Tanmay Pramod Mayekar

Roll No :- 68

Deccan Education Society's

Kirti M. Doongursee College of Arts, Science and Commerce

[NAAC Accredited: "A Grade"]

T.Y.B.Sc. [Computer Science]

Practical Journal

USCS503 : Software Testing and Quality Assurance

Seat Number [ ]

Department of Computer Science

Tanmay Pramod Mayekar
Roll No :- 68

Department of Computer Science

Deccan Education Society's

Kirti M. Doongursee College of Arts, Science and Commerce

[NAAC Accredited: "A Grade"]

C E R T I F I C A T E

This is to certify that Mr. / Miss ____

of T.Y.B.Sc. (Computer Science) with Seat No.________ has completed ___

Practicals of Paper- **USCS503 : Software Testing and Quality Assurance**

under my supervision in this College during the year **2025-2026**.

Lecturer-In-Charge **H.O.D.**

Department of Computer Science

Date: / /2022 Date:

Examined by: Remarks:

Date: ____________________

Tanmay Pramod Mayekar
Roll No :- 68

USCS503 : Software Testing and Quality Assurance

Sr. No.	Date	Title	Page No.	Signature
1		**Practical No. 1:** Installation of Selenium IDE		
2		**Practical No. 2:** Conduct a test suite for any two web sites.		
3		**Practical No. 3:** **I.** Opening chrome from Eclipse using selenium **II.** Search Selenium tutorial in google through code. **III.** Login into Facebook Account using Selenium.		
4		**Practical No. 4:** Write and test the calculator program using JUnit.		
5		**Practical No. 5:** Write and test a program to update 6 students records into Excel file (using TestNG)		
6		**Practical No. 6:** Write & test a program to select the number of students who have scored more than 60 in any one subject (or all subjects) using TestNG.		
7		**Practical No. 7:** Write and test a program to provide total numbers of objects present / available on the page.		
8		**Practical No. 8:** Write and test a program to print total numbers of single drop down list items present / available on the page and print selected item.		
9		**Practical No. 9:** Write and test a program to print total numbers of Multiple drop down list items present / available on the page and print selected item.		
10		**Practical No. 10:** Write and test a program to print total numbers of check boxes present / available on the page and print selected check box.		

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 1:

What is Selenium?

Selenium is a **browser automation framework** that enables testers to perform functional testing of web applications. It automates user actions like clicking buttons, entering text, navigating between pages, and validating outputs. Unlike manual testing, where a human interacts with the application, Selenium executes automated scripts to perform the same actions quickly and repeatedly.

Uses of Selenium:

- Functional Testing: Ensures that the functionality of web applications works as expected.
- Regression Testing: Automates repeated test cases to check that new changes do not break existing functionality.
- Cross-Browser Testing: Verifies compatibility across different browsers (Chrome, Firefox, Edge, Safari).
- Data-Driven Testing: Supports integration with external data sources (Excel, CSV, databases).
- Integration with CI/CD: Works with Jenkins, Maven, GitHub Actions, and other DevOps tools for continuous testing.
- Supports Multiple Languages: Test scripts can be written in Java, Python, C#, Ruby, etc.
- Supports Parallel Testing: Saves time by executing multiple tests simultaneously.

History of Selenium:

The history of Selenium is quite interesting and evolved over time:

- 2004 (Selenium Core): Selenium was originally developed by *Jason Huggins* at ThoughtWorks while testing an internal web application. It was a JavaScript program that could control browser actions.

- 2005 (Selenium RC – Remote Control): Paul Hammant extended Selenium Core into Selenium RC, which allowed testers to write automated tests in multiple programming languages.
- 2006 (Selenium IDE): Shinya Kasatani developed Selenium IDE, a Firefox plugin that enabled record-and-playback testing without writing code.
- 2008 (Selenium WebDriver): Simon Stewart introduced WebDriver, a more powerful and flexible tool that directly communicated with browsers using their native APIs.
- 2011 (Selenium 2.0): Selenium RC and WebDriver were merged into a single tool, known as Selenium 2.0.
- 2016 (Selenium 3.0): Selenium RC was officially deprecated, and WebDriver became the core component.
- 2021 (Selenium 4.0): Released with advanced features like support for W3C WebDriver standard, relative locators, Chrome DevTools Protocol (CDP), and better IDE support

Conclusion:

Selenium has grown from a simple JavaScript testing framework into a powerful, industry-standard automation tool. Its open-source nature, cross-platform compatibility, and integration with modern software development practices make it the preferred choice for testers and developers worldwide. With the release of Selenium 4, it continues to evolve and support advanced testing needs in modern web applications.

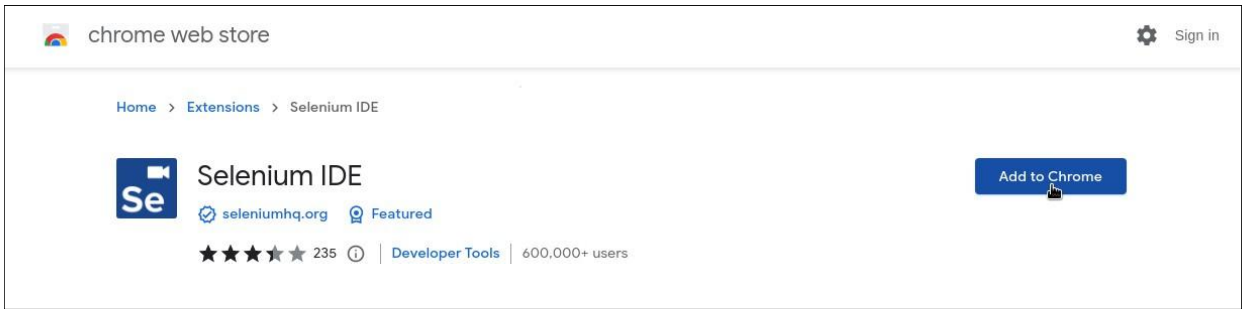
Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 1:

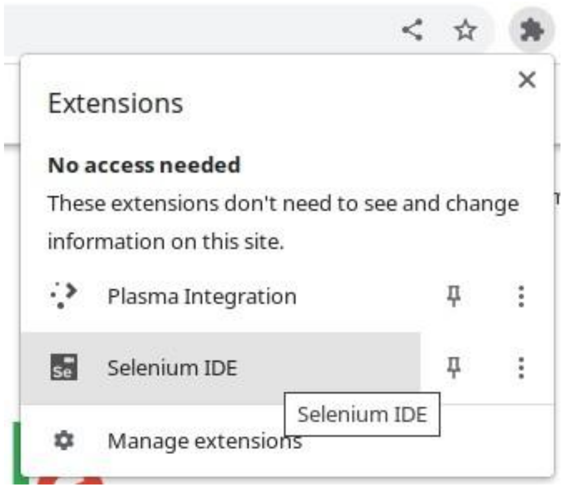
Installation of Selenium IDE:

NOTE: In this practical I will be using chrome browser

1. Open Chrome Browser and type selenium.dev
2. Scroll down you will find Selenium IDE click on read more and click on CHROME DOWNLOAD.
3. This will launch Chrome Web Store click on Add to Chrome.



4. Once done adding Selenium IDE extensions click on icon to launch it on top right corner.



Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 2:

Theory:

A test suite is a collection of test cases grouped logically to test an application's functionality. Instead of running each test individually, suites execute all tests together, saving time and ensuring coverage.

In Selenium IDE:

- A test suite is created by grouping test cases.
- Each test case can represent a particular scenario (e.g., login, search, form submission).
- Running the suite ensures functionality across the application in one go.

Importance:

- Detects defects early by validating multiple functionalities.
- Ensures consistent results in regression testing.
- Helps in continuous testing within CI/CD pipelines.

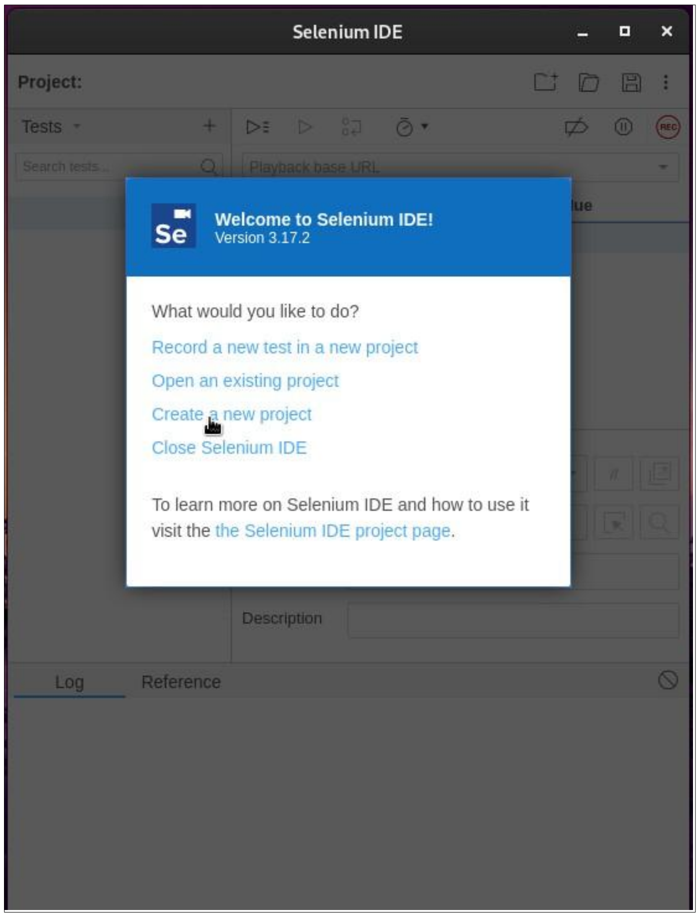
Running a suite on two websites demonstrates Selenium's reusability and flexibility across different domains.

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 2:

Conduct a test suite for any two web sites.:

1. Select "Create a new project" in Selenium IDE



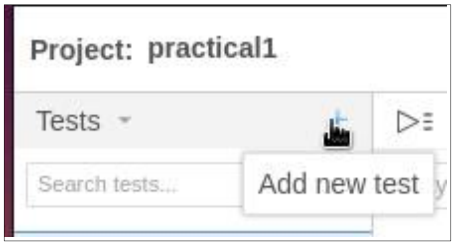
Welcome to Selenium IDE!
Version 3.17.2

What would you like to do?

Record a new test in a new project
Open an existing project
Create a new project
Close Selenium IDE

To learn more on Selenium IDE and how to use it visit the the Selenium IDE project page.

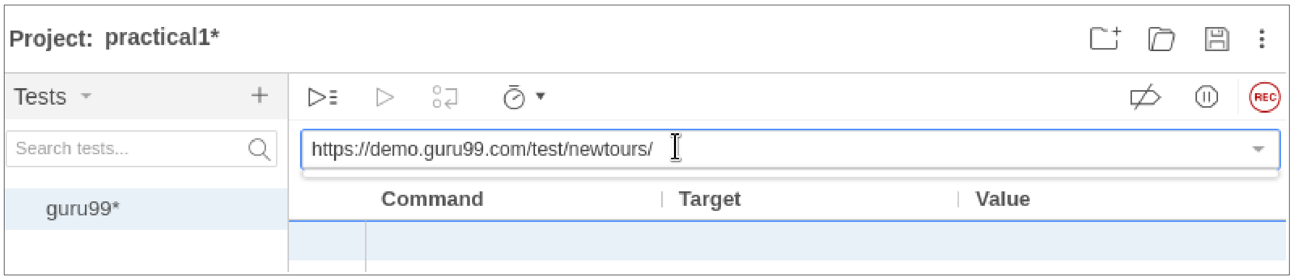
2. Type appropriate Project Name and click on ok
3. Create New Test Case by clicking on the + button.



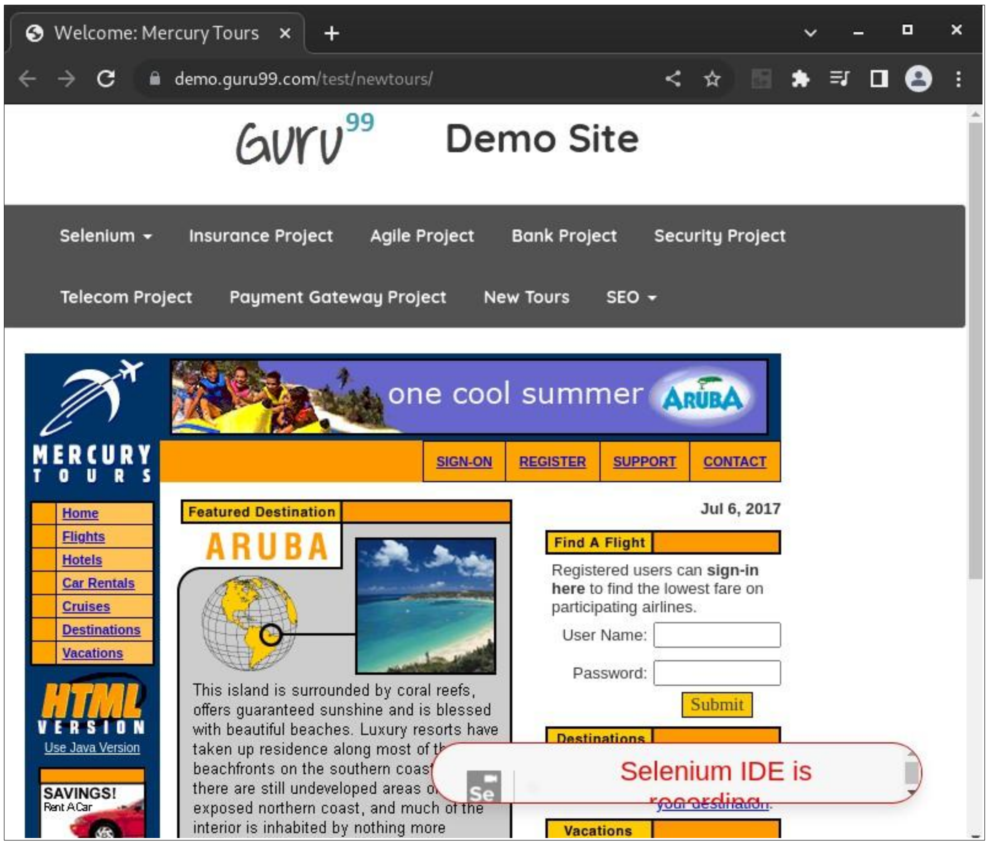
Tanmay Pramod Mayekar
Roll No :- 68

Test Case 1: https://demo.guru99.com/test/newtours/

1. set https://demo.guru99.com/test/newtours/ as Playback base URL



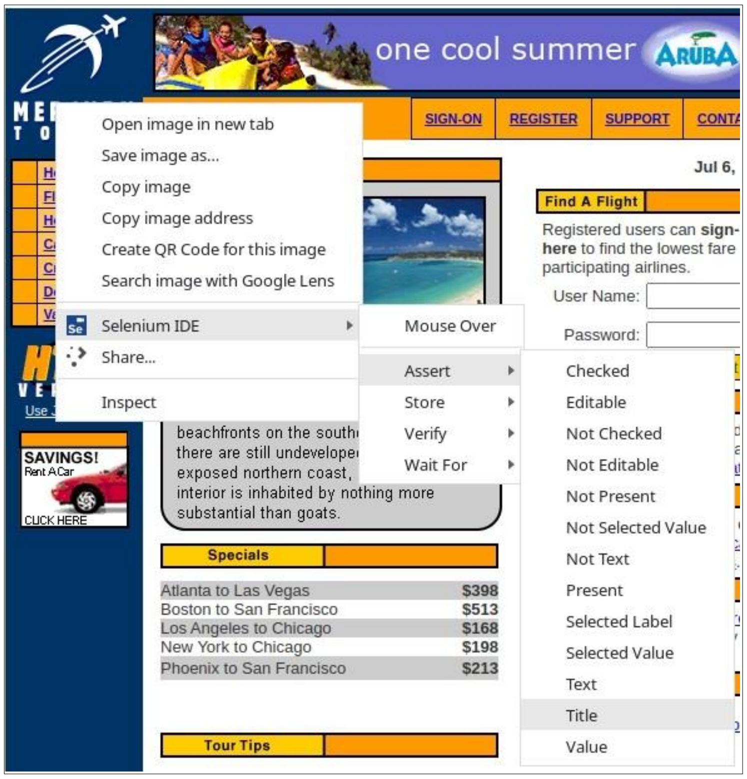
1. Then click on REC (red button) on top right corner.
2. This will Launch Chrome browser with site Entered in Base URL. With Selenium IDE is recording message on mid bottom right corner.

