



**S.I.E.S College of Arts, Science and Commerce**  
**Sion(W), Mumbai – 400 022.**

**CERTIFICATE**

This is to certify that Mr. / Miss. Mondal Shuvo Prallard  
Roll No. TCS2021048 Has successfully completed the necessary course  
of experiments in the subject of Software Testing and Quality Assurance  
during the academic year **2020 – 2021** complying with the requirements of  
**University of Mumbai**, for the course of **T.Y.BSc. Computer Science**  
**[Semester-5]**

Prof. In-Charge

**Miss. Shivani Deopa**

(Software Testing and Quality Assurance)

Examination Date:

Examiner's Signature & Date:

/ / 20

Head of the Department

**Prof. Manoj Singh**

College Seal  
And  
Date|

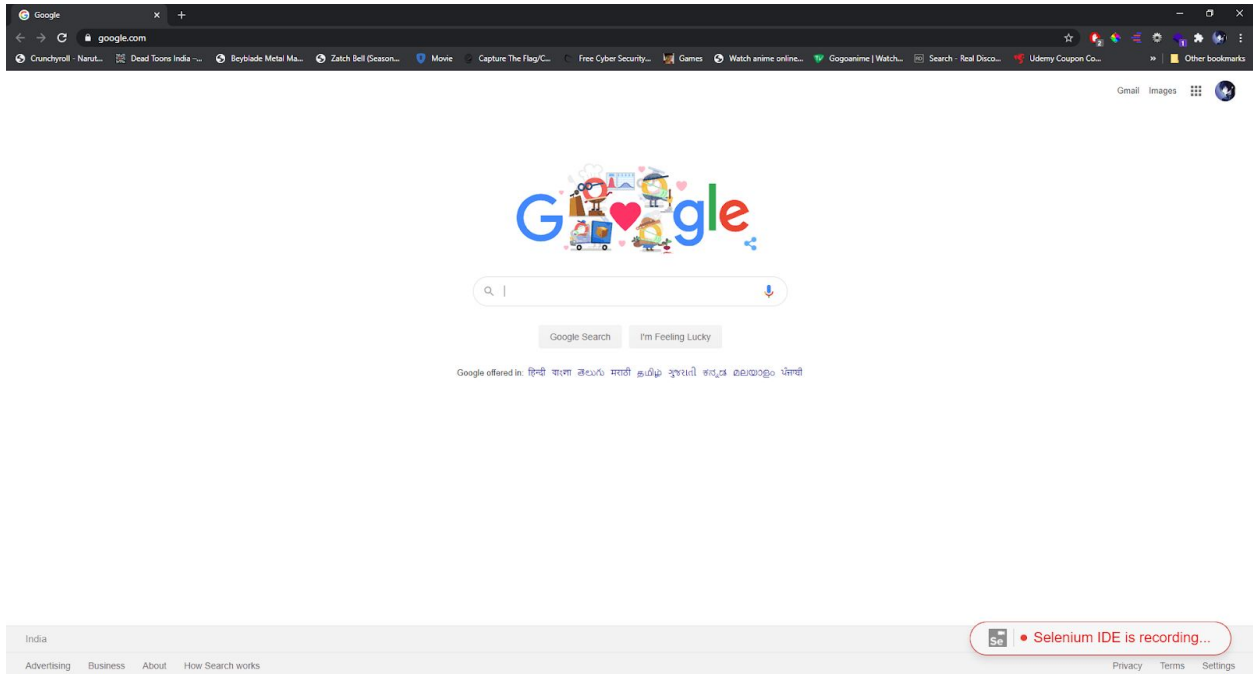
# Index

Sr.No	Title	Pg.No	Remark
1	Practical-1	2	
2	Practical-2	9	
3	Practical-3	14	
4	Practical-4	16	
5	Practical-5	19	
6	Practical-6	22	
7	Practical-7	25	
8	Practical-8	27	
9	Practical-9	30	

