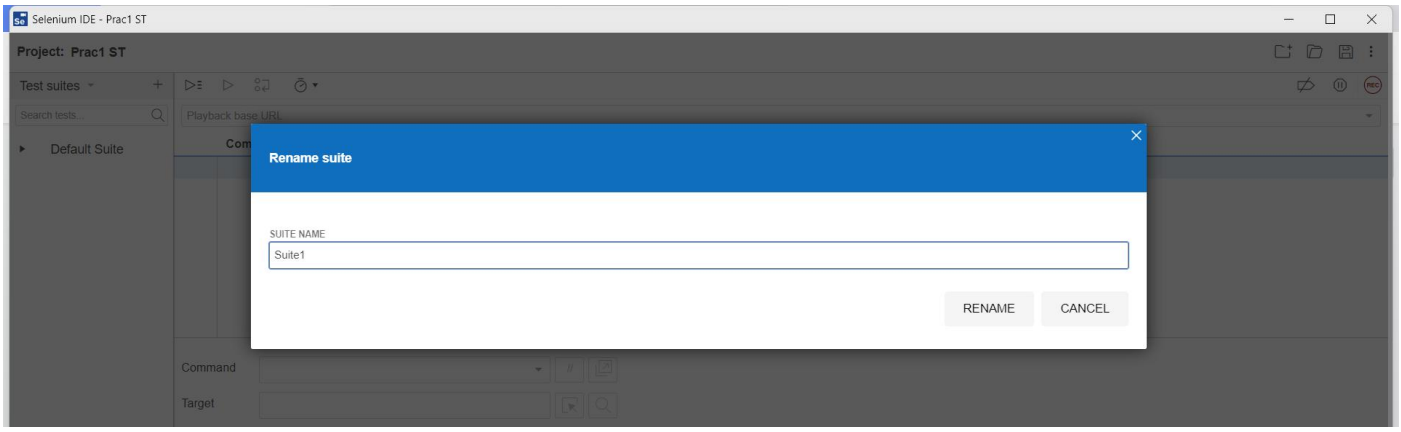
TYCS SEM 5, SUBJECT: Software Testing and Quality Assurance, Name: Subiksh Shashidharan, Roll no: 08

Practical 1

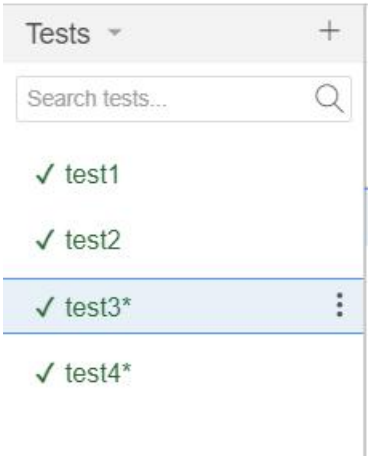
Aim : Install Selenium IDE; Write a test suite containing minimum 4 test cases for different formats.

STEPS:

1) Google “Selenium IDE chrome” and click on the first link. Add the “Selenium IDE” extension to your browser.
2) For testing, we need to create a test case suite, so we create the same in Selenium IDE.
3) Press CTRL+2 and Rename the Suite, “Suite1”



4) Add tests in the Selenium IDE, total 4 tests
5) We will use 4 different websites in each of the tests



6) For Test1:

https://www.google.com

	Command	Target	Value
1	*open*	/	
2	*set window size*	1297x816	
3	*click*	name=q	
4	*type*	name=q	abcdef
5	*send keys*	name=q	${KEY_ENTER}

7) For Test2:

https://classroom.google.com/u/0/

	Command	Target	Value
1	open	/	
2	set window size	1298x816	
3	type	id=sb_form_q	bing search
4	send keys	id=sb_form_q	${KEY_ENTER}
5	mouse over	id=desc	
6	mouse out	id=desc	
7	click	linkText=Bing	

8) For Test3:

https://classroom.google.com/u/0/

	Command	Target	Value
1	open	https://www.calculator.net/	
2	set window size	1298x816	
3	click	css=div:nth-child(1) > .scinm:nth-child(2)	
4	click	css=div:nth-child(3) > .sciop:nth-child(4)	
5	click	css=div:nth-child(3) > .scinm:nth-child(2)	
6	click	css=.scieq:nth-child(4)	

9) For Test4:

https://classroom.google.com/u/0/

	Command	Target	Value
1	✓ open	https://classroom.google.com/u/0/	
2	✓ set window size	1297x816	
3	✓ mouse over	css=.GR7Qld .NMm5M	
4	✓ click	css=.gHz6xd:nth-child(2) .YVvGBb:nth-child(2)	
5	✓ close		

10) Press CTRL+2 to go to Suites and Add 4 Tests
11) Now we can run this suite which will run all the 4 tests

12) **Note** : To play a suite you must select a test case from within that suite

- ✓ Suite1*
  - ✓ test3
  - ✓ test4
  - ✓ test1*
  - ✓ test2*

	Command	Target
1	✓ open	https://www.calculator.net/
2	✓ set window size	1298x816
3	✓ click	css=div:nth-child(1) > .scinm:nth-child(2)
4	✓ click	css=div:nth-child(3) > .sciop:nth-child(4)
5	✓ click	css=div:nth-child(3) > .scinm:nth-child(2)
6	✓ click	css=.scieq:nth-child(4)

Practical 2

Aim : Conduct a test suite for any two web sites.

1) Create a New Test Suite, we name it Suite1
2) Create two tests for two different websites
3) Test 1 (t1) :

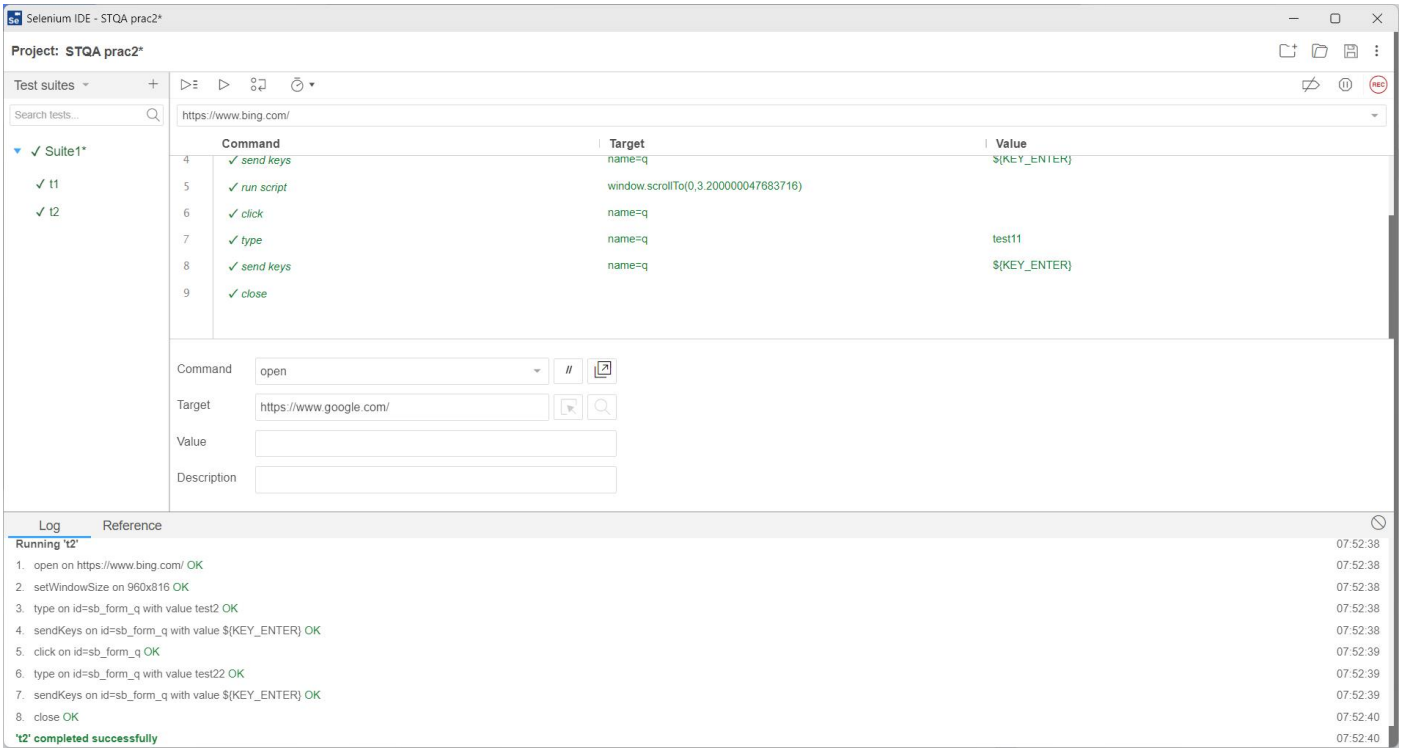
1.open on https://www.google.com/ OK07:41:27
2.setWindowSize on 960x816 OK07:41:27
3.type on name=q with value test1 OK07:41:27
4.sendKeys on name=q with value ${KEY_ENTER} OK07:41:28
5.runScript on window.scrollTo(0,3.200000047683716) OK07:41:28
6.click on name=q OK07:41:30
7.type on name=q with value test11 OK07:41:30
8.sendKeys on name=q with value ${KEY_ENTER} OK07:41:30
9.close OK

4) Test 2 (t2) :

1.open on https://www.bing.com/ OK07:48:10
2.setWindowSize on 960x816 OK07:48:11
3.type on id=sb_form_q with value test2 OK07:48:11
4.sendKeys on id=sb_form_q with value ${KEY_ENTER} OK07:48:11
5.click on id=sb_form_q OK07:48:12
6.type on id=sb_form_q with value test22 OK07:48:12
7.sendKeys on id=sb_form_q with value ${KEY_ENTER} OK07:48:12
8.close OK07:48:13

4) Press CTRL+2 to go to Suites and Add 4 Tests
5) Now we can run this suite which will run all the 4 tests

6) **Note** : To play a suite you must select a test case from within that suite



Practical 3

Aim : Install Selenium server (Selenium RC) and demonstrate it using a script in Java/PHP

Selenium-RC

1. Introduction

Selenium-RCisthesolutionforteststhatneedmorethansimplebrowseractions and linear execution.
Selenium-RCusesthefullpowerofprogramminglanguagestocreatemore complextestslikereadingandwritingfiles,queryingadatabase,emailingtest results.
You'llwanttouseSelenium-RCwheneveryourtestrequireslogicnotsupported by Selenium-IDE.
What logic could this be? For example, Selenium-IDE does not directly support:
• condition statements
• iteration
• logging and reporting of testresults
• error handling, particularly unexpected errors
• database testing
• test case grouping
• re-execution of failed tests
• test case dependency
• screenshot capture of testfailures
Although these tasks are not supported bySelenium directly, all of them can be achieved by using programming techniques with a language-specific Selenium-RC client library.

Pre-requisites :

1) To Download "JDK":

• Visit https://www.oracle.com/technetwork/java/javase/downloads/
• Download the jdk 8 and install it.

2) To Download "Netbeans 8.2 IDE":

• Visit https://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-3413139-esa.html
• Click "Download".
• Installation:
  Open Setup File
  It will automatically locate the JDK. Choose path, and click "Install".
• After installation, click "Launch" or open Netbeans 8.2 from START menu in Windows.

3) To Download "Selenium Server Driver and Client Driver(JAR files)":

a) For "Selenium Server Driver":

• Visit https://www.selenium.dev/downloads
• Under section "Selenium Server (Grid)", click download **Latest stable version 4.3.0**
• You'll get the executable jar file(selenium-server-standalone-4.3.0)

b) For "Selenium Java Driver":

• Visit - https://github.com/SeleniumHQ/selenium/releases/download/selenium-4.3.0/selenium-java-4.3.0.zip
• Extract the file and you'll see two jar files. From them, we'll be using this

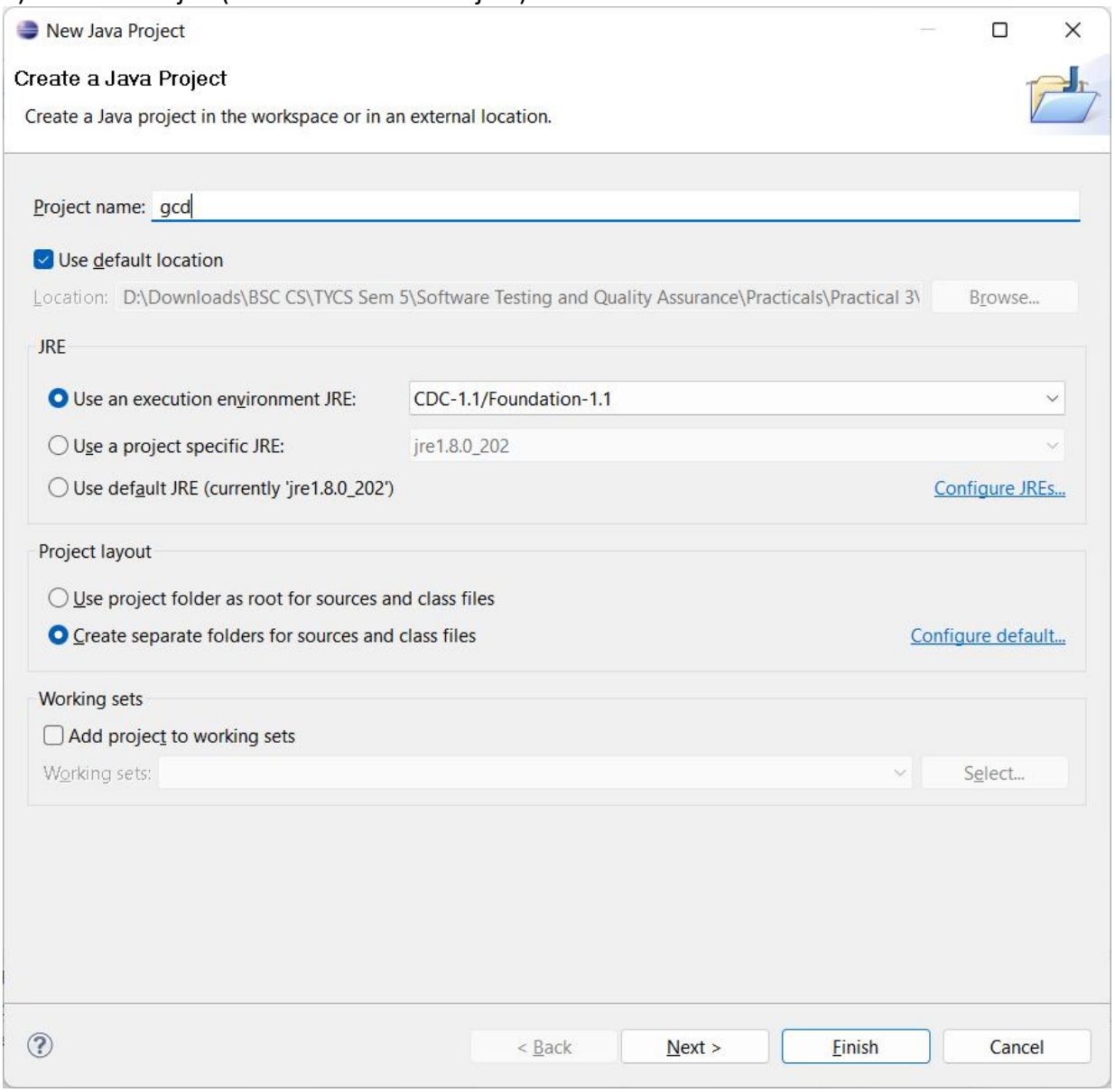
executable jar file

4) **To Download "Chromium Web Driver":**

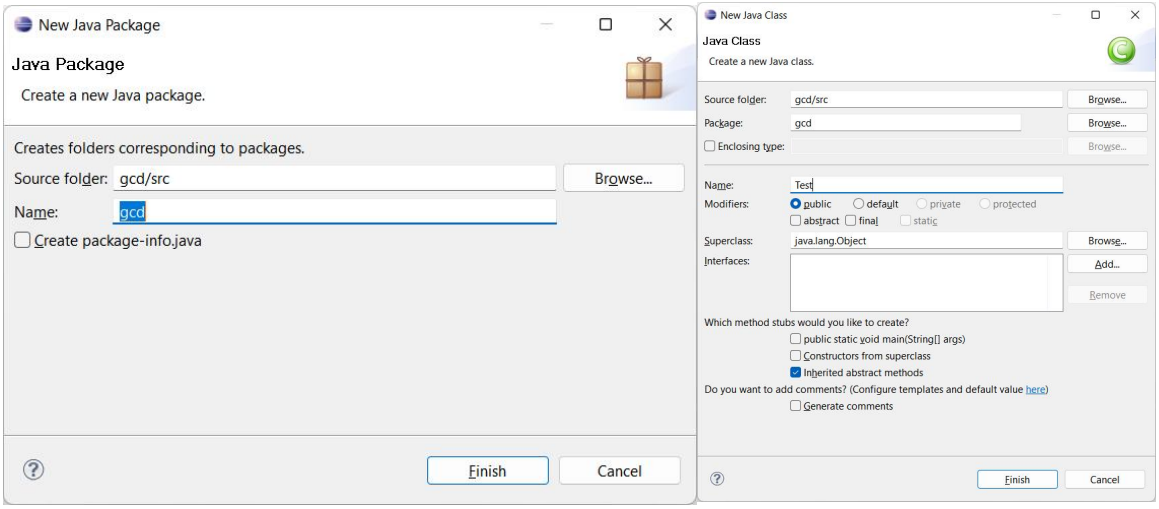
- Visit https://chromedriver.storage.googleapis.com/index.html?path=103.0.5060.134/ for 103 version
- Extract the zip file
- You'll get the application file "chromiumdriver.exe".

