

**Software testing and quality assurance
Practical Journal**

KAUSTUBH NARENDRA SONKUSARE
31011121079
Bsc.Computer Science (hons)
SEM V

Index

Sr.no	Practical	Date
1	Manual Testing of Google Search Browser(google.com) for test cases (Practical 1)	04-07-2023
2	Right and test a program to login a specific web page (Practical 2)	09-08-2023
3	Install the Selenium server (Selenium RC) and demonstrate it using a script in Java/PHP. (Practical 3)	25-08-2023
4	Write an HTML program to calculate the HCF and GCD of two numbers. Then, write a code in Eclipse to automatically check if the HTML calculates the HCF and GCD. (Practical 5)	04-10-2023
5	Write a code in Eclipse to find/calculate the number of different web elements on google page. (Practical 6)	13-10-2023
6	Write an HTML code for a web page with radio buttons as well as checkboxes. Then, write a code in Eclipse to check the number of checkboxes and radio buttons that have been selected/checked on that web page (Practical 7)	13-10-2023
7	Excel Record updation automation using Eclipse and Selenium using the jxl.jar file. (Practical 8)	13-10-2023

Practical 1: Manual Testing of Google Search Browser(google.com) for test cases

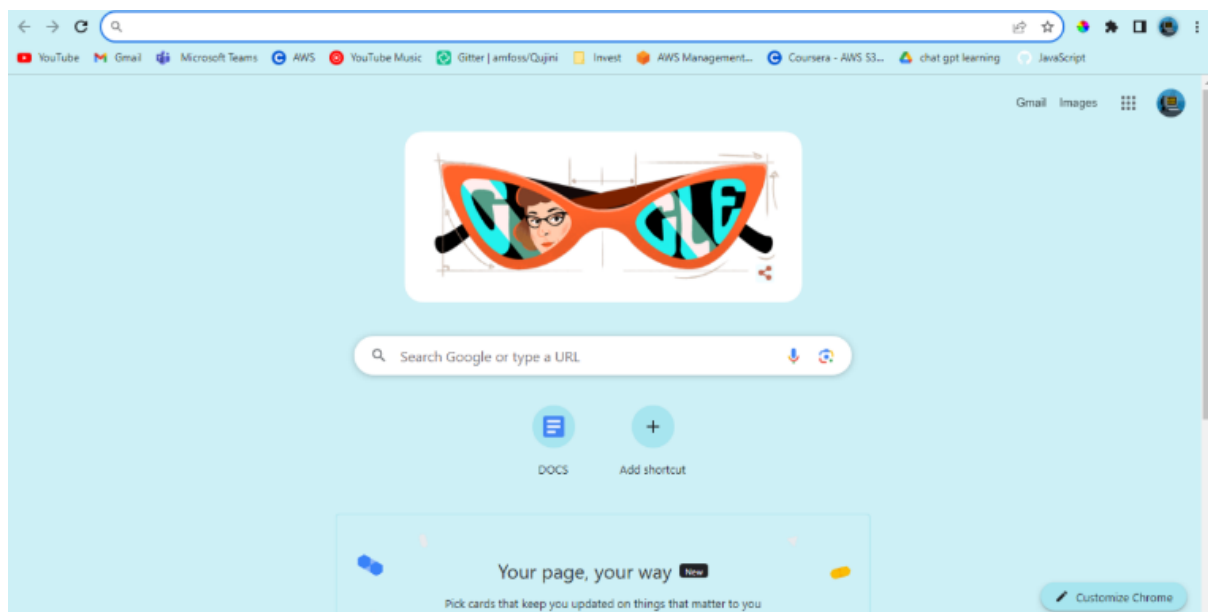
- 1.1 To check website loads properly
- 1.2 To check hyperlinks redirects on the given link
- 1.3 to check whether website detects the regional Language
- 1.4 Multiple languages offered at the home page
- 1.5 Authentication of Google Logo
- 1.6 To check whether website detects the theme
- 1.7 Response Time for each search
- 1.8 Autocomplete of search results
- 1.9 Correction of wrong spelling in search
- 1.10 Translation of Text

