



Project Report  
On  
**Automation Testing On ToolsQA Website**  
By

**GAYATRI CHAUDHARI**

*School of coding certified software tester*

Batch: - 2020-5334

Under the Guidance of,  
Mrs. Priyanka Desle  
(Technical Trainer)  
**EduBridge**

## **Introduction:**

This Project aims to do overall testing like UI testing to check quality of application. It achieves accuracy and reliability of website and saves time for manually doing this testing. In this project our main focus is to check functionalities available on website. Also it includes title check; maximize the screen, screen shot, file uploading and other elements with different locators and their working status.

## **Software Requirements:**

Operating System: - Window 10

Browser: - Latest version of Chrome

IDE: - Eclipse IDE

Programming Language :- Java

Testing Framework :- TestNG

## Project Script:

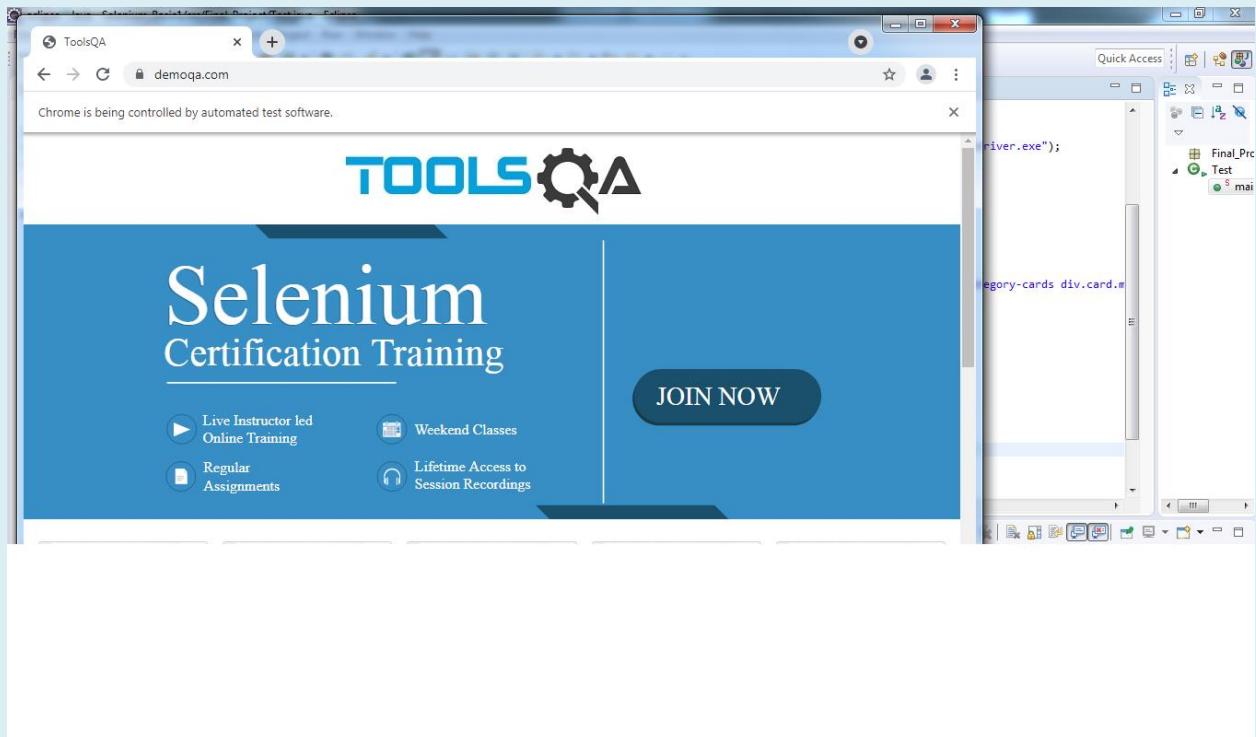
### Module No.1 - Elements

TC01- Launch the chrome browser

```
System.setProperty("webdriver.chrome.driver",  
"C:\\\\Users\\\\Surabhi\\\\Desktop\\\\Selenium jar\\\\chromedriver.exe");  
WebDriver w=new ChromeDriver();
```