STEPS:

1) Open Eclipse. Select your workspace directory. Click Launch:
2) Create a Project(File > New > Java Project):

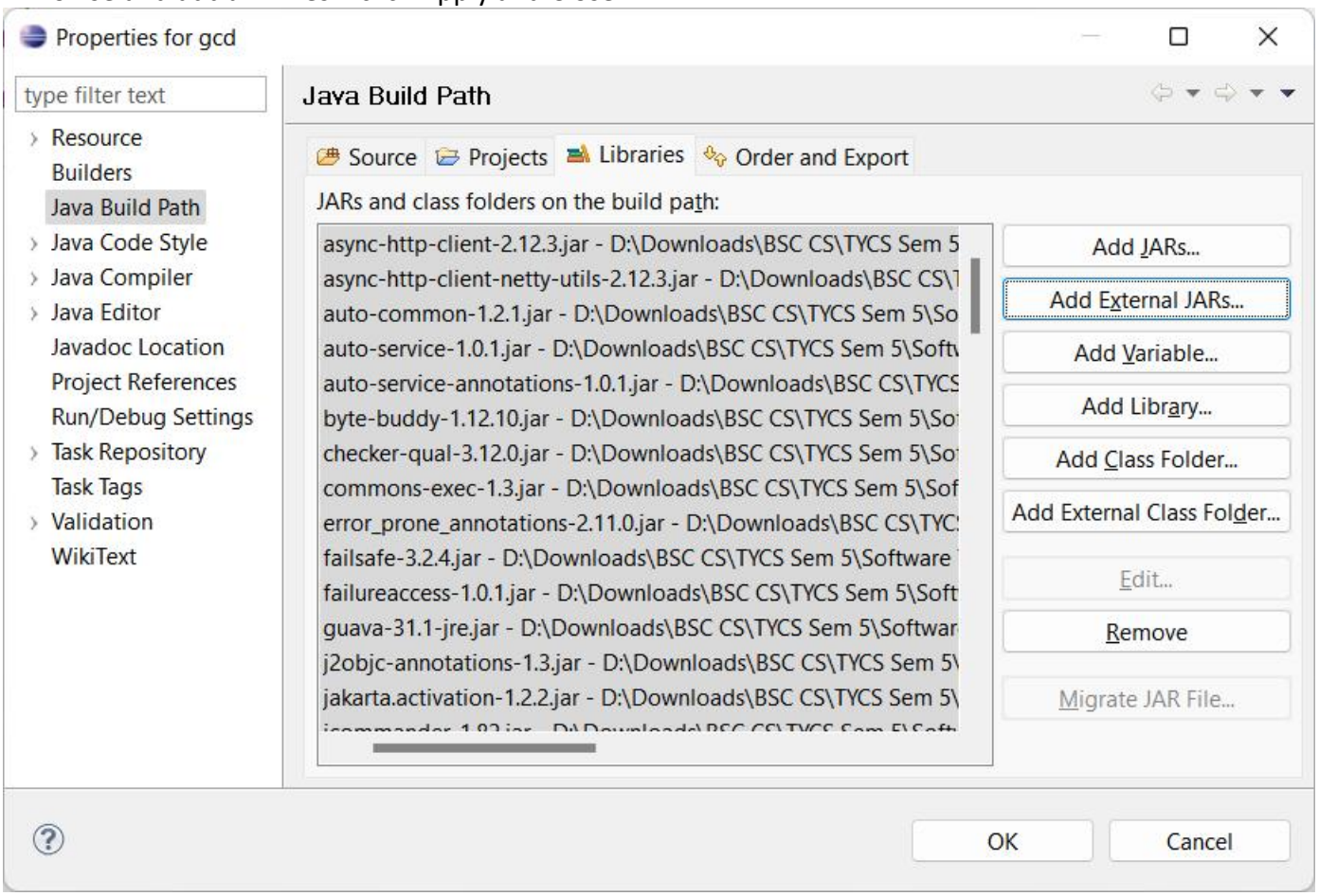


3) Name the project as "gcd" > click Finish > click Don't Create module:
4) Close the "Welcome" tab.
5) Create a Package(right-click on Project Name > New > Package > Name it > Finish):
6) Create a Class(right-click on Project Name > New > Class > Name it > Finish):



7) Adding “Selenium Server Driver and Client Driver(JAR files)” in Eclipse IDE:

- right-click on Project Name > Build Path > Configure Build Path…
- now go under: Java Build Path > Libraries > Classpath > click Add External JARs…
- Browse and add JAR files > click Apply and Close :



8) Creating a link for HTML file(wherein calculation part is present):
(NOTE that this file will be run by the ‘script in JAVA’(which we’ll create later))

- Create a Notepad file with the following code and save it as “gcdhtml.html”:

---(gcdhtml.html)---

```
<html>
<head>
<script type="text/javascript">
```

```
function gcd()
{
var x,y;
x=parseInt(document.myform.n1.value);
y=parseInt(document.myform.n2.value);
while(x!=y)
{
if(x>y){x=x-y;}
else{y=y-x;}
}
document.myform.result.value=x;
}
</script>
</head>
<body>
<center>
<h1>---Program to calculate GCD of two numbers---</h1>
<hr color="red">
<form name="myform">
Enter Number 1: <input type="text" name="n1" value=""> <br> <br>
Enter Number 2: <input type="text" name="n2" value=""> <br> <br>
<input type="button" name="btn" value="Get GCD" onClick="gcd()"><br><br>
GCD: <input type="text" name="result" value="">
</form>
</center>
</body>
</html>
```

• Close the file. Then right-click > Open with > Chrome Browser • Copy URL from the webpage:

9) Creating the script in JAVA:
(NOTE that this script will be run by Eclipse IDE)
(In simple words, it's like we are
-ordering Eclipse to run a script or to do a job
-of opening the HTML file
-and putting the values in the textboxes with the help of Selenium Drivers
-and to show the result.
-Hence automating the work in browser)

• Now we'll put the path of "chromiumdriver" in a String driverPath
• And we'll paste the copied URL in the .get() method of the WebDriver class

---(Test.java)---

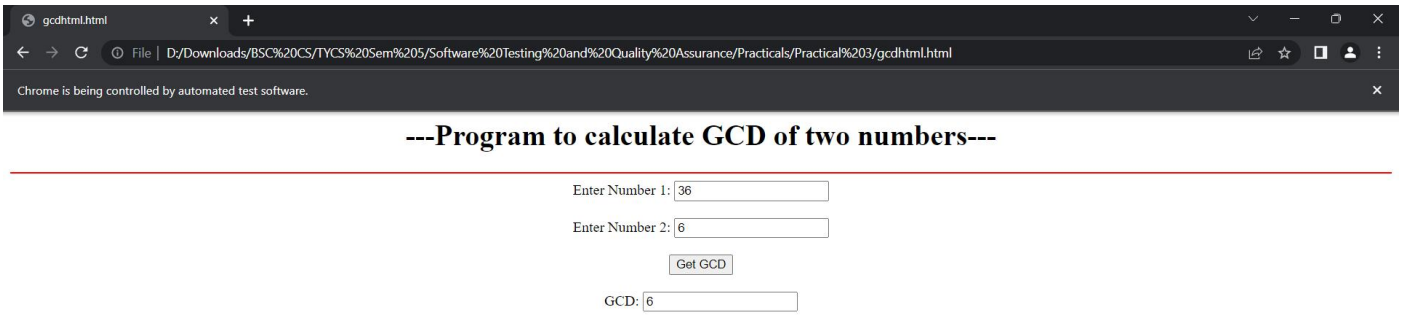
```
package gcd;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Test {
      private static WebDriver driver = null;
    public static void main(String[] args) {
```

```
    System.setProperty("webdriver.chrome.driver", "D:\\Downloads\\BSC CS\\TYCS Sem
5\\Software Testing and Quality Assurance\\Practicals\\Practical 4
php\\chromedriver.exe");
    driver = new ChromeDriver();

      driver=new ChromeDriver();
      driver.get("file:///D:/Downloads/BSC CS/TYCS Sem 5/Software Testing and Quality
Assurance/Practicals/Practical 3/gcdhtml.html");
      driver.manage().window().maximize();
      driver.findElement(By.name("n1")).sendKeys("36");
      driver.findElement(By.name("n2")).sendKeys("6");
      driver.findElement(By.name("btn")).click();
      String
      result=driver.findElement(By.name("result")).getAttribute("name=result");
      System.out.println("GCD="+result);
}
}
```

NOTE[Keep Java Compiler(under Configure Build Path to 1.7) if getting error : `method sendKeys(CharSequence[]) in the type WebElement is not applicable for the arguments (String)`

10) Run the file from Eclipse IDE:

• OUTPUT:



11) Finish!

Practical 4

Aim : Write and test a program to login a specific web page.

1) To Download "JDK":

- Visit https://www.oracle.com/technetwork/java/javase/downloads/
- Download the jdk 8 and install it.

2) To Download "Netbeans 8.2 IDE":

- Visit https://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-3413139-esa.html
- Click "Download".
- Installation:
  Open Setup File
  It will automatically locate the JDK. Choose path, and click "Install".
- After installation, click "Launch" or open Netbeans 8.2 from START menu in Windows.

3) To Download "Selenium Server Driver and Client Driver(JAR files)":

a) For "Selenium Server Driver":

- Visit https://www.selenium.dev/downloads
- Under section "Selenium Server (Grid)", click download **Latest stable version 4.3.0**
- You'll get the executable jar file(selenium-server-standalone-4.3.0)

b) For "Selenium Java Driver":

- Visit - https://github.com/SeleniumHQ/selenium/releases/download/selenium-4.3.0/selenium-java-4.3.0.zip
- Extract the file and you'll see two jar files. From them, we'll be using this executable jar file

4) To Download "Chromium Web Driver":

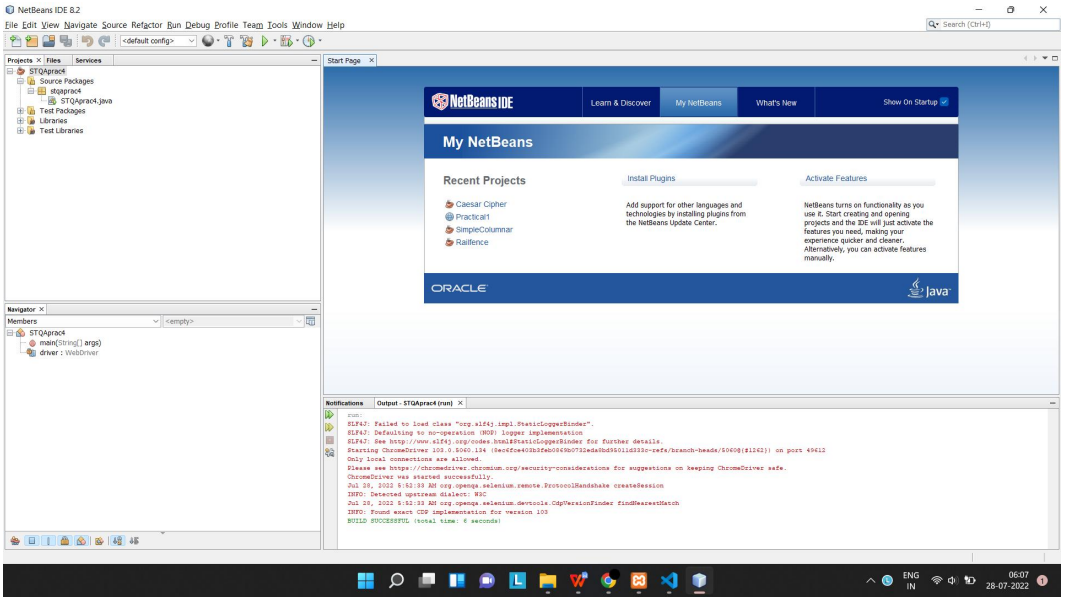
- Visit https://chromedriver.storage.googleapis.com/index.html?path=103.0.5060.134/ for 103 version
- Extract the zip file
- You'll get the application file "chromiumdriver.exe".

5) Download and install XAMPP

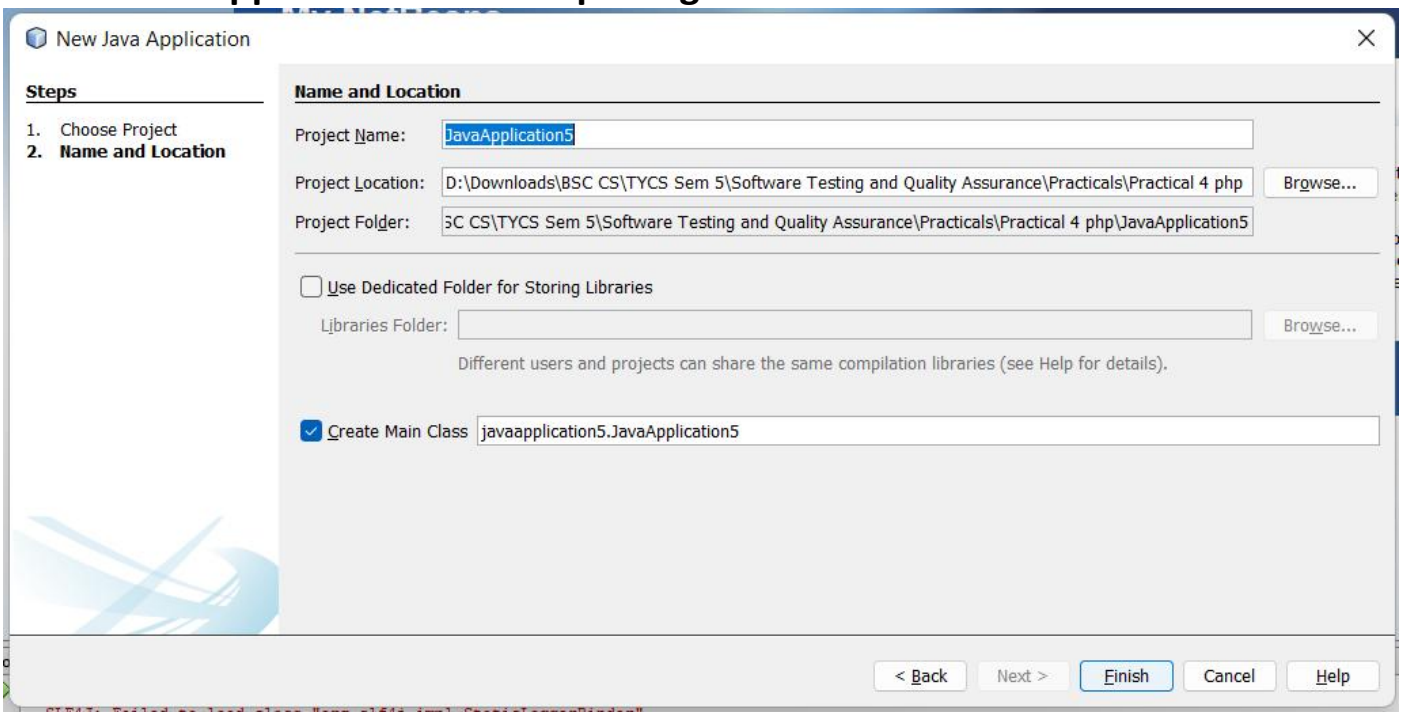
- Visit https://www.apachefriends.org/download.html .