# Practical-1

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

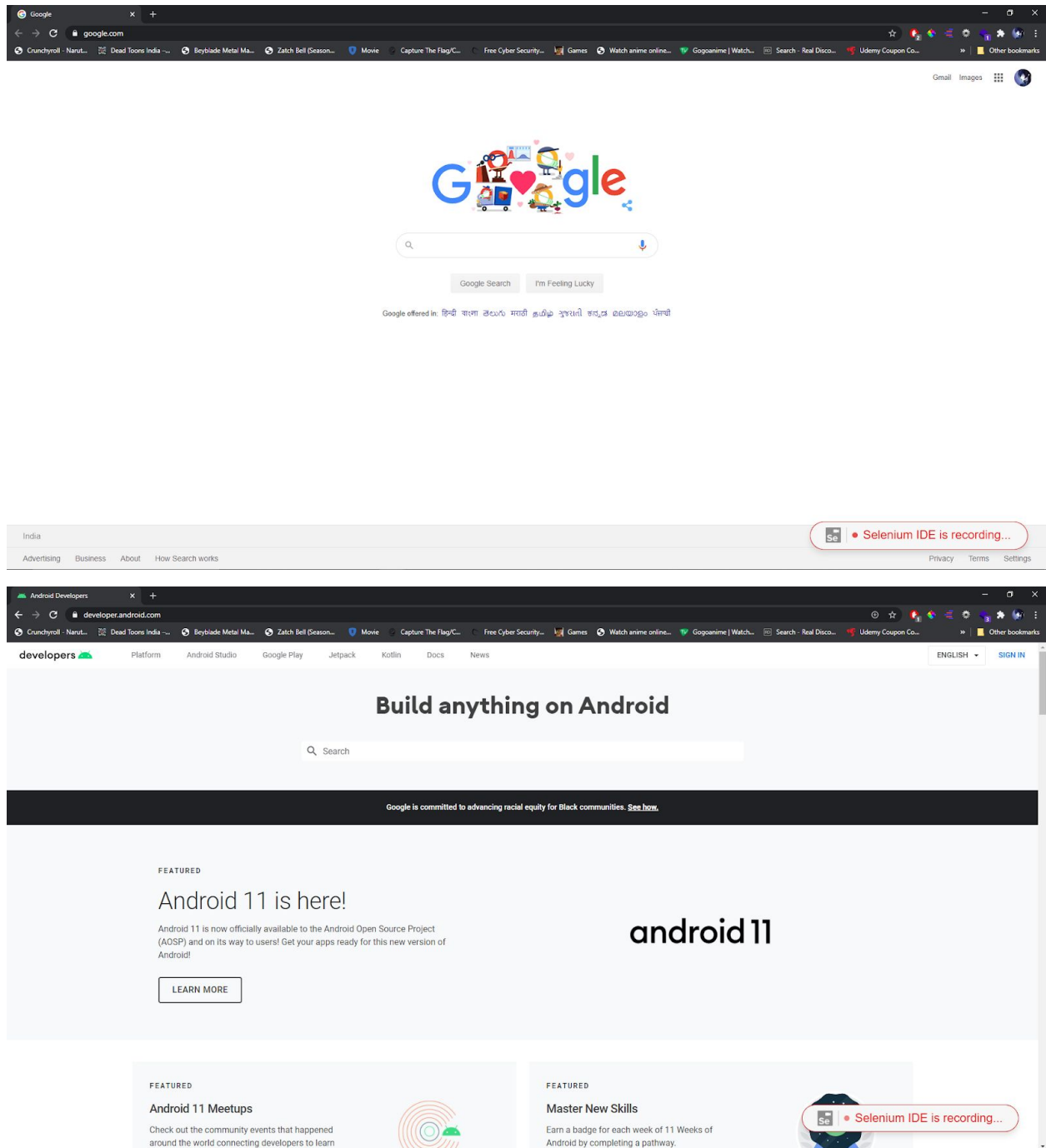
## Test-1:



The screenshot shows the Flutter documentation website (flutter.dev/docs) in a browser window. The website has a dark header with the Flutter logo and navigation links. A sidebar on the left contains a table of contents with categories like 'Get started', 'Samples & tutorials', 'Development', 'Testing & debugging', 'Performance & optimization', 'Deployment', 'Resources', and 'Reference'. The main content area features a 'Flutter documentation' title, a grid of links to 'Get Started', 'Widgets Catalog', 'API Docs', 'Cookbook', 'Samples', and 'Videos', and sections for 'What's new on this site' and 'New to Flutter?'. A Selenium IDE recording bar is visible at the bottom of the browser window, showing a list of commands and their targets/values.

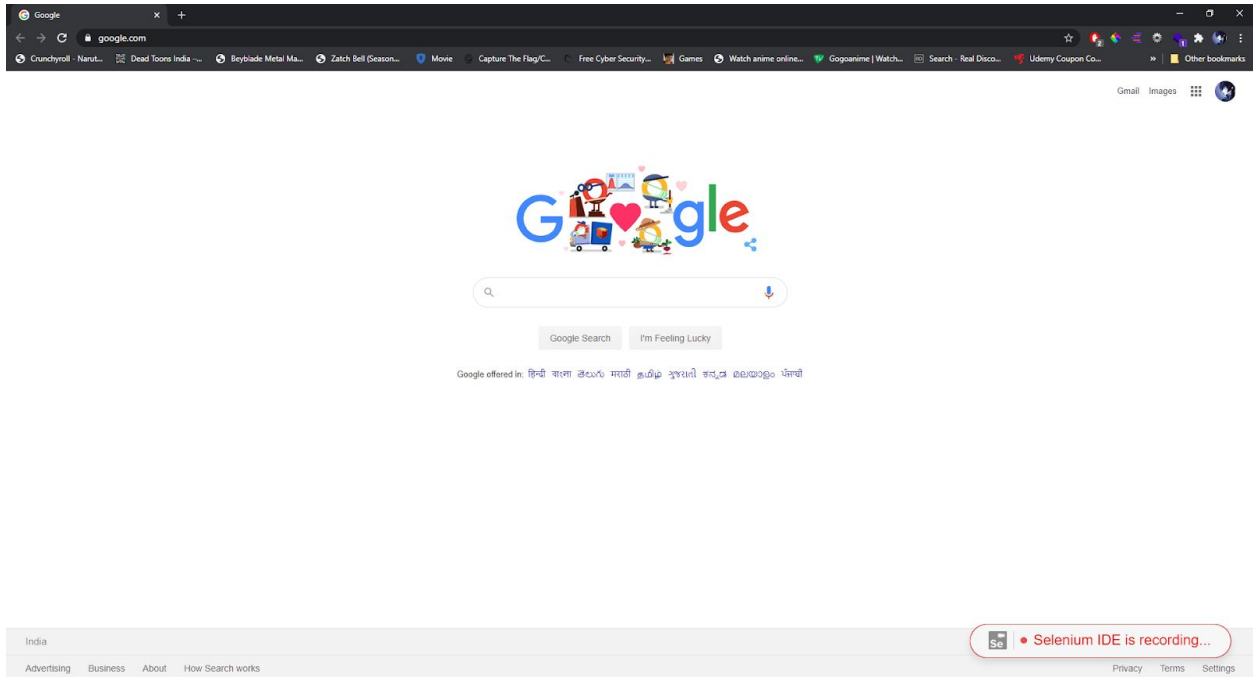
	Command	Target	Value
1	open	https://www.google.com/	
2	set window size	1936x1056	
3	type	name=q	flutter docs
4	send keys	name=q	\${KEY_ENTER}
5	click	css=div:nth-child(2) > .rc:nth-c hild(1) .LC20Ib	