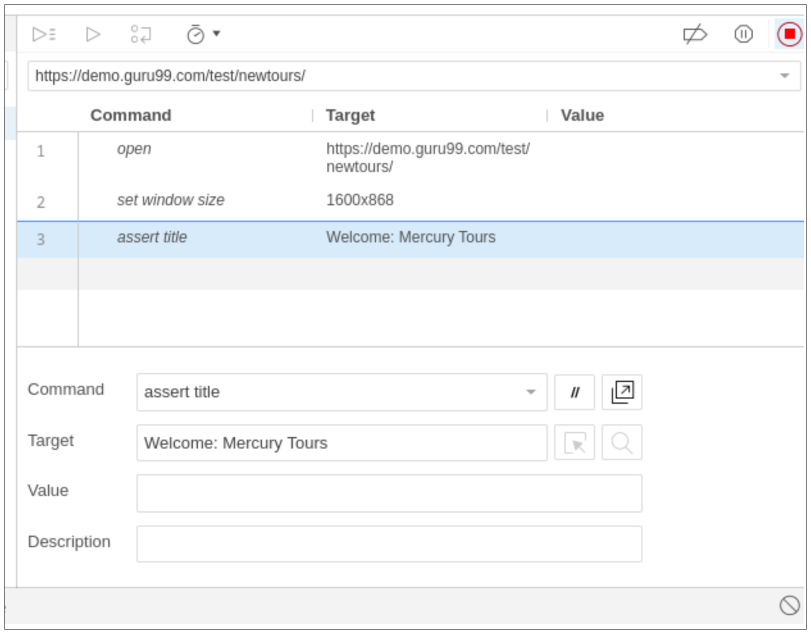


This island is surrounded by coral reefs, offers guaranteed sunshine and is blessed with beautiful beaches. Luxury resorts have taken up residence along most of the beachfronts on the southern coast there are still undeveloped areas on the exposed northern coast, and much of the interior is inhabited by nothing more

Registered users can sign-in here to find the lowest fare on participating airlines.

3. Right click on any blank space within the page, like on the Mercury Tours logo on the upper left corner. This will bring up the Selenium IDE context menu.
4. Then hover on Assert and then select Title on selenium IDE new entries will be added as **assert title Welcome: Mercury Tours**. This is the command that makes sure that the page title is correct.

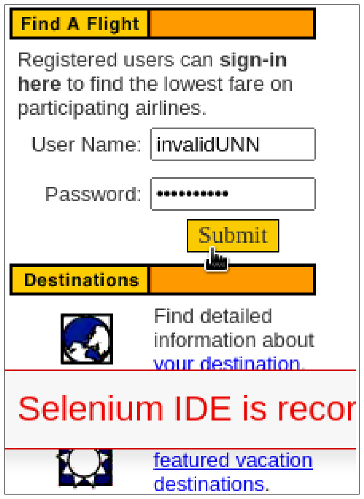




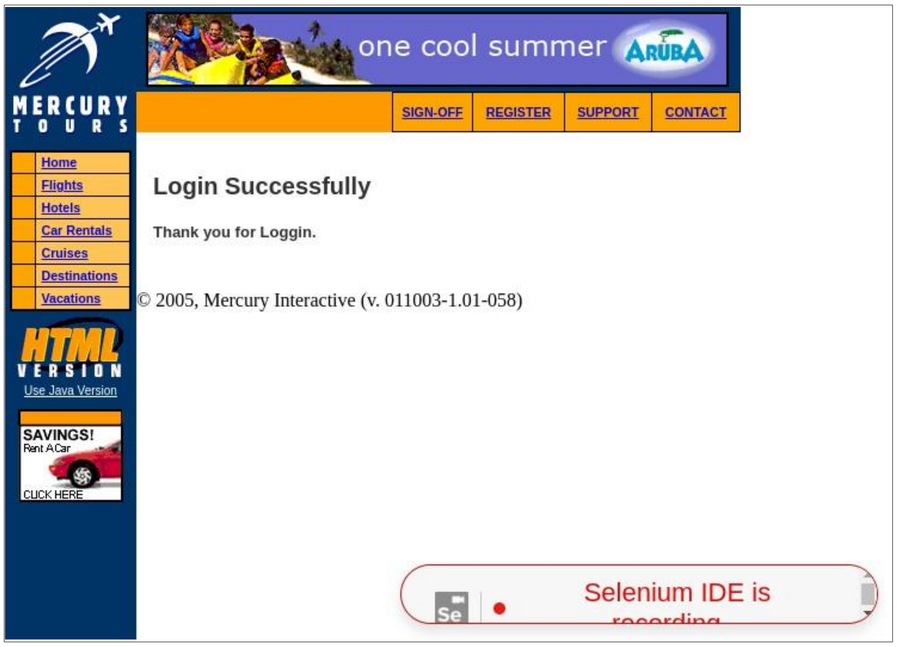
	Command	Target	Value
1	open	https://demo.guru99.com/test/newtours/	
2	set window size	1600x868	
3	assert title	Welcome: Mercury Tours	

Command: assert title
Target: Welcome: Mercury Tours
Value:
Description:

5. Now, In the "User Name" text box of Mercury Tours, type an invalid username, "invalidUNN". In the "Password" text box, type an invalid password, "invalidPWD"


Find A Flight
Registered users can sign-in here to find the lowest fare on participating airlines.
User Name: invalidUNN
Password: ••••••••••
Submit
Destinations
Find detailed information about your destination.
Selenium IDE is recor
featured vacation destinations.

6. Click on "Submit" button. Chrome browser should take you to another page.



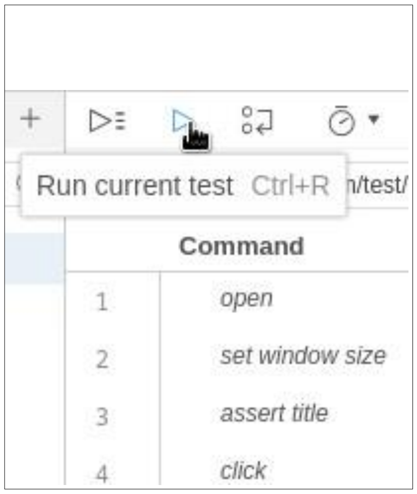
Login Successfully

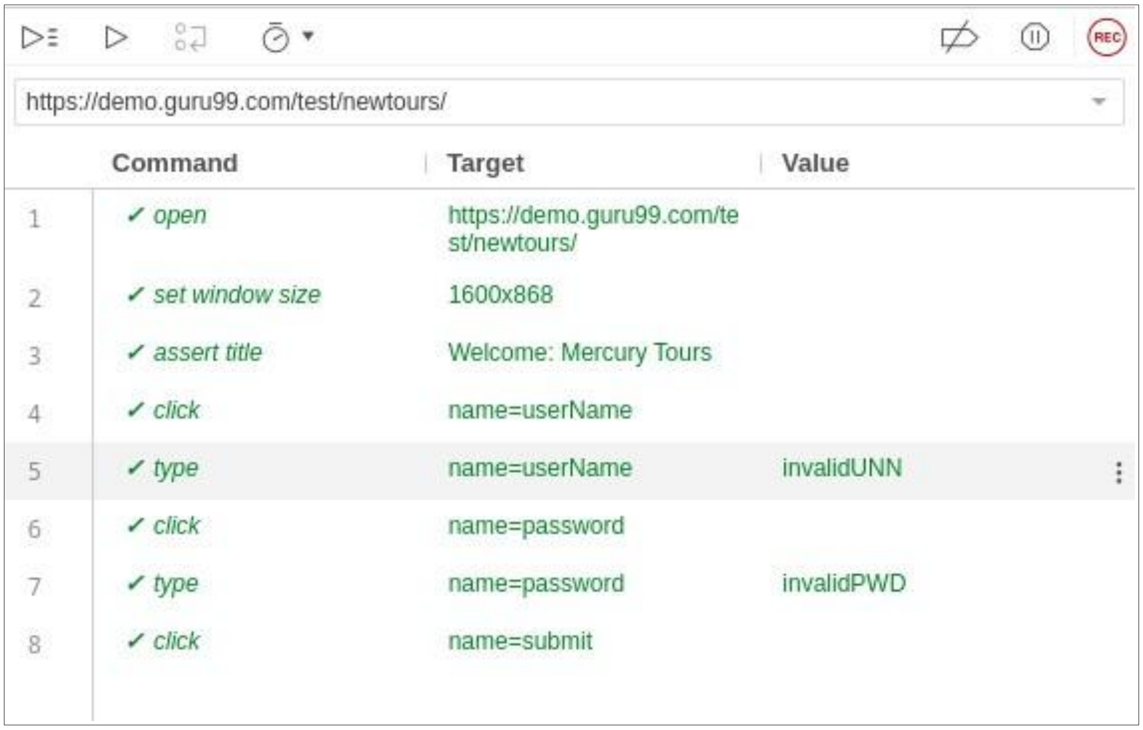
Thank you for Loggin.

© 2005, Mercury Interactive (v. 011003-1.01-058)

7. Click on the Record button to stop recording.
8. Now, click on the Playback button to execute the whole script. Selenium IDE should be able to replicate everything properly.

Tanmay Pramod Mayekar
Roll No :- 68





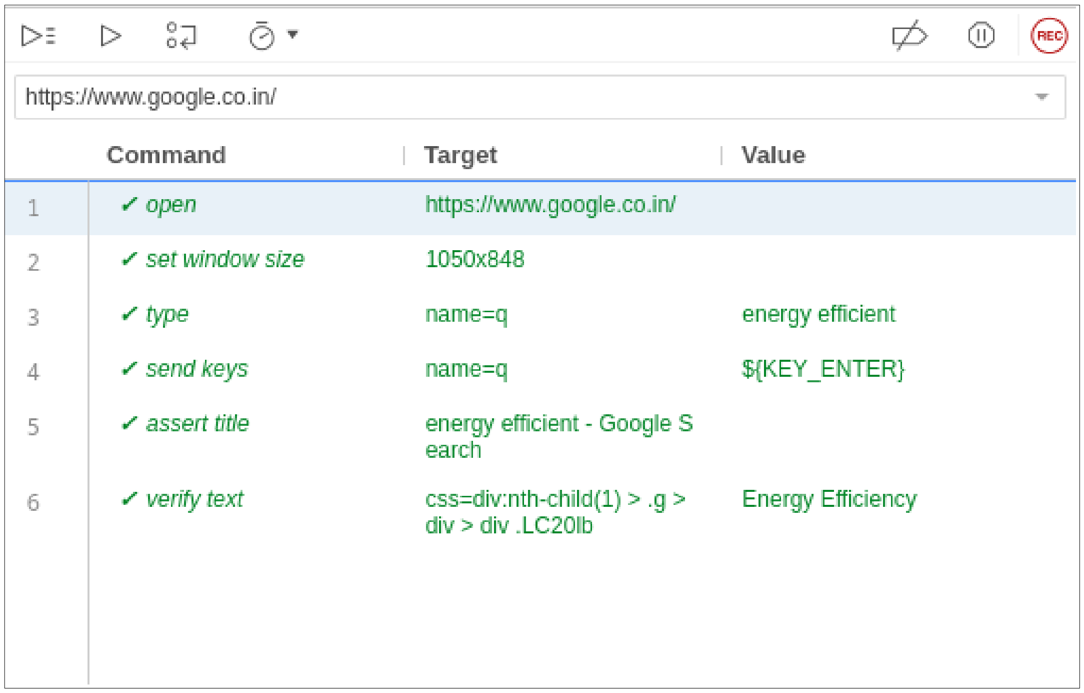
https://demo.guru99.com/test/newtours/

	Command	Target	Value
1	✓ open	https://demo.guru99.com/test/newtours/	
2	✓ set window size	1600x868	
3	✓ assert title	Welcome: Mercury Tours	
4	✓ click	name=userName	
5	✓ type	name=userName	invalidUNN
6	✓ click	name=password	
7	✓ type	name=password	invalidPWD
8	✓ click	name=submit	

Test Case 2: https://www.google.co.in/

1. Type "energy efficient" in the Google Search Input Box.
2. Right click outside on an empty spot and select assets > Title.
3. Right click on Search Button verify the text present as "energy efficient"
4. Save the Test case

Output:



https://www.google.co.in/

	Command	Target	Value
1	✓ open	https://www.google.co.in/	
2	✓ set window size	1050x848	
3	✓ type	name=q	energy efficient
4	✓ send keys	name=q	${KEY_ENTER}
5	✓ assert title	energy efficient - Google S earch	
6	✓ verify text	css=div:nth-child(1) > .g > div > div .LC20lb	Energy Efficiency

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 3

Theory:

This practical demonstrates Selenium WebDriver's real-world usage:

1. Opening Chrome from Eclipse:
   - Requires ChromeDriver and Selenium libraries.
   - WebDriver acts as a bridge between test scripts and the browser.
   - Demonstrates environment setup and execution.
2. Searching Selenium tutorial on Google:
   - Uses locators like name, id, or xpath to identify search fields.
   - Enters query and submits form.
   - Demonstrates navigation and interaction.
3. Login into Facebook account:
   - Automates input handling (username, password).
   - Demonstrates form validation and event handling (button click).

This practical shows how automation can simulate user behavior across websites and verify workflows like searching and logging in.

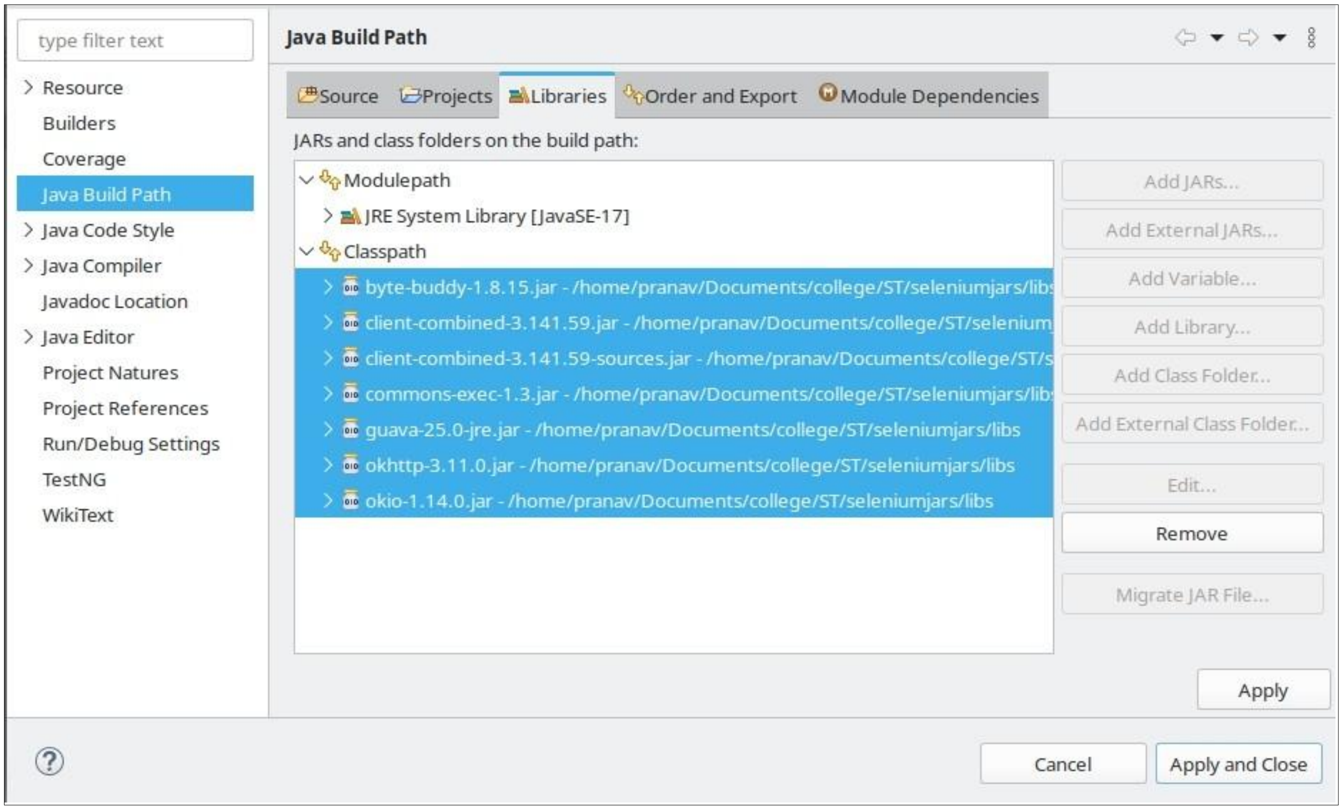
Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 3

3.1. Opening chrome from Eclipse using selenium

NOTE: We are going to use Eclipse IDE for this practical.