STEPS :

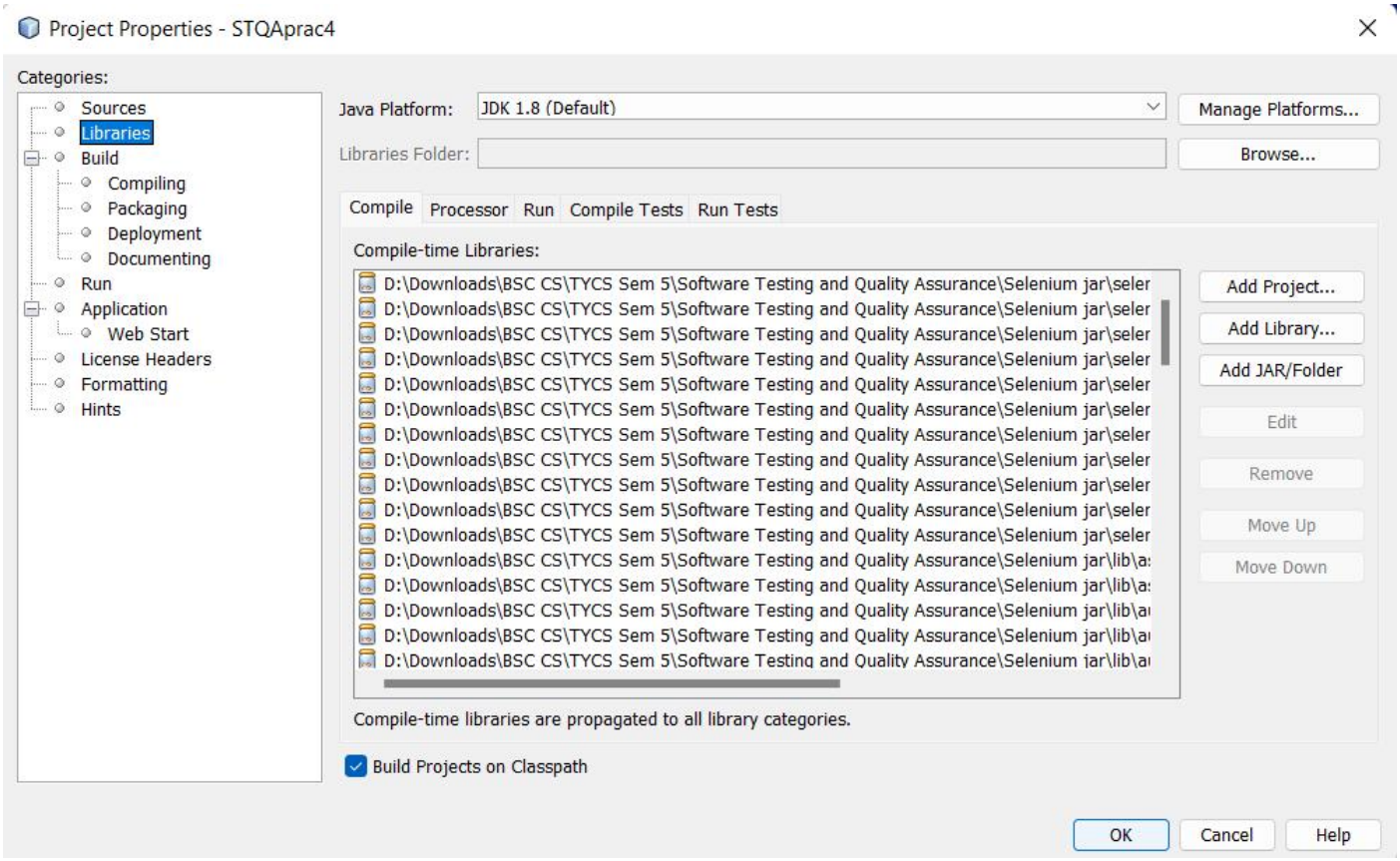
Open Netbeans 8.2



Select **Java Application ->** Give a **package Name** and Click on **Finish Button**



Now Right Click on the **Project**- > Go to **properties** -> Select **Library** -> On the right side select **Add jar/Folder** -> Go to the **location** where **selenium Server and Java Driver jar files** are available and **add** all.



Now we will create an **HTML file** with a **form** and a **Submit** Button.
Code : login.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title></title>
    <meta name="description" content="">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="">
  </head>
  <body>
    <form id="login" action="User.php" method="post" accept-charset="UTF-8">
      Username*:
      <input type="text" name="username" id="username" maxlenght="50">
      Password*:
      <input type="password" name="password" id="password" maxlength="50"><br>
      <input type="submit" name="button" value="Submit">
    </form>
  </body>
```

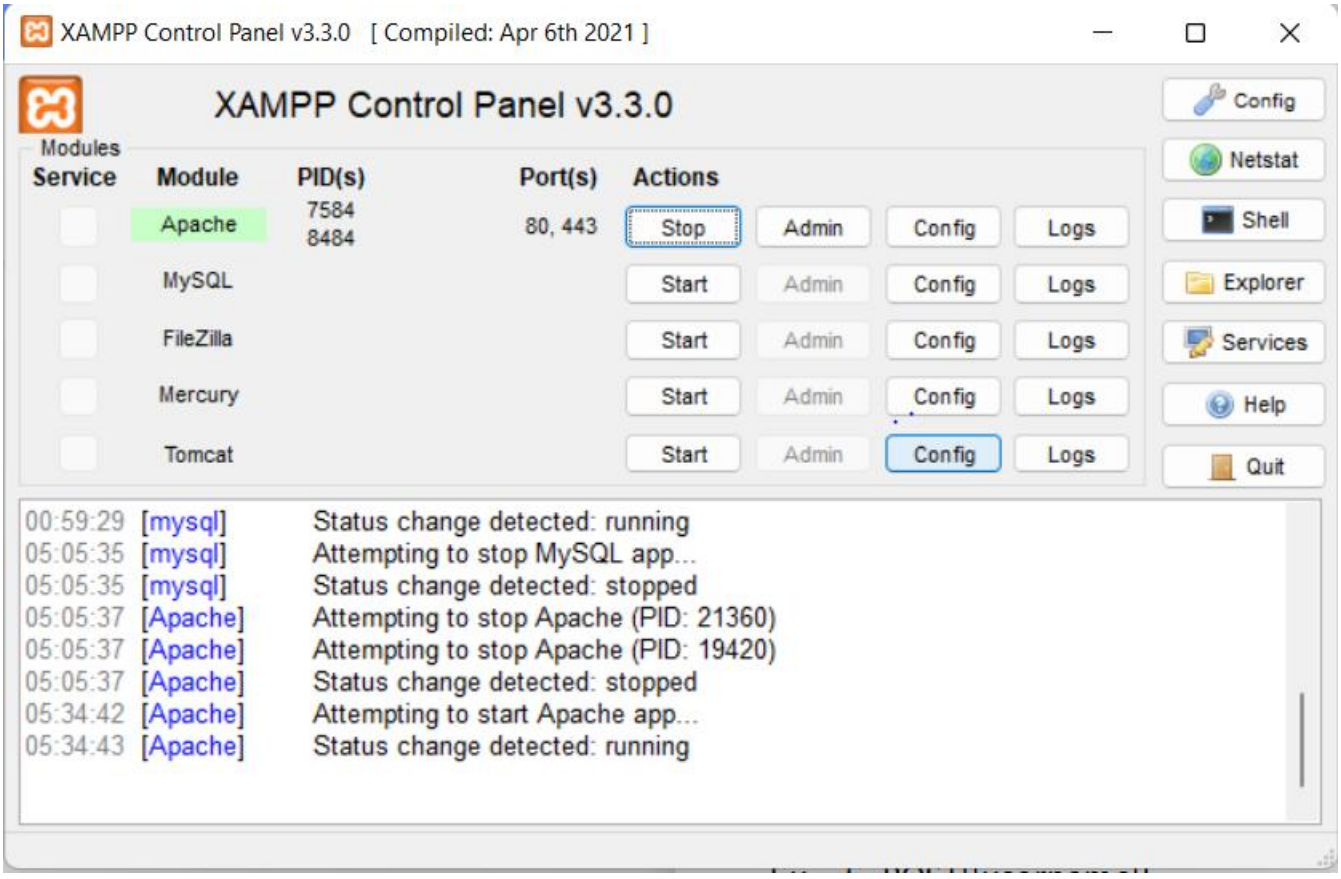
```
</html>
```

Now we will create a **PHP file** with the **form validation**.

Code : User.php

```
<?php
$u = $_POST['username'];
$p = $_POST['password'];
if(($u=="test")&&($p=="test"))
{
   echo "Login Successful";
}
else
{
   echo "Invalid user";
}
```

Now that we have our webpage ready lets open **XAMPP control Panel** and start **localhost services**.(This will allow us to open login.html via localhost)



We will keep both: User.php and login.html inside /htdocs directory where xampp was installed.

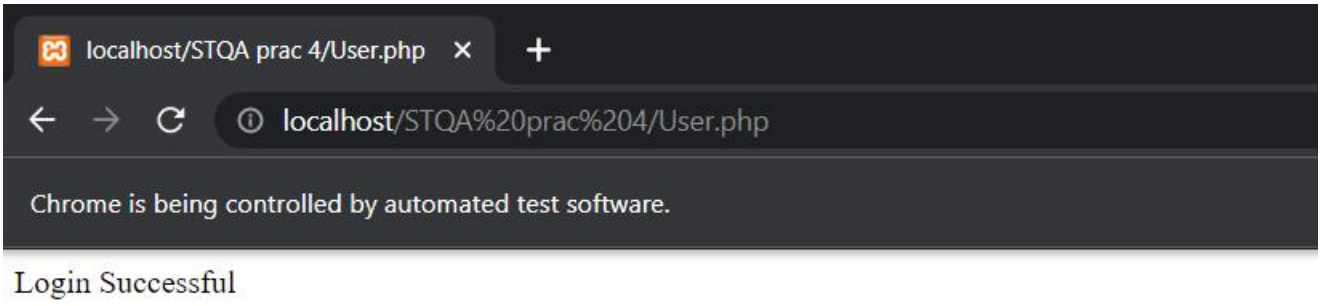
Now we have the following link to access our webpage :

http://localhost/STQA%20prac%204/login.html

Now that our webpage is ready, let's create the **Java file** for **automation**.
Code : STQAprac4.java

```
package stqaprac4;