Case 1 - To check website loads properly

Expected output - loads properly with all embedded CSS

Actual output- loads properly

Remarks - Passed

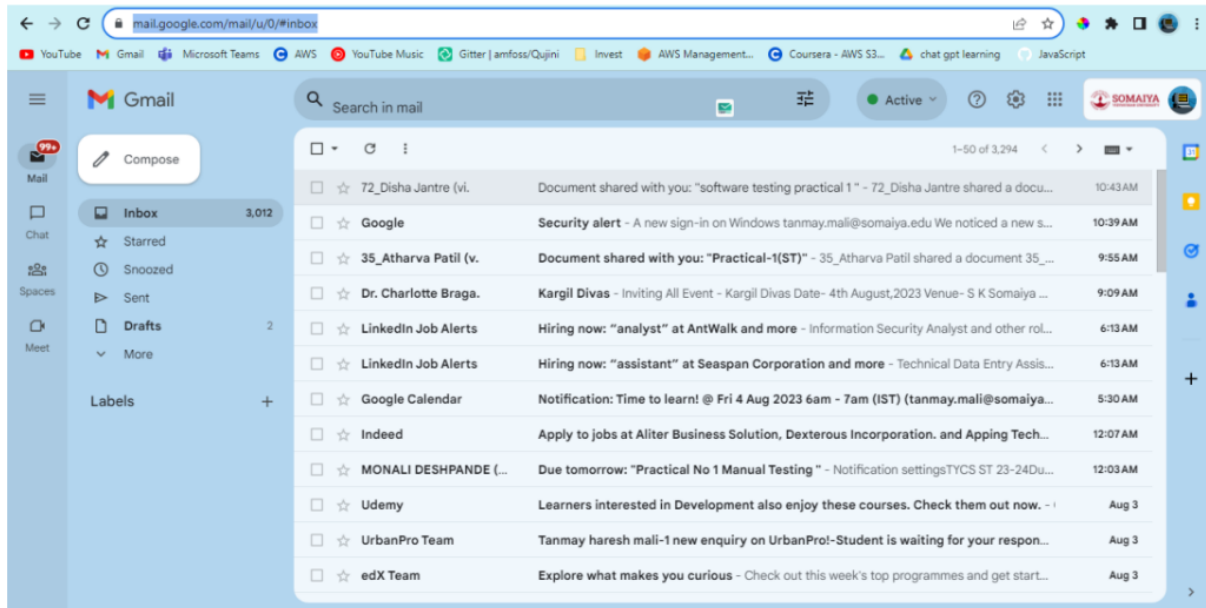


Case 2 -To check hyperlinks redirects on the given link

Expected - redirect to gmail on clicking Gmail

Actual Output - redirecting to <https://mail.google.com/mail/u/0/#inbox>

Remarks - Passed

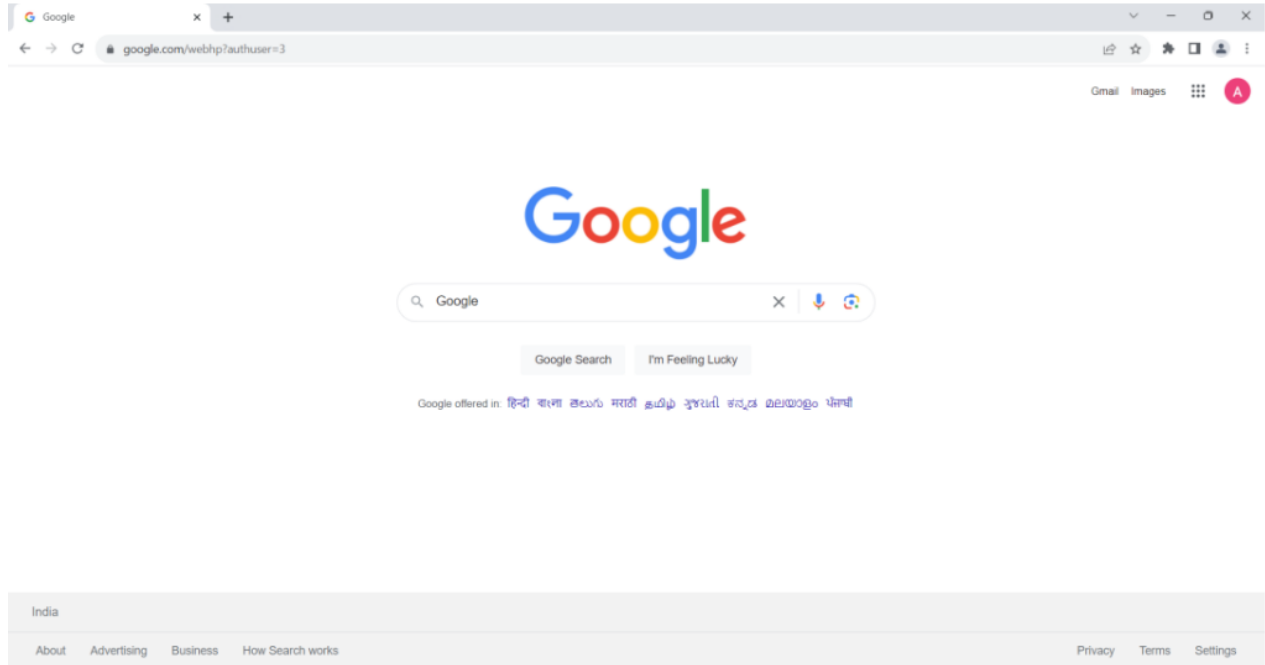


Case 3 - to check whether website detects the regional Language

Expected Output - Loads page on system selected language

Actual Output - loads in English US as the system language

Remarks - Passed

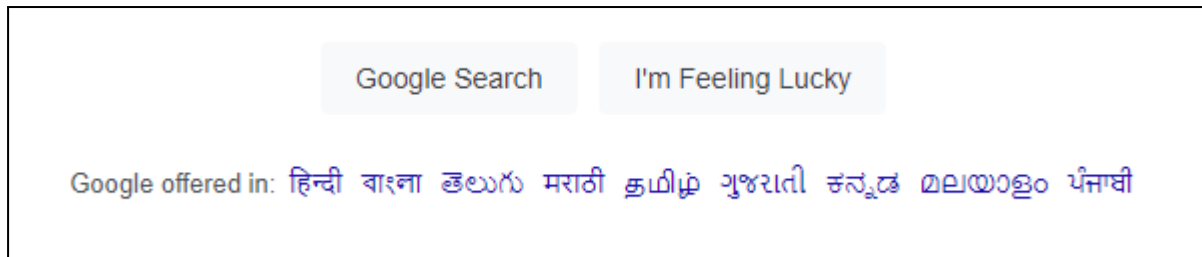


Case 4 - Multiple languages offered at the home page

Expected Output - Found below the search bar for the use

Actual Output - Found below the search bar for the use

Remarks - Passed



Case 5 - Authentication of Google Logo

Expected Output - Proper Colors and Font used for the logo

Actual Output - Standard logo with proper colors and fonts used

Remarks - Passed

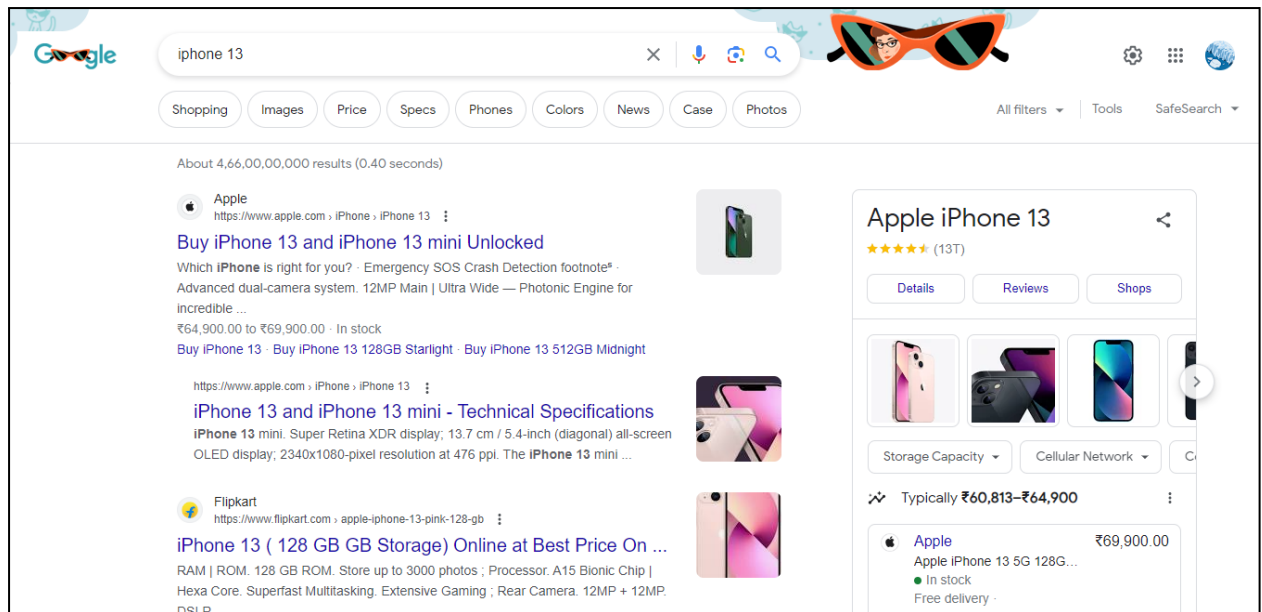


Case 6 - To check whether website detects the theme

Expected Output - when system on dark theme the site loads in dark mode

Actual Output - loads in dark mode

Remarks - Passed



Case 7 - Response Time for each search

Expected Output - Found below the search bar with time displayed in milliseconds

Actual Output - Found just below the ribbon below search bar

Remarks - Passed

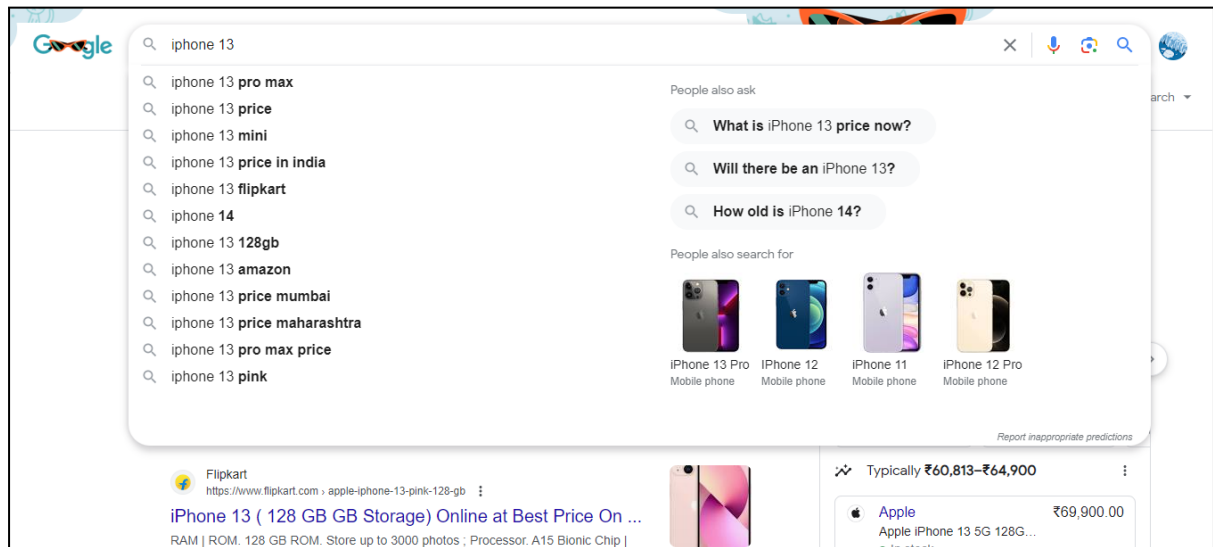
About 4,66,00,00,000 results (0.40 seconds)

Case 8 - Autocomplete of search results

Expected Output - Suggestions made for the word typing

Actual Output - Completion of search while writing/searching

Remarks - Passed

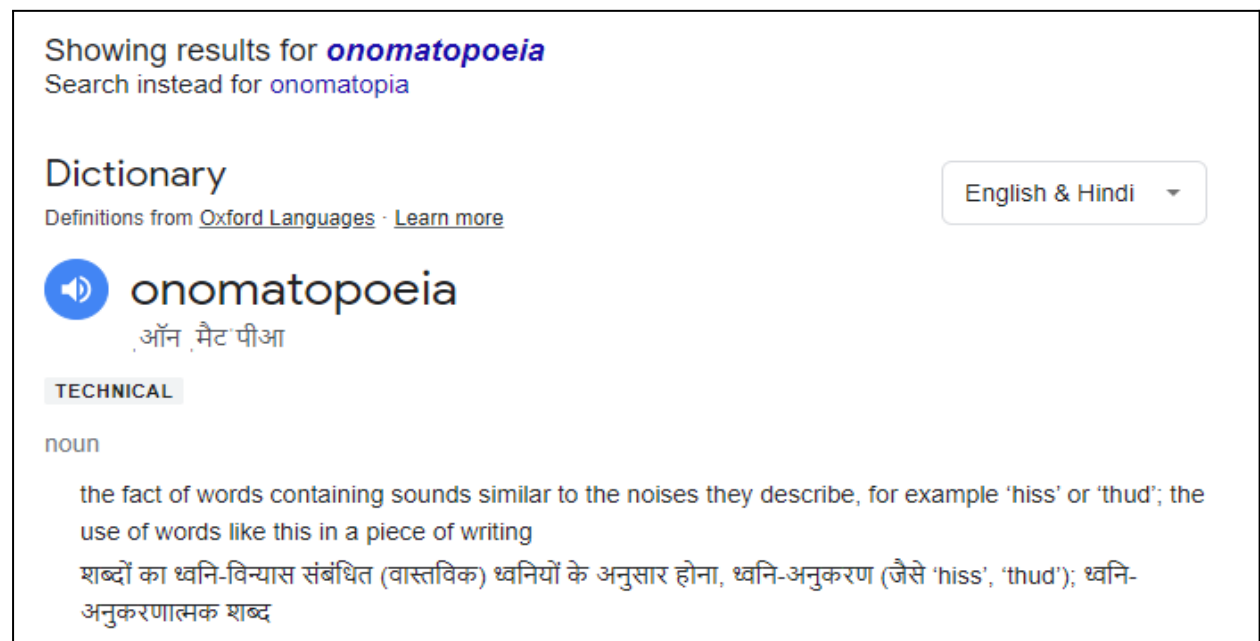


Case 9 - Correction of wrong spelling in search

Expected Output - Display of proper spelling of word

Actual Output - Proper spelling of the word along with result

Remarks - Passed



Case 10 - Translation of Text

Expected Output - Translation of various different languages into preferred language

Actual Output - Translation of spanish into english

Remarks - Passed



Test Case No.	Title	Input	Expected Output	Actual Output	Remarks
1	To check website loads properly	Google	loads properly with all embedded CSS	loads properly	Passed
2	To check hyperlinks redirects on the given link	Google	redirect to gmail on clicking Gmail	redirecting to https://mail.google.com/mail/u/0/#inbox	Passed
3	to check whether website detects the regional Language	Google	Loads page on system selected language	loads in English US as the system language	Passed
4	Multiple languages offered at the home page	Google	Found below the search bar for the use	Found below the search bar for the use	Passed
5	Authentication of Google Logo	Google	Standard logo with proper colors and fonts used	Proper Colors and Font used for the logo	Passed
6	To check whether website detects the theme	Google	when system on dark theme the site loads in dark mode	loads in dark mode	Passed
7	Response Time for each search	Google	Found below the search bar with time displayed in milliseconds	Found just below the ribbon below search bar	Passed
8	Autocomplete of search results	Google	Suggestions made for the word typing	Completion of search while writing/searching	Passed
9	Correction of wrong spelling in search	Google	Display of proper spelling of word	Proper spelling of the word along with result	Passed
10	Translation of Text	Google	Translation of various different languages into preferred language	Translation of spanish into english	Passed

Practical-2: Right and test a program to login a specific web page

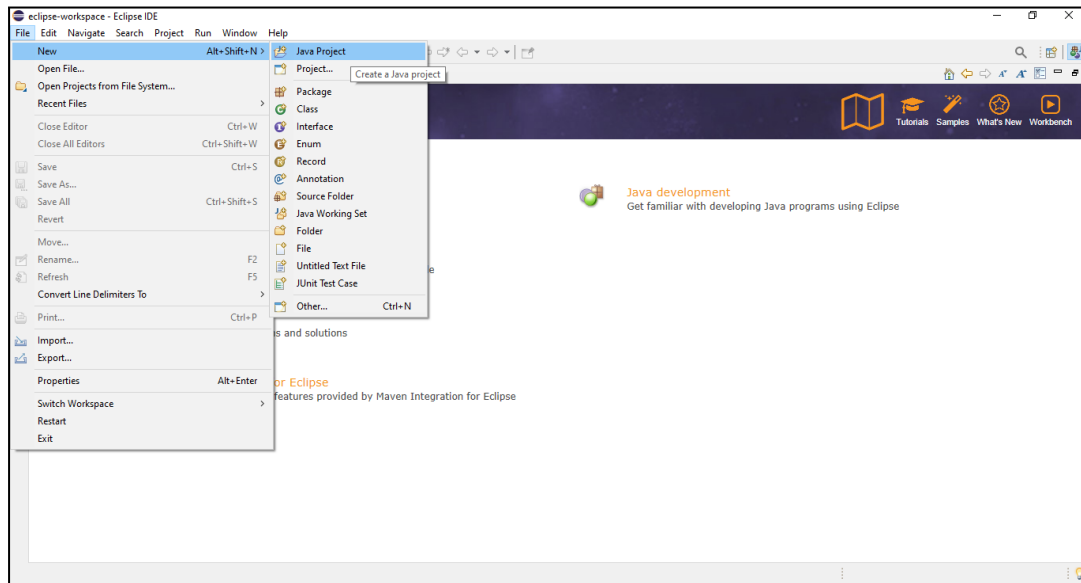
Aim: Program to login Google account

Procedure:

- 1) Download and setup Eclipse workspace
- 2) Create a new project

Steps:

i) New → Java project



ii) Fill up project name, module name

The screenshot shows the 'New Java Project' dialog box in the Eclipse IDE. The dialog is titled 'New Java Project' and has a subtitle 'Create a Java Project'. Below the subtitle, it says 'Create a Java project in the workspace or in an external location.' The main area of the dialog is divided into several sections: 'Project name', 'Location', 'JRE', 'Project layout', 'Working sets', and 'Module'. The 'Project name' field is filled with 'SeleniumTest'. The 'Use default location' checkbox is checked, and the 'Location' field shows the path 'C:\Users\Admin\Desktop\SELENIUM1\SeleniumTest'. In the 'JRE' section, the 'Use an execution environment JRE' radio button is selected, and the dropdown menu shows 'JavaSE-17'. The 'Project layout' section has the 'Create separate folders for sources and class files' radio button selected. The 'Working sets' section has the 'Add project to working sets' checkbox unchecked. The 'Module' section has the 'Create module-info.java file' checkbox unchecked. At the bottom of the dialog, there are four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'. The 'Finish' button is highlighted with a blue border.