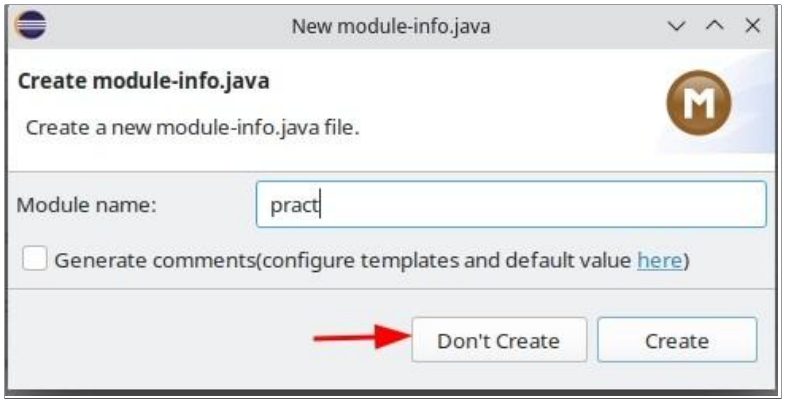
1. Download latest Eclipse IDE (java development) from official site.
2. Install Eclipse IDE and launch it, set its workspace directory.
3. Create a new Java Project, set appropriate project name and click on next



JARs and class folders on the build path:

```
Modulepath
  JRE System Library [JavaSE-17]
Classpath
  byte-buddy-1.8.15.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  client-combined-3.141.59.jar - /home/pranav/Documents/college/ST/selenium
  client-combined-3.141.59-sources.jar - /home/pranav/Documents/college/ST/s
  commons-exec-1.3.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  guava-25.0-jre.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  okhttp-3.11.0.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  okio-1.14.0.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
```

4. Now navigate to Libraries and click on Add External JARs
5. Add these JARs and click on Finish and a window will popup saying do you want to create module? Just click on don't create



Create module-info.java

Create a new module-info.java file.

Module name: pract

☐ Generate comments(configure templates and default value here)

Now we are ready to code just create the new class file by right clicking on the “src” folder and set appropriate class name like I am using “LaunchChrome” as my class name. Make sure you have latest ChromeDriver.

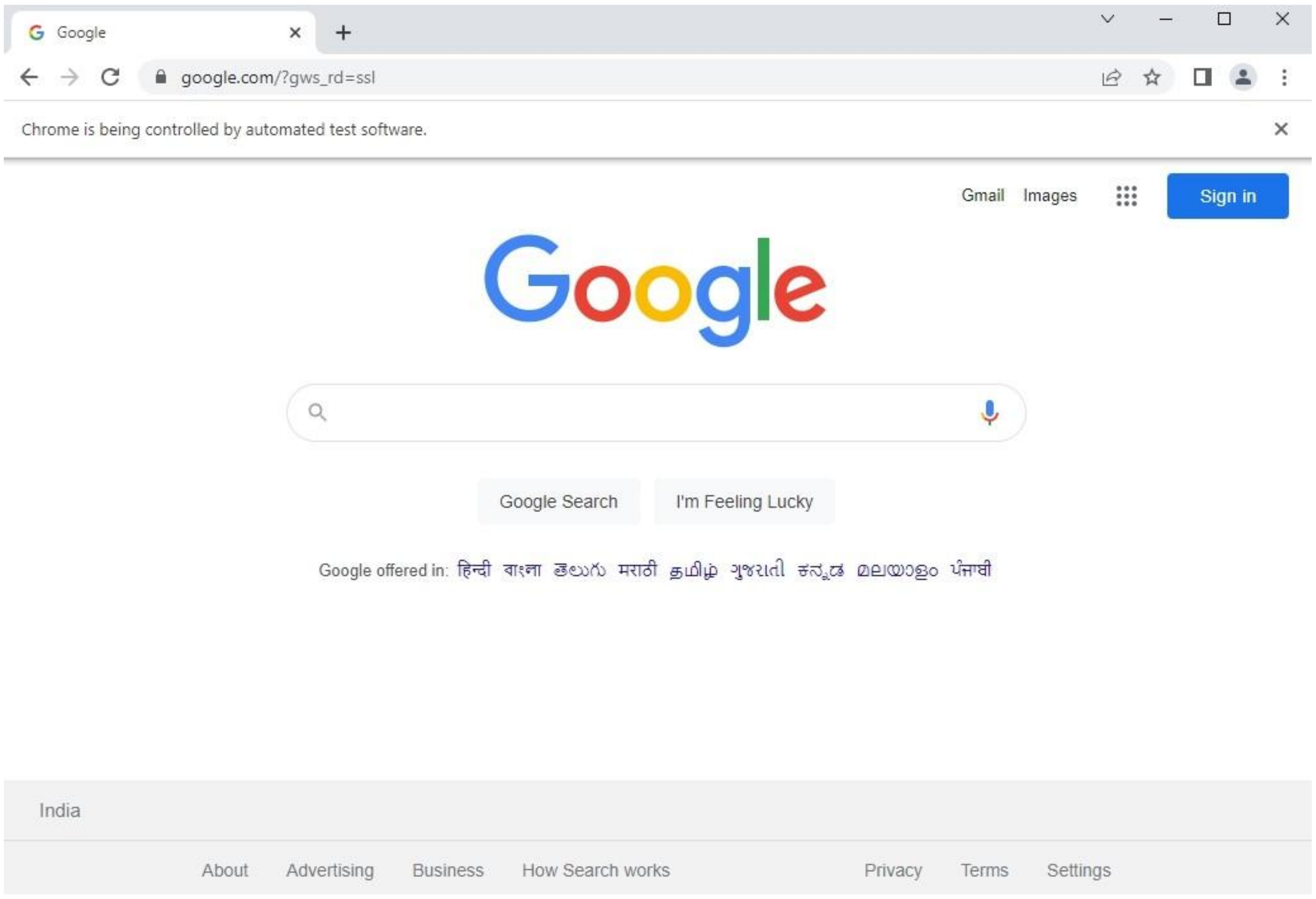
Code:

```
package TYCS68;

import org.openqa.selenium.WebDriver;

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

class LaunchChrome {

        public static WebDriver driver;

        public static void main(String[] args) {

                System.out.println("Launching Chrome...");


        System.setProperty("webdriver.chrome.driver","D:\\chromedriver.exe1\\chromedriver-
win64\\chromedriver.exe");

                driver=new ChromeDriver();

                driver.navigate().to("http://google.com");



        }
```

}

Now, lets run the program by right clicking on class name click on Run As > 1 Java Application

output:



Chrome is being controlled by automated test software.

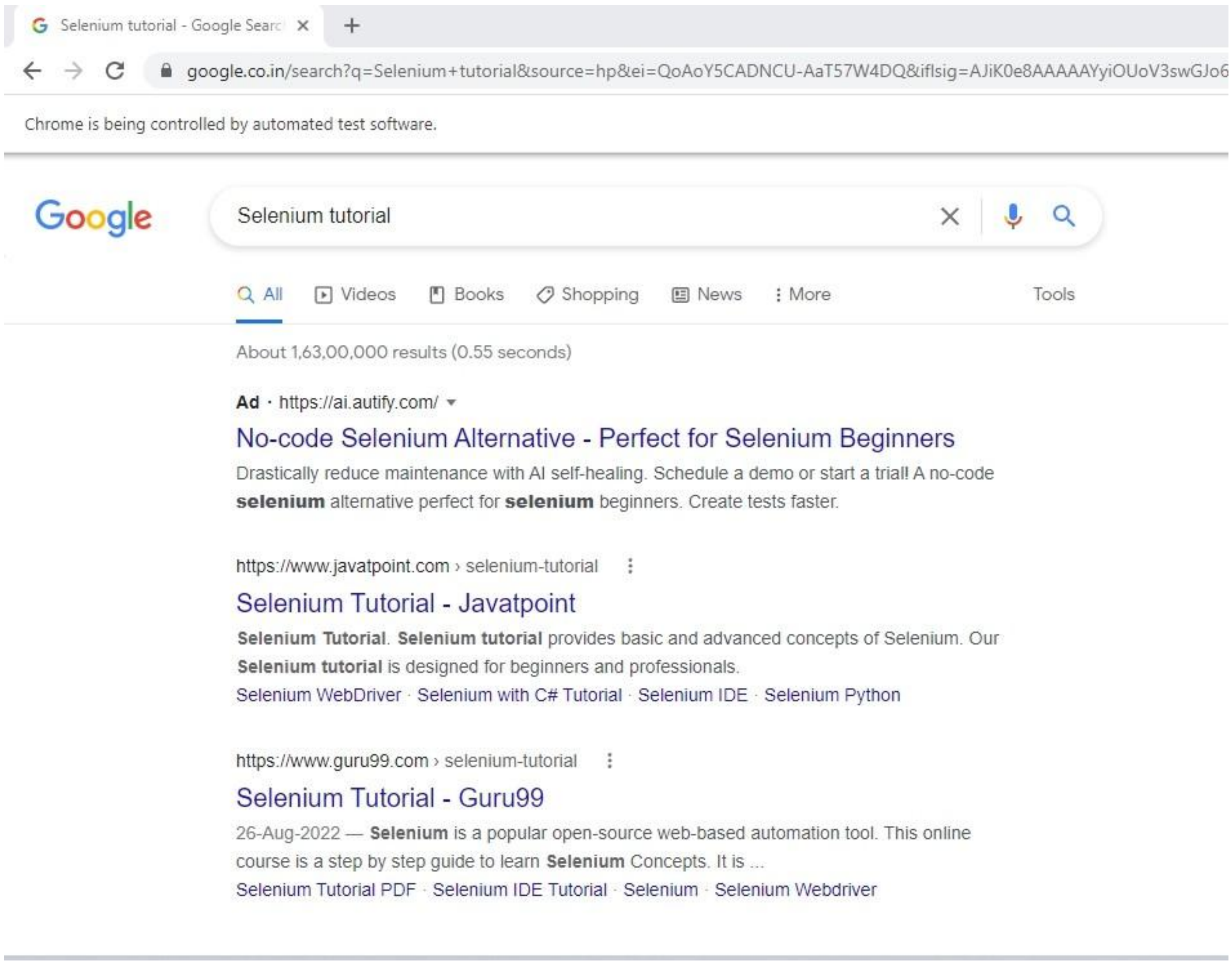
3.2. write a code to search Selenium tutorial in google through code.

Code:

```
package TYCS68;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver; public
class search {
        public static void main(String[] args) {

        System.setProperty("webdriver.chrome.driver","D:\\chromedriver.exe1\\chromedriver-
win64\\chromedriver.exe");
                WebDriver  driver  =  new  ChromeDriver();
                driver.navigate().to("http://google.co.in/");
                driver.manage().window().maximize();
                WebElement        element=driver.findElement(By.name("q"));
                element.sendKeys("junit");
        element.submit();
        }
}
```

Tanmay Pramod Mayekar
Roll No :- 68

Output:


Selenium tutorial - Google Searc
google.co.in/search?q=Selenium+tutorial&source=hp&ei=QoAoY5CADNCU-AaT57W4DQ&iflsig=AJiK0e8AAAAAYyiOUoV3swGJo6
Chrome is being controlled by automated test software.
Google
Selenium tutorial
All
Videos
Books
Shopping
News
More
Tools

About 1,63,00,000 results (0.55 seconds)

Ad · https://ai.autify.com/

No-code Selenium Alternative - Perfect for Selenium Beginners

Drastically reduce maintenance with AI self-healing. Schedule a demo or start a trial! A no-code **selenium** alternative perfect for **selenium** beginners. Create tests faster.

https://www.javatpoint.com › selenium-tutorial

Selenium Tutorial - Javatpoint

Selenium Tutorial. **Selenium tutorial** provides basic and advanced concepts of Selenium. Our **Selenium tutorial** is designed for beginners and professionals.

Selenium WebDriver · Selenium with C# Tutorial · Selenium IDE · Selenium Python

https://www.guru99.com › selenium-tutorial

Selenium Tutorial - Guru99

26-Aug-2022 — **Selenium** is a popular open-source web-based automation tool. This online course is a step by step guide to learn **Selenium** Concepts. It is ...

Selenium Tutorial PDF · Selenium IDE Tutorial · Selenium · Selenium Webdriver

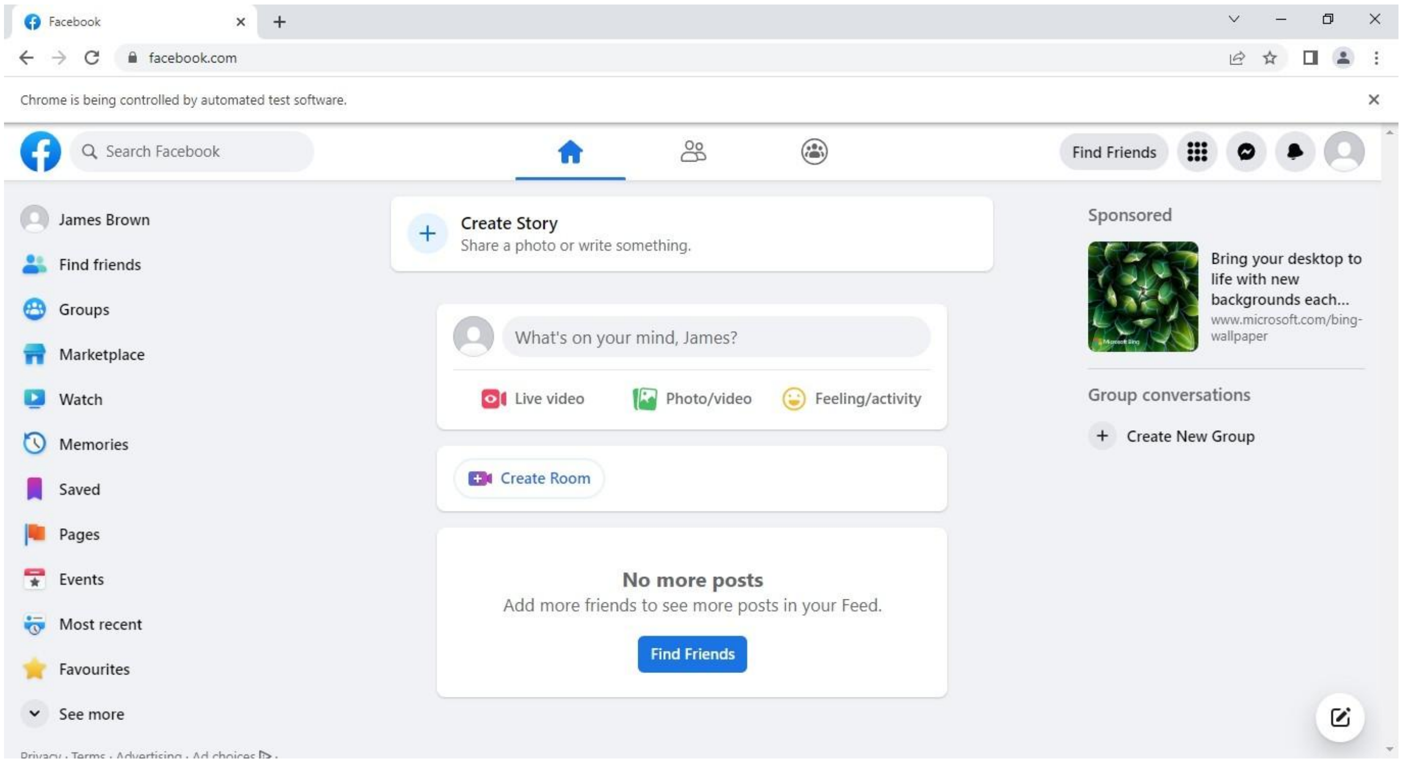
3.3. write and test a program to login into Facebook Account use Selenium.

Code:

```
package TYCS6;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver; public
class LoginIntoFB {
        public static void main(String[] args) {

        System.setProperty("webdriver.chrome.driver","D:\\chromedriver.exe1\\chromedriver-
win64\\chromedriver.exe");
                WebDriver driver = new ChromeDriver(); driver.navigate().to("http://www.facebook.com/");
                driver.manage().window().maximize();
                WebElement elementEmail=driver.findElement(By.name("email"));
                elementEmail.sendKeys("student.unity@proton.me");  WebElement
                elementpwd=driver.findElement(By.name("pass"));
                elementpwd.sendKeys("Unity@1234");
                WebElement elementSubmit=driver.findElement(By.name("login"));
                elementSubmit.submit();


        }


}
```

Tanmay Pramod Mayekar
Roll No :- 68

Output:


Facebook
facebook.com
Chrome is being controlled by automated test software.
Search Facebook
Find Friends
James Brown
Find friends
Groups
Marketplace
Watch
Memories
Saved
Pages
Events
Most recent
Favourites
See more
Privacy · Terms · Advertising · Ad choices
Create Story
Share a photo or write something.
What's on your mind, James?
Live video
Photo/video
Feeling/activity
Create Room
No more posts
Add more friends to see more posts in your Feed.
Find Friends
Sponsored
Bring your desktop to life with new backgrounds each...
www.microsoft.com/bing-wallpaper
Group conversations
Create New Group

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 4

Theory:

JUnit is one of the most popular Java unit testing frameworks. Unit testing is the process of testing individual components or methods of a program in isolation.

In this practical:

- A calculator program (basic arithmetic operations) is tested using JUnit test cases.
- Assertions like assertEquals() are used to compare expected vs. actual results.
- Test annotations (@Test, @Before, @After) provide structure and lifecycle to the tests.