import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class STQAprac4 {
        private static WebDriver driver = null;
    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver", "D:\\Downloads\\BSC CS\\TYCS Sem
5\\Software Testing and Quality Assurance\\Practicals\\Practical 4
php\\chromedriver.exe");
        driver = new ChromeDriver();
        driver.get("http://localhost/STQA%20prac%204/login.html");
        driver.findElement(By.name("username")).sendKeys("test");
        System.out.println("Username Entered");
        driver.findElement(By.name("password")).sendKeys("test");
        System.out.println("Password Entered");
        driver.findElement(By.name("button")).click();
        System.out.println("Clicked on Submit");
        //driver.quit();
    }
}
```

Now let's run our Java file :-

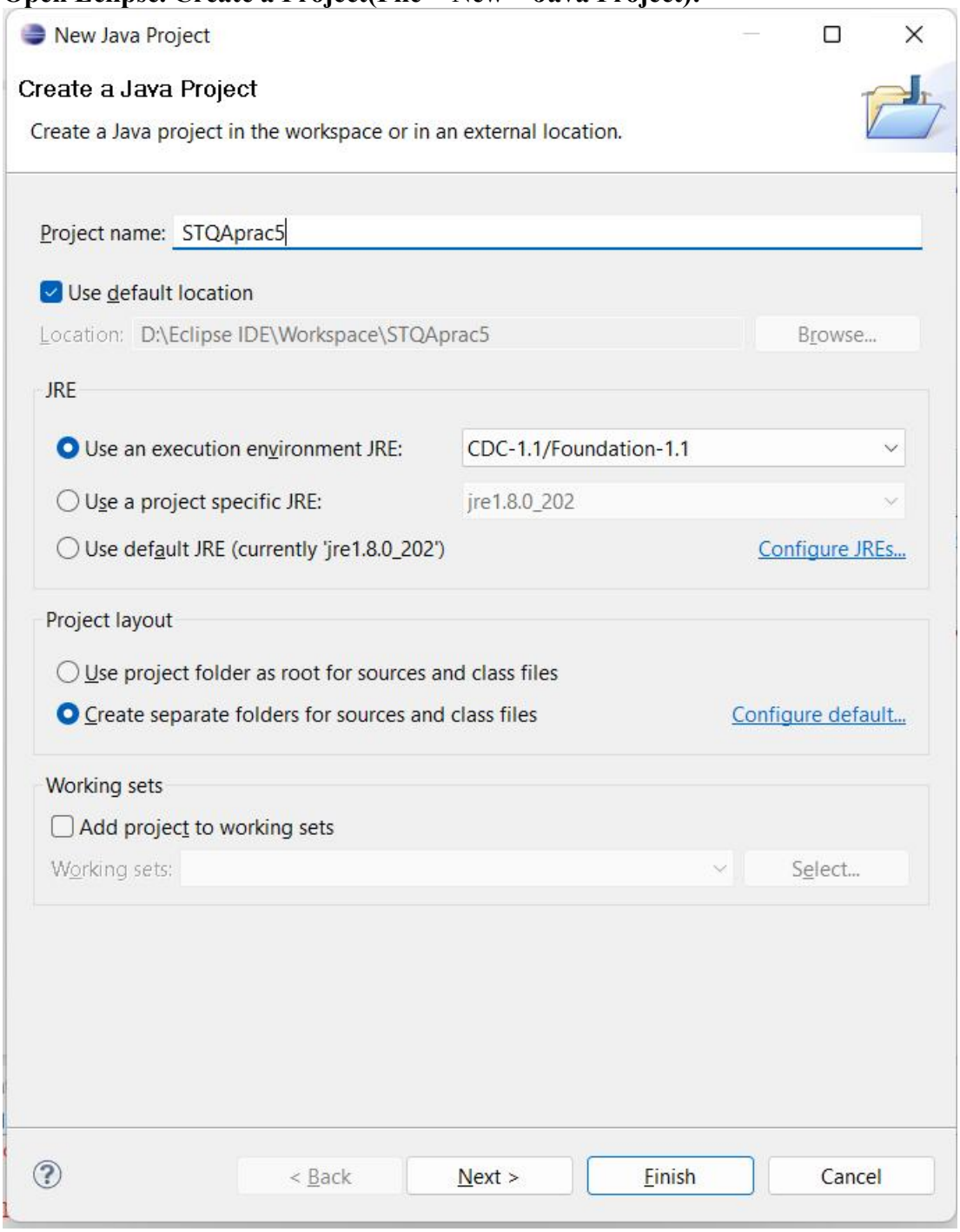


Practical 5

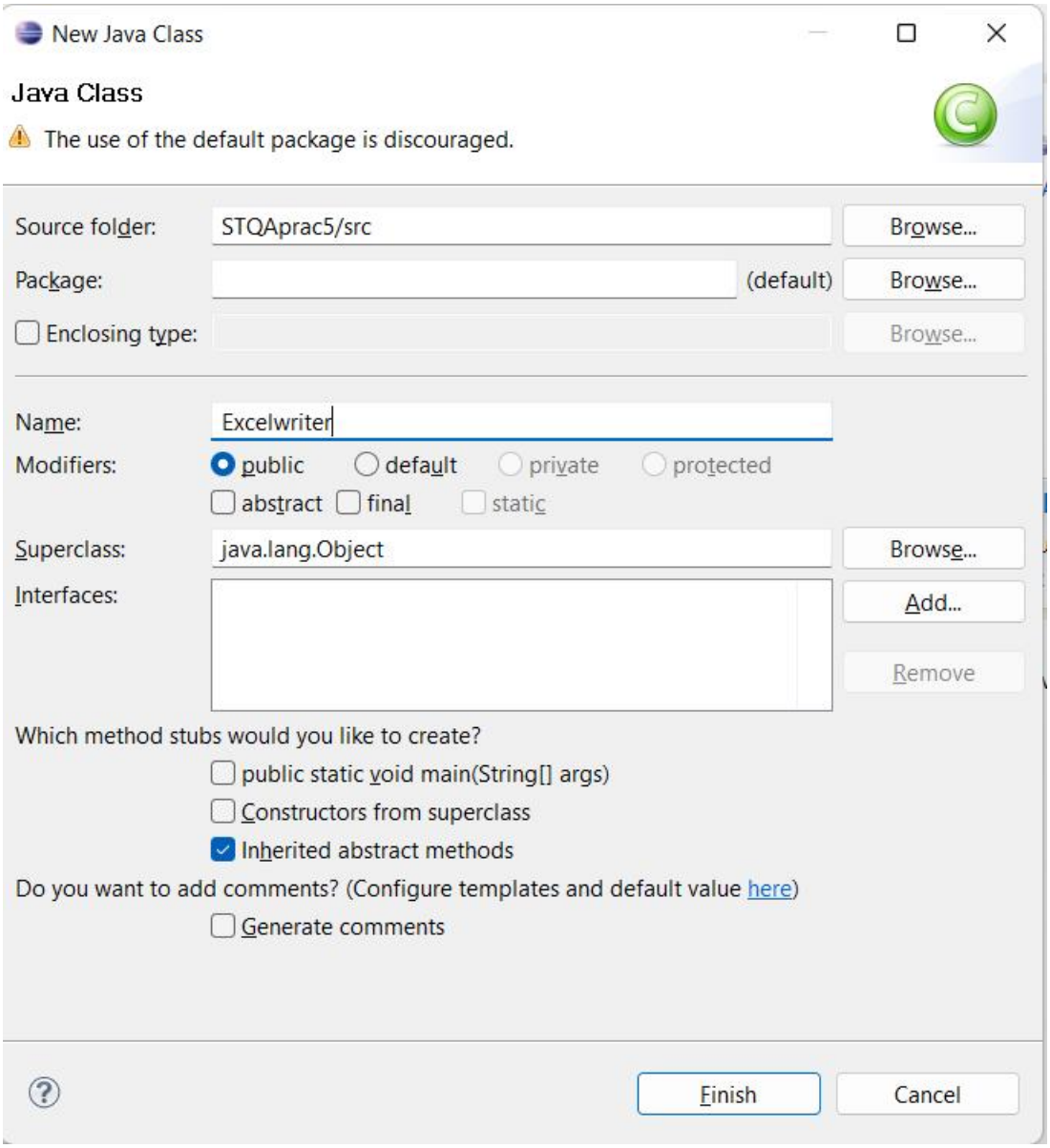
Aim : Write and test a program to update 10 student records into table into Excel file

Download "JXL.JAR":
• Visit http://www.java2s.com/Code/Jar/j/Downloadjxl26jar.htm
• Download this file: "**jxl/jxl-2.6.jar.zip(603 k)**" and **extract** it.(you'll get the .jar file)

Open Eclipse. Create a Project(File > New > Java Project):

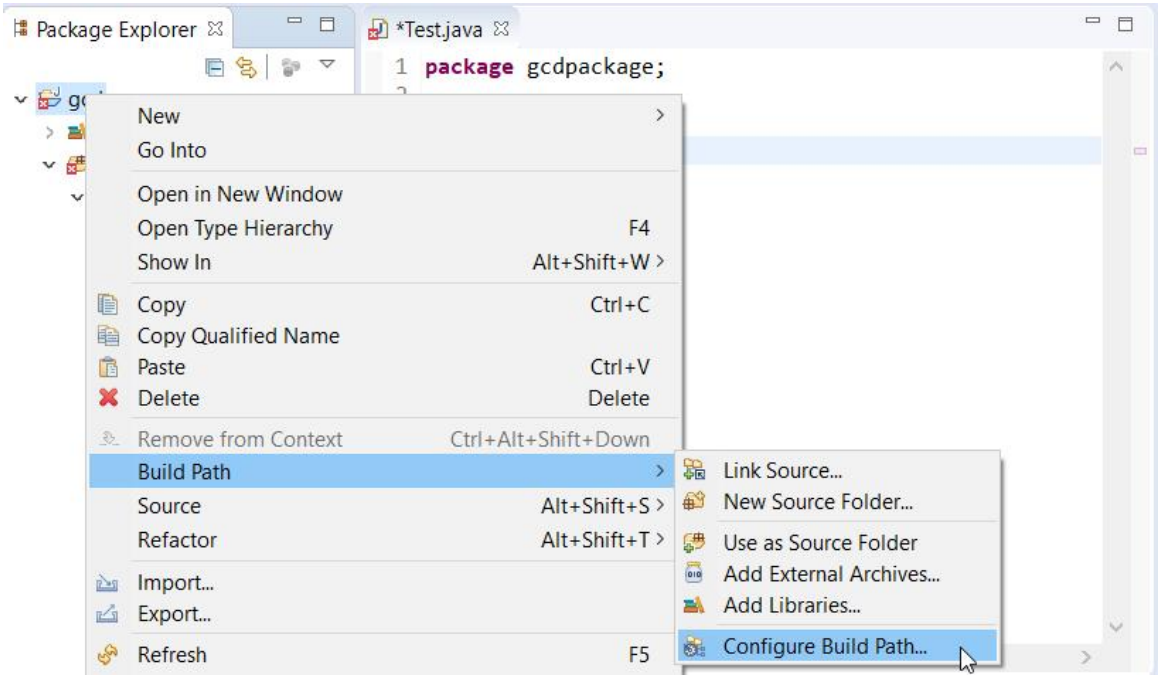


Create a Class(Project> New> Java Class):

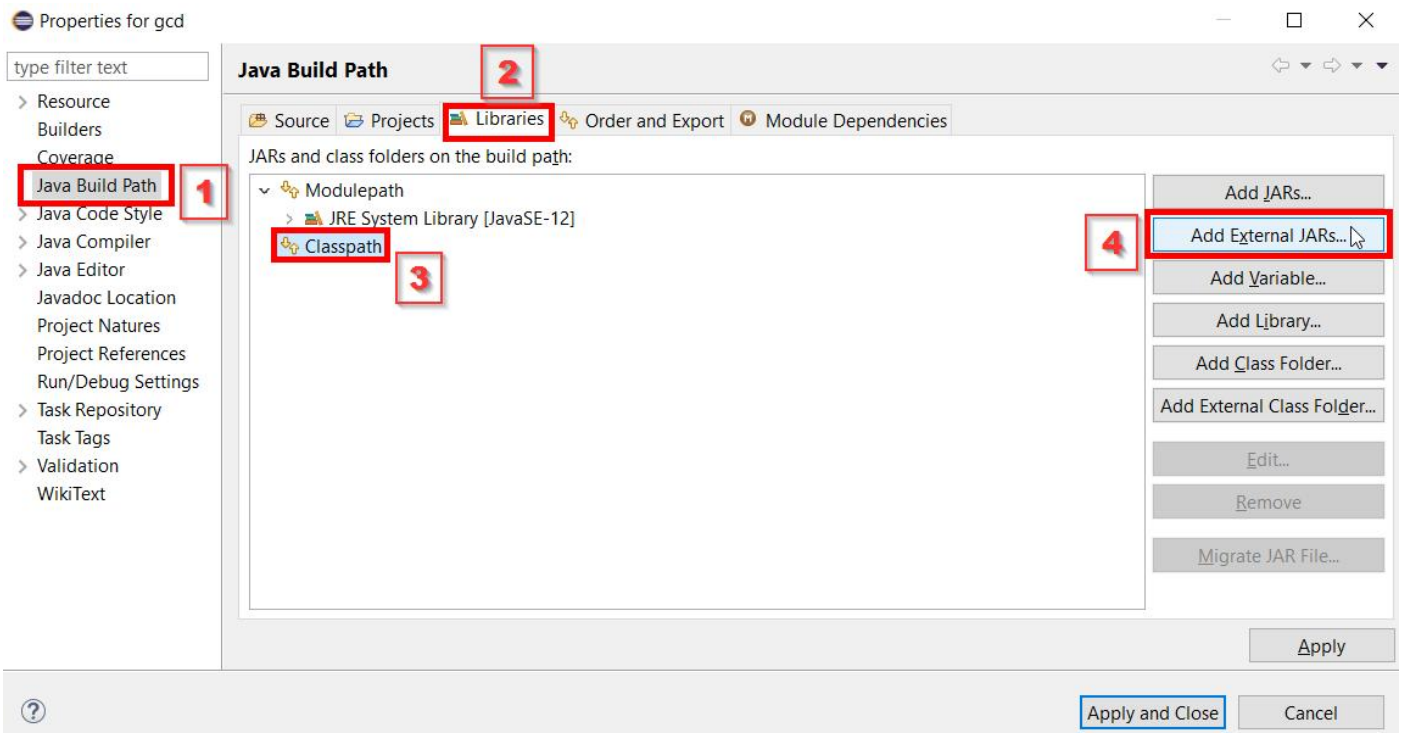


Adding "JXL(JAR file)" in Eclipse IDE:

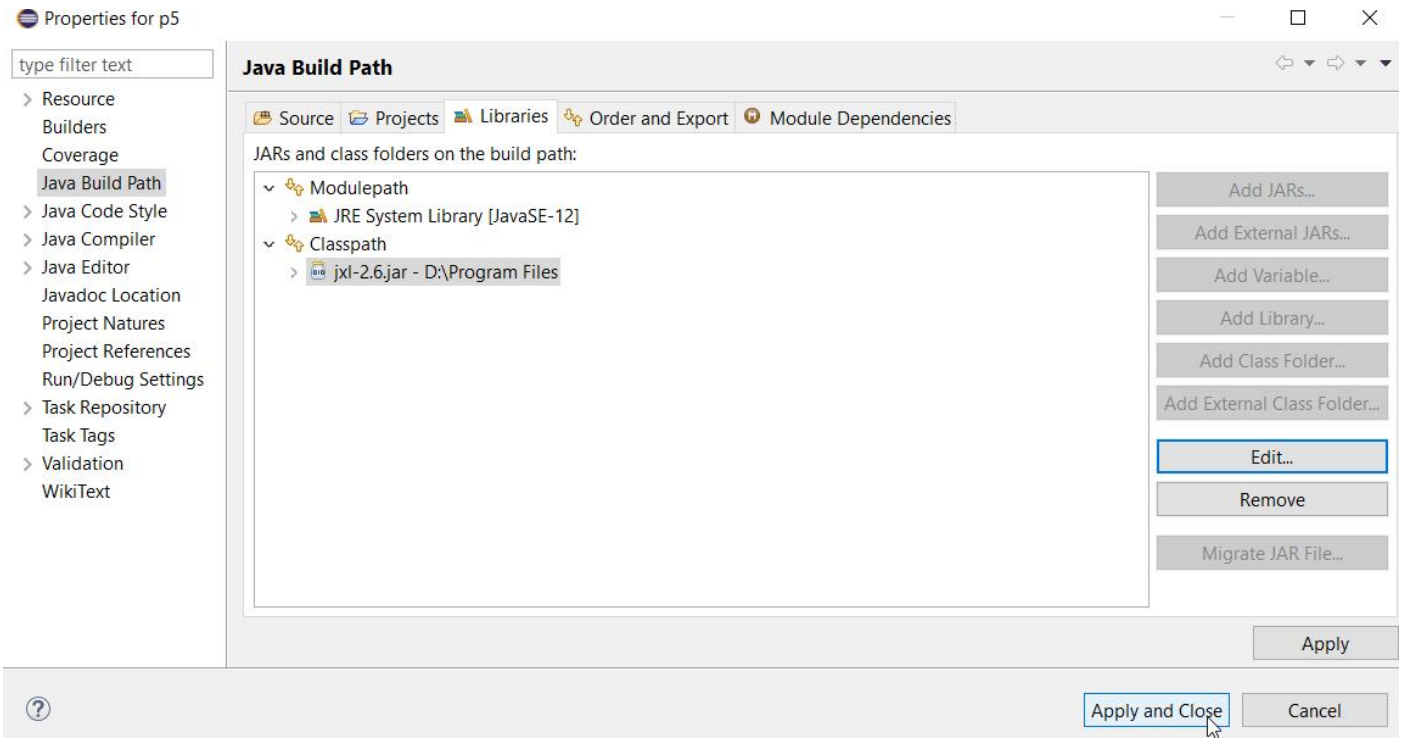
- **right-click on Project Name > Build Path > Configure Build Path…**



now go under: Java Build Path > Libraries > Classpath > click Add External JARs…



Browse and add JAR file > click Apply and Close :



Code : STQAprac4.java

```
import jxl.*; //used for WorkbookSettings,Workbook
import jxl.write.*; //used for WriteException,WritableWorkbook,WritableSheet,Label
import jxl.write.Number; //used for Number
```

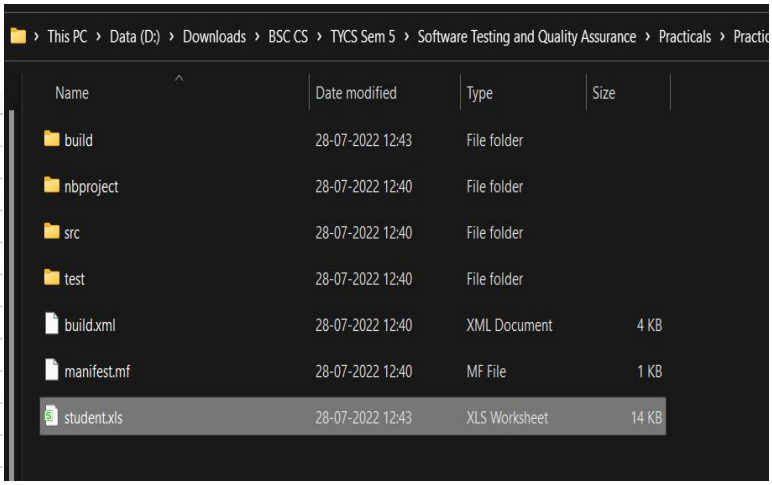
```
import java.io.*; //used for IOException,File
import java.util.Locale; //used for Locale
public class Excelwriter {
public static void main(String[] args) throws IOException,WriteException {
// TODO Auto-generated method stub
int r=0,c=0;
String header[]={"Student
Name","Subject1","Subject2","Subject3","Total"};
String
sname[]={"Carls","James","Paul","Philip","Smith","Thomson","Rhodey","Stark","Gary"
,"AnneMarie"};
int marks[]={50,45,60,55,70,45,67,78,89,90,30};
File file = new File("student.xls");
WorkbookSettings wbSettings = new WorkbookSettings();
wbSettings.setLocale(new Locale("en", "EN"));
WritableWorkbook workbook = Workbook.createWorkbook(file,
wbSettings);
workbook.createSheet("Report", 0);
WritableSheet excelSheet = workbook.getSheet(0);
//creating header row
for(r=0;r<1;r++) {
for(c=0;c<header.length;c++) {
Label l=new Label(c,r,header[c]);
excelSheet.addCell(l);
}
}
//filling name in column1
for(r=1;r<=sname.length;r++) {
for(c=0;c<1;c++) {
Label l=new Label(c,r,sname[r-1]);
excelSheet.addCell(l);
}
}
//filling name in column2,3,4
for(r=1;r<=sname.length;r++) {
for(c=1;c<4;c++) {
Number num = new Number(c, r, marks[r-1]);
excelSheet.addCell(num);
}
}
//filling name in total
for(r=1;r<=sname.length;r++) {
for(c=4;c<5;c++) {
int total=marks[r-1]+marks[r-1]+marks[r-1];
```