## Test-2:



https://www.google.com/			
	Command	Target	Value
1	open	https://www.google.com/	
2	set window size	1936x1056	
3	type	name=q	andr
4	send keys	name=q	\${KEY_DOWN}
5	type	name=q	android docs
6	send keys	name=q	\${KEY_ENTER}
7	mouse over	css=.g > div > .rc.LC20lb	
8	mouse out	css=.g > div > .rc.LC20lb	
9	click	css=.g:nth-child(2) > .rc:nth-child(2) .LC20lb	

## Test-3:



The screenshot displays the Selenium IDE interface. At the top, the browser window shows the URL `selenium-python.readthedocs.io`. The main content area displays the 'Selenium with Python' page, which includes a navigation menu on the left, a 'Note' section, and a list of links for installation and getting started. The Selenium IDE toolbar at the bottom shows various icons for playback, undo, redo, and recording. A status bar at the bottom indicates 'Selenium IDE is recording...'. Below the toolbar, a table lists the recorded commands and their targets and values.

**Selenium with Python**

**Author:** Baiju Muthukadan  
**License:** This document is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

**Note:**  
 This is not an official documentation. If you would like to contribute to this documentation, you can fork this project in Github and send pull requests. You can also send your feedback to my email: baiju.m.mail AT gmail DOT com. So far 40+ community members have contributed to this project (See the closed pull requests). I encourage contributors to add more sections and make it a good documentation!

**Navigation**

1. Installation
2. Getting Started
3. Navigating
4. Locating Elements
5. Waits
6. Page Objects
7. WebDriver API
8. Appendix: Frequently Asked Questions

**Related Topics**

Documentation overview  
 • Next: 1. Installation

**Quick search**

**1. Installation**

- 1.1. Introduction
- 1.2. Downloading Python bindings for Selenium
- 1.3. Drivers
- 1.4. Detailed instructions for Windows users
- 1.5. Downloading Selenium server
- 1.6. Installing from Git sources

**2. Getting Started**

- 2.1. Simple Usage
- 2.2. Example Explained
- 2.3. Using Selenium to write tests
- 2.4. Walk through of the example
- 2.5. Using Selenium with remote WebDriver

**3. Navigating**

- 3.1. Interacting with the page
- 3.2. Filling in forms
- 3.3. Drag and drop
- 3.4. Moving between windows and frames
- 3.5. Popup dialogs
- 3.6. Navigation: history and location
- 3.7. Cookies

**Command | Target | Value**

1	<code>open</code>	<code>https://www.google.com/</code>	
2	<code>set window size</code>	<code>1936x1056</code>	
3	<code>send keys</code>	<code>name=q</code>	<code>\${KEY_DOWN}</code>
4	<code>type</code>	<code>name=q</code>	<code>selenium docs</code>
5	<code>send keys</code>	<code>name=q</code>	<code>\${KEY_ENTER}</code>
6	<code>click</code>	<code>css=.rc:nth-child(3) .LC20lb</code>	
7	<code>mouse over</code>	<code>css=ul:nth-child(3) &gt; .toctree-l1:nth-child(1) &gt; .reference</code>	



## Test-4:

Web Scraping Explained

webharvy.com/articles/what-is-web-scraping.html#~:text=Web%20Scraping%20(also%20termed%20Screen.in%20table%20spreadsheet%20format, Crunchyroll - Naru... Dead Toons India ~... Beyblade Metal Ma... Zatch Bell (Season... Movie Capture The FlagC... Free Cyber Security... Games Watch anime online... Goganime | Watch... Search - Real Disco... Udemu Coupon Co... Other bookmarks

support@webharvy.com | sales@webharvy.com | YouTube Channel | KB Articles

Select Language

**WebHarvy** Home Help Blog Download Buy Contact

### What is Web Scraping ?

Articles Home  
Product Help  
YouTube Channel  
WebHarvy Blog

**Web Scraping** (also termed **Screen Scraping**, **Web Data Extraction**, **Web Harvesting** etc.) is a technique employed to extract large amounts of data from websites whereby the data is extracted and saved to a local file in your computer or to a database in table (spreadsheet) format.

Data displayed by most websites can only be viewed using a web browser. They do not offer the functionality to save a copy of this data for personal use. The only option then is to manually copy and paste the data - a very tedious job which can take many hours or sometimes days to complete. **Web Scraping** is the technique of automating this process, so that instead of manually copying the data from websites, the **Web Scraping** software will perform the same task within a fraction of the time.

Structured data in file or database  
(JSON, CSV)

How to easily scrape data from websites using WebHarvy ?

A **web scraping software** will automatically load and extract data from multiple pages of websites based on your requirement. It is either custom built for a specific website or is one which can be configured to work with any website. With the click of a button you can easily save the data available in the website in a file in your

Selenium IDE is recording...