Advantages:

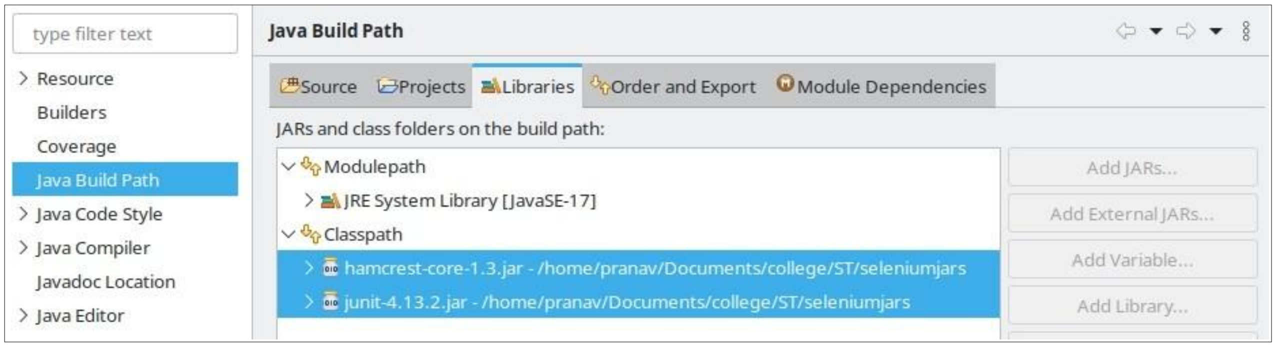
- Ensures correctness of small code units.
- Improves code reliability and maintainability.
- Facilitates Test-Driven Development (TDD).

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 4

4.1 Write and test the calculator program using JUnit.

1. Download JUnit via https://search.maven.org/search?q=g:junit%20AND%20a:junit
2. Download hamcrest-core 1.3 via https://search.maven.org/artifact/org.hamcrest/hamcrest-core/1.3/jar
3. Move these 2 JARs to your preferred location or move them to the Selenium JARs folder.
4. Now launch the Eclipse with your workspace and
5. Import these JARs by right click on your project (in Package Explorer) > Build Path > Configure Build Path.
6. The window will appear
7. Click on Classpath and click on Add External JARs Button and add those 2 JARs.
8. Once done click on Apply and Close.



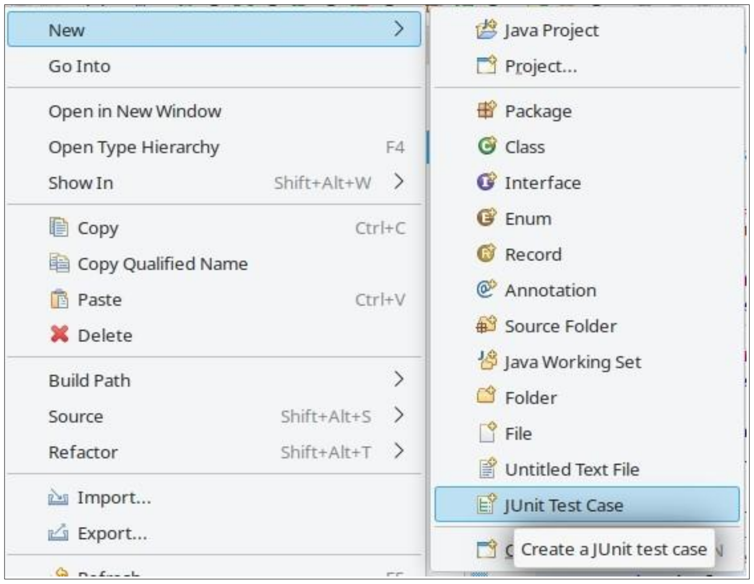
JARs and class folders on the build path:

```
Modulepath
  JRE System Library [JavaSE-17]
Classpath
  hamcrest-core-1.3.jar - /home/pranav/Documents/college/ST/seleniumjars
  junit-4.13.2.jar - /home/pranav/Documents/college/ST/seleniumjars
```

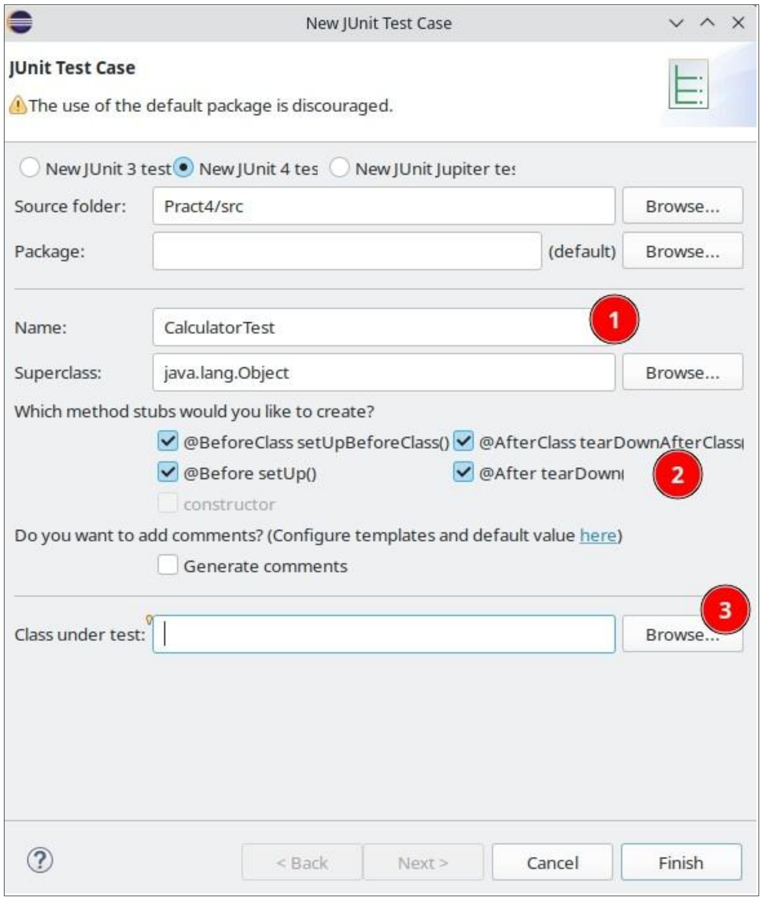
1. Create new class file named as Calculator, and write the following code:

```
package TYCS68;

public class calculator {

        public int add(int a,int b) {

                return a+b;
        }
        public int subtract(int a,int b) { return
                a-b;
        }
        public int mul(int a,int b) {
                return a*b;
        }
        public int div(int a,int b) {
                return a/b;
        }
}
```

2. Now, Create JUnit Test Case,



3. In class under test field click on browse.



The use of the default package is discouraged.

New JUnit 3 test ( ) [✓ New JUnit 4 tes] New JUnit Jupiter te:

Source folder: Pract4/src

Package: (default)

Name: CalculatorTest

Superclass: java.lang.Object

Which method stubs would you like to create?

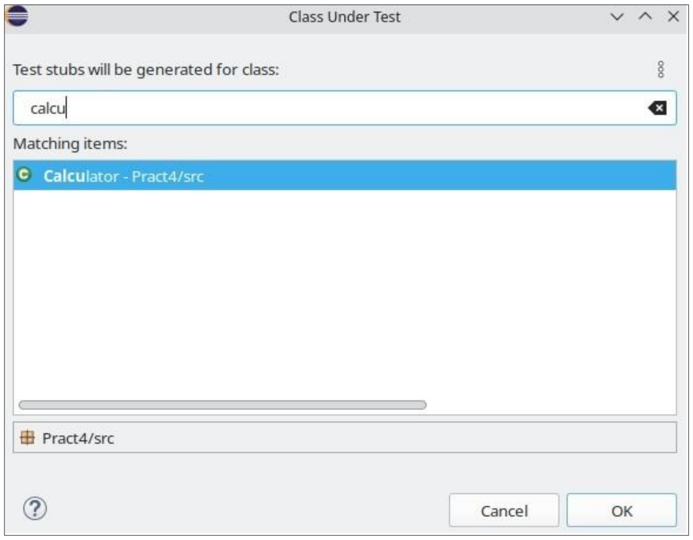
- [✓ @BeforeClass setUpBeforeClass()] [✓ @AfterClass tearDownAfterClass(]
- [✓ @Before setUp()] [✓ @After tearDown(]
- constructor

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

- Generate comments

Class under test:

4. Enter Calculator and select appropriate class from the search.



Test stubs will be generated for class:

calcu

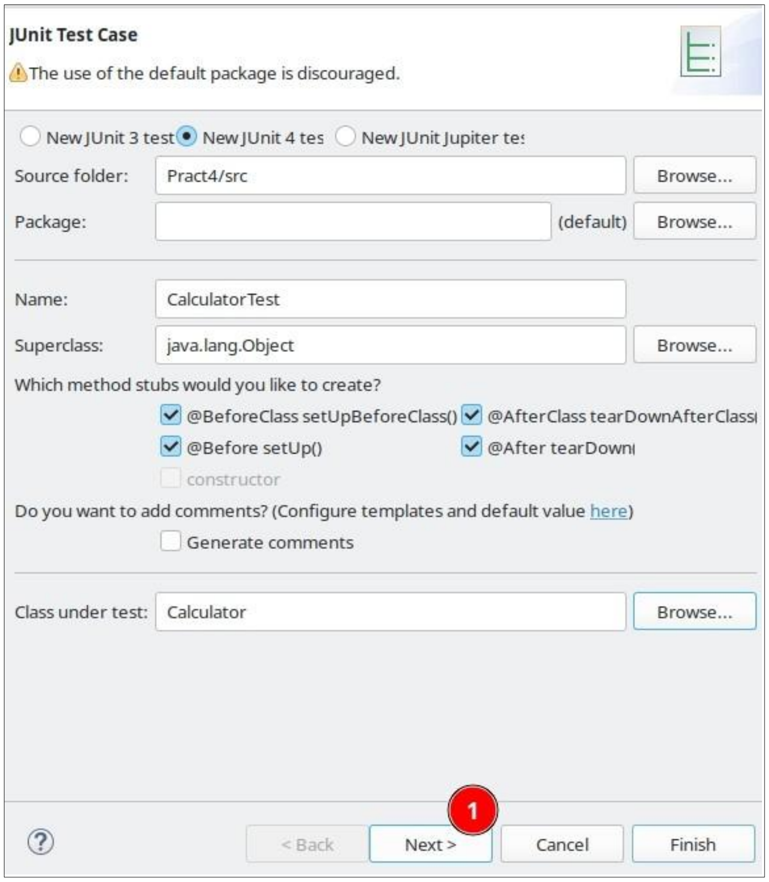
Matching items:

Calculator - Pract4/src

Pract4/src

5. Click on OK.

6. Click on Next



JUnit Test Case

⚠The use of the default package is discouraged.

○ New JUnit 3 test [● New JUnit 4 tes] ○ New JUnit Jupiter te:

Source folder: Pract4/src

Package: (default)

Name: CalculatorTest

Superclass: java.lang.Object

Which method stubs would you like to create?

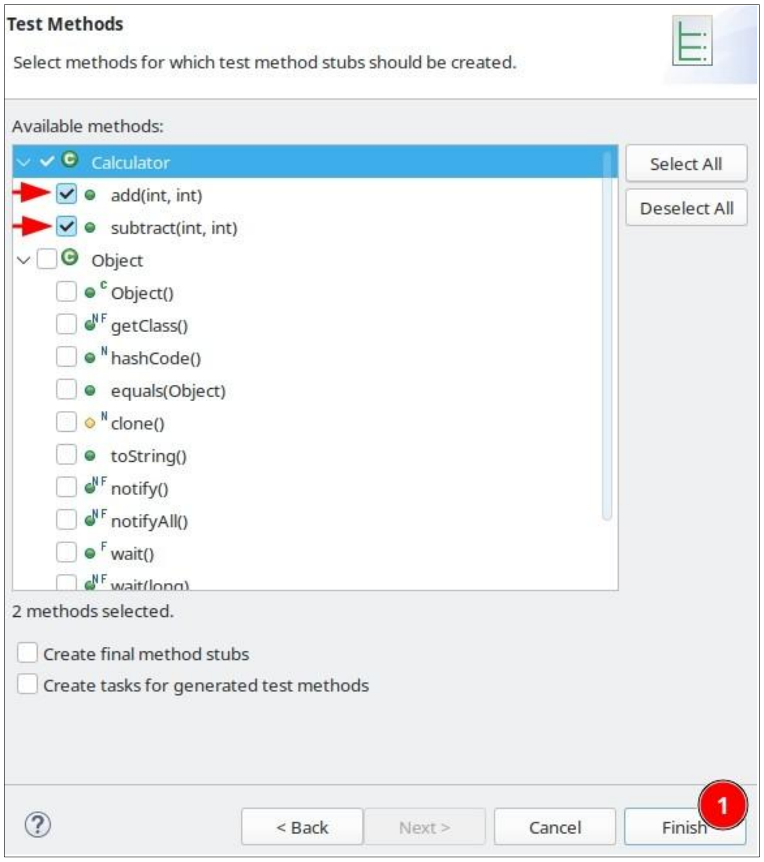
- [✓] @BeforeClass setUpBeforeClass()
- [✓] @AfterClass tearDownAfterClass
- [✓] @Before setUp()
- [✓] @After tearDown
- [ ] constructor

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

- [ ] Generate comments

Class under test: Calculator

7. Select the 2 Methods of the Calculator, and click on Finish


Test Methods

Select methods for which test method stubs should be created.

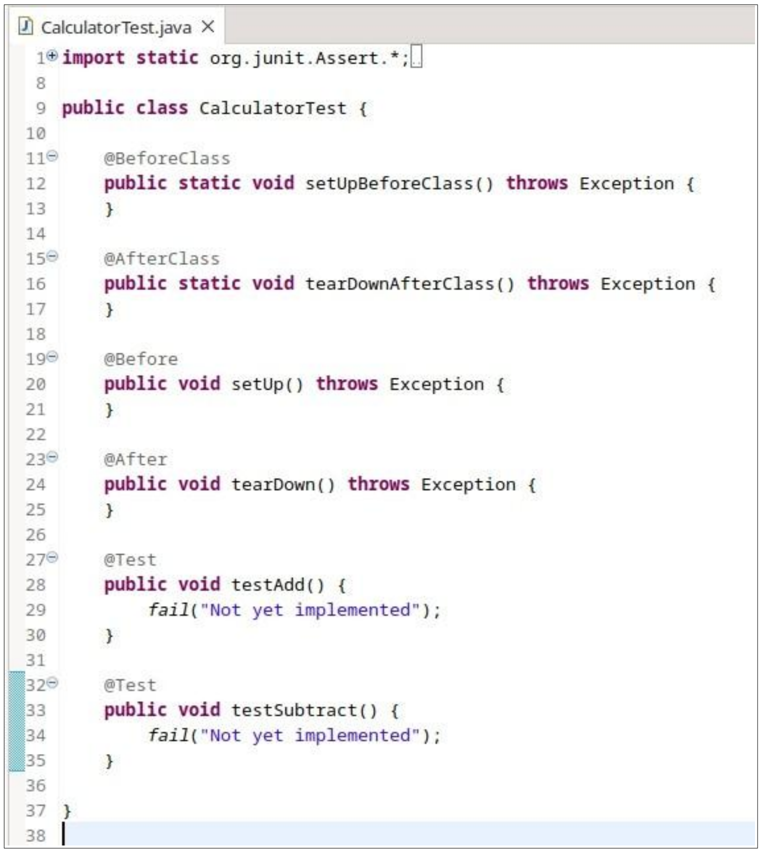
Available methods:

- [✓] Calculator
  - [✓] add(int, int)
  - [✓] subtract(int, int)
- [ ] Object
  - [ ] Object()
  - [ ] getClass()
  - [ ] hashCode()
  - [ ] equals(Object)
  - [ ] clone()
  - [ ] toString()
  - [ ] notify()
  - [ ] notifyAll()
  - [ ] wait()
  - [ ] wait(long)

2 methods selected.