New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location: [Browse...](#)

JRE

☒ Use an execution environment JRE: [Configure JREs...](#)

☐ Use a project specific JRE:

☐ Use default JRE 'jre' and workspace compiler preferences

Project layout

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets [New...](#)

Working sets: [Select...](#)

Module

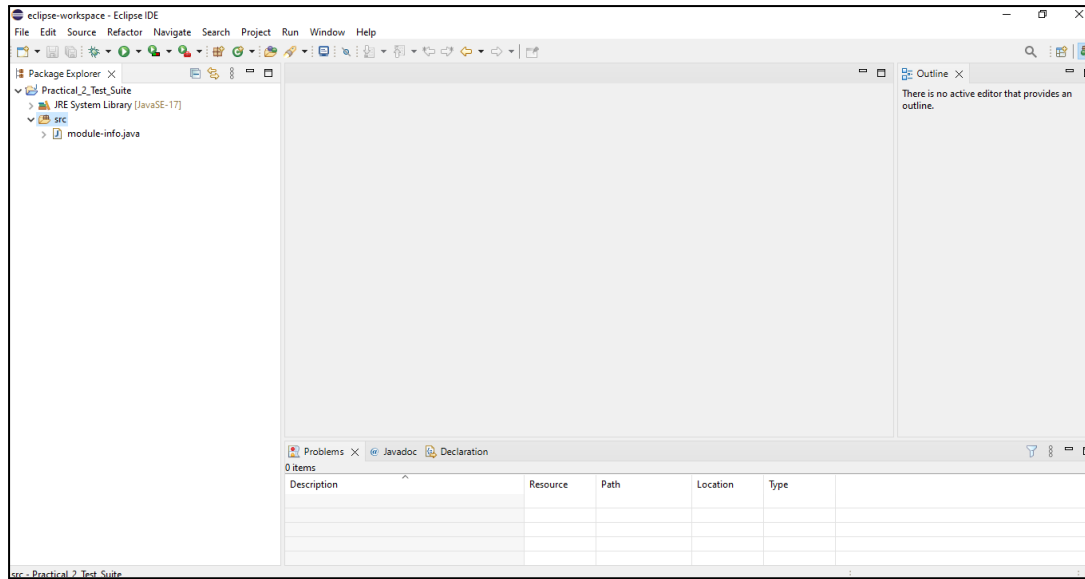
☐ Create module-info.java file

Module name:

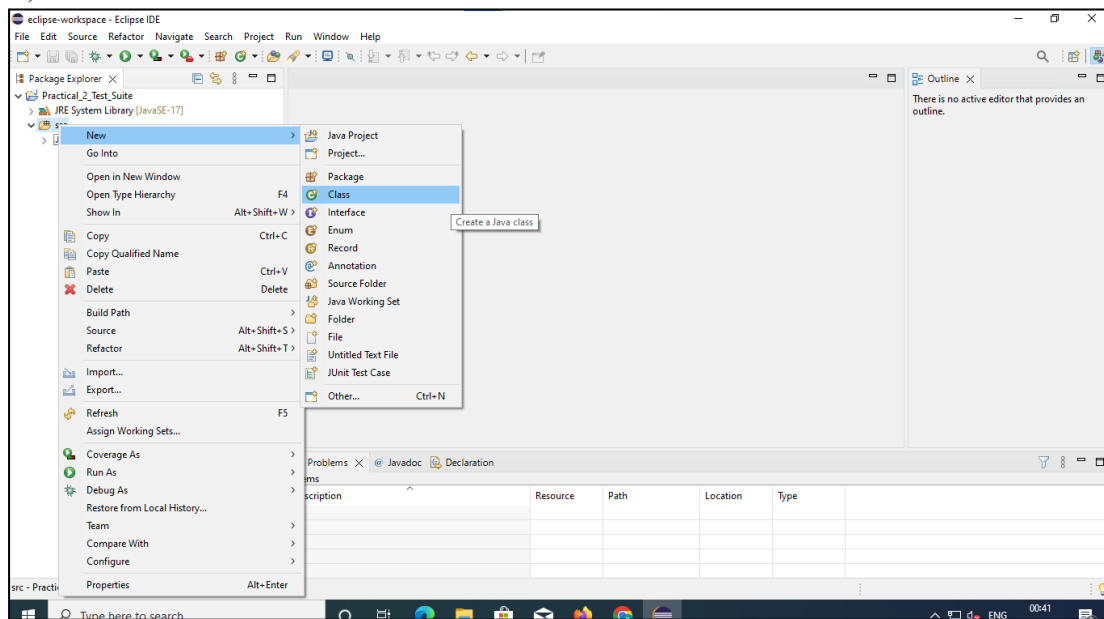
☐ Generate comments

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

iii) Project created



iv) create class



v) Add name for class

New Java Class

The use of the default package is discouraged.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

Which method stubs would you like to create?

☐ public static void main(String[] args)

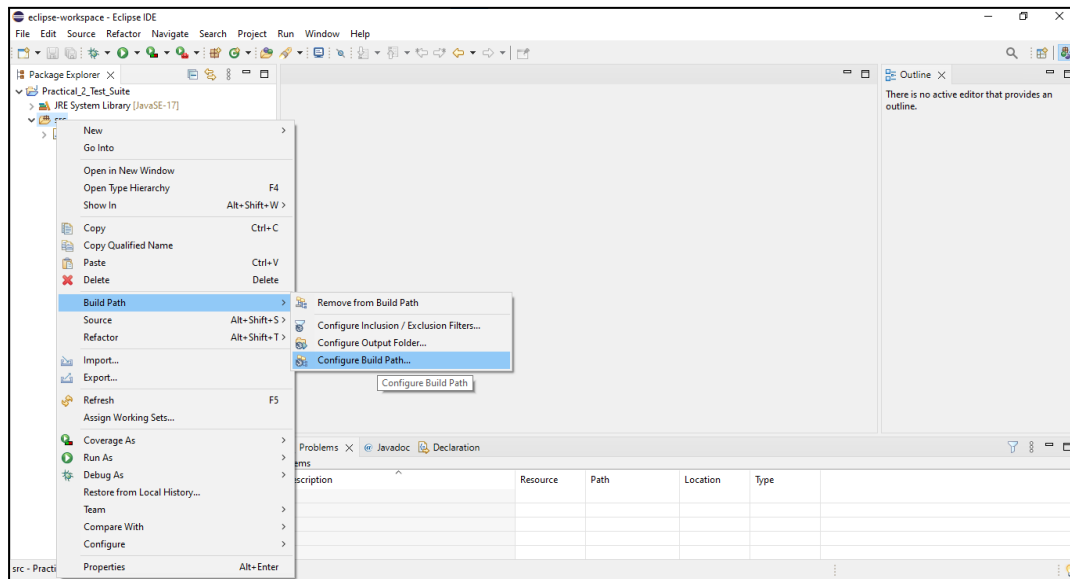
☐ Constructors from superclass

☒ Inherited abstract methods

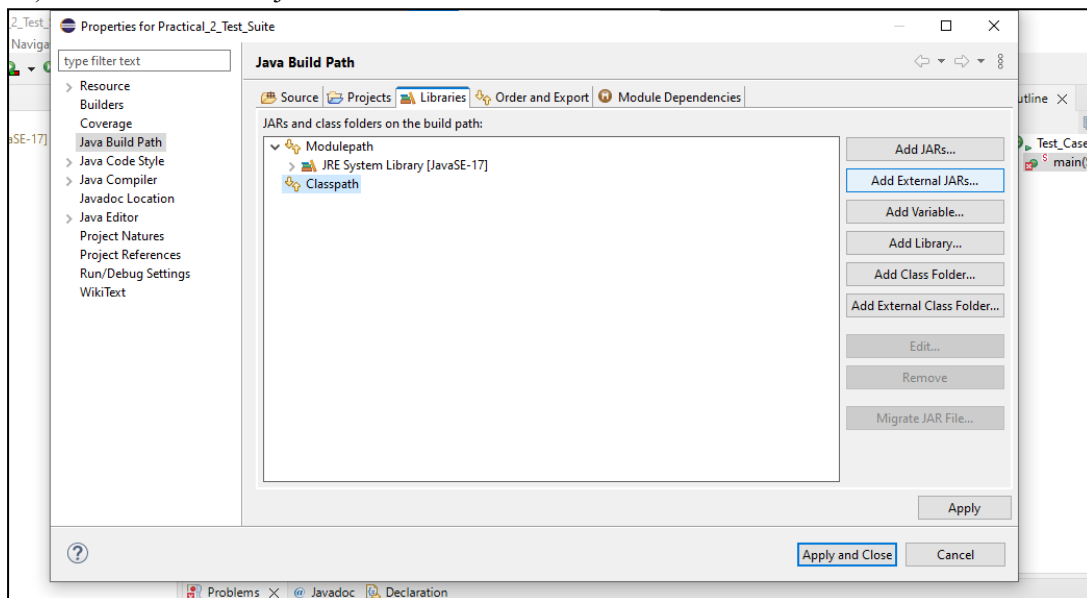
Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

