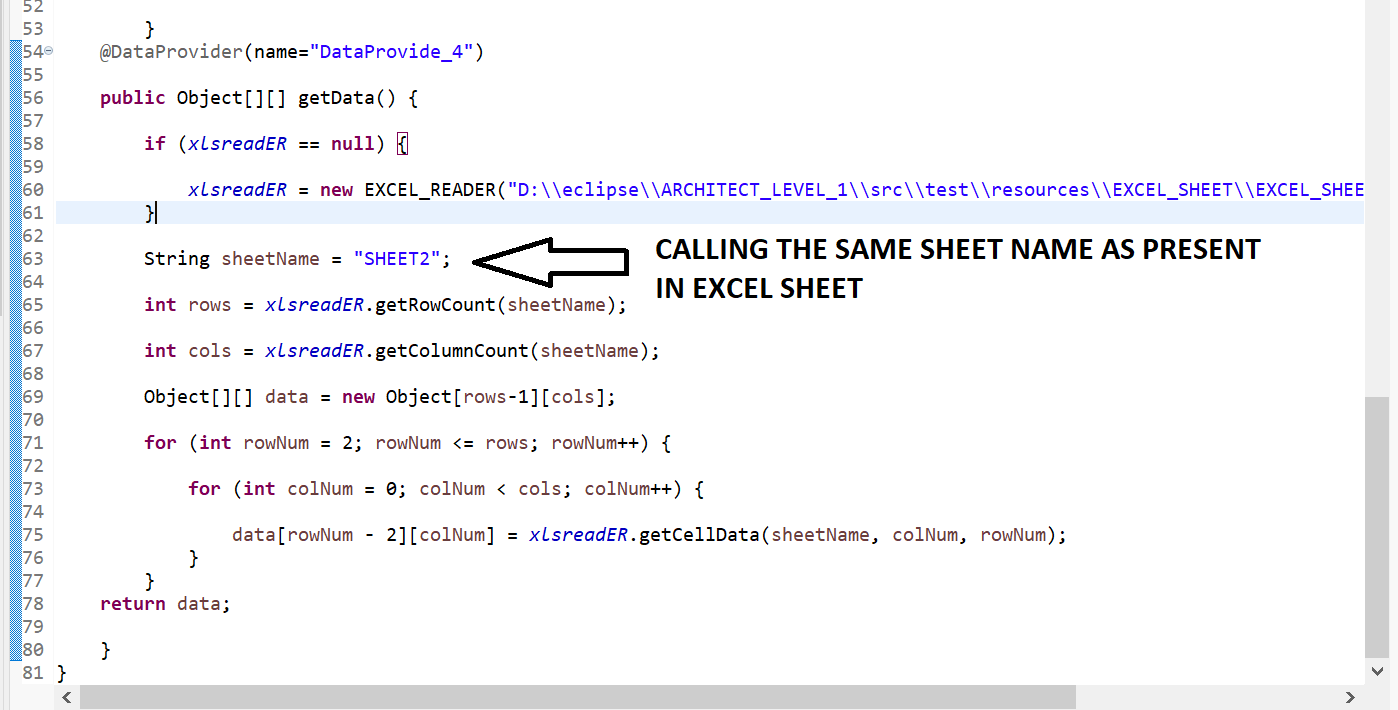
****

**GIVE SAME THE NAME OF EACH FIELD HERE SHOULD BE SAME AS IT IS PRESENT IN EXCEL SHEET THEN CALL THE SAME SHEET NAME AS PRESENT IN EXCEL SHEET**