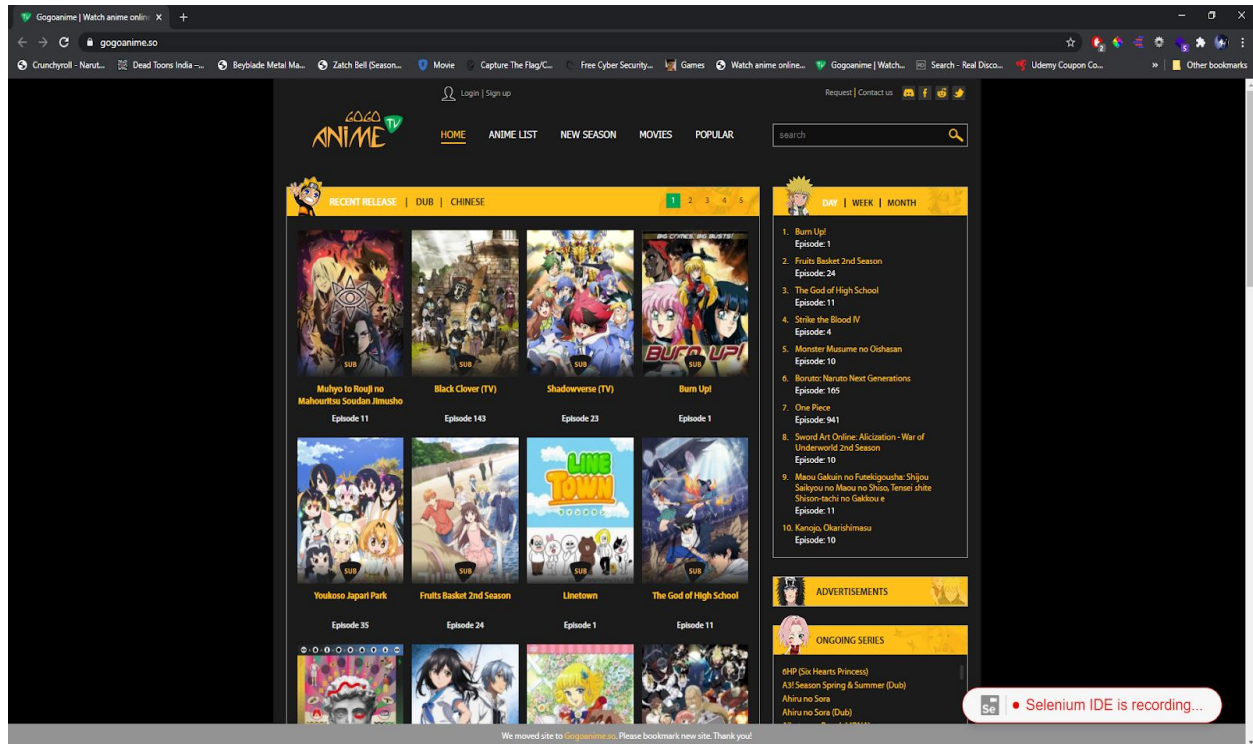
https://www.google.com/

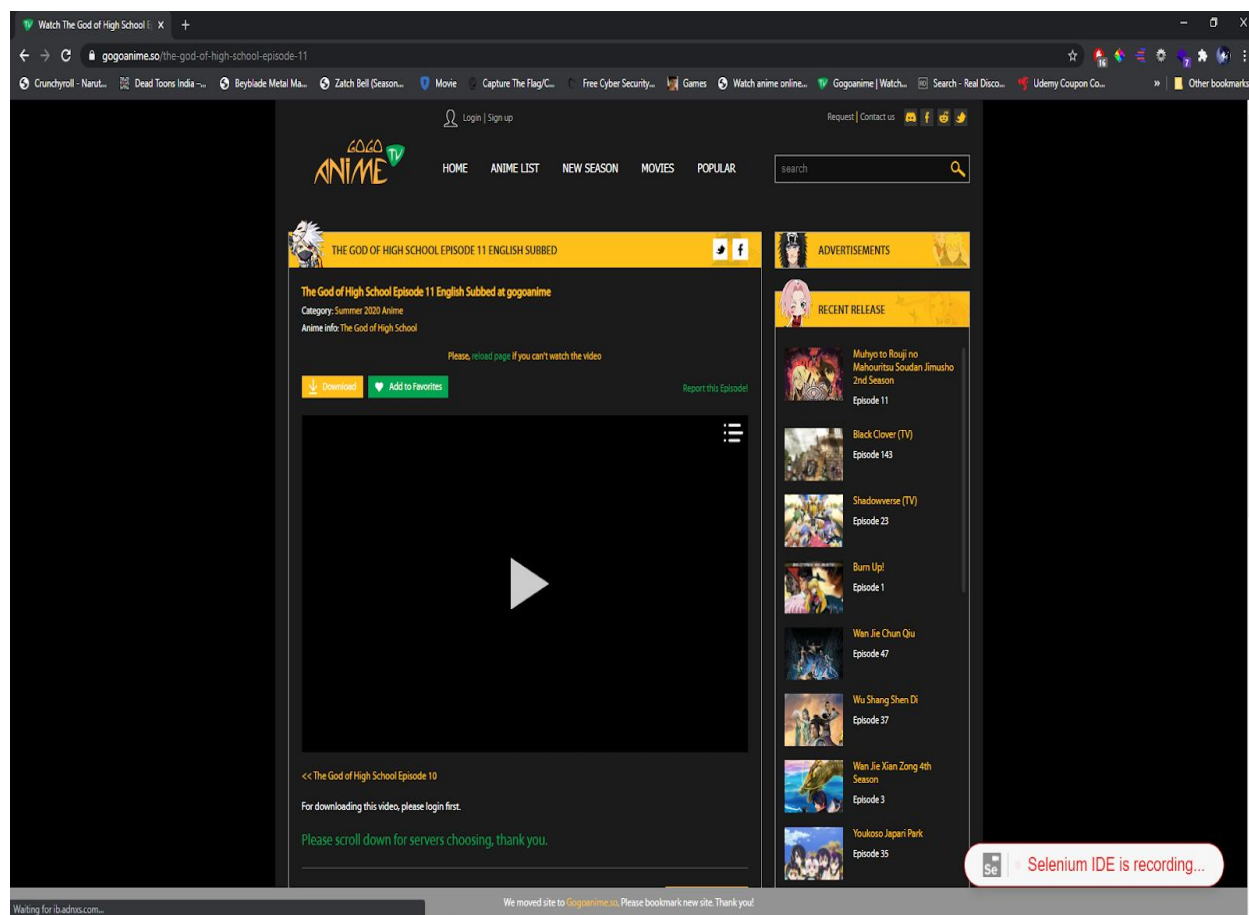
	Command	Target	Value
1	open	https://www.google.com/	
2	set window size	1936x1056	
3	send keys	name=q	#{KEY_DOWN}
4	type	name=q	web scraping
5	send keys	name=q	#{KEY_ENTER}
6	mouse over	css=g:nth-child(3) > .rc:nth-child(1) .LC201b	
7	click	css=g:nth-child(3) > .rc:nth-child(1) .LC201b	
8	mouse out	css=g:nth-child(3) > .rc:nth-child(1) .LC201b	