- [ ] Create final method stubs
- [ ] Create tasks for generated test methods

8. Template will generate. We will be writing the code in **@Test** block:



```
CalculatorTest.java
import static org.junit.Assert.*;

public class CalculatorTest {

    @BeforeClass
    public static void setUpBeforeClass() throws Exception {
    }

    @AfterClass
    public static void tearDownAfterClass() throws Exception {
    }

    @Before
    public void setUp() throws Exception {
    }

    @After
    public void tearDown() throws Exception {
    }

    @Test
    public void testAdd() {
        fail("Not yet implemented");
    }

    @Test
    public void testSubtract() {
        fail("Not yet implemented");
    }

}
```

9. Write the code given below in place of fail("Not yet implemented");

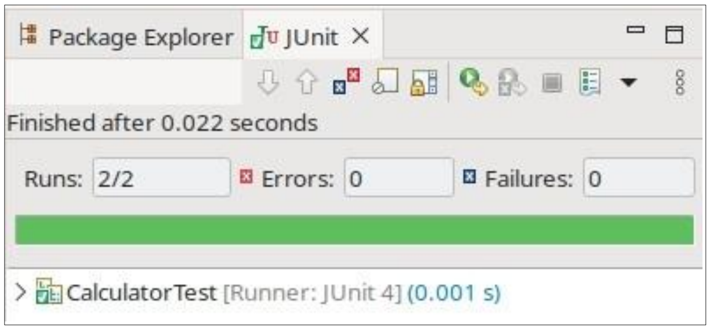
```
@Test

        public void testAdd() {

                int a=12;

                int b=32;

                int expected=44;

                calculator c=new calculator(); int

                actual=c.add(a, b);

                assertEquals(expected,actual);


        }
```

```
@Test

public void testSubtract() { int

        a=12;

        int b=10;

        int expected=2;

        calculator c=new calculator(); int

        actual=c.subtract(a, b);

        assertEquals(expected,actual);

}

@Test

public void testmul() {

        int a=12;

        int b=10;

        int expected=120;

        calculator c=new calculator(); int

        actual=c.mul(a, b);

        assertEquals(expected,actual);

}

@Test

public void testdiv() { int

        a=10;

        int b=5;

        int expected=2;

        calculator c=new calculator(); int

        actual=c.div(a, b);
```

```
                    assertEquals(expected,actual);

            }
```

10. Right click > Run as > 1 JUnit File.

Output:



```
Finished after 0.022 seconds
Runs: 2/2    Errors: 0    Failures: 0
> CalculatorTest [Runner: JUnit 4] (0.001 s)
```

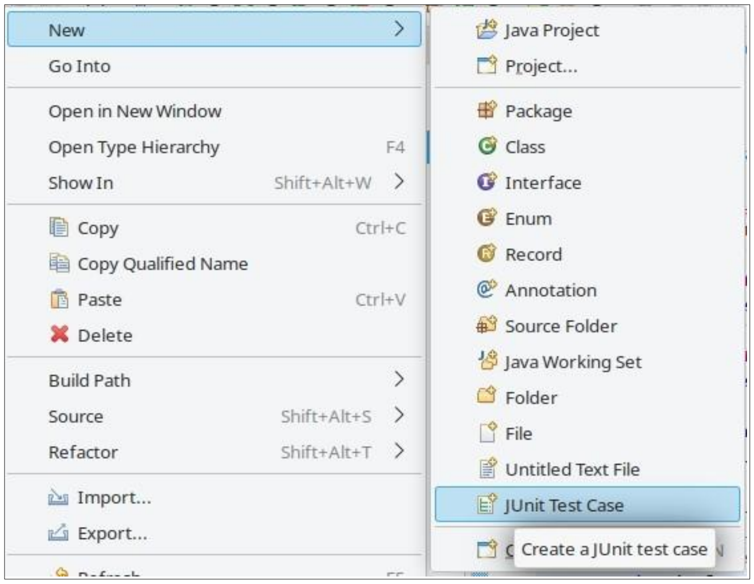
Tanmay Pramod Mayekar
Roll No :- 68

4.2 Write and test the program to calculate the area of triangle using JUnit.

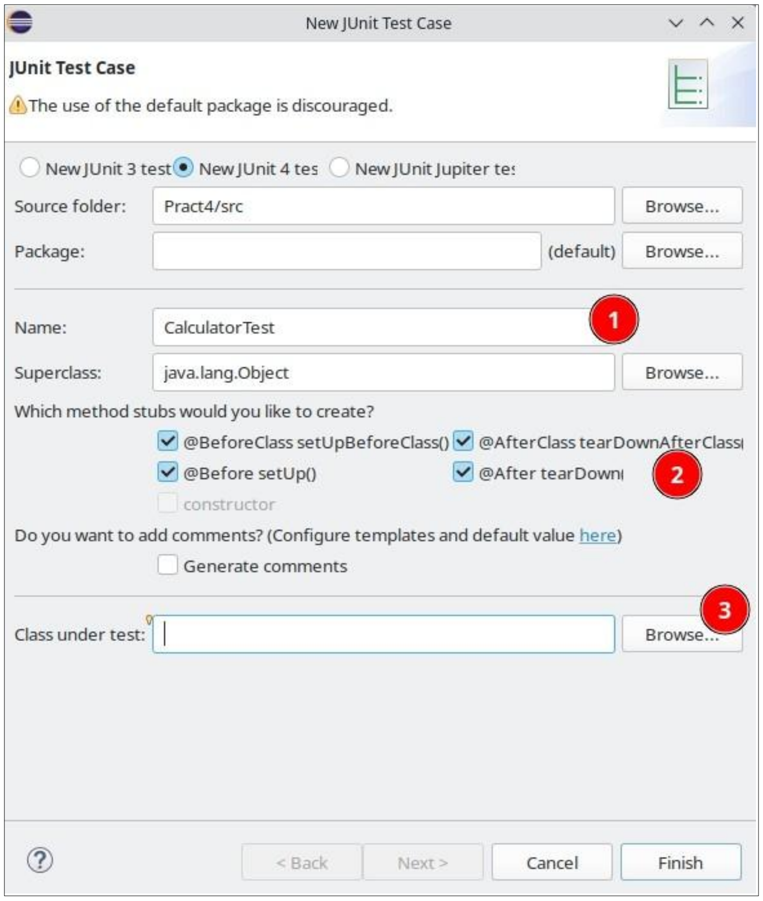
1. Create new class file named as Area , and write the following code:

```
package tycs68;

public class Area{
   public int triangleArea(int b, int h){
}
}
```

2. Now, Create JUnit Test Case,



3. In class under test field click on browse.


New JUnit Test Case

JUnit Test Case

The use of the default package is discouraged.

New JUnit 3 test [✓ New JUnit 4 tes] New JUnit Jupiter tes

Source folder: Pract4/src

Package: (default)

Name: CalculatorTest

Superclass: java.lang.Object

Which method stubs would you like to create?

[✓] @BeforeClass setUpBeforeClass() [✓] @AfterClass tearDownAfterClass(

[✓] @Before setUp() [✓] @After tearDown(

[ ] constructor

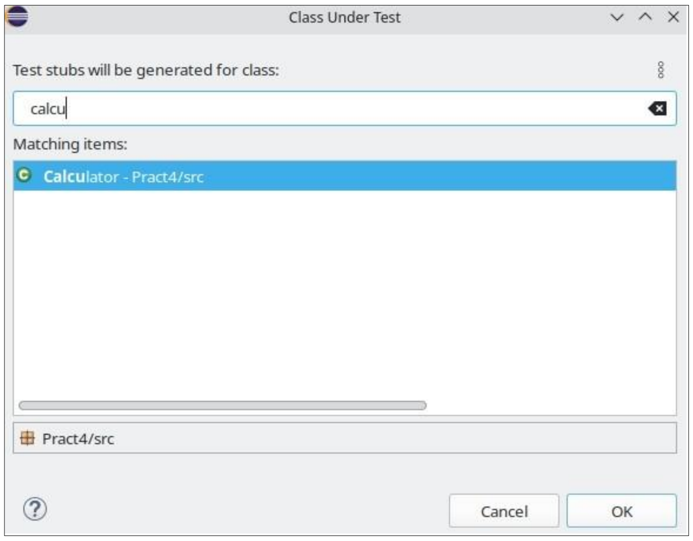
Do you want to add comments? (Configure templates and default value here)

[ ] Generate comments

Class under test:

4. Enter Area and select appropriate class from the search.

Tanmay Pramod Mayekar
Roll No :- 68



Test stubs will be generated for class:

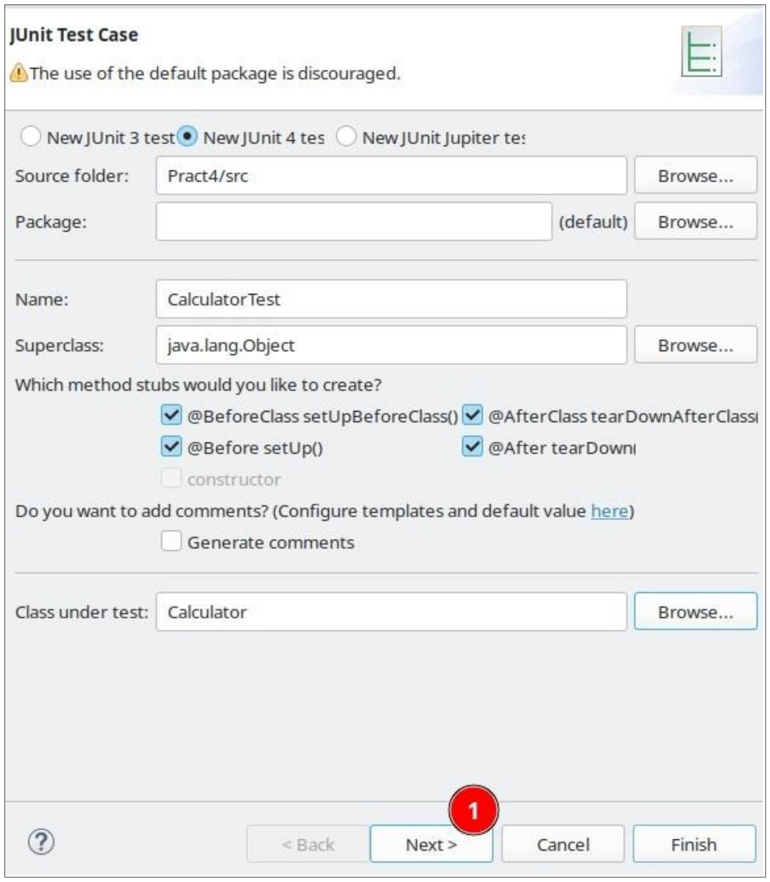
calcu

Matching items:

Calculator - Pract4/src

Pract4/src

5. Click on OK.
6. Click on Next



JUnit Test Case

The use of the default package is discouraged.

New JUnit 3 test [✓ New JUnit 4 tes] New JUnit Jupiter te

Source folder: Pract4/src

Package: (default)

Name: CalculatorTest

Superclass: java.lang.Object

Which method stubs would you like to create?

✓ @BeforeClass setUpBeforeClass() ✓ @AfterClass tearDownAfterClass

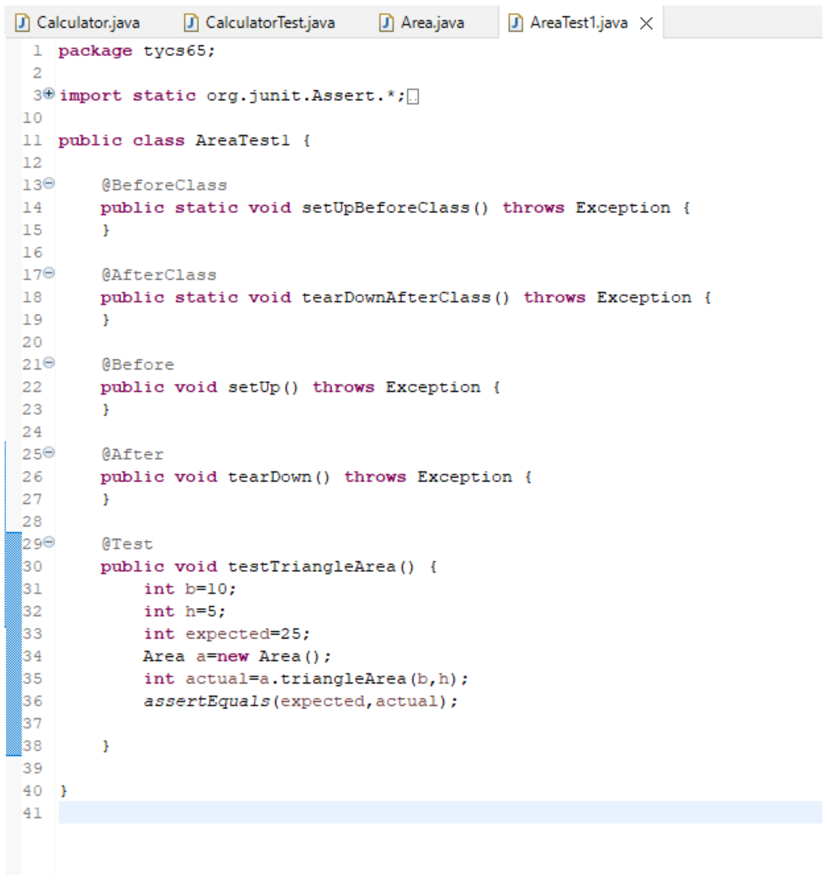
✓ @Before setUp() ✓ @After tearDown

constructor

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

Generate comments

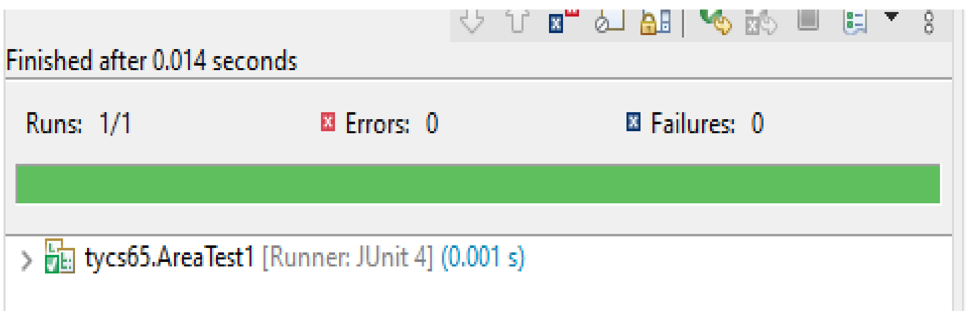
Class under test: Calculator



```java
package tycs65;

import static org.junit.Assert.*;

public class AreaTest1 {

    @BeforeClass
    public static void setUpBeforeClass() throws Exception {
    }

    @AfterClass
    public static void tearDownAfterClass() throws Exception {
    }

    @Before
    public void setUp() throws Exception {
    }

    @After
    public void tearDown() throws Exception {
    }

    @Test
    public void testTriangleArea() {
        int b=10;
        int h=5;
        int expected=25;
        Area a=new Area();
        int actual=a.triangleArea(b,h);
        assertEquals(expected,actual);

    }

}
```

Tanmay Pramod Mayekar
Roll No :- 68

Output:


Finished after 0.014 seconds

Runs: 1/1 Errors: 0 Failures: 0

tycs65.AreaTest1 [Runner: JUnit 4] (0.001 s)

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 5

Theory:

TestNG (Next Generation) is an advanced testing framework that overcomes limitations of JUnit. It supports data-driven testing, parallel execution, and detailed reporting.

In this practical:

- Six students' records are updated into an Excel file using Apache POI (a library for handling MS Office files).
- TestNG integrates test execution with external data storage.

Key points:

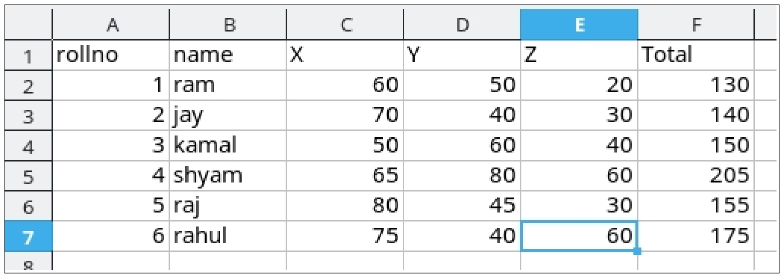
- Demonstrates parameterization with data providers.
- Ensures data persistence through Excel files.
- Useful in scenarios where test data comes from external sources (common in enterprise testing).