```
Number num = new Number(c, r, total);
excelSheet.addCell(num);
}
}
workbook.write();
workbook.close();
System.out.println("Excel File Created!!!!!");
}
}
```

Run the file from Eclipse IDE:

• **OUTPUT**:

	A	B	C	D	E
1	Student Na	Subject1	Subject2	Subject3	Total
2	Carls	50	50	50	150
3	James	45	45	45	135
4	Paul	60	60	60	180
5	Philip	55	55	55	165
6	Smith	70	70	70	210
7	Thomson	45	45	45	135
8	Rhodey	67	67	67	201
9	Stark	78	78	78	234
0	Gary	89	89	89	267
1	AnneMarie	90	90	90	270



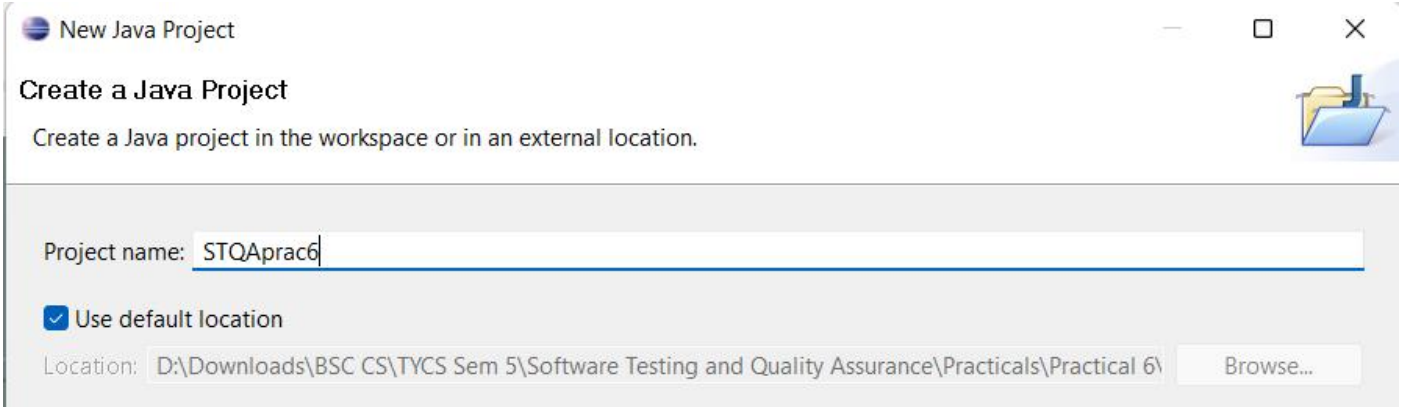
Practical 6

Aim : Write and test a program to select the number of students who have scored more than 60 in any one subject (or all subjects).

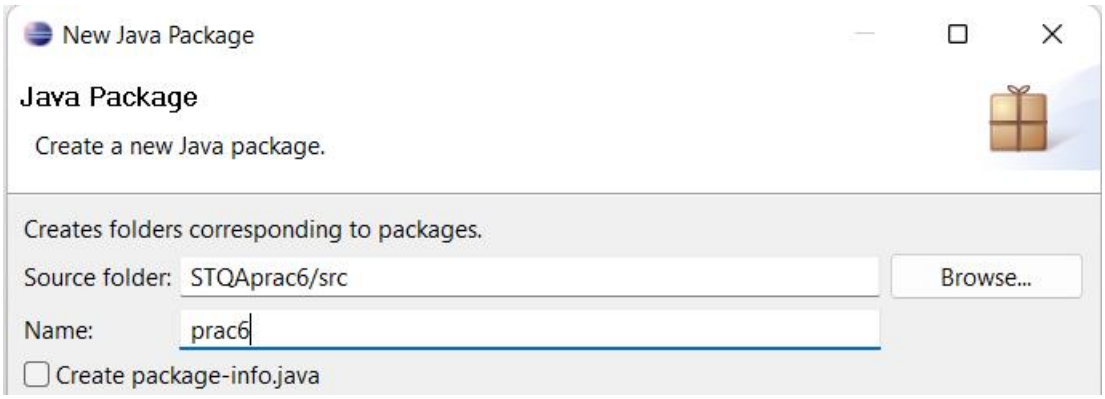
PRE-REQUISITES:

1) Check that you have Eclipse IDE.
2) Check that you have JXL.JAR.
3) Check that you've the Excel file("student.xls") that we'll be working on for reading data.

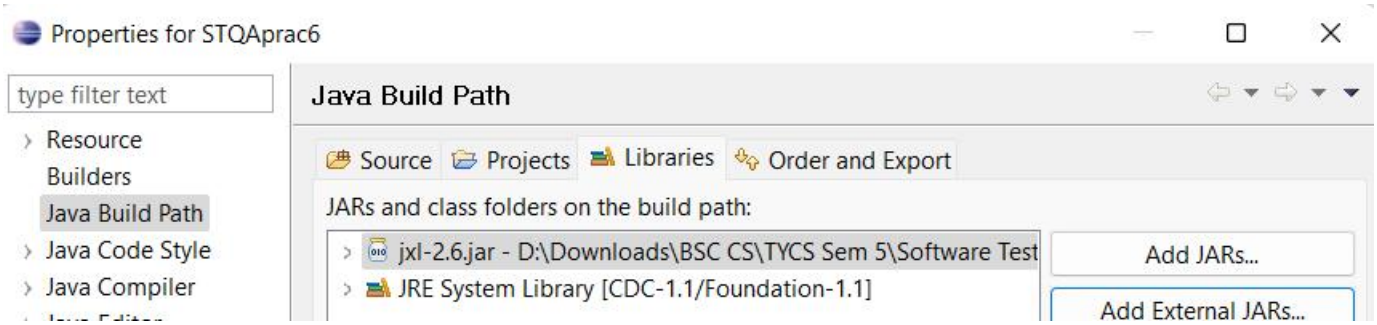
STEPS:
1) Open Eclipse. Select your workspace directory. Click Launch:
2) Create a Project(File > New > Java Project):
3) Name the project as "STQAprac6" > click Finish > click Don't Create module:



4) Close the "Welcome" tab.
5) Create a Package(right-click on Project Name > New > Package > Name it > Finish)
6) Create a Class(right-click on Project Name > New > Class > Name it > Finish):



7) Adding "JXL(JAR file)" in Eclipse IDE:
• right-click on Project Name > Build Path > Configure Build Path…
• now go under: Java Build Path > Libraries > Classpath > click Add External JARs…
• Browse and add JAR file > click Apply and Close :



8) Creating the script in JAVA:
(NOTE that this script will be run by Eclipse IDE)
(In simple words, it's like we are
-ordering Eclipse to run a script or to do a job
-of opening .xls file
-and fetching the marks count>=60 from the cells with the help of jxl.jar
-and to show the result.
-Hence automating the work in a local system(PC)).

---(Excelreader.java)---

```
package prac6;

import java.io.File;
import java.io.IOException;
import jxl.Cell;
import jxl.CellType;
import jxl.Sheet;
import jxl.Workbook;
import jxl.read.biff.BiffException;

public class Excelreader {
	private String inputFile;
	public void setInputFile(String inputFile) {this.inputFile = inputFile;}
	public void read() throws IOException {
		File inputWorkbook = new File(inputFile);
		Workbook w;
		boolean flag=false;
		int count=0;
		try {
			w = Workbook.getWorkbook(inputWorkbook);
			// Get the first sheet
			Sheet sheet = w.getSheet(0);
			// Loop over first 10 column and lines
			for (int j = 0; j < sheet.getRows(); j++)
			{
				for (int i = 0; i < sheet.getColumns()-1; i++)
				{
					Cell cell = sheet.getCell(i, j);
					if (cell.getType() == CellType.NUMBER)
					{
						if(Integer.parseInt(cell.getContents())>=60)
						{
							flag = true;
							if(flag == true)
							{
								count++;
								flag=false;
							}
							break;
						}
					}
```

```
                        }
                    }
            System.out.println("Total number of students who scored more than 60 in one
or more subjects: " +count);
            }
            catch (BiffException e) {e.printStackTrace();}
      }
      public static void main(String[] args) throws IOException {
            Excelreader test = new Excelreader();
            test.setInputFile("D://Downloads//BSC CS//TYCS Sem 5//Software Testing and
Quality Assurance//Practicals//Practical 6//STQAprac6//student.xlsx");
            test.read();
      }
}
```

9) Run the file from Eclipse IDE

Inputfile provded is-

	A	B	C	D	E
1	Student Na	Subject1	Subject2	Subject3	Total
2	Carls	50	50	50	150
3	James	45	45	45	135
4	Paul	60	60	60	180
5	Philip	55	55	55	165
6	Smith	70	70	70	210
7	Thomson	45	45	45	135
8	Rhodey	67	67	67	201
9	Stark	78	78	78	234
10	Gary	89	89	89	267
11	AnneMarie	90	90	90	270

• **OUTPUT:**