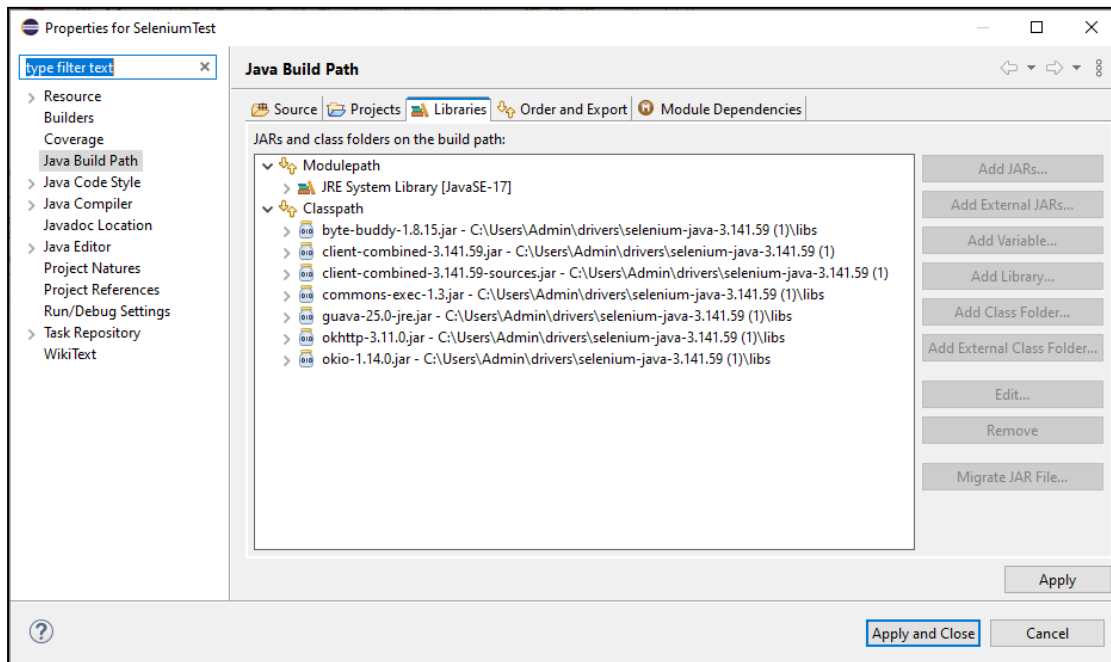
vi) Configure build path



vii) click Add external jar



viii) select selenium files

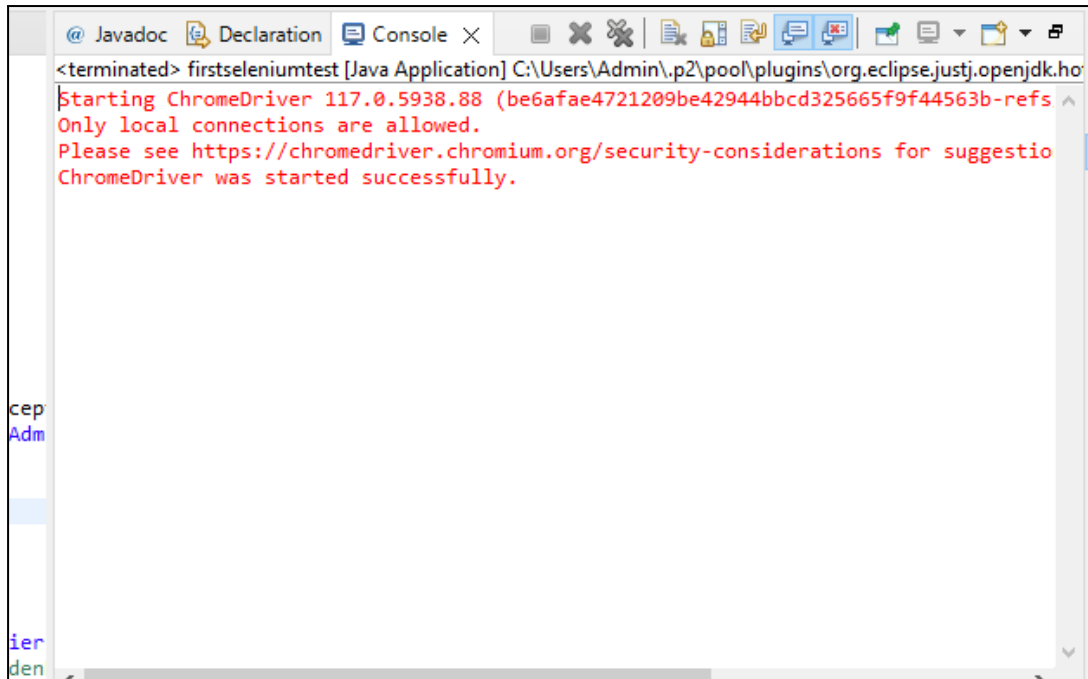


Java code:

```
package com.Test_Case_1.scripts;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
public class Test_Case_1 {
static WebDriver driver;
public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver","c:/Users/Admin/drivers/chromedriver.exe");
driver = new ChromeDriver();
driver.get("https://www.gmail.com");
//driver.get("https://www.mycontactform.com");
driver.manage().window().maximize();
Thread.sleep(3000);
//driver.quit();
WebElement username = driver.findElement(By.id("identifierId"));
//WebElement password = driver.findElement(By.name("hiddenPassword"));
//name="hiddenPassword"
username.sendKeys("tanmay.mali@somaiya.edu");
//password.sendKeys("");
WebElement login = driver.findElement(By.id("DH6Rkf"));
login.click();
try
{
Thread.sleep(2000);
//WebElement navBar = driver.findElement(By.id("js_u"));
System.out.println("Test Case Pass");
driver.close();
}
catch(Exception e)
{
System.out.print("Test Case Failed");
}
}
}
```

Output:

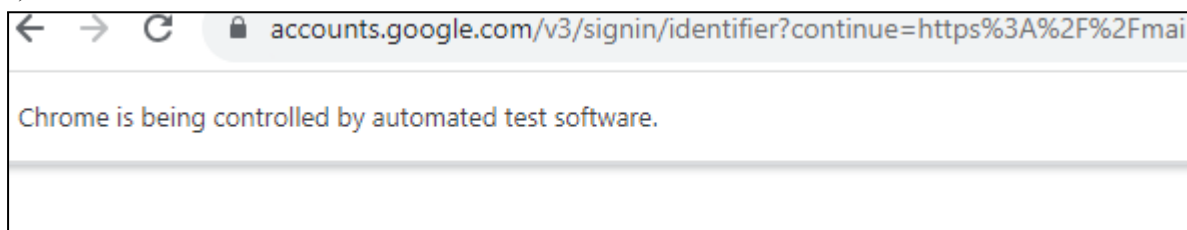
i) in eclipse



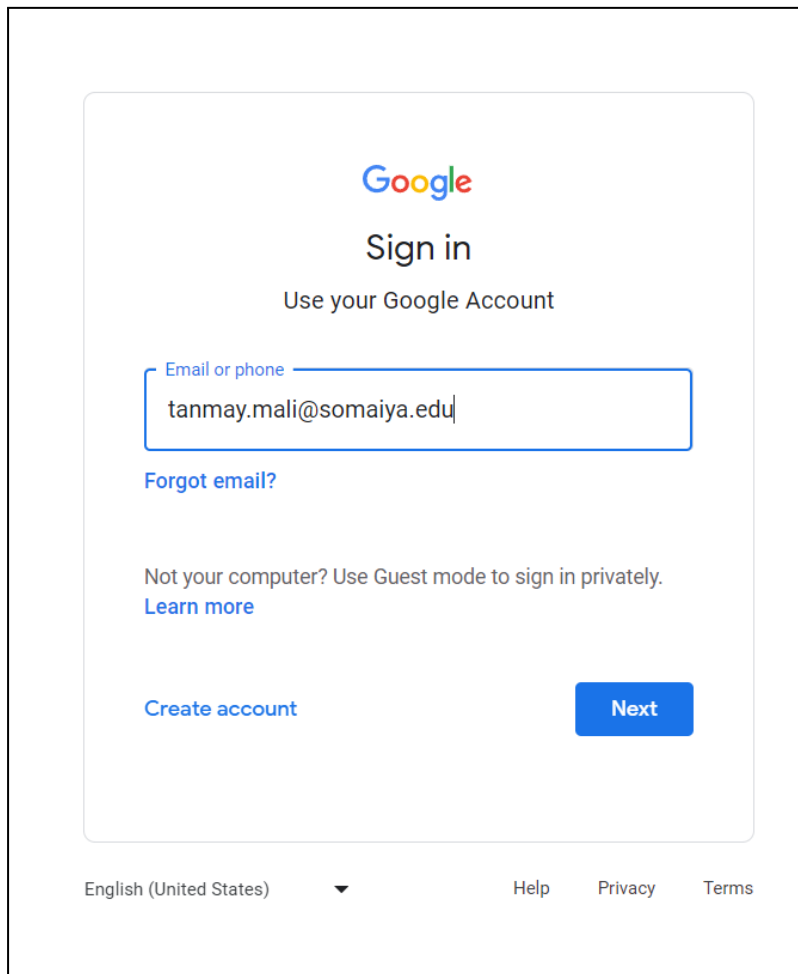
The screenshot shows the Eclipse IDE's Console window. The title bar includes tabs for 'Javadoc', 'Declaration', and 'Console'. The console output is as follows:

```
<terminated> firstseleniumtest [Java Application] C:\Users\Admin\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full\jre\bin\java.exe  
Starting WebDriverManager 117.0.5938.88 (be6afae4721209be42944bbcd325665f9f44563b-refs)  
Only local connections are allowed.  
Please see https://chromedriver.chromium.org/security-considerations for suggestions  
ChromeDriver was started successfully.
```

ii) in chrome



iii) Sign-in activity



It automatically enters the given email address

```
WebElement username = driver.findElement(By.id("identifierId"));  
//WebElement password = driver.findElement(By.name("hiddenPassword"));  
//name="hiddenPassword"
```

For this part —> the identifier id

ID

Right click on login page(labe)>inspect>take the name and ID from there

```
<input type="email" class="whsOnd zHQkBf" jsname="YPqjbf" autocomplete="username" spellcheck="false" tabindex="0"  
aria-label="Email or phone" name="identifier" value autocapitalize="none" id="identifierId" dir="ltr" data-initial-  
dir="ltr" data-initial-value="tanmay.mali@somaiya.edu" badinput="false"> == $0
```

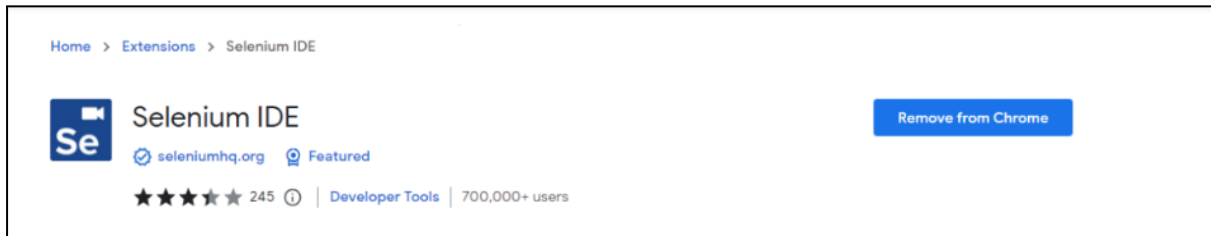
Same thing for password

```
<input type="password" class="whsOnd zHQkBf" jsname="YPqjbf" autocomplete="current-password" spellcheck=  
"false" tabindex="0" aria-label="Enter your password" name="Passwd" autocapitalize="off" dir="ltr" data-  
initial-dir="ltr" data-initial-value> == $0
```

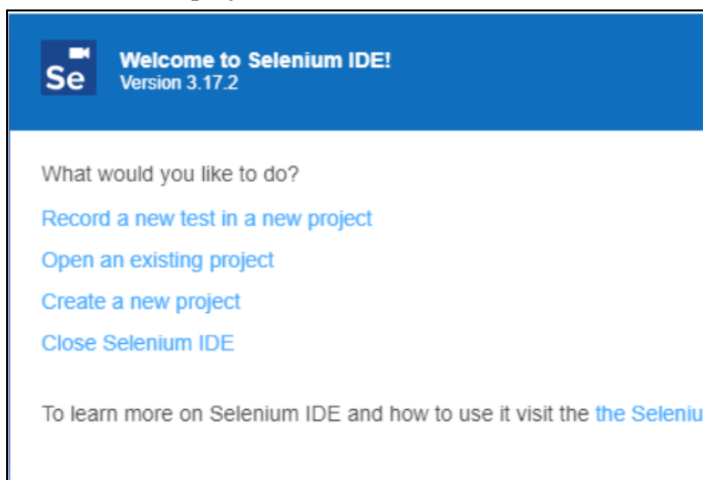
Practical 3: Install the Selenium server (Selenium RC) and demonstrate it using a script in Java/PHP.