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 5

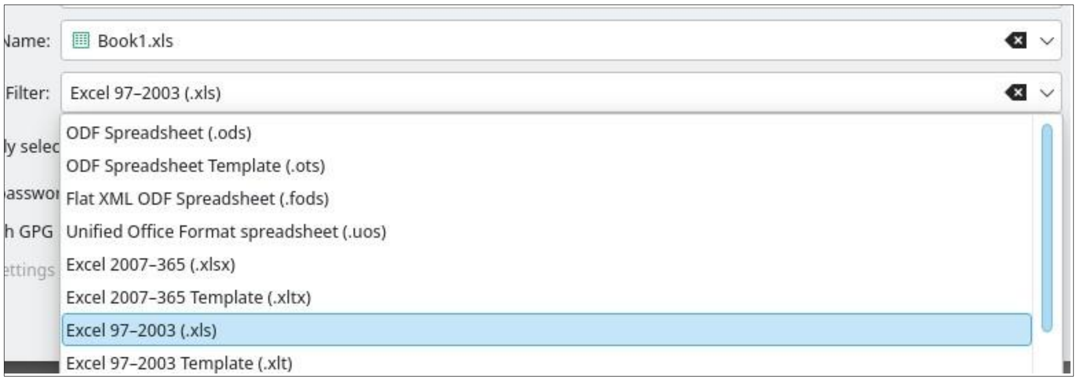
Write and test a program to update 6 students records into table into Excel file (using TestNG)

1. Create new Excel Workbook and add the following data:



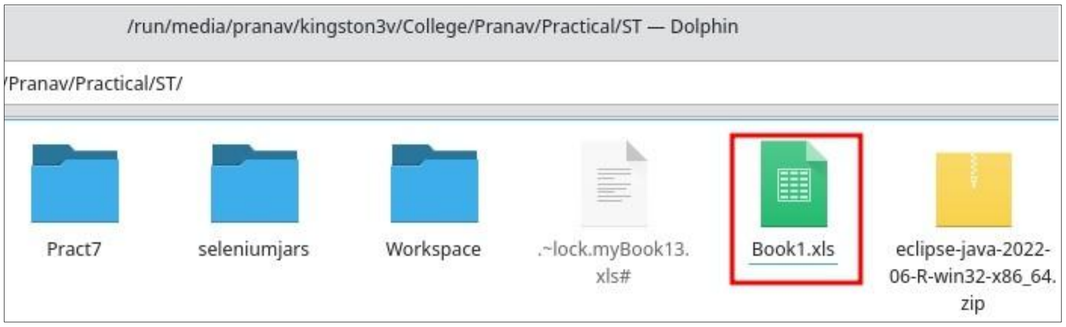
	A	B	C	D	E	F
1	rollno	name	X	Y	Z	Total
2	1	ram	60	50	20	130
3	2	jay	70	40	30	140
4	3	kamal	50	60	40	150
5	4	shyam	65	80	60	205
6	5	raj	80	45	30	155
7	6	rahul	75	40	60	175

2. And now save as



Name: Book1.xls
Filter: Excel 97–2003 (.xls)

- ODF Spreadsheet (.ods)
- ODF Spreadsheet Template (.ots)
- Flat XML ODF Spreadsheet (.fods)
- Unified Office Format spreadsheet (.uos)
- Excel 2007–365 (.xlsx)
- Excel 2007–365 Template (.xltx)
- Excel 97–2003 (.xls)
- Excel 97–2003 Template (.xlt)

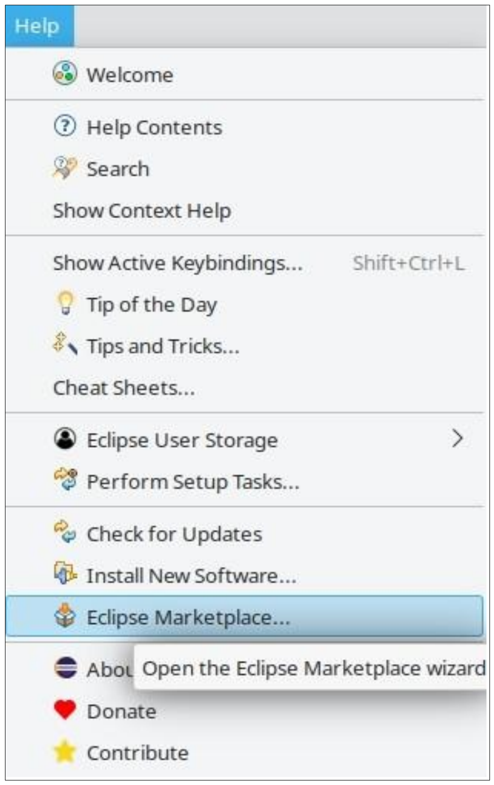


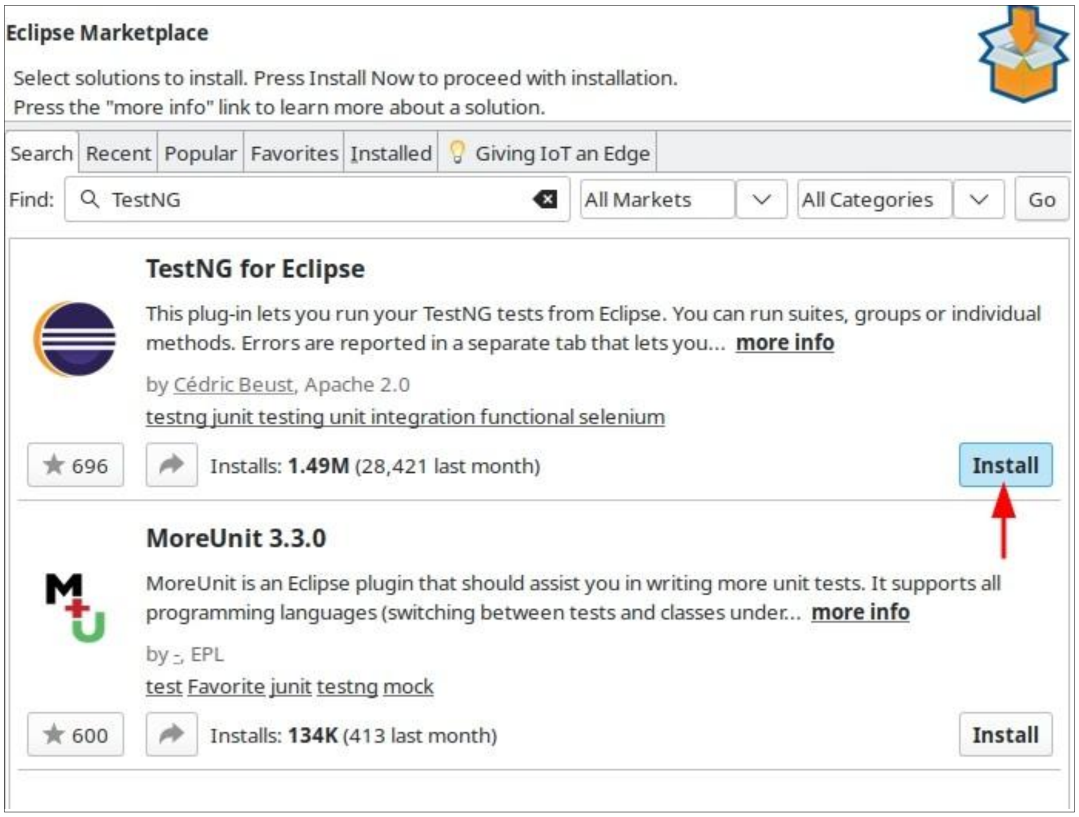
/run/media/pranav/kingston3v/College/Pranav/Practical/ST — Dolphin

/Pranav/Practical/ST/

2. Now start Eclipse and install TestNG by click on Help and select Eclipse Marketplace.

3. Search TestNG and hit Enter, and click on Install





Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.
Press the "more info" link to learn more about a solution.

Find: TestNG

TestNG for Eclipse

This plug-in lets you run your TestNG tests from Eclipse. You can run suites, groups or individual methods. Errors are reported in a separate tab that lets you... more info

by Cédric Beust, Apache 2.0

testng junit testing unit integration functional selenium

696 Installs: **1.49M** (28,421 last month) Install

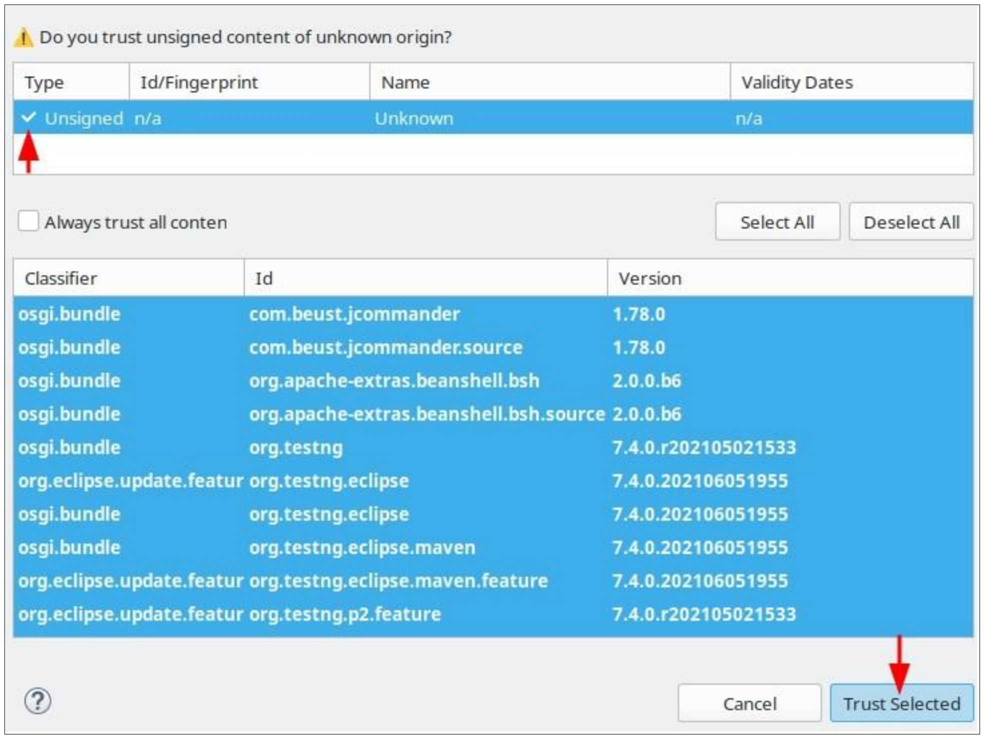
MoreUnit 3.3.0

MoreUnit is an Eclipse plugin that should assist you in writing more unit tests. It supports all programming languages (switching between tests and classes under... more info

by -, EPL

test Favorite junit testng mock

600 Installs: **134K** (413 last month) Install

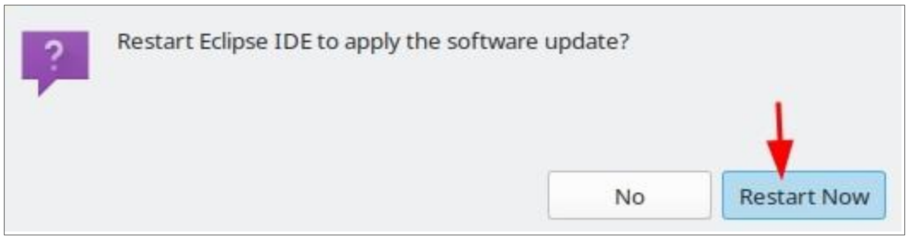


Do you trust unsigned content of unknown origin?

Type	Id/Fingerprint	Name	Validity Dates
✓ Unsigned	n/a	Unknown	n/a

Always trust all conten

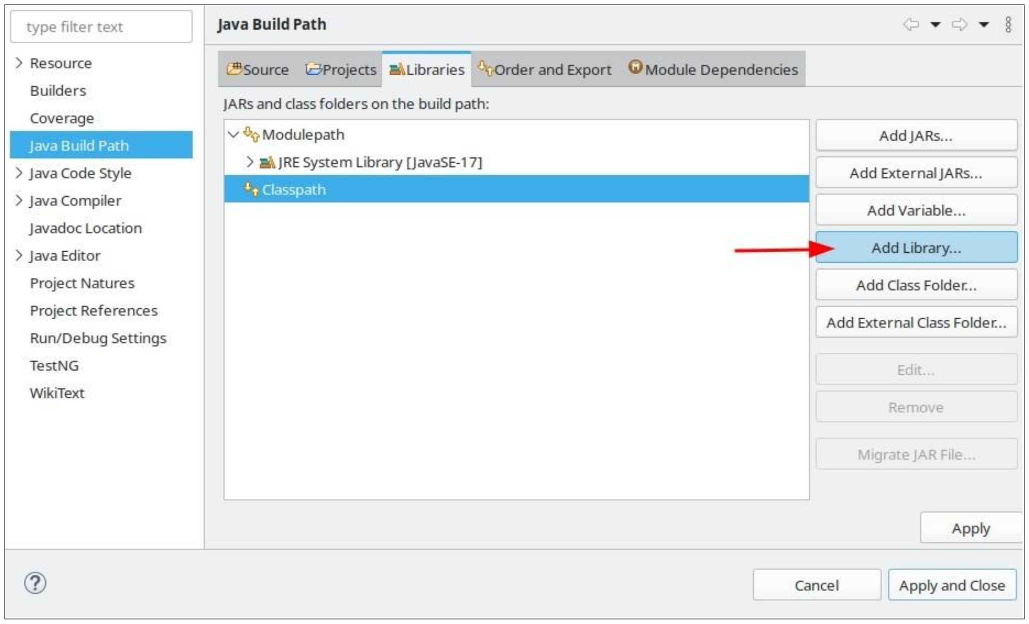
Classifier	Id	Version
osgi.bundle	com.beust.jcommander	1.78.0
osgi.bundle	com.beust.jcommander.source	1.78.0
osgi.bundle	org.apache-extras.beanshell.bsh	2.0.0.b6
osgi.bundle	org.apache-extras.beanshell.bsh.source	2.0.0.b6
osgi.bundle	org.testng	7.4.0.r202105021533
org.eclipse.update.featur	org.testng.eclipse	7.4.0.202106051955
osgi.bundle	org.testng.eclipse	7.4.0.202106051955
osgi.bundle	org.testng.eclipse.maven	7.4.0.202106051955
org.eclipse.update.featur	org.testng.eclipse.maven.feature	7.4.0.202106051955
org.eclipse.update.featur	org.testng.p2.feature	7.4.0.r202105021533



Restart Eclipse IDE to apply the software update?

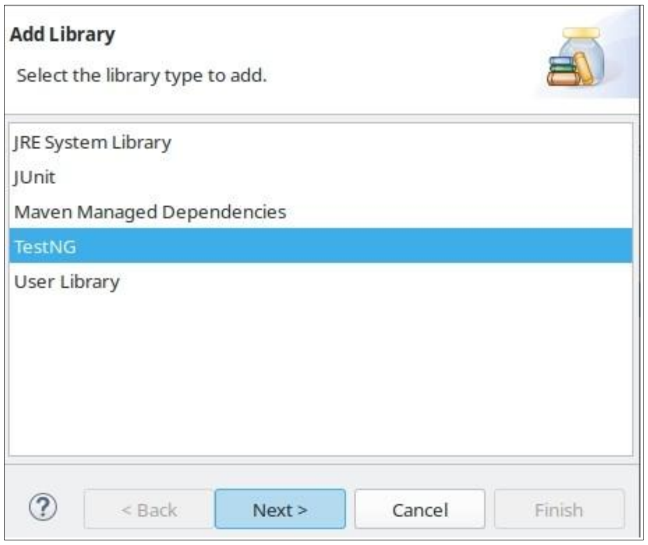
3. Now, create new Java Project, create new Java Class.

4. And Add TestNG Library and jexcelapi JARs.



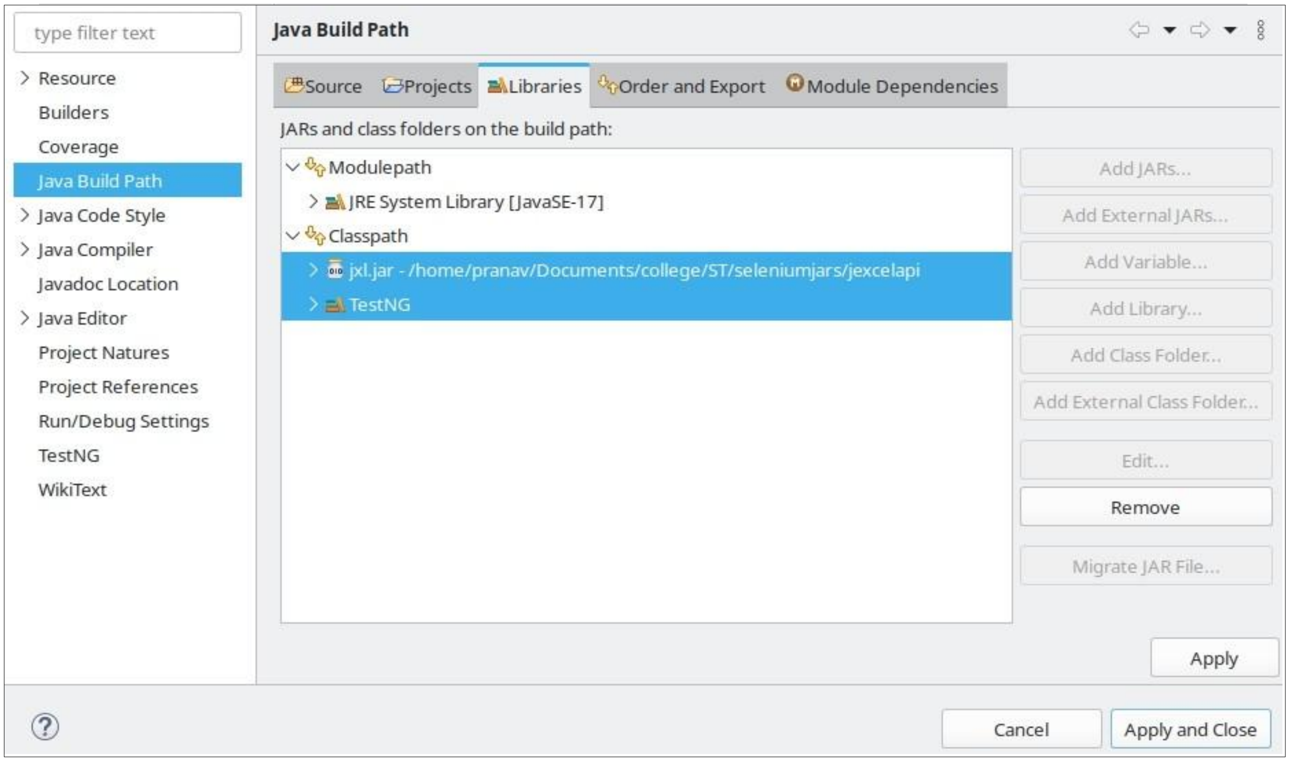
JARs and class folders on the build path:

- Modulepath
  - JRE System Library [JavaSE-17]
- Classpath



Select the library type to add.