```
<terminated> Excelreader [Java Application] C:\Program Files\Java\jre1.8.0_202\bin\javaw.exe (10-Sep-2022 11:36:45 am)
Total number of students who scored more than 60 in one or more subjects: 6
```

10) Finish!

Practical 7

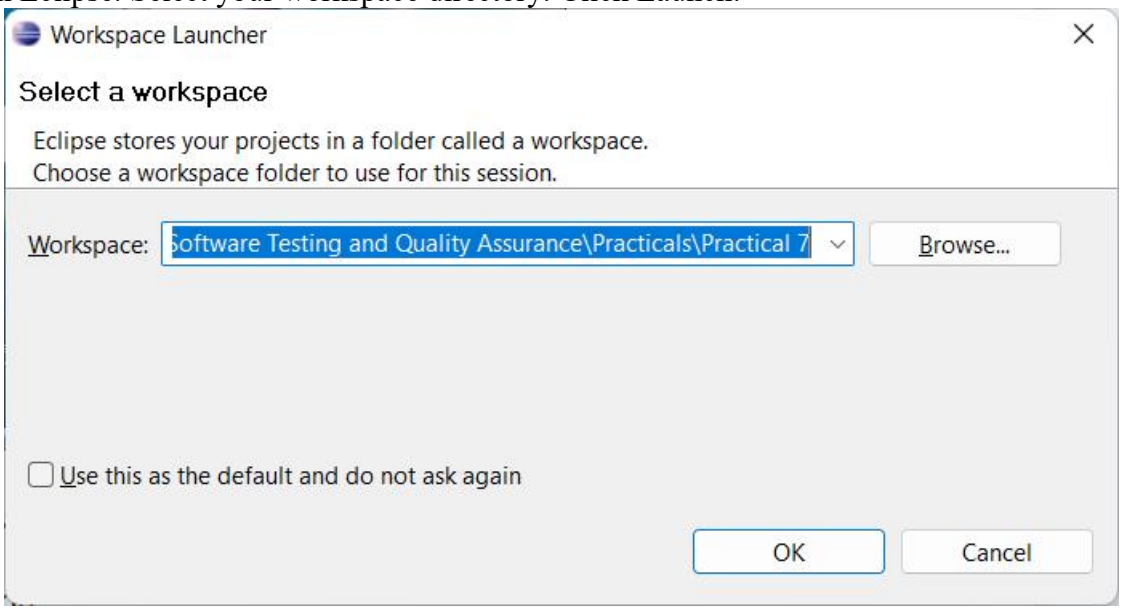
Aim : Write and test a program to provide total number of objects present / available on the page.

PRE-REQUISITES:

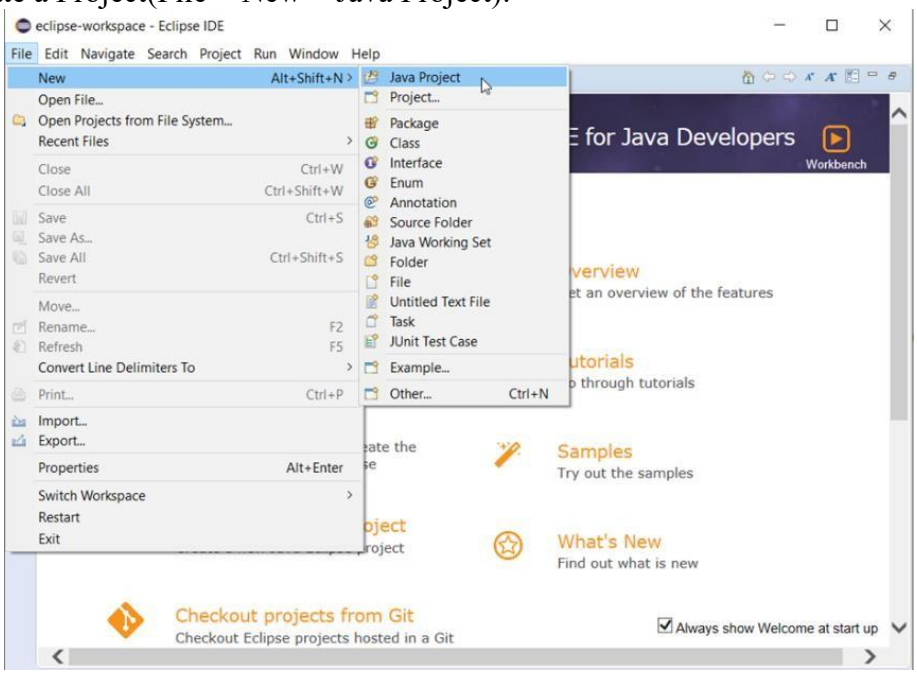
1) Check that you have JDK.
2) Check that you have Eclipse IDE.
3) Check that you have Selenium Server Driver and Client Driver (JAR files).
4) Check that you have Chromium Driver.
5) Check that you have a stable Internet connection.

STEPS:

1) Open Eclipse. Select your workspace directory. Click Launch:

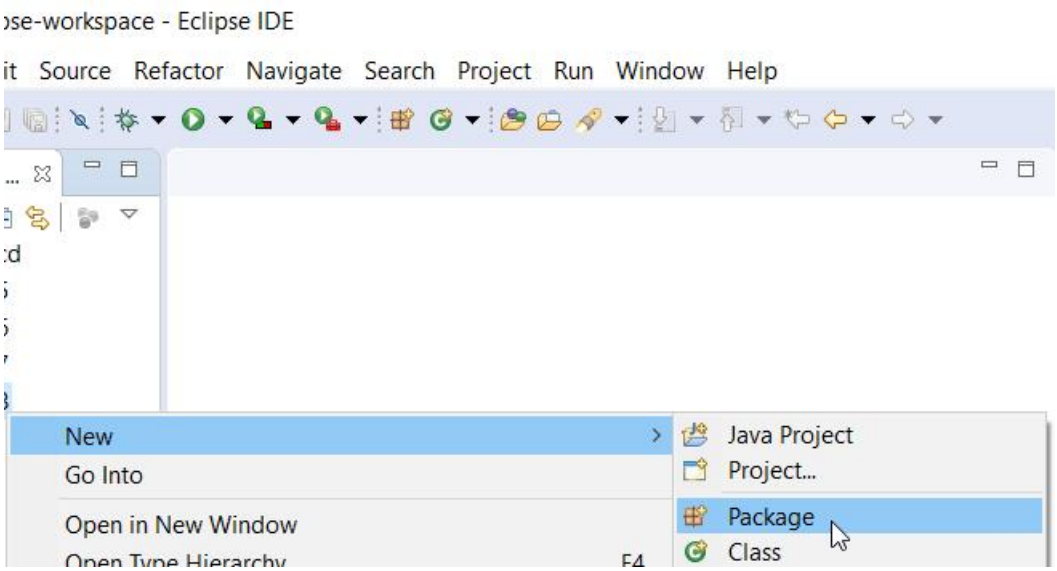


2) Create a Project(File > New > Java Project):

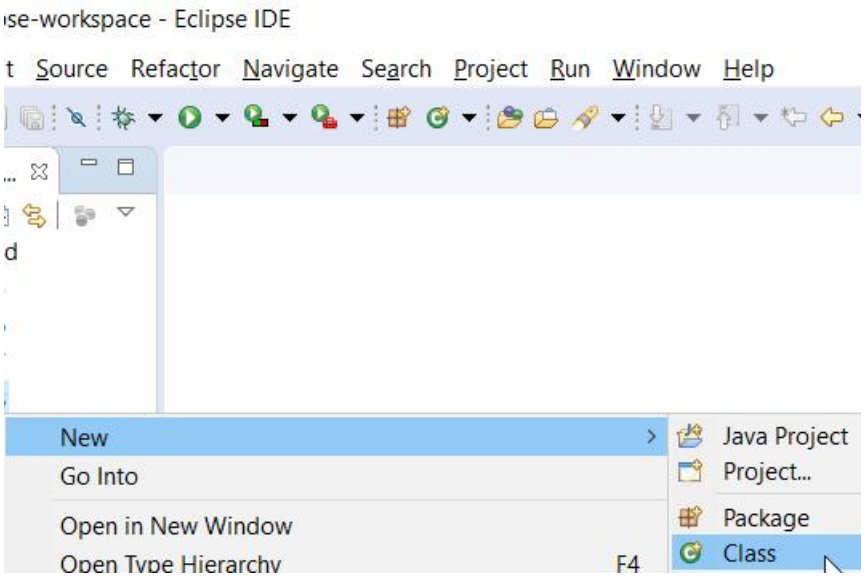


3) Name the project as “STQAprac7” > click Finish > click Don’t Create module
4) Close the “Welcome” tab.

5) Create a Package(right-click on Project Name > New > Package > Name it > Finish):

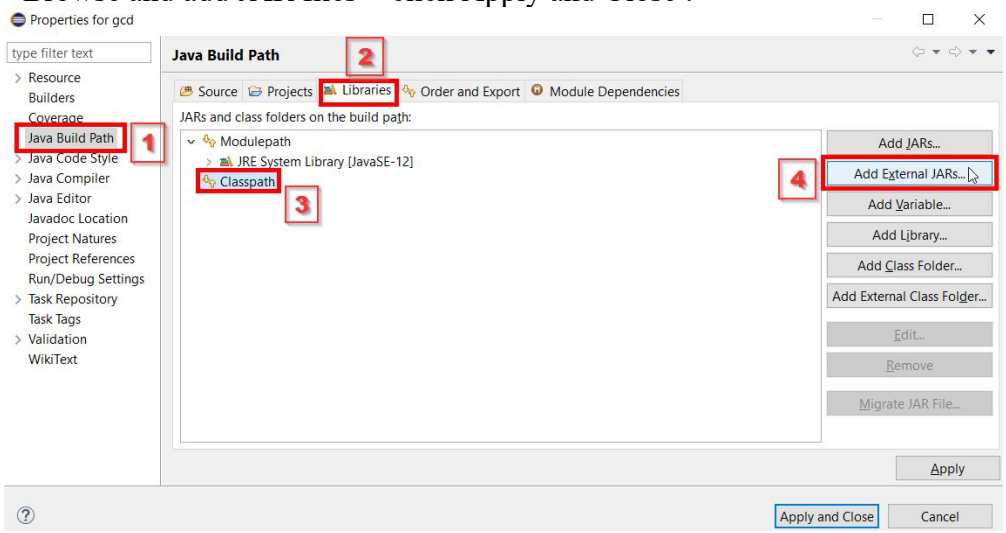


6) Create a Class(right-click on Project Name > New > Class > Name it > Finish):



7) Adding “Selenium Server Driver and Client Driver(JAR files)” in Eclipse IDE:

- right-click on Project Name > Build Path > Configure Build Path…
- now go under: Java Build Path > Libraries > Classpath > click Add External JARs…
- Browse and add JAR files > click Apply and Close :



8) Creating the script in JAVA:

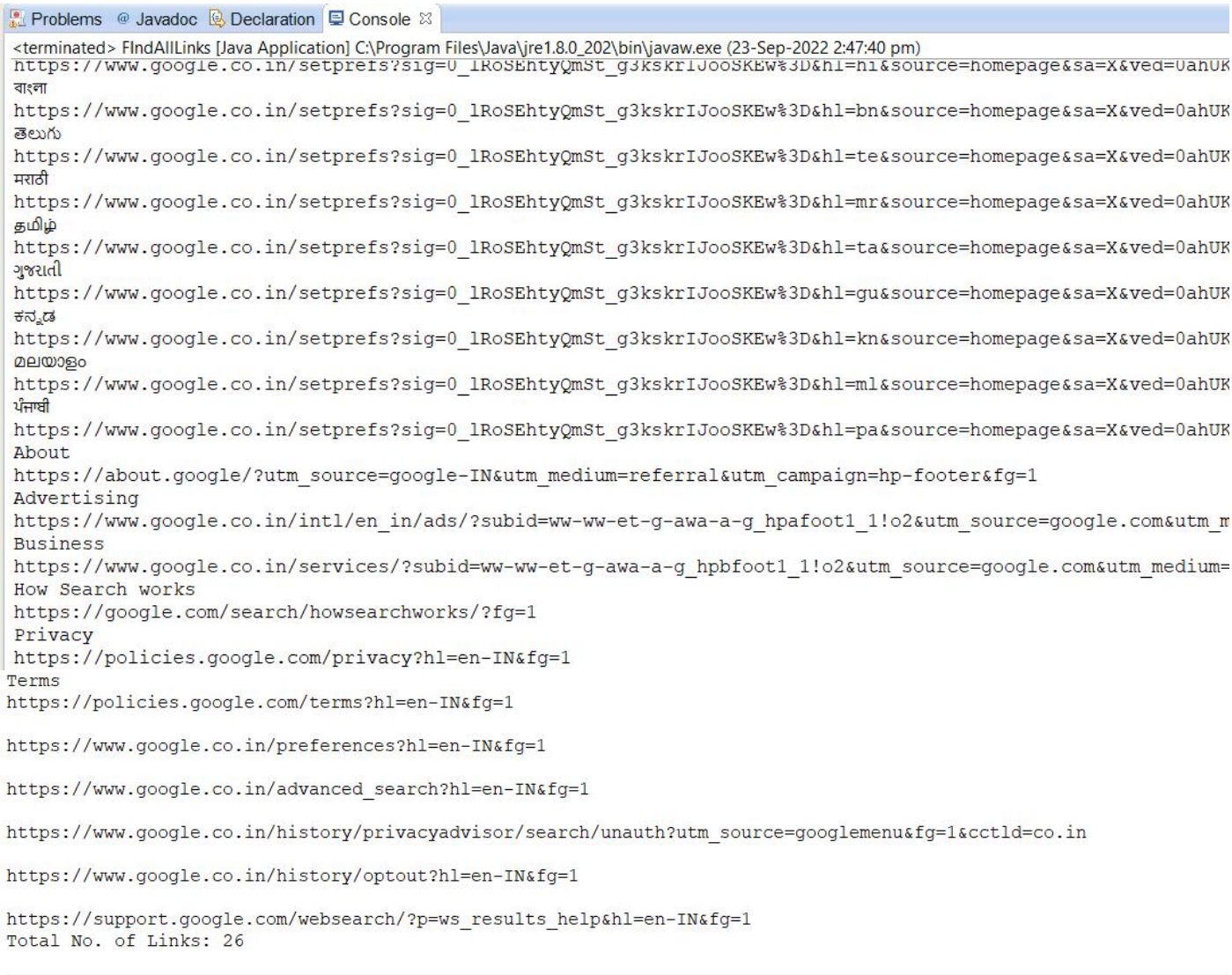
(NOTE that this script will be run by Eclipse IDE)
(In simple words, it's like we are
-ordering Eclipse to run a script or to do a job
-of opening the browser, visiting the URL
-and finding the <a> tag WebElements with the help of Selenium Drivers
-and to show the result.
-Hence automating the work in browser)

- Now we'll put the path of "ChromiumDriver" in a String driverPath

---(FindAllLinks.java)---

```
package Prac7STQA;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
//import org.openqa.selenium.chrome.*;
//import org.openqa.selenium.chrome.ChromeOptions;
//import org.openqa.selenium.chrome.ChromeProfile;
//import org.openqa.selenium.chrome.internal.ProfilesIni;