Steps:

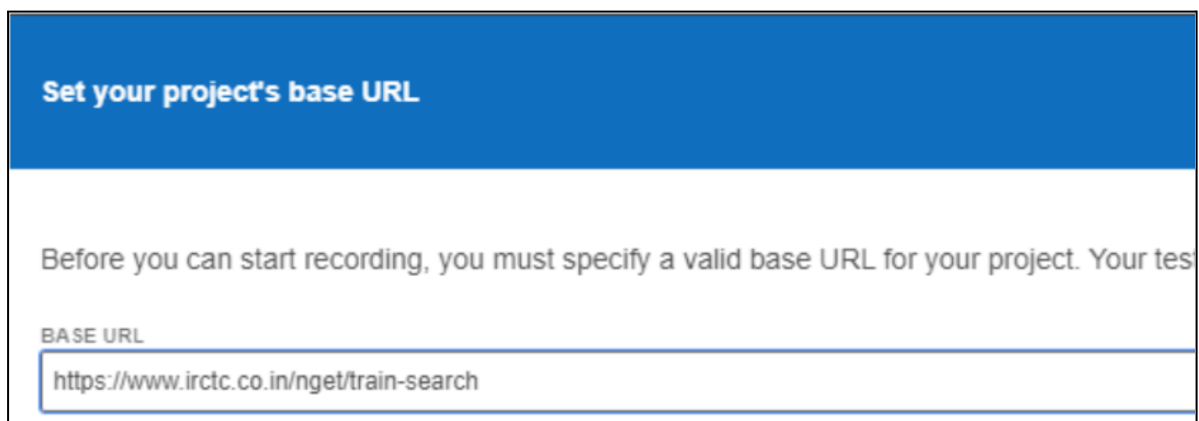
1. Add a Selenium IDE extension in your google chrome.



2. Create a new project and test case.



3. For 1st test, add the url of IRTC site.



4. Start recording and do the following tasks for testing of IRCTC.

PNR STATUS

CHARTS / VACANCY

From

C SHIVAJI MAH T - C

DD/MM/YYYY *

25/08/2023

To

SIMLA - SML

AC 2 Tier (2A)

GENERAL

☐ Person With Disability Concession
 ☐ Flexible With Date
 ☐ Train with Available Berth
 ☐ Railway Pass Concession

Search

Try booking in Ask DISHA 2.0

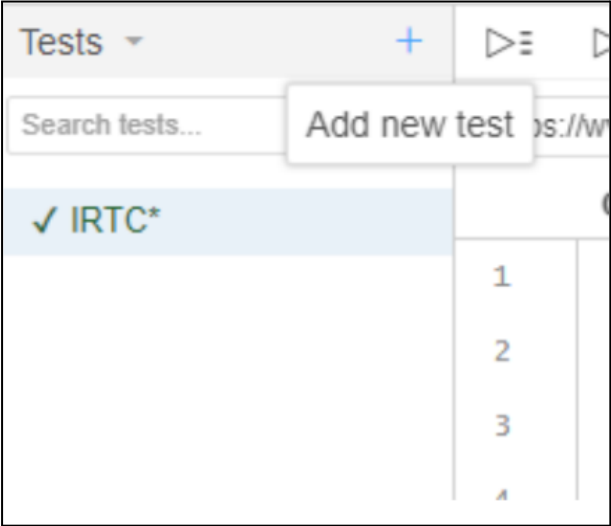
5. Check the test case, by clicking run :

Tests			
Run current test Ctrl+R			
✓ IRTC*	Command	Target	Value
1	✓ open	/ngel/train-search	
2	✓ set window size	1050x840	
3	✓ click	css= ui-autocomplete > .ng-tns-c57-8	
4	✓ type	css= .ng-tns-c57-8 > ui-inputtext	csmt
5	✓ click	css= .ng-star-inserted:nth-child(1) > strong	
6	✓ click	css= ui-autocomplete > .ng-tns-c57-9	
7	✓ type	css= .ng-tns-c57-9 > ui-inputtext	shimla
8	✓ click	css=#p-highlighted-option > .ng-star-inserted > .ng-star-inserted > strong	
9	✓ click	css= ui-dropdown-label-container > .ng-tns-c65-11	
10	✓ click	css= .ng-tns-c65-11:nth-child(6) > ui-dropdown-item	
11	✓ click	css= .train_Search	
<div> <div>Command</div> <div>Target</div> <div>Value</div> <div>Description</div> </div>			
<div> <div>Log</div> <div>Reference</div> </div>			
<div> <div>8. click on css=#p-highlighted-option > .ng-star-inserted > .ng-star-inserted > strong OK</div> <div>9. click on css= ui-dropdown-label-container > .ng-tns-c65-11 OK</div> <div>10. click on css= .ng-tns-c65-11:nth-child(6) > ui-dropdown-item OK</div> <div>11. click on css= .train_Search OK</div> <div>12. click on css= ui-confirm-dialog-acceptbutton > ui-button-text OK</div> <div>13. close OK</div> <div>IRTC* completed successfully</div> </div>			

	Command	Target	Value
1	✓ open	/ngel/train-search	
2	✓ set window size	1050x840	
3	✓ click	css= ui-autocomplete > .ng-tns-c57-8	
4	✓ type	css=.ng-tns-c57-8 > .ui-inputtext	csmt
5	✓ click	css=.ng-star-inserted:nth-child(1) > strong	
6	✓ click	css= ui-autocomplete > .ng-tns-c57-9	
7	✓ type	css=.ng-tns-c57-9 > .ui-inputtext	shimla
8	✓ click	css=#p-highlighted-option > .ng-star-inserted > .ng-star-inserted > strong	
9	✓ click	css= ui-dropdown-label-container > .ng-tns-c65-11	
10	✓ click	css= .ng-tns-c65-11:nth-child(6) > .ui-dropdown-item	
11	✓ click	css=train_Search	

Log	Reference
8. click on css=#p-highlighted-option > .ng-star-inserted > .ng-star-ins	
9. click on css=.ui-dropdown-label-container > .ng-tns-c65-11	OK
10. click on css=.ng-tns-c65-11:nth-child(6) > .ui-dropdown-item	OK
11. click on css=.train_Search	OK
12. click on css=.ui-confirmdialog-acceptbutton > .ui-button-text	OK
13. close	OK
'IRTC' completed successfully	

6. Add a new test case named python and add url of python



7. Check if the test case works properly, by clicking run :

Project: ST_Pract_2*

Tests +

Search tests... Q

✓ IRTC*

✓ Python*

https://www.python.org/

	Command	Target	Value
1	✓ open	https://www.python.org/	
2	✓ set window size	1050x840	
3	✓ click	linkText=Python 3.11.5	
4	✓ click	linkText=Windows installer (64-bit)	
5	✓ close		

Command

Target

Value

Description

Log Reference

Running 'Python'

1. open on https://www.python.org/ OK 00:06:52

2. setWindowSize on 1050x840 OK 00:06:52

3. click on linkText=Python 3.11.5 OK 00:06:52

4. click on linkText=Windows installer (64-bit) OK 00:06:53

5. close OK 00:06:54

'Python' completed successfully 00:06:54

python-3.11.5-amd64 (2).exe

24.7 MB • Done

python-3.11.5-amd64 (1).exe

24.7 MB • Done

Tests +

Search tests... Q

✓ IRTC*

✓ Python*

https://www.python.org/

	Command	Target
1	✓ open	https://www.python.org/
2	✓ set window size	1050x840
3	✓ click	linkText=Python 3.11.5
4	✓ click	linkText=Windows installer (64-bit)
5	✓ close	

Log Reference

Running 'Python'