## Practical-2:

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

Test-1:





https://gogoanime.so/		
Command	Target	Value
1	open	https://gogoanime.so/
2	set window size	1936x1056
3	click	css=li:nth-child(8) img
4	select frame	index=0
5	run script	window.scrollTo(0,0)

## Test-2:

The image displays two screenshots of a YouTube interface. The top screenshot shows the YouTube homepage with various video recommendations. The bottom screenshot shows a specific video player for 'Web Developer Salaries are UP in 2020!' by Stefan Mischock, with a 'Up next' sidebar.

**Top Screenshot: YouTube Homepage**

- Search Bar:** Search
- Navigation:** Home, Trending, Subscriptions, Library, History, Watch later, Flutter Tutorials, Show more
- Subscriptions:** Spinint Records, freeCodeCamp.org, Traversy Media, DesignCourse, T-Series, CodeWithHarry, Abhishek Sahani, Show 620 more
- More from YouTube:** YouTube Premium
- Latest YouTube posts:** Teachingmensfashion (1 day ago), Beebom (6 days ago), RVCI Media (1 hour ago), Netflix (15 hours ago)
- Video Recommendations:**
  - Anthem ON RTX 2060 GAMEPLAY - CRAZY GRAPHIC DLSS ONI (Abhishek Sahani, 17 views • 6 hours ago)
  - Graph Theory Algorithms NO ADS (freeCodeCamp.org, 514K views • 11 months ago)
  - Top 10 Best Manhwa Where MC Reborn / Reincarnated to Complete His... (Animan, 160K views • 3 weeks ago)
  - using 600 IQ Impostor negotiation tactics to survive... (Disguised Toast, 960K views • 10 hours ago)
  - [MV] Yuki no Hana (Nakashima Mika) Cover by yurisa (yurisa, 3.8M views • 7 months ago)
  - Bedroom Masters Degree Day 0 Day ~1,097 (Daniel Bourke, 13K views • 2 days ago)
  - 2 TEAMS! PERFECT! (Apex Legends Funny Moments - Gaming Cur..., 43K views • 16 hours ago)
  - JS (Stefan Mischock, 1.2K views • 3 hours ago)
  - Web Developer Salaries are UP in 2020! (Stefan Mischock, 1.2K views • 3 hours ago)

**Bottom Screenshot: Video Player**

- Video Title:** Web Developer Salaries are UP in 2020!
- Channel:** Stefan Mischock
- Views:** 1,211 views • Sep 15, 2020
- Engagement:** 171 likes, 2 dislikes, SHARE, SAVE, ...
- Up next:**
  - Developer Job Prospects in 2020 (Stefan Mischock, 18K views • 2 weeks ago)
  - How I Became a Web Developer in 3 Months (Chris Sean, 22K views • 1 week ago)
  - NEW Apex Legends FUNNY & EPIC Moments #425 (Apex Legends Funny Moments - Gaming Cur..., Recommended for you)
  - DAKQO x Kenahi Yonezu "Tireworks" MUSIC VIDEO (Dakko, Recommended for you)
  - Your Name/Kimi no Na wa/君の名は。 Orchestra Concert... (K28M042, Recommended for you)
  - What is a RESTful API? Explanation of REST & HTTP (Traversy Media, 1M views • 3 years ago)
  - Hacking challenge at DEFCON (Conflict International, 385K views • 3 years ago)

https://www.youtube.com/		
	Command	Target   Value
1	open	https://www.youtube.com/
2	set window size	1936x1056
3	mouse over	css=.ytd-thumbnail > #thumb nail
4	mouse out	css=.ytd-thumbnail > #thumb nail
5	mouse over	css=.ytd-rich-grid-renderer:nt h-child(8) > #content #video-t itle
6	mouse out	css=.ytd-rich-grid-renderer:nt h-child(8) > #content #video-t itle
7	mouse over	css=.ytd-rich-grid-renderer:nt h-child(8) > #content #thumb nail #img
8	mouse out	css=.ytd-rich-grid-renderer:nt h-child(8) > #content #thumb nail #img
9	click	css=.style-scope:nth-child(8) > #content #thumbnail #thum bnail
10	mouse over	css=.style-scope:nth-child(8) > #content #thumbnail #thum bnail
11	mouse out	css=.style-scope:nth-child(8) > #content #thumbnail #thum bnail
12	mouse over	css=.style-scope:nth-child(5) > #dismissable #thumbnail
13	mouse out	css=.style-scope:nth-child(5) > #dismissable #thumbnail
14	mouse over	css=.ytd-thumbnail:nth-child (1) > #icon

15	<i>mouse out</i>	css=.ytd-thumbnail:nth-child(1) > #icon
16	<i>mouse over</i>	css=.ytd-thumbnail:nth-child(1) > #tooltip
17	<i>mouse out</i>	css=.ytd-thumbnail:nth-child(1) > #tooltip
18	<i>mouse over</i>	css=.ytd-thumbnail:nth-child(1) > #icon
19	<i>mouse out</i>	css=.ytd-thumbnail:nth-child(1) > #icon
20	<i>mouse over</i>	css=#hover-overlays > .style-scope:nth-child(1)
21	<i>mouse out</i>	css=#hover-overlays > .style-scope:nth-child(1)
22	<i>mouse over</i>	css=.style-scope:nth-child(4) > #dismissable #img
23	<i>mouse out</i>	css=.style-scope:nth-child(4) > #dismissable #img
24	<i>click</i>	css=.video-stream