****

****

**THEN CALL THE SAME SHEET NAME AS PRESENT IN EXCEL SHEET**

****

**STEPS 4=> CREATE “NEW TESTNG.XML” FILE WITH CALLING THE ABOVE TEST CASE**

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

<suite name="Suite">

<test name="TESTCASE1">

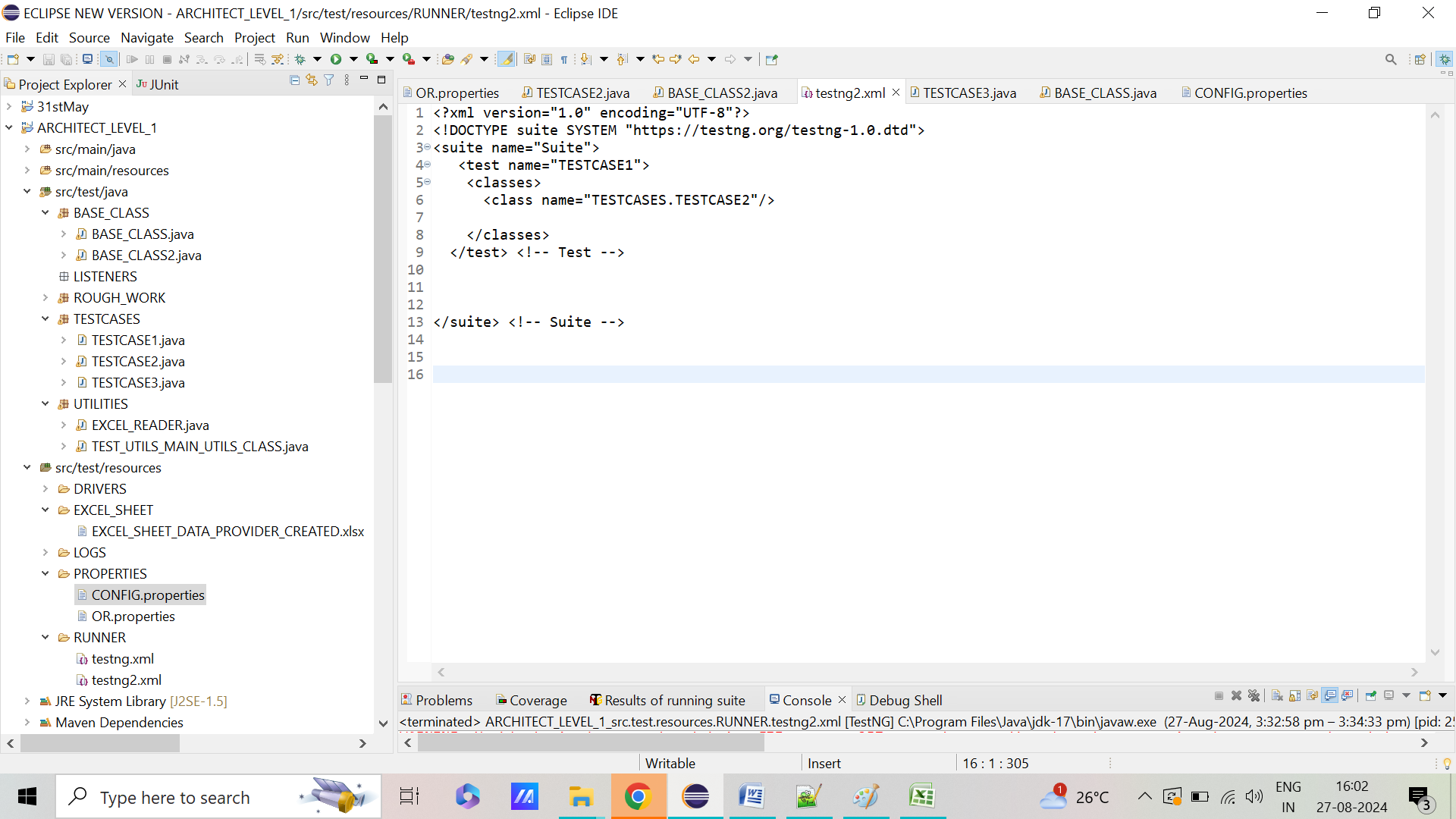
<classes>

<class name="TESTCASES.TESTCASE2"/>

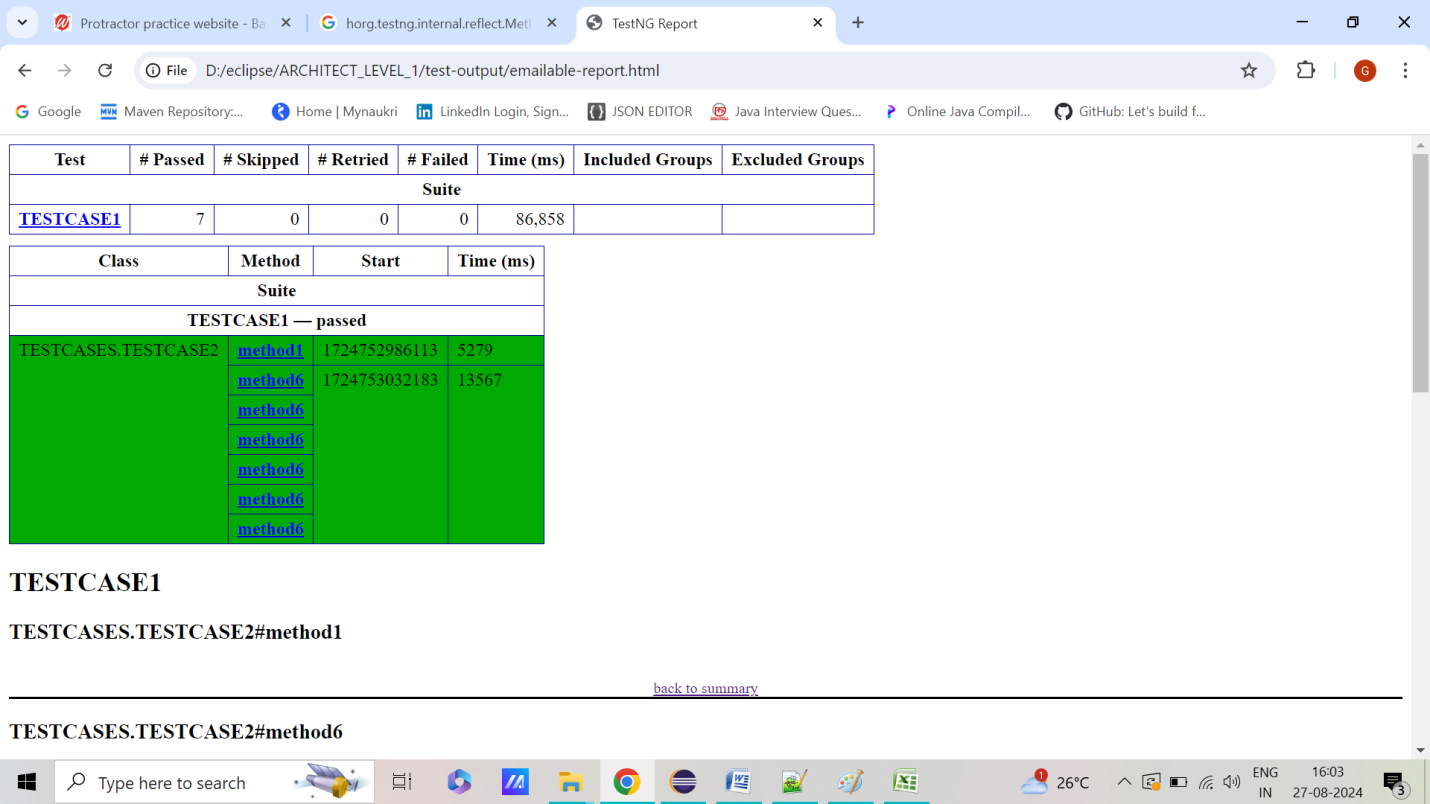
</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

****

**STEPS 5=> RIGHT CLICK ON PROJECT AND RUN AS “TESTNG”.IT SHOULD WORK AS EXPECTED**

****