1. open on https://www.python.org/ OK

2. setWindowSize on 1050x840 OK

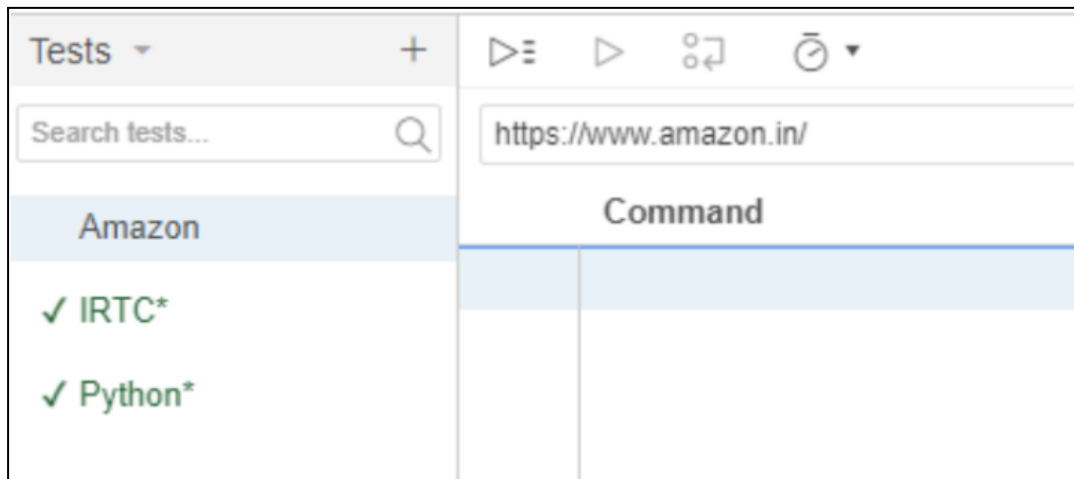
3. click on linkText=Python 3.11.5 OK

4. click on linkText=Windows installer (64-bit) OK

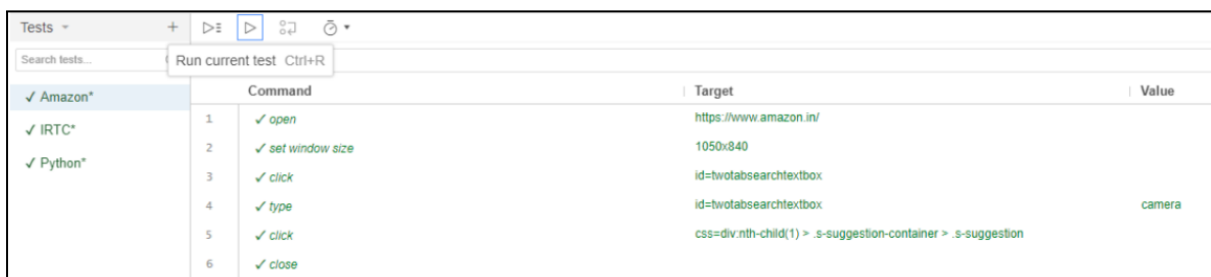
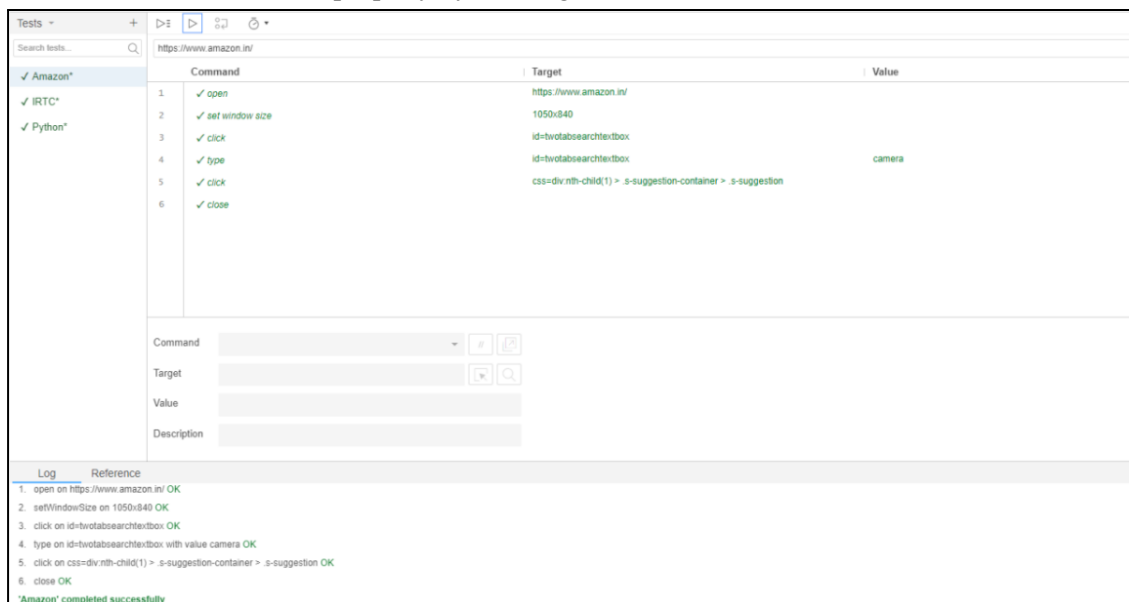
5. close OK

'Python' completed successfully

8. Create a new test for Amazon too



9. Check if the test case run properly by clicking run:-



Running 'Amazon'

1. open on <https://www.amazon.in/> OK
2. setWindowSize on 1050x840 OK
3. click on id=twotabsearchtextbox OK
4. type on id=twotabsearchtextbox with value camera OK
5. click on css=div:nth-child(1) > .s-suggestion-container > .s-suggestion OK
6. close OK

'Amazon' completed successfully

Practical 5: Write an HTML program to calculate the HCF and GCD of two numbers. Then, write a code in Eclipse to automatically check if the HTML calculates the HCF and GCD.

HTML CODE:

```
<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>
<h1>Program to calculate gcd of 2 numbers</h1>

Enter Two numbers:
<form name="myform">
Number 1 :<input type="text" name="n1" value="">
<br><br>
Number 2 :<input type="text" name="n2" value="">
<br><br>
<input type="button" value="get GCD"
onClick="gcd()"><br><br>
GCD is :<input type="text" name="result" value="">
</body>
</html>
```

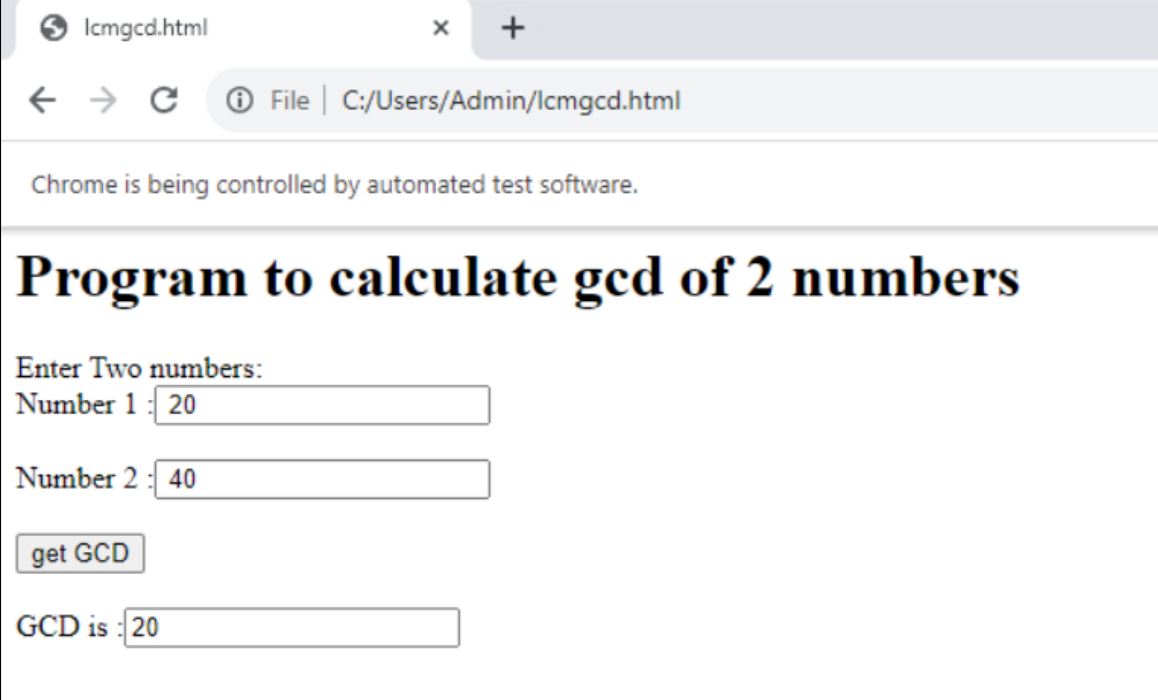
ECLIPSE JAVA CODE :

```
Gcd.java

package practical7;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
public class Gcd {
public static void main (String args[]) {
System.setProperty("webdriver.chrome.driver","C:/Users/Admin/Downloads/chromedriver.exe");

WebDriver driver = new ChromeDriver();
driver.get("file:///C:/Users/Admin/lcmgcd.html");
driver.manage().window().maximize();
driver.findElement(By.name("n1")).sendKeys("20");
driver.findElement(By.name("n2")).sendKeys("40");
driver.findElement(By.name("Get GCD")).click();
String result =
driver.findElement(By.name("result")).getAttribute("Values");
System.out.println("Result is:"+result);
}
}
```

OUTPUT:



The screenshot shows a web browser window with the title 'lcmgcd.html'. The address bar shows the file path 'C:/Users/Admin/lcmgcd.html'. A message at the top states 'Chrome is being controlled by automated test software.' The main heading is 'Program to calculate gcd of 2 numbers'. Below this, there is a form with the label 'Enter Two numbers:'. It contains two input fields: 'Number 1' with the value '20' and 'Number 2' with the value '40'. A button labeled 'get GCD' is positioned below these fields. At the bottom, there is a label 'GCD is :' followed by an input field containing the value '20'.

elements on google page

ECLIPSE CODE:

```
package practical6;
import java.util.List;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

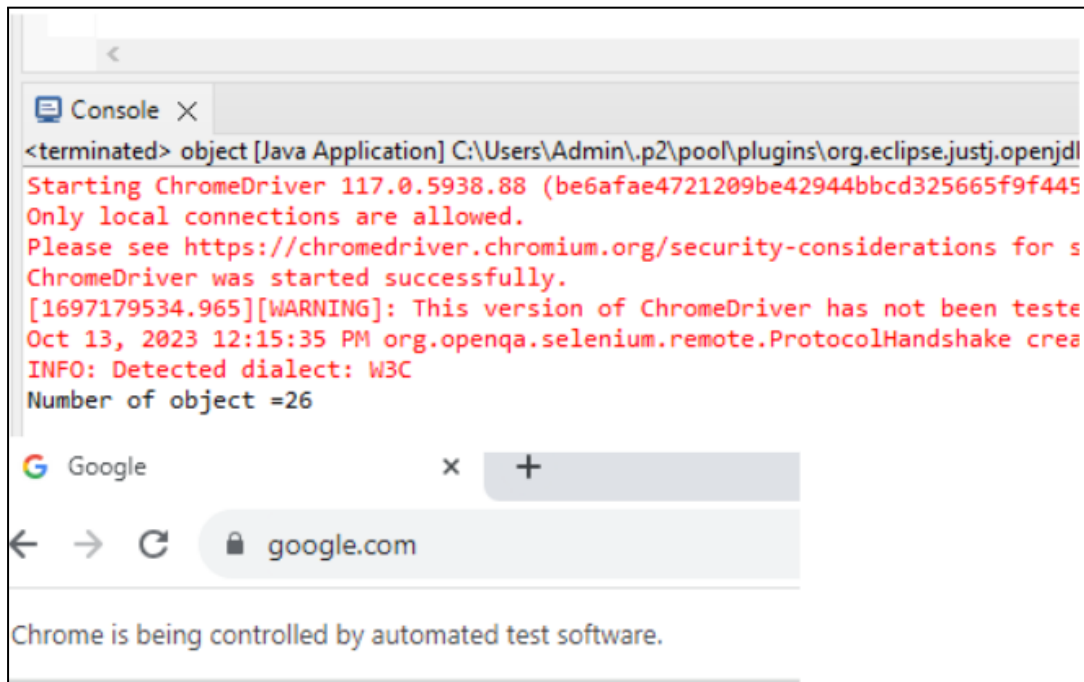
public class object {
    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver","C:/Users/Admin/Downloads/chromedriver.exe");

        WebDriver driver=new ChromeDriver();
        driver.get("https://www.google.com");

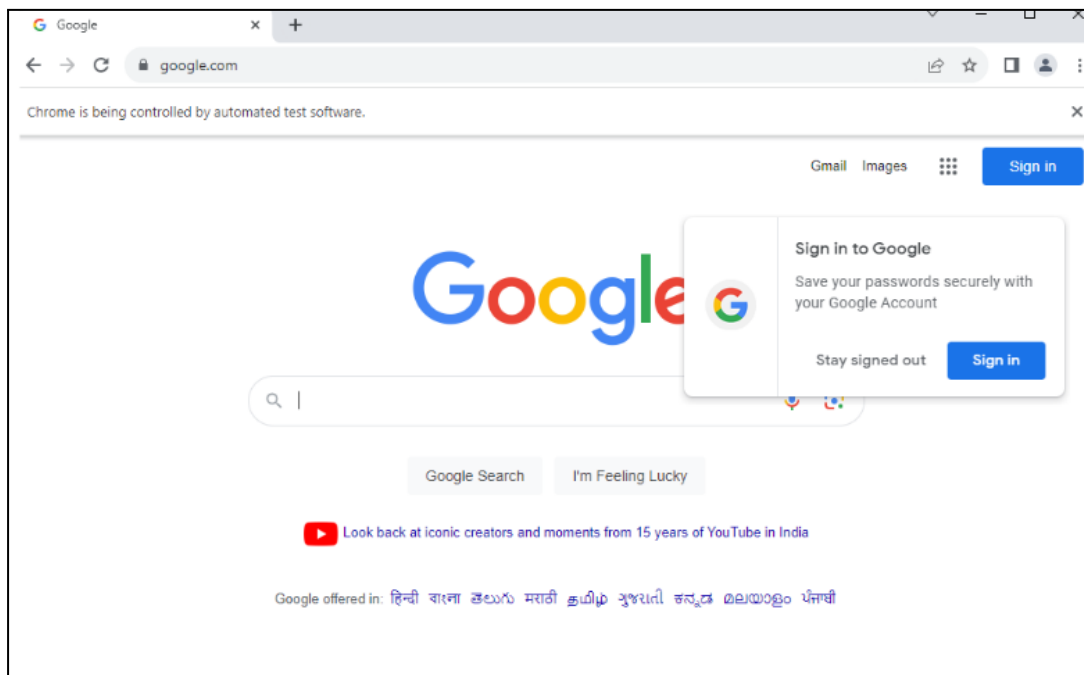
        List <WebElement> mylist= driver.findElements(By.xpath("//a"));
        System.out.println("Number of object =" +mylist.size());
    }
}
```