JRE System Library
JUnit
Maven Managed Dependencies
TestNG
User Library



Java Build Path

JARs and class folders on the build path:

```
Modulepath
  JRE System Library [JavaSE-17]
Classpath
  jxl.jar - /home/pranav/Documents/college/ST/seleniumjars/jexcelapi
  TestNG
```

5. Add these JARs and click on Finish and a window will popup saying do you want to create module? Just click on don't create.

6. Now create new JAVA Class.

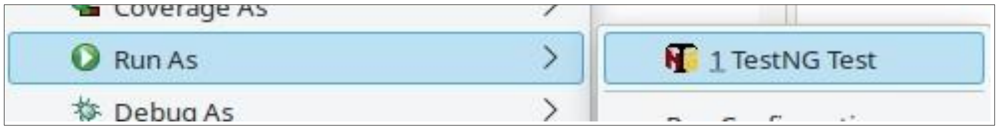
7. Now type the code given below in Pract5.java

```
package TYCS68;

import jxl.*;

import jxl.write.*;

import org.testng.annotations.*; import

java.io.*;

public class pract5 {

        @BeforeClass

        public void f1() {}

        @Test

        public void testImportExport() throws Exception{
```

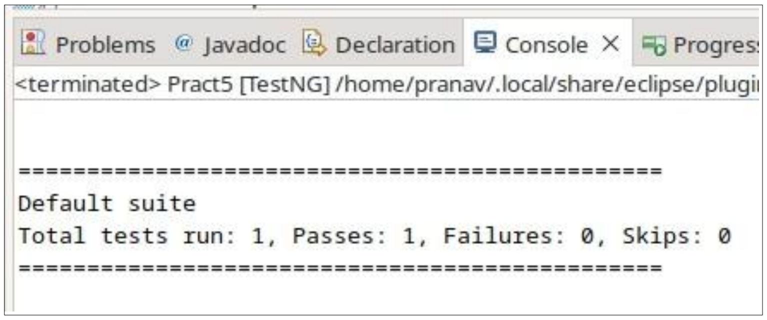
```
FileInputStream fi=new FileInputStream("D:\\Book1.xls"); Workbook

w=Workbook.getWorkbook(fi);

Sheet s=w.getSheet(0);

String a[][]=new String[s.getRows()][s.getColumns()]; FileOutputStream

fo=new FileOutputStream("D:\\Book1res.xls"); WritableWorkbook

wwb= Workbook.createWorkbook(fo); WritableSheet

ws=wwb.createSheet("result",0);

for(int i=0;i<s.getRows();i++)

        for(int j=0;j<s.getColumns();j++) {

                a[i][j]=s.getCell(j,i).getContents(); Label

                l2=new Label(j,i,a[i][j]); ws.addCell(l2);

                Label l1=new Label(6,0,"Result");

        }

for(int i=1;i<s.getRows();i++)

        for(int j=2;j<s.getColumns();j++) {

                        a[i][j]=s.getCell(j,i).getContents(); int

                        x= Integer.parseInt(a[i][j]); if(x>35) {

                                Label l1=new Label(6,i,"Pass");

                                ws.addCell(l1);

                        }

                        else {

                                Label l1=new Label(6,i,"Fail");

                                ws.addCell(l1);
```

```
                                        break;
                              }
                    }
          wwb.write();
          wwb.close();
     }
}
```

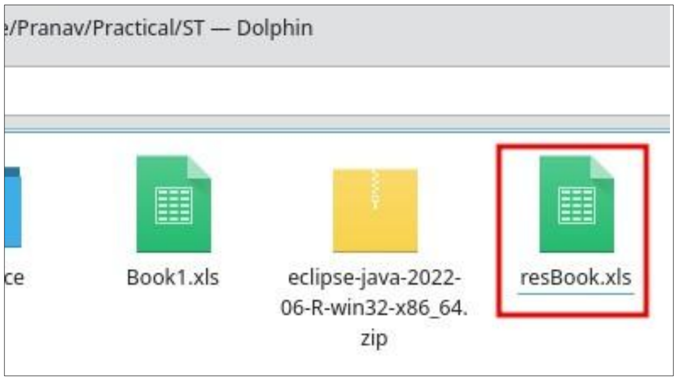
8. Now run the file as 1 TestNG test



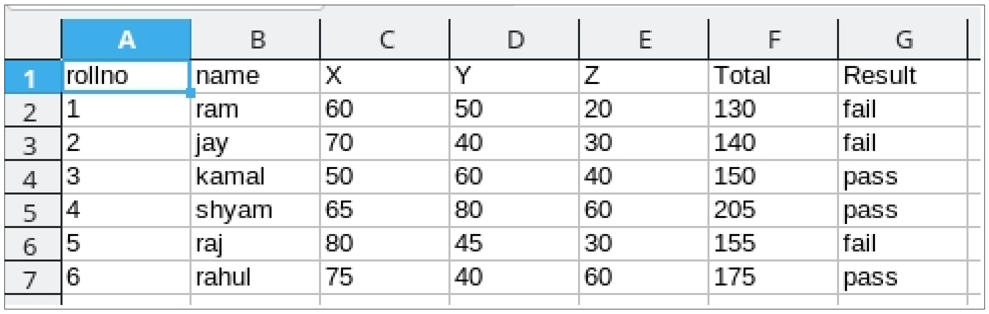
Output:



```
<terminated> Pract5 [TestNG] /home/pranav/.local/share/eclipse/plugi

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


/Pranav/Practical/ST — Dolphin
ce
Book1.xls
eclipse-java-2022-06-R-win32-x86_64.zip
resBook.xls

Tanmay Pramod Mayekar
Roll No :- 68



	A	B	C	D	E	F	G
1	rollno	name	X	Y	Z	Total	Result
2	1	ram	60	50	20	130	fail
3	2	jay	70	40	30	140	fail
4	3	kamal	50	60	40	150	pass
5	4	shyam	65	80	60	205	pass
6	5	raj	80	45	30	155	fail
7	6	rahul	75	40	60	175	pass

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 6

Theory:

This practical demonstrates **conditional logic and validation** in TestNG:

- Student records (marks) are stored in a data file or array.
- The program checks which students scored above 60 in one subject or all subjects.
- TestNG generates a report highlighting the selected students.

Applications:

- Filtering and validating test data.
- Automating result analysis.
- Demonstrating how TestNG can combine logic with automated testing.

It emphasizes the importance of data validation in quality assurance.

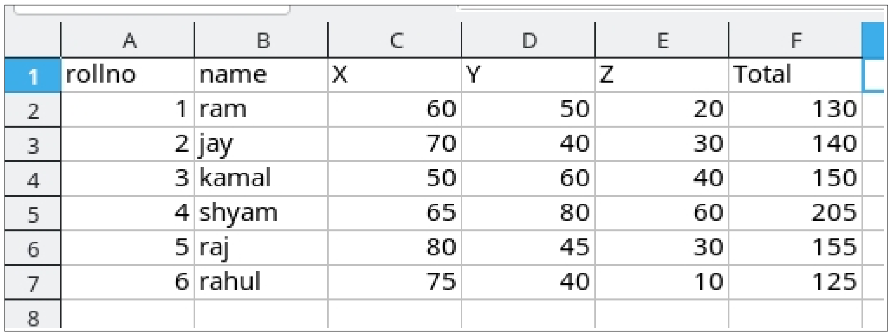
Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 6

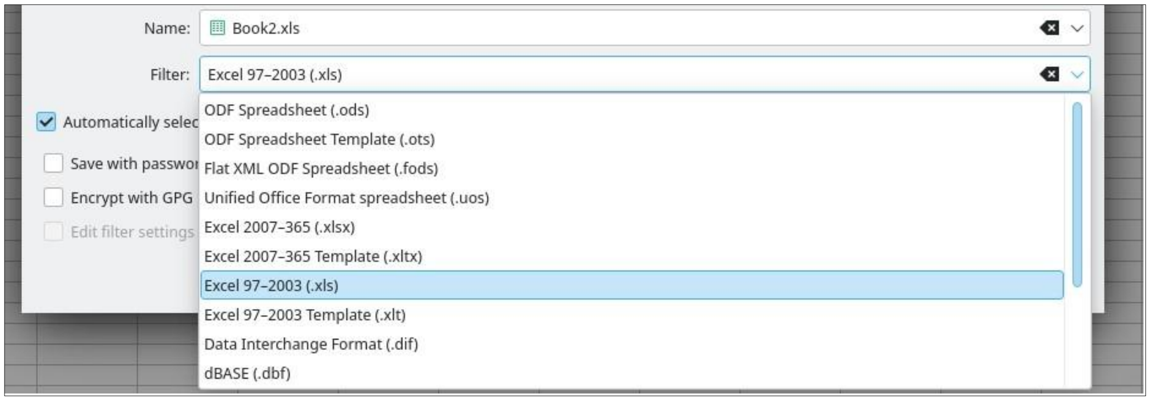
Write & test a program to select the number of students who have scored more than 60 in any one subject (or all subjects) using TestNG.

1. Create new Excel Workbook and add the following data:

2. And now save as



	A	B	C	D	E	F
1	rollno	name	X	Y	Z	Total
2	1	ram	60	50	20	130
3	2	jay	70	40	30	140
4	3	kamal	50	60	40	150
5	4	shyam	65	80	60	205
6	5	raj	80	45	30	155
7	6	rahul	75	40	10	125
8						

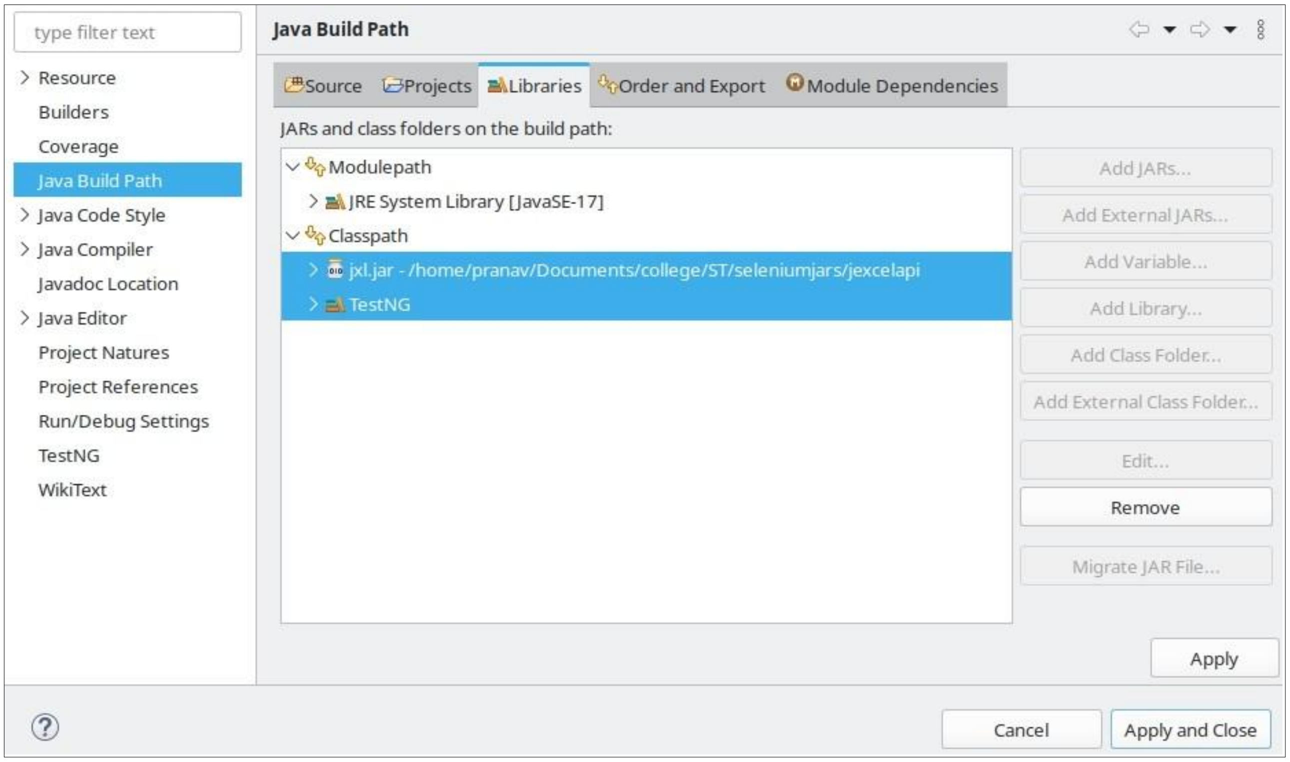


Name: Book2.xls
Filter: Excel 97–2003 (.xls)

Automatically selec
Save with passwor
Encrypt with GPG
Edit filter settings
ODF Spreadsheet (.ods)
ODF Spreadsheet Template (.ots)
Flat XML ODF Spreadsheet (.fods)
Unified Office Format spreadsheet (.uos)
Excel 2007–365 (.xlsx)
Excel 2007–365 Template (.xltx)
Excel 97–2003 (.xls)
Excel 97–2003 Template (.xlt)
Data Interchange Format (.dif)
dBASE (.dbf)

3. Now, create new Java Project, create new Java Class.

4. And Add TestNG Library and jexcelapi JARs. Into buildpath (classpath)


type filter text

Resource
Builders
Coverage
Java Build Path
Java Code Style
Java Compiler
Javadoc Location
Java Editor
Project Natures
Project References
Run/Debug Settings
TestNG
WikiText

Java Build Path

Source | Projects | Libraries | Order and Export | Module Dependencies

JARs and class folders on the build path:

```
Modulepath
  JRE System Library [JavaSE-17]
Classpath
  jxl.jar - /home/pranav/Documents/college/ST/seleniumjars/jexcelapi
  TestNG
```

5. Add these JARs and click on Finish and a window will popup saying do you want to create module? Just click on don't create.

6. Now create new JAVA Class and type following code:

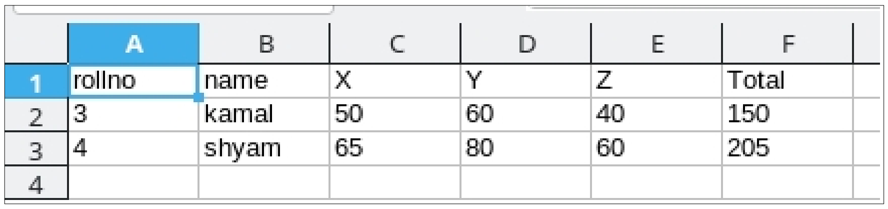
```
package TYCS68;

import jxl.*;

import jxl.write.*;

import java.io.*;

import org.testng.annotations.Test;

public class pract6 {

        @Test

        public void testimportexport1() throws Exception{ FileInputStream

                fi=new FileInputStream("D:\\Book2.xls"); Workbook

                w=Workbook.getWorkbook(fi);

                Sheet s=w.getSheet(0);
```

```
String a[][]=new String[s.getRows()][s.getColumns()]; FileOutputStream

fo=new FileOutputStream("D:\\countBook2res.xls"); WritableWorkbook

wwb= Workbook.createWorkbook(fo); WritableSheet

ws=wwb.createSheet("result",0);

int c=0;

for(int i=0;i<s.getRows();i++) {

        for(int j=0;j<s.getColumns();j++)

        {

                if(i>=1)

                {

                        String b=new String();

                        b=s.getCell(3,i).getContents(); int

                        x=Integer.parseInt(b); if(x<60)

                        {

                                c++;

                                break;

                        }

                }

                a[i][j]=s.getCell(j,i).getContents();

                Label l2=new Label(j,i-c,a[i][j]);

                ws.addCell(l2);


        }

}
```

```
		wwb.write();

		wwb.close();