---



## Practical-3

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

### Code:

```
package practical.pkg3;

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

/**
 *
 * @author shuvo
 */
public class Practical3 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        System.setProperty("webdriver.chrome.driver",
"D:\\Practicals\\SQAT\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.get("http://www.google.com");
        driver.manage().window().maximize();
        List lists =
driver.findElements(By.xpath("//select/option"));
        System.out.println("Total no. of lists: " + lists.size());
        driver.quit();
    }
}
```

## Output:

The screenshot displays the NetBeans IDE environment where a Selenium WebDriver test is being executed. The source code in `Practical3.java` defines a `Practical3` class with a `main` method that initializes a `ChromeDriver`, navigates to `https://www.google.com/`, maximizes the window, and finds all `select` elements on the page. The output window shows the following log:

```
run:
Starting ChromeDriver 85.0.4103.37 (c8b713e5f2f4cc0f1a59d230099af05d7-refs/branch-heads/85@{head})
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on how to use Chromedriver in a production environment.
ChromeDriver was started successfully.
Sep 15, 2020 5:31:41 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
```

The browser window shows the Google search page with the 'I'm Feeling Lucky' button highlighted. The page title is 'Google' and the URL is `google.com/?gws_rd=ssl`. The page content includes the Google logo, a search bar, and the text 'Google offered in: हिन्दी বাংলা తెలుగు मराठी தமிழ் ગુજરાતી ಕನ್ನಡ ಕೊಂಕಣಿ བོད་སྐད་ཡིག་'.



## Practical-4

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

### PHP Code:

```
<?php
    if(isset($_POST['login'])) {
        $user = $_POST['user'];
        $pass = $_POST['pass'];
        if($user=="shuvo" && $pass=="1234") {
            echo "<h1>Welcome Shuvo</h1>";
        }
        else{
            echo "<h1>Invalid username or password</h1>";
        }
    }
?>
<html>
    <body><br><br>
        <form method="POST">
            <input placeholder="Enter Username" type="text"
name="user" id="user"/><br>
            <input placeholder="Enter Password" type="password"
name="pass" id="pass"/><br>
            <button name="login" id="login"
type="submit">Login</button>
        </form>
    </body>
</html>
```

## Java Code:

```
package prcatical.pkg4;

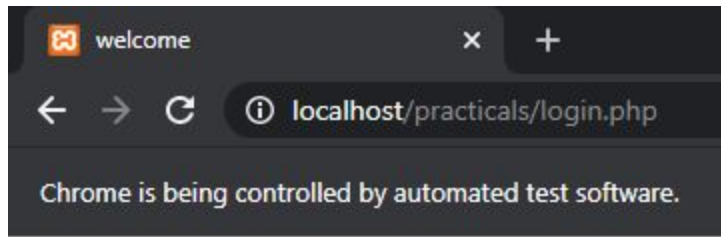
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

/**
 *
 * @author shuvo
 */
public class Prcatical4 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        System.setProperty("webdriver.chrome.driver",
"D:\\Practicals\\SQAT\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.get("http://localhost/practicals/login.php");
        driver.manage().window().maximize();
        driver.findElement(By.id("user")).sendKeys("shuvo");
        driver.findElement(By.id("pass")).sendKeys("1234");
        driver.findElement(By.id("login")).click();

        if(driver.getTitle().contains("welcome")){
            System.out.print("Login Successful!");
        }
        else{
            System.out.print("Login Failed!");
        }
    }
}
```

## Output:



# Welcome Shuvo

```
Output - Prcatical 4 (run) X
run:
Starting ChromeDriver 85.0.4183.87 (cd6713ebf92falcacc0fla598df280093af
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations fo
ChromeDriver was started successfully.
Oct 11, 2020 11:21:20 AM org.openqa.selenium.remote.ProtocolHandshake c
INFO: Detected dialect: W3C
Login Successful!BUILD SUCCESSFUL (total time: 18 seconds)
```

## Practical-5

**Aim: Write and Test a program to update 10 student record excel sheets.**

### Code:

```
package sepract5;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import jxl.Sheet;
import jxl.Workbook;
import jxl.write.Label;
import jxl.write.WritableSheet;
import jxl.write.WritableWorkbook;

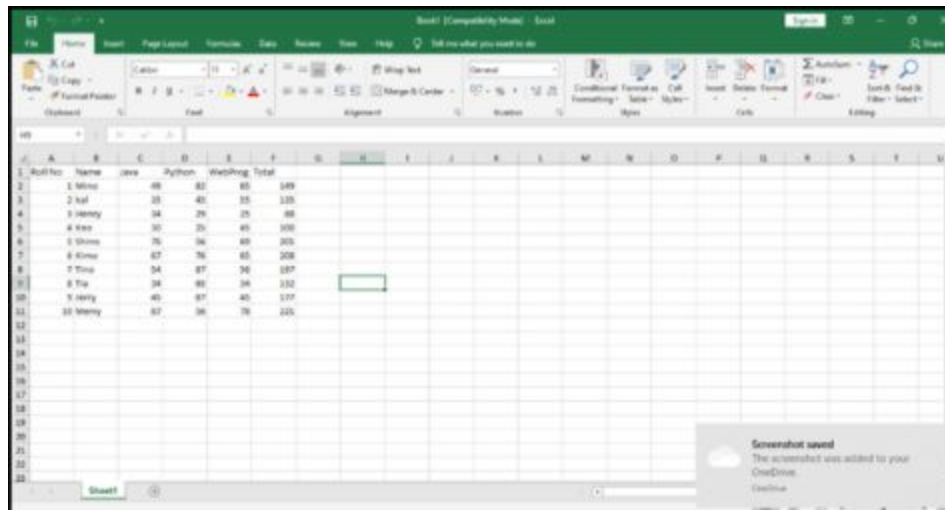
/**
 *
 * @author shuvo
 */
public class SEPract5 {
    private static Object WorkBook;

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) throws Exception{
        // TODO code application logic here
        FileInputStream fi = new
        FileInputStream("D:\\pracs\\STQA\\Book1.xls");
        Workbook w=Workbook.getWorkbook(fi);
        Sheet s=w.getSheet(0);
        String a[][]=new String[s.getRows()][s.getColumns()];
        FileOutputStream to=new
        FileOutputStream("D:\\pracs\\STQA\\Book2(646).xls");
        WritableWorkbook wwb=Workbook.createWorkbook(to);
```