OUTPUT:

i) eclipse output



ii)chrome browser output



Practical 7: Write an HTML code for a web page with radio buttons as well as checkboxes. Then, write a code in Eclipse to check the number of checkboxes and radio buttons that have been selected/checked on that web page

HTML CODE (Radio.html):

```
<html>
<body>
<form>

<h2>Select Gender</h2>
<input type="radio" name="gender" value="male"
checked> Male<br>
<input type="radio" name="gender" value="female">
Female<br>
<input type="radio" name="gender" value="other"
checked> Other<br>
<h2>Select Languages Known</h2>
<input type="checkbox" name="lang" value="Java"
checked="checked"> Java<br>
<input type="checkbox" name="lang" value="PHP"
checked="checked"> PHP<br> <input type="checkbox"
name="lang" value="C#"> C#<br>
<input type="checkbox" name="lang" value="Python"
checked="checked"> Python<br> <input
type="checkbox" name="lang" value="Ruby"> Ruby<br>
<br><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

Name the package while creating the class

New Java Class

Java Class

⚠ Type name is discouraged. By convention, Java type names usually start with an uppercase letter

Source folder: RadioOnHtml/src Browse...

Package: practical7 Browse...

☐ Enclosing type: Browse...

Name: checkradio

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass: java.lang.Object Browse...

Interfaces: Add...
Remove

Which method stubs would you like to create?

☒ public static void main(String[] args)
☐ Constructors from superclass
☐ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

? Finish Cancel

ECLIPSE JAVA CODE:

```
package practical7;
import java.util.*;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
public class checkradio {
public static void main(String args[])
{
System.setProperty("webdriver.chrome.driver","C:/Users/Admin/Downloads/chromedriver.exe");
WebDriver driver = new ChromeDriver();
driver.get("file:///C:/Users/Admin/radio.html");
int chk = 0;
int unchk = 0;
int rchk = 0;
int runchk = 0;
List<WebElement> els =
driver.findElements(By.xpath("//input[@type='checkbox']")); for(WebElement el :els)
{
if(el.isSelected()) {
chk++;
}
else{
unchk++;
}
}
System.out.println("Total checked items" + chk);
System.out.println("Total unchecked items" + unchk);
List<WebElement> elm =
driver.findElements(By.xpath("//input[@type='radio']")); for(WebElement ell : elm)
{
if(ell.isSelected()) {
rchk++;
}
else{
runchk++;
}
}
System.out.println("Total checked radio items" + rchk);
System.out.println("Total unchecked radio items" + runchk);}}
```


OUTPUT:

i) Eclipse output

```
Console X
<terminated> checkradio [Java Application] C:\Users\Admin\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.
ChromeDriver was started successfully.
[1697180846.046][WARNING]: This version of ChromeDriver has not been tested with Chrome version 118.
Oct 13, 2023 12:37:27 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Total checked items3
Total unchecked items2
Total checked radio items1
Total unchecked radio items2

Chrome is being controlled by automated test software.
```

ii) Browser output

radio.html x +

← → ↻ ⓘ File | C:/Users/Admin/radio.html

Chrome is being controlled by automated test software.

Select Gender

☐ Male
☐ Female
☒ Other

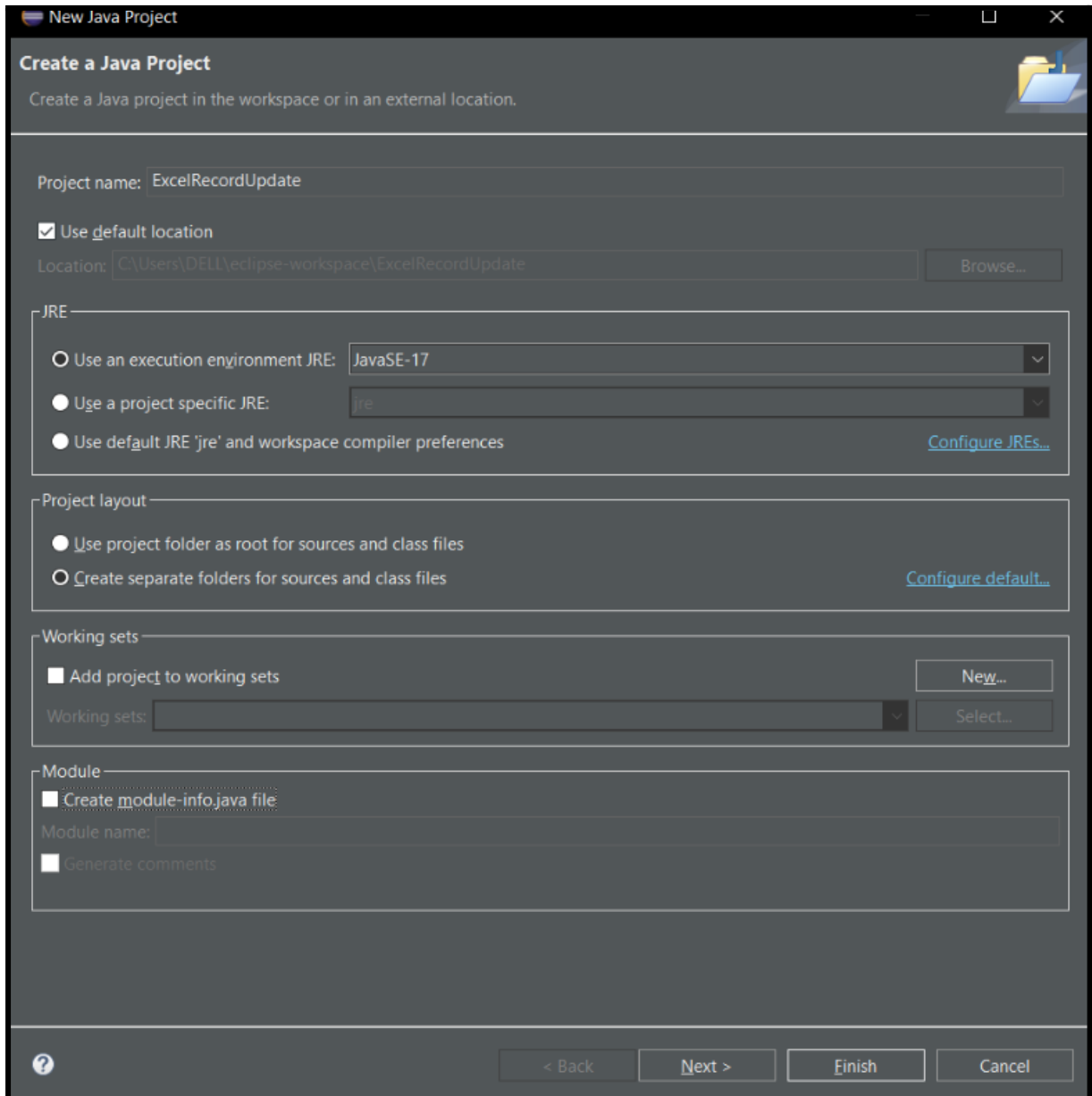
Select Languages Known

☒ Java
☒ PHP
☐ C#
☒ Python
☐ Ruby

Submit

Practical 8: Excel Record updation automation using Eclipse and Selenium using the jxl.jar file

- File > New Project > ExcelRecordUpdation(Project name)



The screenshot shows the 'New Java Project' dialog box in Eclipse. The title bar reads 'New Java Project'. The main heading is 'Create a Java Project' with a subtitle 'Create a Java project in the workspace or in an external location.' and a folder icon. The 'Project name' field contains 'ExcelRecordUpdate'. The 'Use default location' checkbox is checked, and the 'Location' field shows 'C:\Users\DELL\eclipse-workspace\ExcelRecordUpdate' with a 'Browse...' button. The 'JRE' section has three radio buttons: 'Use an execution environment JRE:' (selected) with a dropdown showing 'JavaSE-17', 'Use a project specific JRE:' with a dropdown showing 'jre', and 'Use default JRE 'jre' and workspace compiler preferences' with a 'Configure JREs...' link. The 'Project layout' section has two radio buttons: 'Use project folder as root for sources and class files' (selected) and 'Create separate folders for sources and class files' with a 'Configure default...' link. The 'Working sets' section has a checked 'Add project to working sets' checkbox, a 'New...' button, a 'Working sets:' dropdown, and a 'Select...' button. The 'Module' section has a checked 'Create module-info.java file' checkbox, a 'Module name:' field, and a 'Generate comments' checkbox. At the bottom are buttons for '?', '< Back', 'Next >', 'Finish', and 'Cancel'.

Create a Java Project
Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location
Location:

JRE

☐ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE 'jre' and workspace compiler preferences [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

☐ Create separate folders for sources and class files [Configure default...](#)

Working sets

☒ Add project to working sets

Working sets:

Module

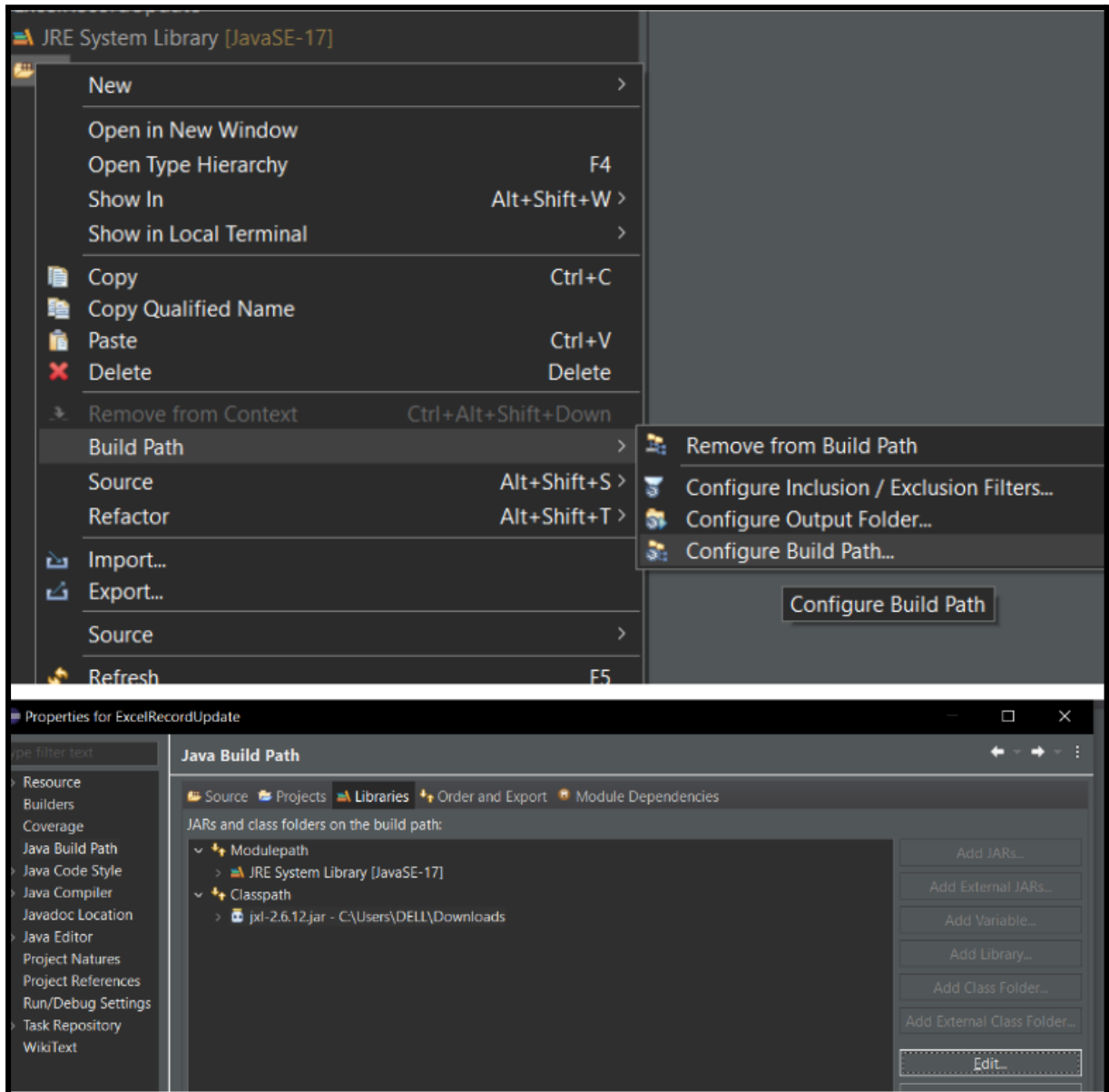
☒ Create module-info.java file

Module name:

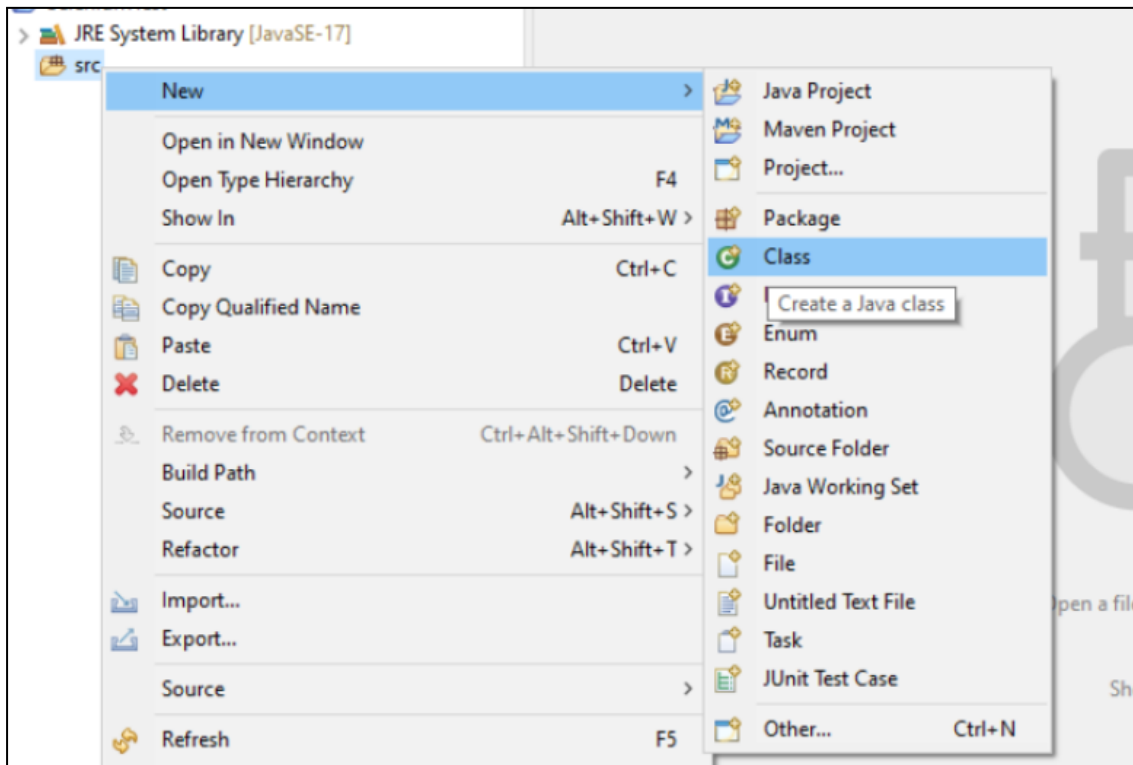
☐ Generate comments

Add the jxl.jar file using build classpath:

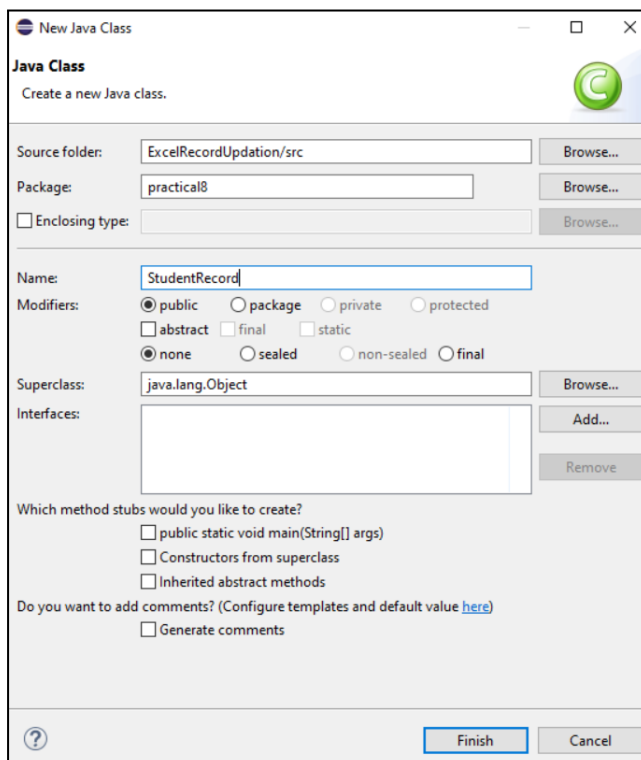
- src > Right click > Configure Build Path > Libraries > Classpath > Add External JARs



Create a Java Class: Right click on src > New > Class



Java class details



ECLIPSE JAVA CODE:

```
StudentRecord.java
package practical8;
import java.io.File;
import java.io.IOException;
import java.util.Locale;
import jxl.CellView;
import jxl.Workbook;
import jxl.WorkbookSettings;
import jxl.format.UnderlineStyle;
import jxl.write.Formula;
import jxl.write.Label;
import jxl.write.Number;
import jxl.write.WritableCellFormat;
import jxl.write.WritableFont;
import jxl.write.WritableWorkbook;
import jxl.write.WritableSheet;
import jxl.write.WriteException;
import jxl.write.biff.RowsExceededException;
public class StudentRecord {
    private WritableCellFormat timeBoldUnderline;
    private WritableCellFormat times;
    private String inputFile;
    public void setOutputFile(String inputFile) {
        this.inputFile = inputFile;
    }
    public void write() throws IOException, WriteException {
        File file = new File(inputFile);
        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);
        createLabel(excelSheet);
        createContent(excelSheet);
        workbook.write();
        workbook.close();
    }
    private void createLabel(WritableSheet sheet) throws WriteException {
        WritableFont time10pt = new WritableFont(WritableFont.TIMES, 10);
        times = new WritableCellFormat(time10pt);
        times.setWrap(true);
```

```

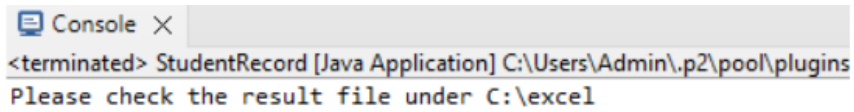
WritableFont times10ptBoldUnderline = new
WritableFont(WritableFont.TIMES , 10 , WritableFont.BOLD , false,
UnderlineStyle.SINGLE);
timeBoldUnderline = new WritableCellFormat(times10ptBoldUnderline);
timeBoldUnderline.setWrap(true);
CellView cv = new CellView();
cv.setFormat(times);
cv.setFormat(timeBoldUnderline);
addCaption(sheet, 0,0, "Student Name");
addCaption(sheet, 1,0, "Subject 1");
addCaption(sheet, 2,0, "Subject 2");
addCaption(sheet, 3,0, "Subject 3");
}
private void createContent(WritableSheet sheet) throws WriteException,
RowsExceededException {
for (int i = 1; i<10;i++) {
addLabel(sheet, 0, i, "Student " +i);
addNumber(sheet, 1, i, ((i*i)+10));
addNumber(sheet, 2, i, ((i*i)+4));
addNumber(sheet, 3, i, ((i*i)+3));
}
}
private void addCaption(WritableSheet sheet, int column, int row, String s)
throws RowsExceededException, WriteException {
Label label;
label = new Label(column, row, s, timeBoldUnderline);
sheet.addCell(label);
}
private void addNumber(WritableSheet sheet, int column, int row, Integer
integer)
throws RowsExceededException, WriteException {
Number number;
number = new Number(column, row, integer, times);
sheet.addCell(number);
}
private void addLabel(WritableSheet sheet, int column, int row, String s)
throws RowsExceededException, WriteException {
Label label;
label = new Label(column, row, s, times);
sheet.addCell(label);
}
public static void main(String[] args) throws WriteException, IOException {
StudentRecord test = new StudentRecord();

```

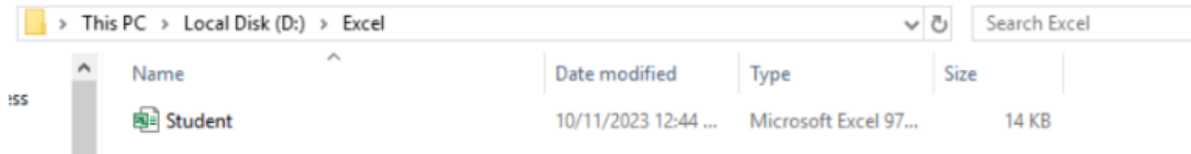
```
test.setOutputFile("E:\\Excel\\Student.xlsx");  
test.write();  
System.out.println("Please check the result file under C:\\excel");  
}  
}
```

OUTPUT:

i) eclipse



Console X
<terminated> StudentRecord [Java Application] C:\Users\Admin\.p2\pool\plugins
Please check the result file under C:\excel



ii) Student.xls (excel file)

Student Name					
	A	B	C	D	E
1	<u>Student Name</u>	<u>Subject 1</u>	<u>Subject 2</u>	<u>Subject 3</u>	
2	Student 1	11	5	4	
3	Student 2	14	8	7	
4	Student 3	19	13	12	
5	Student 4	26	20	19	
6	Student 5	35	29	28	
7	Student 6	46	40	39	
8	Student 7	59	53	52	
9	Student 8	74	68	67	
10	Student 9	91	85	84	
11					
12					
13					