	}

}
```

7. Now run the file as 1 TestNG test

Output:



	A	B	C	D	E	F
1	rollno	name	X	Y	Z	Total
2	3	kamal	50	60	40	150
3	4	shyam	65	80	60	205
4						

Practical No. 7

Theory:

Web pages are made up of multiple elements such as links, buttons, images, checkboxes, input fields, etc. In automation testing, verifying the presence and count of elements ensures the UI is complete.

Selenium WebDriver:

- findElements() retrieves all objects of a given type.
- The size() function gives the total count.

Applications:

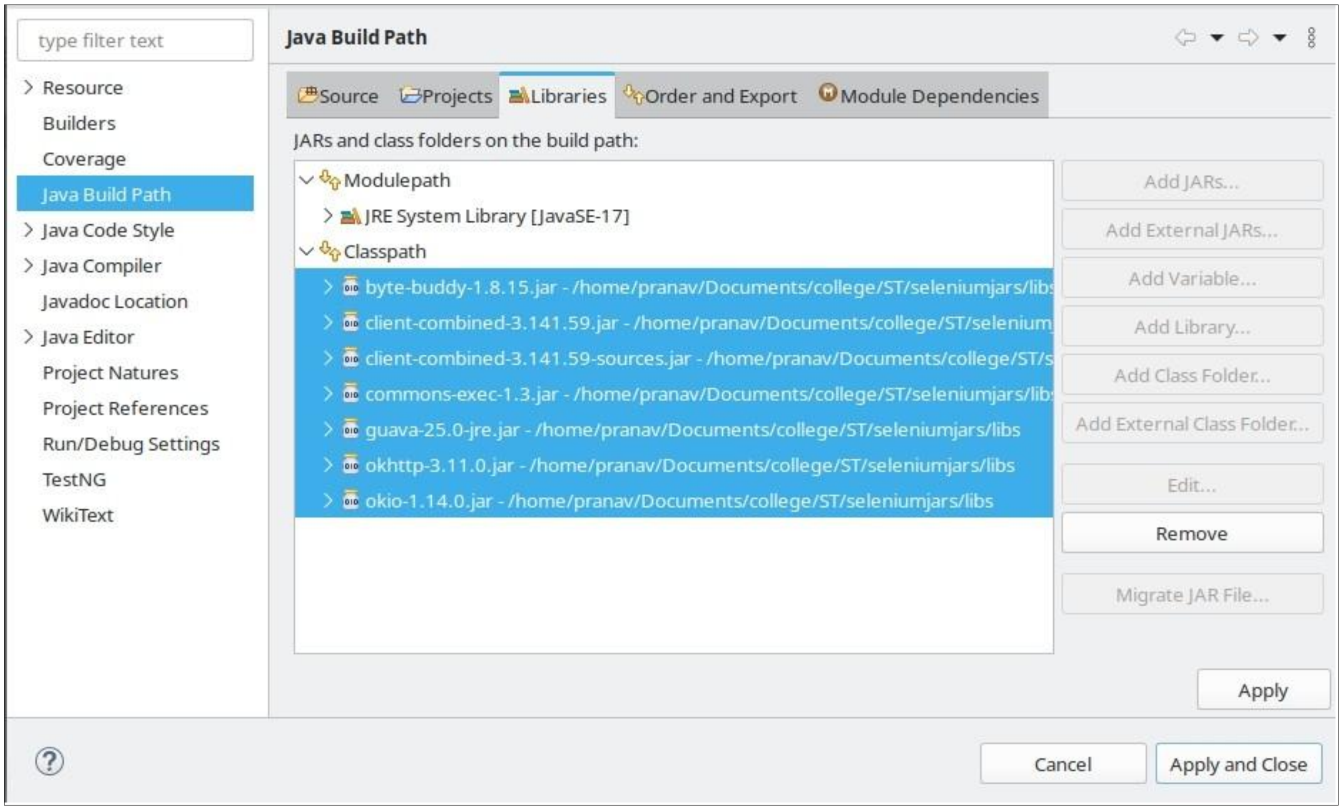
- Checking that all elements are loaded correctly.
- Ensuring no missing components after deployment.
- Useful in layout verification and smoke testing.

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 7

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

1. Create a new Java Project and create a new Class then add External Selenium JARs



JARs and class folders on the build path:

```
Modulepath
  JRE System Library [JavaSE-17]
Classpath
  byte-buddy-1.8.15.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  client-combined-3.141.59.jar - /home/pranav/Documents/college/ST/selenium
  client-combined-3.141.59-sources.jar - /home/pranav/Documents/college/ST/s
  commons-exec-1.3.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  guava-25.0-jre.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  okhttp-3.11.0.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
  okio-1.14.0.jar - /home/pranav/Documents/college/ST/seleniumjars/libs
```

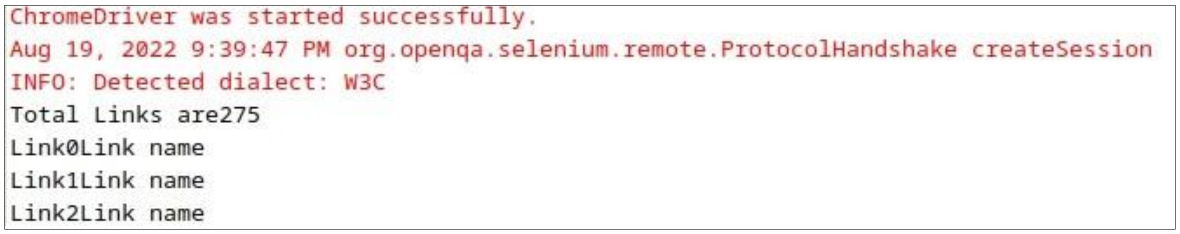
2. Click on Finish and a window will popup saying do you want to create module? Just click on don't create.

3. Now add the following code in newly created Class.

```
package TYCS68;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

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

```
  public class pract7 {


        public static WebDriver driver; public

        static void main(String args[])

        {


        System.setProperty("webdriver.chrome.driver","D:\\chromedriver.exe1\\chromedriver-
win64\\chromedriver.exe");

                driver=new ChromeDriver();

                driver.get("http://amazon.com/");

                java.util.List<WebElement>links=driver.findElements(By.tagName("a"));

                System.out.println("Total links are"+links.size());

                for(int i=0;i<links.size();i=i+1)

                        {

                        System.out.println("Link "+i+" Link name "+links.get(i).getText());

                        }

        }


}
```

Output:

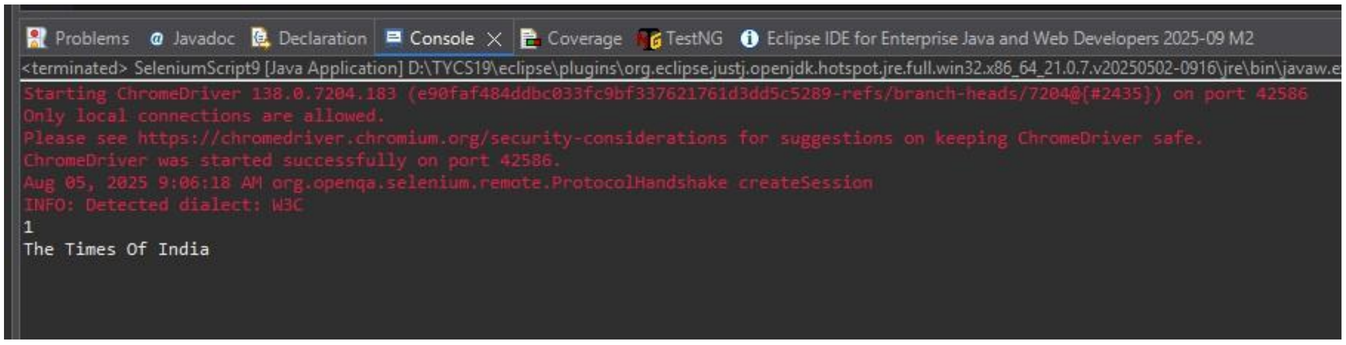


```
ChromeDriver was started successfully.
Aug 19, 2022 9:39:47 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Total Links are275
Link0Link name
Link1Link name
Link2Link name
```

Extra Code: In this program browser will open and close simultaneously.

```
package TYCS68;

import java.util.Set;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

class SeleniumScript9
{
  public static void main(String[] args)
  {
           System.setProperty("webdriver.chrome.driver","D:\\chromedriver.exe1\\chromedriver-
win64\\chromedriver.exe");
           WebDriver driver=new ChromeDriver();
    driver.get("http://www.timesofindia.indiatimes.com//");
    Set<String> allWH=driver.getWindowHandles();
    int count=allWH.size();
    System.out.println(count); for(String
    wh:allWH)
    {
      driver.switchTo().window(wh);
      String title=driver.getTitle();
      System.out.println(title);
      driver.close();
    }
  }
}
```

Output:


Problems
Javadoc
Declaration
Console
Coverage
TestNG
Eclipse IDE for Enterprise Java and Web Developers 2025-09 M2

```
<terminated> SeleniumScript9 [Java Application] D:\TYCS19\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.7.v20250502-0916\jre\bin\javaw.e
Starting ChromeDriver 138.0.7204.183 (e90faf484ddbc033fc9bf337621761d3dd5c5289-refs/branch-heads/7204@{#2435}) on port 42586
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully on port 42586.
Aug 05, 2025 9:06:18 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
1
The Times Of India
```

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 8

Theory:

Drop-down menus are common UI components for selecting one option. Selenium provides the Select class to handle drop-downs.

Methods:

- getOptions() → Retrieves all available options.
- getFirstSelectedOption() → Gets the currently selected item.
- selectByVisibleText() → Chooses an option programmatically.

Applications:

- Validating drop-down values (e.g., countries, states).
- Checking default selections.
- Ensuring user inputs are restricted to valid options.

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 8

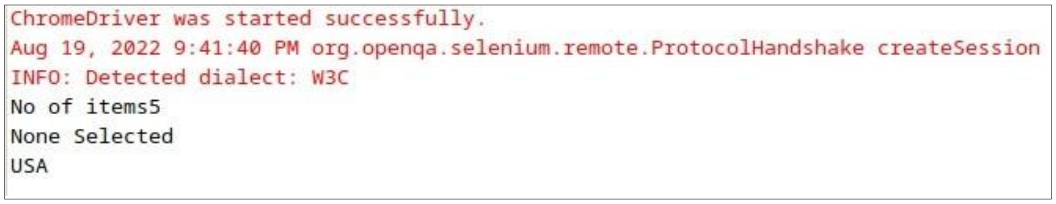
Write and test a program to print total numbers of single drop down list items present / available on the page and print selected item.

NOTE: Follow the same steps to create new project and class as Practical 7 for Practical 8, Practical 9 and Practical 10.

code:

```
package pract; import

java.util.List;

import java.util.concurrent.TimeUnit;

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.support.ui.Select; public

class prac8 {

        public static WebDriver driver;

        public static void main(String[] args)throws InterruptedException{

                System.setProperty("webdriver.chrome.driver","C:\\chromedriver.exe\\");

                WebDriver driver = new ChromeDriver();

                driver.manage().timeouts().implicitlyWait(10,TimeUnit.SECONDS);


        driver.get("https://www.tutorialspoint.com/selenium/practice/selenium_automation_practi ce.php");
                Select oSelect =new Select(driver.findElement(By.name("state")));

                oSelect.selectByVisibleText("NCR");

                Thread.sleep(2000); oSelect.selectByIndex(2);

                Thread.sleep(2000);

                List<WebElement>oSize=oSelect.getOptions(); int

                iListSize=oSize.size(); System.out.println("No of

                items"+iListSize); for(int i=0;i<iListSize;i++) {

                    String sValue = oSelect.getOptions().get(i).getText();

                    System.out.println(sValue); if(sValue.contentEquals("Haryana"))

                    {

                        oSelect.selectByIndex(i); break;

                    }

                }

                driver.quit();

        }

}
```

Output:



```
ChromeDriver was started successfully.
Aug 19, 2022 9:41:40 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
No of items5
None Selected
USA
```

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 9

Theory:

A multiple drop-down allows more than one selection at a time. Selenium handles these with the same Select class.

Methods:

- isMultiple() → Confirms multi-select property.
- getAllSelectedOptions() → Returns all chosen options.
- deselectAll() → Clears selections.

Applications:

- Validating multi-choice forms (e.g., skills, hobbies).
- Ensuring multiple selections work correctly.
- Testing UI flexibility and form validations.

Tanmay Pramod Mayekar
Roll No :- 68

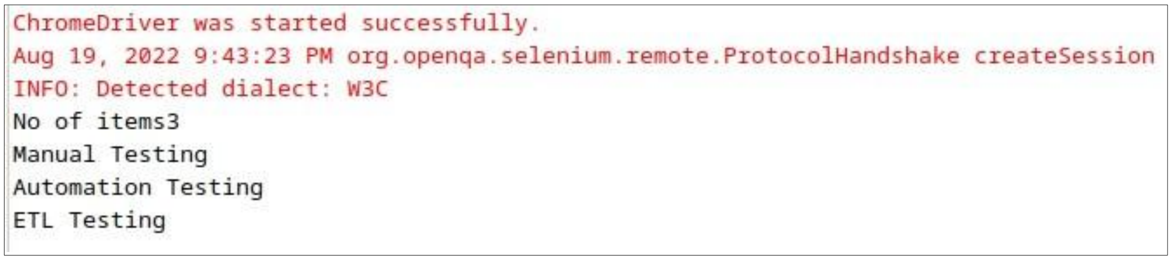
Practical No. 9

Write and test a program to print total numbers of Multiple drop down list items present / available on the page and print selected item.

NOTE: Follow the same steps to create new project and class as Practical 7 for Practical 8, Practical 9 and Practical 10.

```
package pract9; import

java.util.List;

import java.util.concurrent.TimeUnit;

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.support.ui.Select; public

class pract9 {

        public static WebDriver driver;

        public static void main(String[] args)throws InterruptedException{

                System.setProperty("webdriver.chrome.driver","C:\\chromedriver.exe\\");

                WebDriver driver = new ChromeDriver();

                driver.manage().timeouts().implicitlyWait(10,TimeUnit.SECONDS);

                driver.get("https://www.techlistic.com/p/selenium-practice-form.html");

                Select oSelect =new Select(driver.findElement(By.id("selenium_commands")));

                oSelect.selectByIndex(0);

                Thread.sleep(2000);
                oSelect.deselectByIndex(0);
                oSelect.selectByVisibleText("Browser Commands");
                Thread.sleep(2000);
                oSelect.deselectByVisibleText("Browser Commands");
                List<WebElement>oSize=oSelect.getOptions();
                int iListSize=oSize.size(); System.out.println("No of
                items"+iListSize); for(int i=0;i<iListSize;i++) {
                        String sValue = oSelect.getOptions().get(i).getText();
                        System.out.println(sValue);   oSelect.selectByIndex(i);
                        Thread.sleep(2000);
            }
oSelect.deselectAll();
driver.close();
}
}
```

Output:



```
ChromeDriver was started successfully.
Aug 19, 2022 9:43:23 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
No of items3
Manual Testing
Automation Testing
ETL Testing
```

Tanmay Pramod Mayekar
Roll No :- 68

Practical No. 10

Theory:

Checkboxes are used when multiple inputs are required. Selenium can locate them using attributes like id, name, or xpath.

Key concepts:

- findElements() retrieves all checkboxes.
- isSelected() checks whether a checkbox is ticked.
- Automation can simulate checking/unchecking options.

Applications:

- Form testing (e.g., terms & conditions, subscription options).
- Ensuring defaults are selected/unselected as expected.
- Validating correct number of checkboxes are displayed.

Tanmay Pramod Mayekar
Roll No :- 68

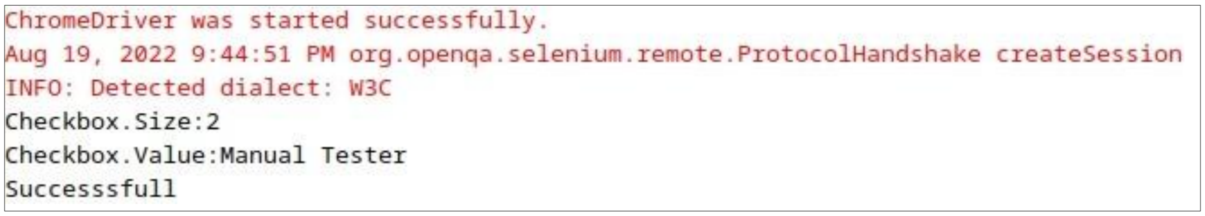
Practical No. 10

Write and test a program to print total numbers of check boxes present / available on the page and print selected check box.

NOTE: Follow the same steps to create new project and class as Practical 7 for Practical 8, Practical 9 and Practical 10.

```
package pratical10; import

java.util.List;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

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

class prac10 {

   public static void main(String[] args) {

        System.setProperty("webdriver.chrome.driver","C:\\chromedriver.exe\\");

      WebDriver driver = new ChromeDriver();

      driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

      driver.get("https://www.techlistic.com/p/selenium-practice-form.html");

      List<WebElement> rdBtn_Gender = driver.findElements(By.id("sex-0"));

      boolean bValue = false;

      bValue = rdBtn_Gender.get(0).isSelected(); if

      (bValue == true) {

         rdBtn_Gender.get(1).click();

      } else {
            rdBtn_Gender.get(0).click();

        }

        List<WebElement> chkBx_Profession = driver.findElements(By.name("profession")); int

        iSize = chkBx_Profession.size();

        System.out.println("Checkbox Size:" + iSize); for

        (int i = 0; i < iSize; i++) {

            String sValue = chkBx_Profession.get(i).getAttribute("value"); System.out.println("Checkbox

            Value:" + sValue);

            if (sValue.equalsIgnoreCase("Manual Tester")) {

                chkBx_Profession.get(i).click();

                break;

            }

        }

        System.out.println("Successsfull");

        driver.quit();

    }

}
```

Output:



```
ChromeDriver was started successfully.
Aug 19, 2022 9:44:51 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Checkbox.Size:2
Checkbox.Value:Manual Tester
Successsfull
```