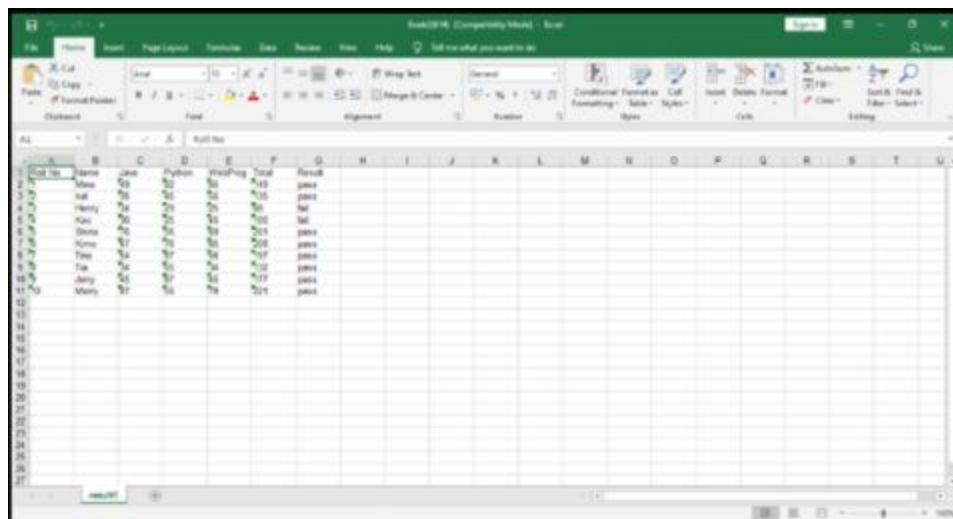
```
WritableSheet ws=wwb.createSheet("result1", 0);
for(int i=0; i<s.getRows();i++){
    for(int j=0;j<s.getColumns();j++){
        a[i][j]=s.getCell(j,i).getContents();
        Label l=new Label(j, i, a[i][j]);
        ws.addCell(l);
    }
}
Label l=new Label(6,0,"Result");
ws.addCell(l);
for(int i=1;i<s.getRows();i++){
    for(int j=5;j<s.getColumns();j++){
        a[i][j]=s.getCell(j,i).getContents();
        int x=Integer.parseInt(a[i][j]);
        if(x/3>=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();

}
}
```

## Output:



Roll No	Name	Java	Python	JavaScript	Total
1	Mina	88	82	85	155
2	Kul	35	40	35	110
3	Henry	34	29	25	88
4	Kari	30	25	40	95
5	Shona	76	74	60	210
6	Klara	67	76	65	208
7	Tim	34	87	36	157
8	Tia	34	80	34	148
9	Arly	40	87	40	167
10	Manny	87	36	76	199



Roll No	Name	Java	Python	JavaScript	Total	Result
1	Mina	88	82	85	155	pass
2	Kul	35	40	35	110	pass
3	Henry	34	29	25	88	fail
4	Kari	30	25	40	95	fail
5	Shona	76	74	60	210	pass
6	Klara	67	76	65	208	pass
7	Tim	34	87	36	157	pass
8	Tia	34	80	34	148	pass
9	Arly	40	87	40	167	pass
10	Manny	87	36	76	199	pass

## Practical-6

**Aim: Write and test a program to select the no. of students who have scored more than 60 in anyone subject.**

### Code:

```
package staqpract6;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import jxl.Sheet;
import jxl.Workbook;
import jxl.read.biff.BiffException;
import jxl.write.Label;
import jxl.write.WritableSheet;
import jxl.write.WritableWorkbook;
import jxl.write.WriteException;

/**
 *
 * @author shuvo
 */
public class Staqpract6 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) throws Exception{
        FileInputStream fi = new
FileInputStream("D:\\pracs\\STQA\\Book1 (AutoRecovered).xls");
        Workbook w = Workbook.getWorkbook(fi);
        Sheet s = w.getSheet(0);
        String a[][]= new String[s.getRows()][s.getColumns()];
```

```
        FileOutputStream fo = new
FileOutputStream("D:\\pracs\\STQA\\Book2 (AutoRecovered).xls");
        WritableWorkbook wwb = Workbook.createWorkbook(fo);
        WritableSheet ws = wwb.createSheet("result1", 0);
        System.out.println();
        for (int i=0; i<s.getRows(); i++){
            String temp[] = new String[s.getColumns()];
            boolean flag = false;
            for(int j=0; j<s.getColumns(); j++){
                temp[j] = s.getCell(j,i).getContents();
                if(i >= 1 && j>=2 && j<=4){
                    if(Integer.parseInt(temp[j]) >= 60){
                        flag = true;
                    }
                }
            }
            if(flag){
                for(int k=0; k<temp.length;k++){
                    Label l2 = new Label(k,i,temp[k]);
                    ws.addCell(l2);
                }
            }
        }
        wwb.write();
        wwb.close();
    }
}
```



## Output:

Excel spreadsheet showing student performance data. The table has columns for Roll No., Name, Java, Python, WebProg, and Total. The data is as follows:

Roll No.	Name	Java	Python	WebProg	Total
1	Shree	77	88	40	217
2	Bhatt	45	69	70	184
3	Maurya	80	48	33	219
4	Shru	79	55	89	223
5	Chir	90	45	59	194
6	Pani	97	69	60	252
7	Balachala	80	50	69	199
8					
9					
10	Padma	70	90	79	229
11	Krish	33	44	62	182
12					
13					
14					
15					
16					

## Practical-7

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

### Code:

```
package practicals7;

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;
/**
 *
 * @author shuvo
 */
public class Practicals7 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        System.setProperty("webdriver.chrome.driver",
"D:\\Practicals\\SQAT\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.get("http://www.google.com");
        driver.manage().window().maximize();
        List<WebElement> links =
driver.findElements(By.tagName("a"));
        List<WebElement> buttons =
driver.findElements(By.tagName("button"));
        List<WebElement> fields =
driver.findElements(By.tagName("input"));
```

```
        System.out.println("Total No. of links = " +
links.size());
        System.out.println("Total No. of buttons = " +
buttons.size());
        System.out.println("Total No. of fields = " +
fields.size());
        driver.quit();
    }

}
```

## Output:

```
run:
Starting ChromeDriver 85.0.4183.87 (cd6713ebf92falcacc0f1a598df280093af0c5d7-refs/branch-heads/41830(#1689)) on port 20713
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Sep 15, 2020 5:33:11 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Total No. of links = 29
Total No. of buttons = 1
Total No. of fields = 13
BUILD SUCCESSFUL (total time: 20 seconds)
```

## Practical-8

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

### Code:

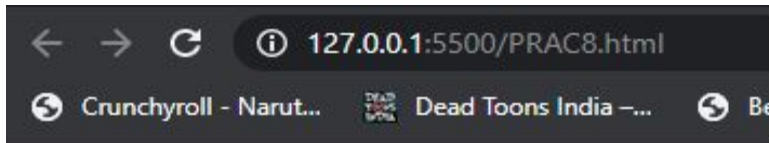
```
package practical8;

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

/**
 *
 * @author shuvo
 */
public class Practical8 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        System.setProperty("webdriver.chrome.driver",
"D:\\Practicals\\SQAT\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.get("http://127.0.0.1:5500/PRAC8.html");
        driver.manage().window().maximize();
        List lists =
driver.findElements(By.xpath("//select/option"));
        List lists1 = driver.findElements(By.xpath("//ul/li"));
        System.out.println("Total no. of Wifus: " +
lists1.size());
```

```
        System.out.println("Total no. of Character: " +  
lists.size());  
        driver.quit();  
  
    }  
  
}
```

**Output:**

# Prac8

## Wifus

- Emilia
- Erza
- Megumi Tadokoro
- Lucy Heartfilia
- Rem
- Akame

Choose a Character:

```
run:
Starting ChromeDriver 85.0.4183.87 (cd6713ebf
Only local connections are allowed.
Please see https://chromedriver.chromium.org/
ChromeDriver was started successfully.
Sep 15, 2020 9:09:48 PM org.openqa.selenium.r
INFO: Detected dialect: W3C
Total no. of Wifus: 6
Total no. of Character: 5
BUILD SUCCESSFUL (total time: 8 seconds)
```

## Practical-9

**Aim:** Write and test a program to count the number of checkboxes on the page checked and unchecked count.

### Code:

```
package practical9;

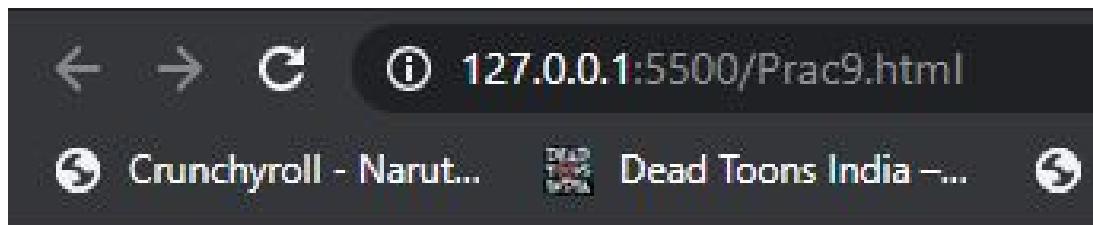
import java.util.List;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;

/**
 *
 * @author shuvo
 */
public class Practical9 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        System.setProperty("webdriver.chrome.driver",
"D:\\Practicals\\SQAT\\chromedriver.exe");
        WebDriver driver = new
org.openqa.selenium.chrome.ChromeDriver();
        driver.get("http://127.0.0.1:5500/Prac9.html");
        driver.manage().window().maximize();
        List check =
driver.findElements(By.xpath("//input[@type='checkbox']"));
        System.out.println("Total no of checkboxes : " + check.size());
        int checked_Count=0,unchecked_Count=0;
        for(int i=0;i<check.size();i++) {
            if(check.get(i).isSelected()){
                checked_Count++;
            }
        }
    }
}
```

```
        else{ unchecked_Count++; }
    }
    System.out.println("Number of selected checkboxes are : " +
checked_Count);
    System.out.println("Number of unselected checkboxes are : " +
unchecked_Count);
    driver.quit();
}
}
```

## Output:



# Show Checkboxes

- ☒ Bike
- ☒ Car
- ☐ Boat
- ☐ Scooter
- ☐ Cycle
- ☒ Jet
- ☒ Train



```
run:
Starting ChromeDriver 85.0.4183.87 (cd6713e9f9a970000000000000000000)
Only local connections are allowed.
Please see https://chromedriver.chromium.org/
ChromeDriver was started successfully.
Sep 15, 2020 5:40:08 PM org.openqa.selenium.remote
INFO: Detected dialect: W3C
Total no of checkboxes : 7
Number of selected checkboxes are : 4
Number of unselected checkboxes are : 3
BUILD SUCCESSFUL (total time: 10 seconds)|
```