public class FIndAllLinks {
	static String driverPath = "D:\\Downloads\\BSC CS\\TYCS Sem 5\\Software Testing and Quality
Assurance\\Practicals\\Practical 4 php\\chromedriver.exe";
	public static void main(String[] args) {
	System.setProperty("webdriver.chrome.driver",driverPath);
	WebDriver driver = new ChromeDriver();
	String appUrl ="https://www.google.co.in/";
	driver.get(appUrl);
	java.util.List<WebElement> links = driver.findElements(By.tagName("a")); //a is anchor tag
	for (int i = 1; i<links.size(); i=i+1)
	{
	System.out.println(links.get(i).getText());
	System.out.println(links.get(i).getAttribute("href"));
	}
	System.out.println("Total No. of Links: "+links.size());
	driver.quit();
	}
}
```

9) Run the file from Eclipse IDE: OUTPUT:


Problems
Javadoc
Declaration
Console
<terminated> FIndAllLinks [Java Application] C:\Program Files\Java\jre1.8.0_202\bin\javaw.exe (23-Sep-2022 2:47:40 pm)
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=hi&source=homepage&sa=X&ved=0ahUK
বাংলা
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=bn&source=homepage&sa=X&ved=0ahUK
తెలుగు
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=te&source=homepage&sa=X&ved=0ahUK
मराठी
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=mr&source=homepage&sa=X&ved=0ahUK
தமிழ்
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=ta&source=homepage&sa=X&ved=0ahUK
ગુજરાતી
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=gu&source=homepage&sa=X&ved=0ahUK
ಕನ್ನಡ
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=kn&source=homepage&sa=X&ved=0ahUK
മലയാളം
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=ml&source=homepage&sa=X&ved=0ahUK
ਪੰਜਾਬੀ
https://www.google.co.in/setprefs?sig=0_lRoSEhtyQmSt_g3kskrIJooSKEw%3D&hl=pa&source=homepage&sa=X&ved=0ahUK
About
https://about.google/?utm_source=google-IN&utm_medium=referral&utm_campaign=hp-footer&fg=1
Advertising
https://www.google.co.in/intl/en_in/ads/?subid=ww-ww-et-g-awa-a-g_hpafoot1_1!o2&utm_source=google.com&utm_m
Business
https://www.google.co.in/services/?subid=ww-ww-et-g-awa-a-g_hpbfoot1_1!o2&utm_source=google.com&utm_medium=
How Search works
https://google.com/search/howsearchworks/?fg=1
Privacy
https://policies.google.com/privacy?hl=en-IN&fg=1
Terms
https://policies.google.com/terms?hl=en-IN&fg=1
https://www.google.co.in/preferences?hl=en-IN&fg=1
https://www.google.co.in/advanced_search?hl=en-IN&fg=1
https://www.google.co.in/history/privacyadvisor/search/unauth?utm_source=googlemenu&fg=1&cctld=co.in
https://www.google.co.in/history/optout?hl=en-IN&fg=1
https://support.google.com/websearch/?p=ws_results_help&hl=en-IN&fg=1
Total No. of Links: 26

Practical 8

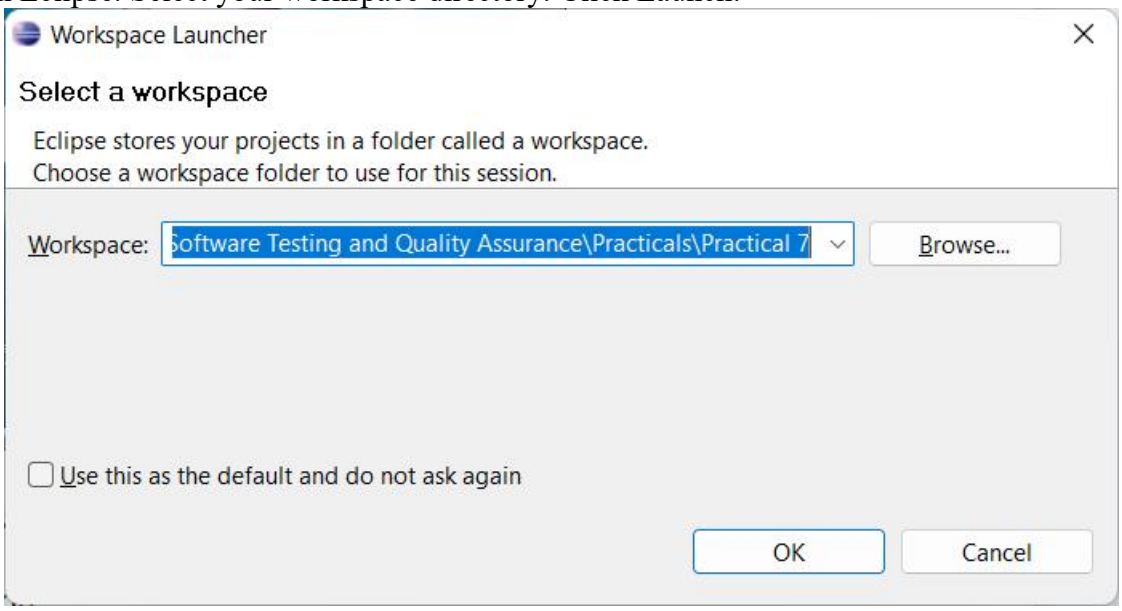
Aim : Write and test a program to get the number of items in a list / combo box.

PRE-REQUISITES:

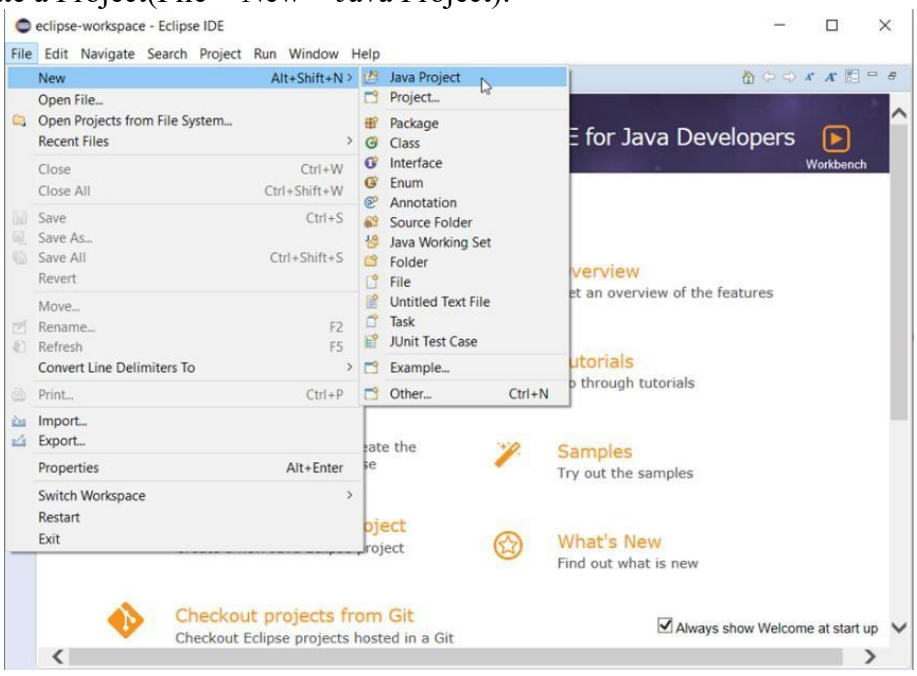
1) Check that you have JDK.
2) Check that you have Eclipse IDE.
3) Check that you have Selenium Server Driver and Client Driver (JAR files).
4) Check that you have Chromium Driver.
5) Check that you have a stable Internet connection.

STEPS:

1) Open Eclipse. Select your workspace directory. Click Launch:

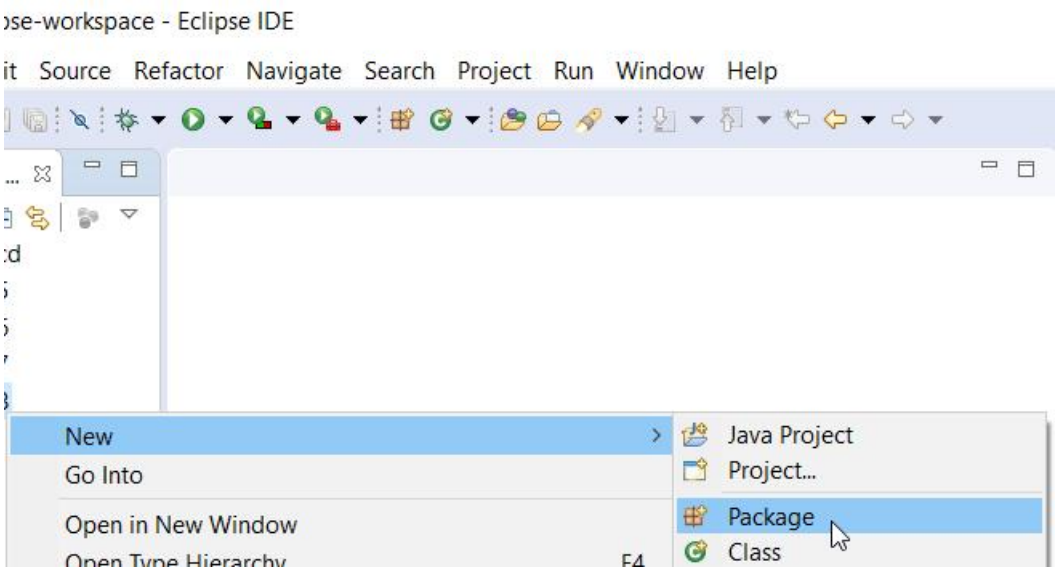


2) Create a Project(File > New > Java Project):

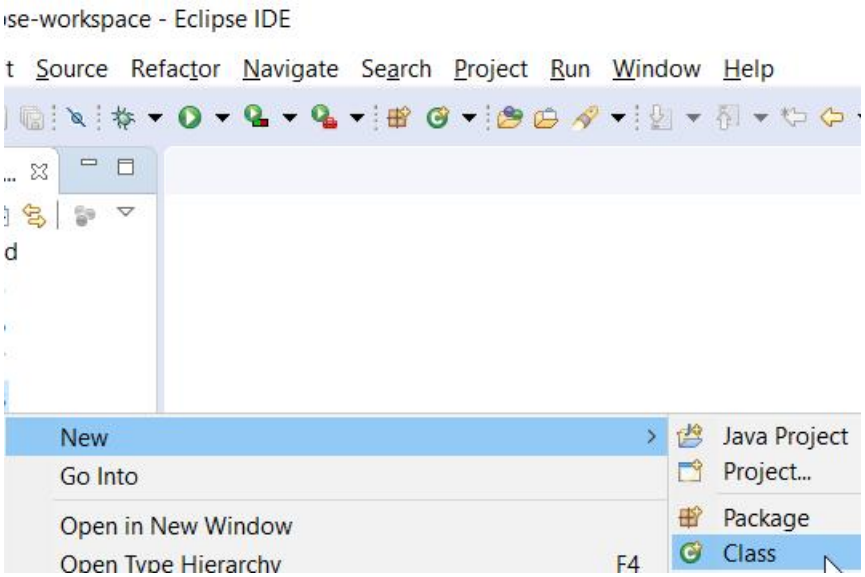


3) Name the project as “STQAprac7” > click Finish > click Don’t Create module
4) Close the “Welcome” tab.

5) Create a Package(right-click on Project Name > New > Package > Name it > Finish):

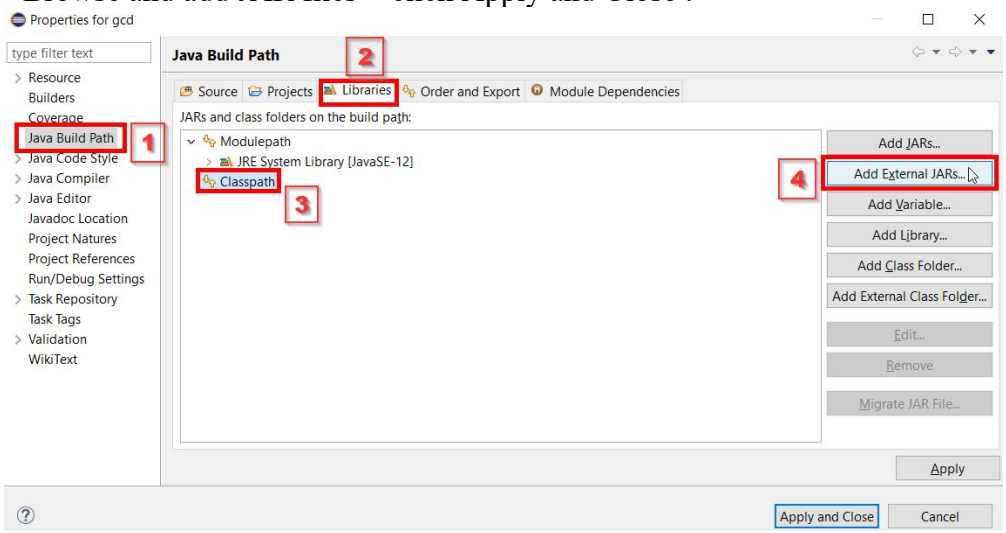


9) Create a Class(right-click on Project Name > New > Class > Name it > Finish):



7) Adding "Selenium Server Driver and Client Driver(JAR files)" in Eclipse IDE:

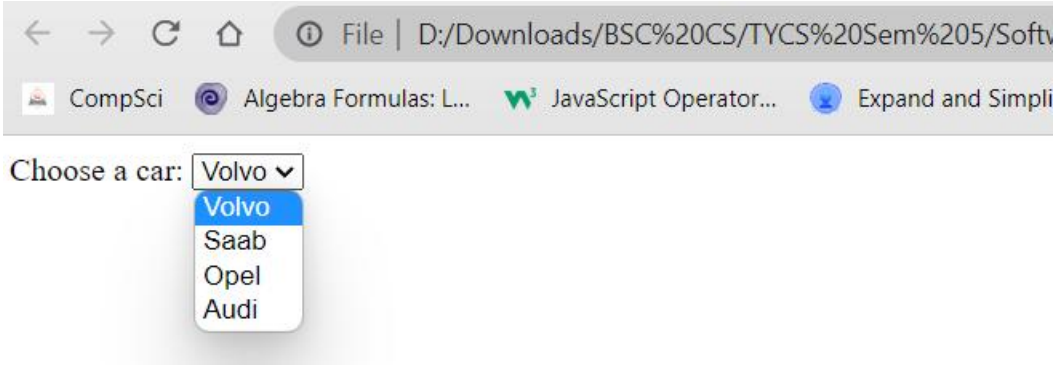
- right-click on Project Name > Build Path > Configure Build Path…
- now go under: Java Build Path > Libraries > Classpath > click Add External JARs…
- Browse and add JAR files > click Apply and Close :



8) Create HTML file in a notepad > Save it > Open it in browser > Copy the URL:
(NOTE THAT: as this is our LOCAL FILE, this file will be opened thru STATIC URL in script)

---(combobox.html)---

```
<!DOCTYPE html>
<html>
<body>
<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select name="cars" id="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="opel">Opel</option>
    <option value="audi">Audi</option>
  </select>
</form>
</body>
</html>
```



9) Creating the script in JAVA:

(NOTE that this script will be run by Eclipse IDE)
(In simple words, it's like we are
-ordering Eclipse to run a script or to do a job
-of opening the browser, visiting the URL
-and finding the WebElement By "ID" with the help of "Select" class and Selenium Drivers
-and to show the result.
-Hence automating the work in browser)

- Now we'll put the path of our LOCAL FILE(combobox.html) in a string
- Also put the path of "ChromiumDriver" in a String driverPath

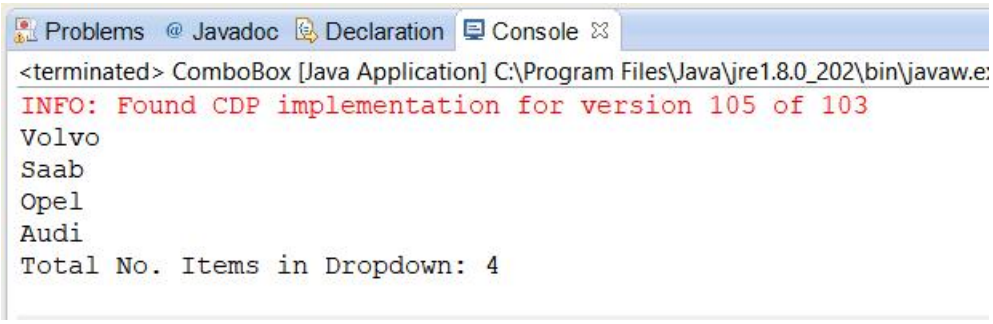
---(ComboBox.java)---

```
package prac8STQA;

import java.util.List;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.*;
import org.openqa.selenium.support.ui.Select;

public class ComboBox {
```

```
static String driverPath = "D:\\Downloads\\BSC CS\\TYCS Sem 5\\Software Testing and
Quality Assurance\\Practicals\\Practical 4 php\\chromedriver.exe";
       public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver",driverPath);
      WebDriver driver = new ChromeDriver();
      String appUrl = "file:///D:/Downloads/BSC CS/TYCS Sem 5/Software Testing and
Quality Assurance/Practicals/Practical 8/combobox.html"; //STATIC URL(LOCAL FILE)
      driver.get(appUrl);
      Select oSelect = new Select(driver.findElement(By.id("cars")));
      List<WebElement> oSize = oSelect.getOptions();
      int iListSize = oSize.size();
      for(int i =0; i < iListSize ; i++)
      {
      // Storing the value of the option
      String sValue = oSelect.getOptions().get(i).getText();
      // Printing the stored value
      System.out.println(sValue);
      }
      System.out.println("Total No. Items in Dropdown: "+iListSize);
      driver.quit();
      }
}
```

10) Run the file from Eclipse IDE:

- **OUTPUT**:



Practical 9

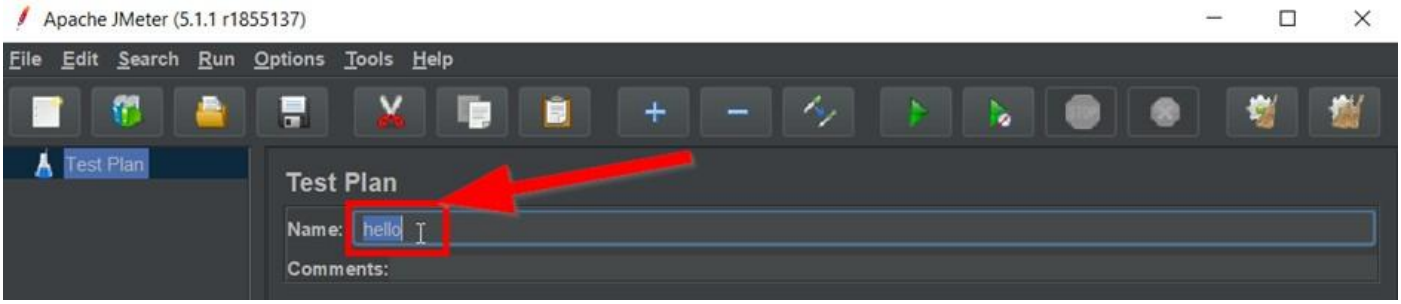
Aim : Load Testing using JMeter.

PRE-REQUISITES:

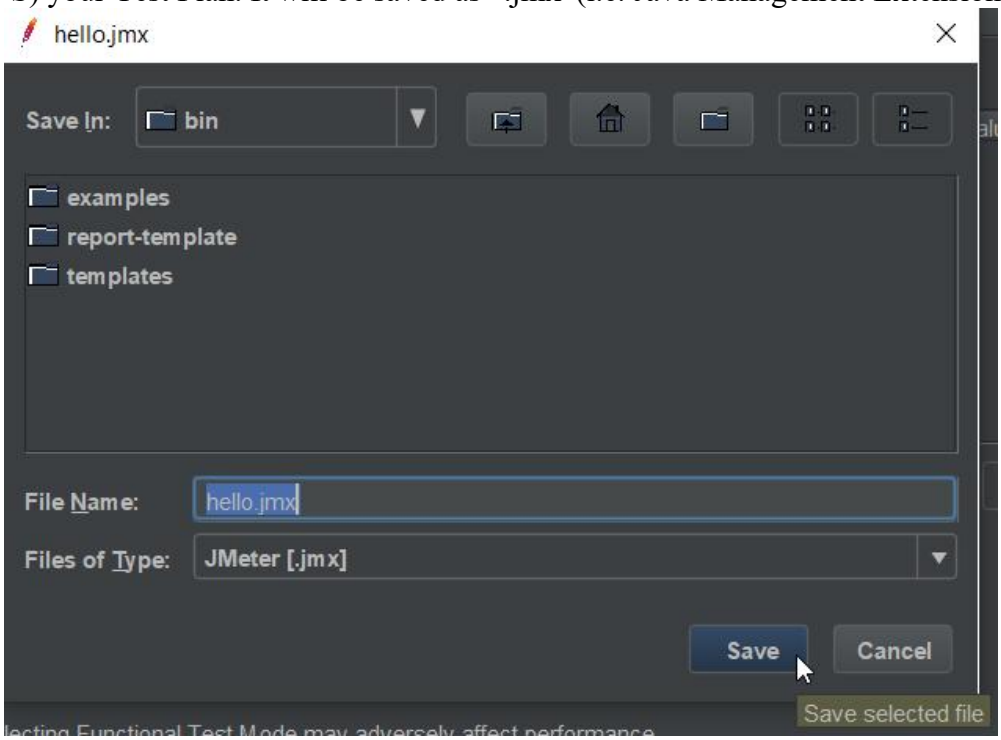
1) To Download "JDK":
   - Visit https://www.oracle.com/technetwork/java/javase/downloads/jdk12-downloads-5295953.html
   - Download this file "jdk-12.0.2_windows-x64_bin.exe" and install it.
2) To Download "Apache JMeter":
   - Visit https://jmeter.apache.org/download_jmeter.cgi
   - Under section "Apache JMeter 5.1.1 (Requires Java 8+)"
   - Under sub-section "Binaries", download "apache-jmeter-5.1.1.zip" file.

• **Installation:**
➢ Extract the "apache-jmeter-5.1.1.zip" file.
➢ Navigate to: apache-jmeter-5.1.1 > bin > the "ApacheJMeter.jar" file is your working space.

STEPS:
1) Open "ApacheJMeter.jar" file. Name your Test Plan as "hello":

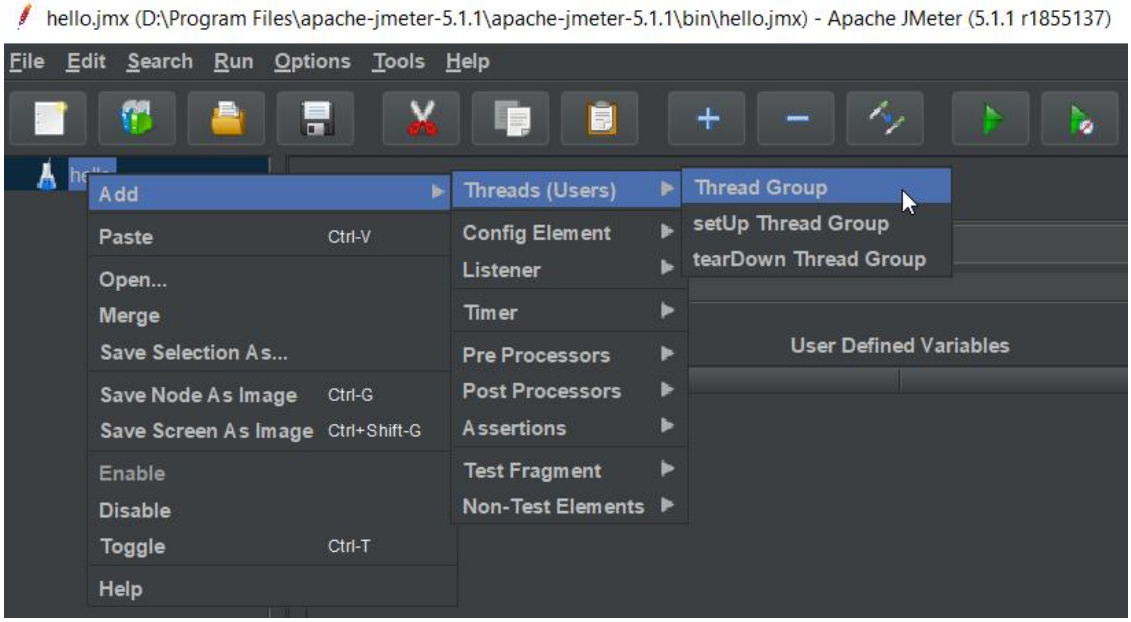


2) Save(CTRL+S) your Test Plan. It will be saved as ".jmx"(i.e. Java Management Extensions):

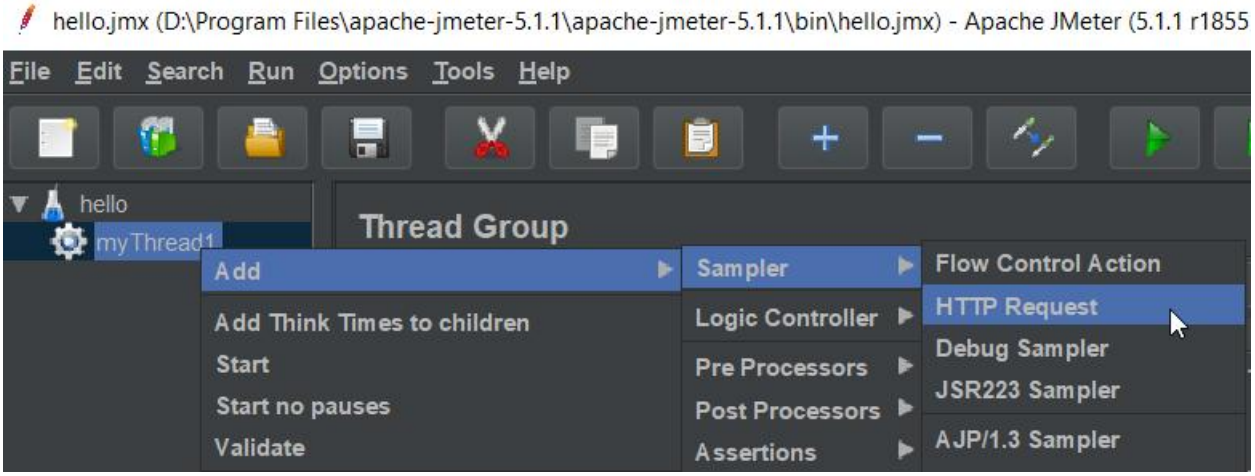


3) Create a Thread Group

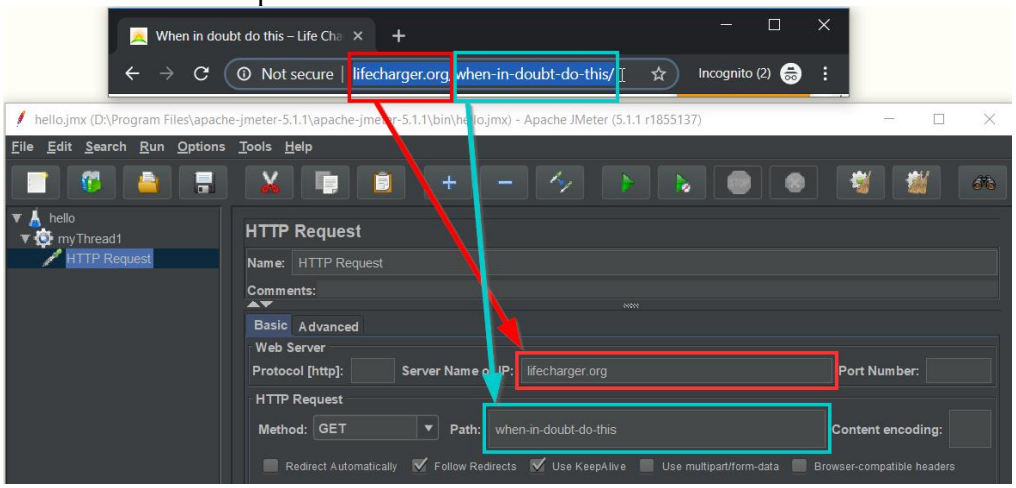
(right-click over “hello(Test Plan)” > Add > Thread(Users) > Thread Group >
Name your Thread Group as “myThread1” > Save(CTRL+S)):



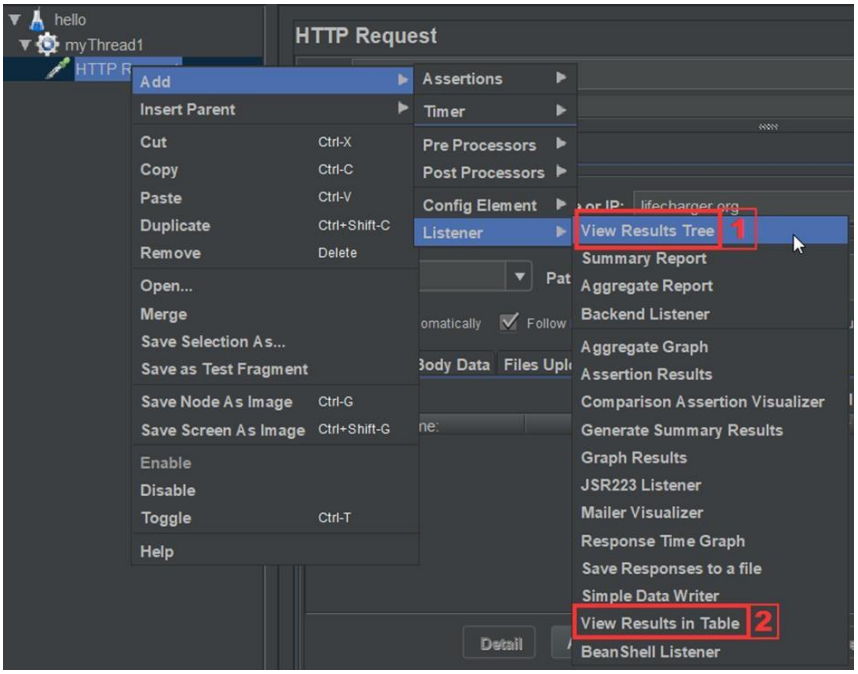
4) Under Thread Group(myThread1), add a “Sampler”, namely HTTP Request:
(right-click over “myThread1(Thread Group)” > Add > Sampler > HTTP Request):



5) Now visit any website page with a next path. Then set website name as “Server Name or IP”
& next path as “Path” in the Basic panel:

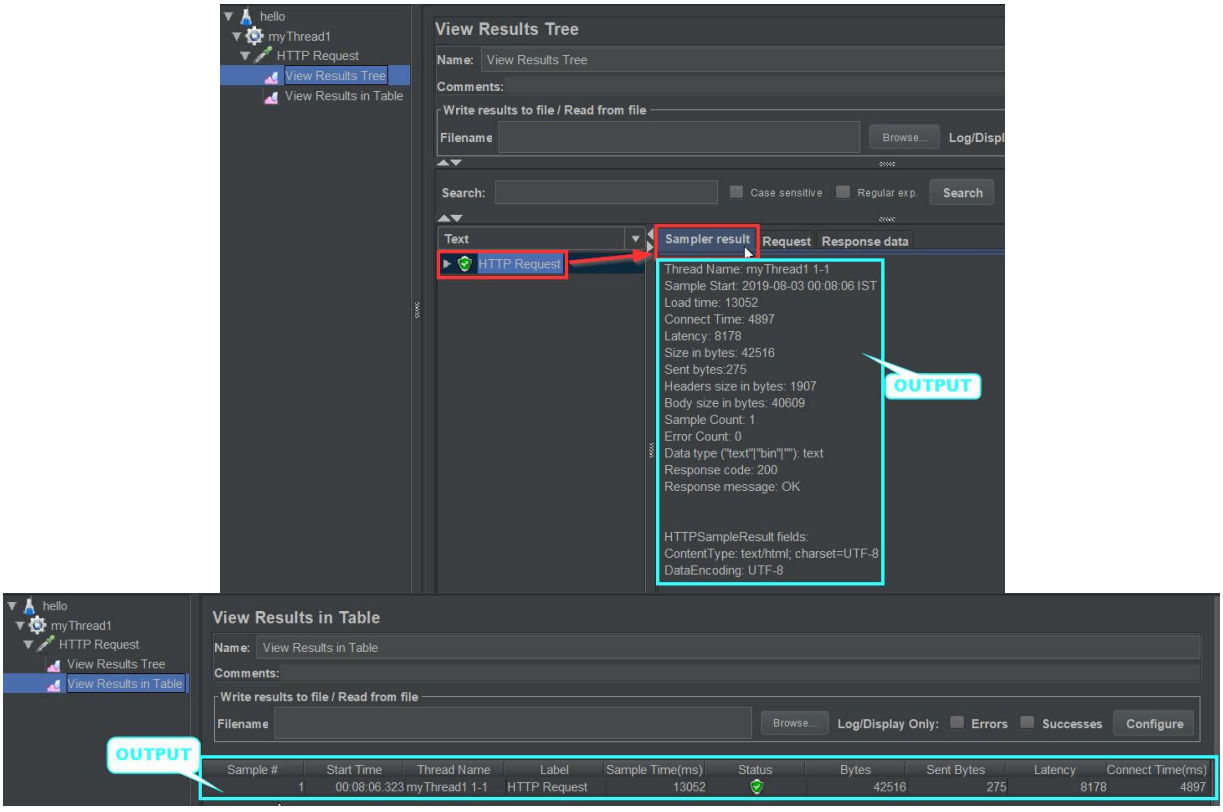


6) Under HTTP Request Sampler, add 2 “Listeners”, namely View Results Tree & View Results
in Table:
(right-click over HTTP Request(Sampler) > Add > Listener > View Results Tree) &
(right-click over HTTP Request(Sampler) > Add > Listener > View Results in Table):

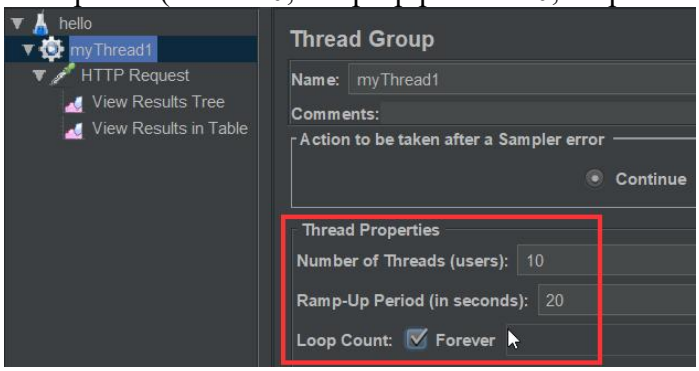


7) **Save(CTRL+S) & Run(CTRL+R)** the file and wait for a while. You'll see some:

• **OUTPUTS:**

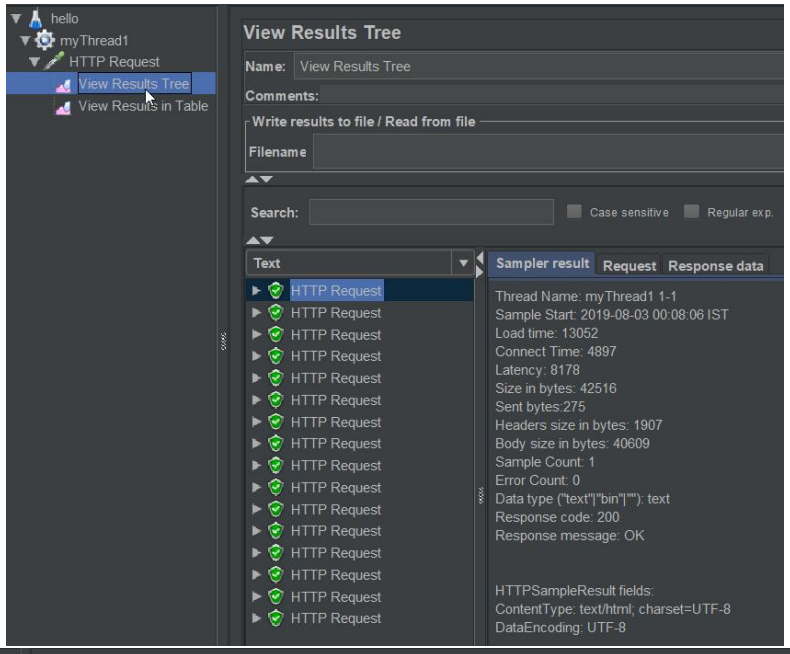


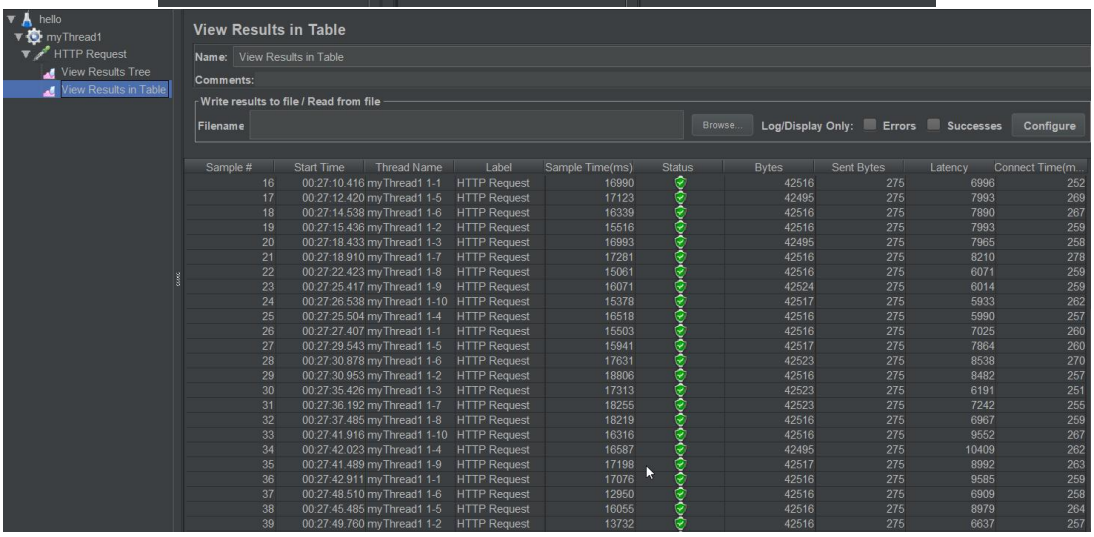
8) Now change some Thread Properties(users=10; ramp-up period=20; loop count=Forever):



9) **Save(CTRL+S) & Run(CTRL+R)** the file and wait for a while. You'll see some infinite:

• **OUTPUTS:**





10) Finish!

Apache JMeter is:

- -designed to load test functional behavior and measure performance.
- -used to test performance both on static and dynamic resources, web dynamic applications.
- Some features include: ability to test many apps/server/protocol types; Test Plan recording, building,
- debugging; complete report; portable; pure java; multi-threading; offline analysis of results; highly extensible
- core.