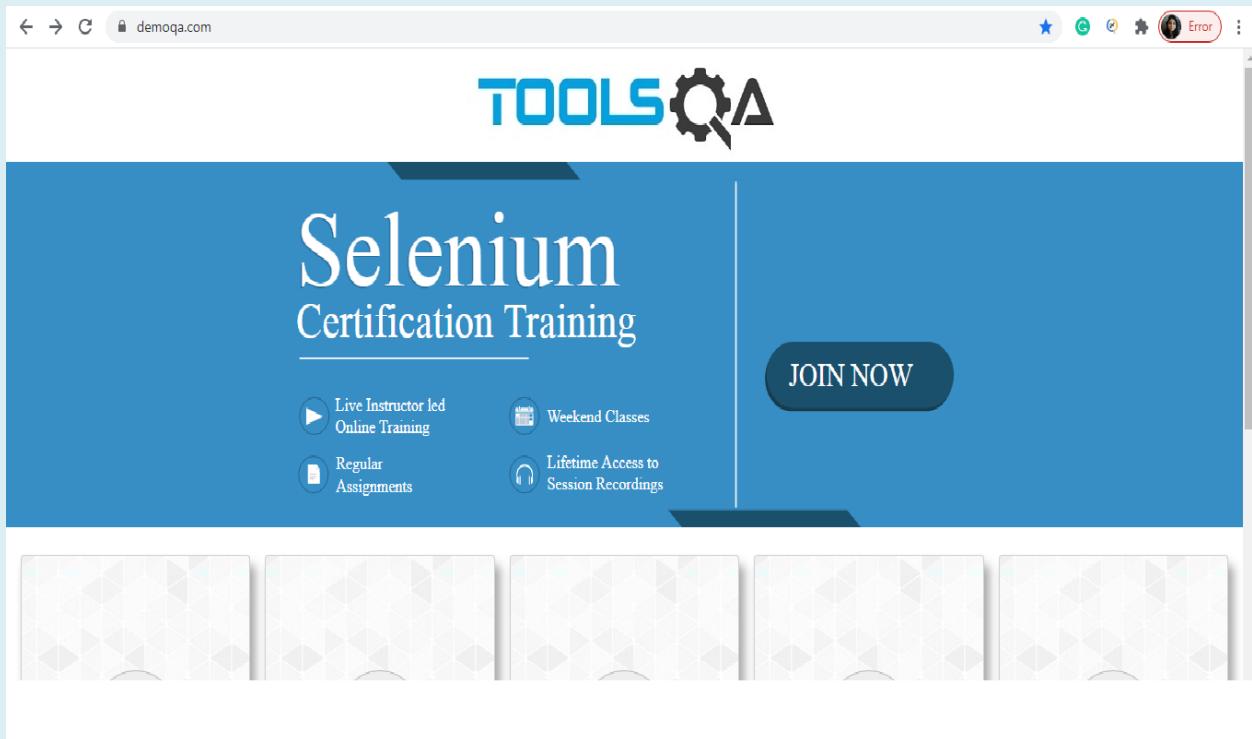
TC02- Open toolsQA URL

```
w.get ("https://www.demoqa.com/");
```



## TC03- To Maximize the window

```
w.manage().window().maximize(); //for maximize the window
```



## TC04- To Get current page URL

```
System.out.println (w.getCurrentUrl());
```

**OUTPUT-** https://demoqa.com/

## TC05- To scroll down

```
JavascriptExecutor js = (JavascriptExecutor) w;  
js.executeScript ("window.scrollTo (0, 1000);")
```

## TC06- To click on Elements field

```
w.findElement(By.cssSelector("div.body-height:nth-child(2) div.home-content  
div.home-body div.category-cards div.card.mt-4.top-card:nth-child(1) div:nth-child(1)  
div.avatar.mx-auto.white > svg:nth-child(1)").click();
```

## TC07- To get page title

```
System.out.println (w.getTitle ());
```

**OUTPUT-** ToolsQA

## TC08- To click on Text box field

```
w.findElement(By.id("item-0")).click();
```

## TC09- To enter name in name field

```
w.findElement(By.id("username")).sendKeys("Sharayu D");
```

## TC10- To enter Email

```
w.findElement(By.id("userEmail")).sendKeys("surabhi759@gmail.co  
m");
```

**TC11- To enter current address in text field**

```
w.findElement(By.id("currentAddress")).sendKeys("New koradi");
```

**TC12- To enter permanent address in text field**

```
w.findElement(By.id("permanentAddress")).sendKeys("Plot. no. 175,New Koradi");
```

**TC13- To scroll down**

```
JavascriptExecutor js1 = (JavascriptExecutor) w;  
js1.executeScript("window.scrollTo(0,200);");
```

**TC14- To click on submit button**

```
w.findElement(By.xpath("//button[@id='submit']")).click();
```

**TC15- To print output in console**

```
System.out.println("Registration done successfully!!!!");
```

**OUTPUT- Registration done successfully!!!!**

**TC16- To click on check boxes**

```
w.findElement(By.id("item-1")).click();
```

## TC17- To click on home check box

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/ol[1]/li[1]/span[1]/button[1]/*[1]")).click();
```

## TC18- To click on desktop check box

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/ol[1]/li[1]/ol[1]/li[1]/span[1]/label[1]/span[1]/*[1]")).click();
```

## TC19- To click on documents check box

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/ol[1]/li[1]/ol[1]/li[2]/span[1]/label[1]/span[1]/*[1]")).click();
```

## TC20- To click on documents check box

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/ol[1]/li[1]/span[1]/button[1]/*[1]")).click();
```

## TC21- To click on radio button field

```
w.findElement(By.xpath("//span[contains(text(),'Radio Button')]")).click();
```

TC22- To click on “Impressive” radio button

```
w.findElement(By.xpath("//label[contains(text(),'Impressive')]")).click();
```

TC23- To print and get text of the radio button in console

```
System.out.println(w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/p[1]")).getText());
```

**OUTPUT-** You have selected Impressive

TC24- To click on “Yes” radio button

```
w.findElement(By.xpath("//*[@id=\"app\"]/div/div/div[2]/div[2]/div[1]/div[2]/label")).click();
```

TC25- To print and get text of the radio button in console

```
System.out.println(w.findElement(By.xpath("//*[@id=\"app\"]/div/div/div[2]/div[2]/div[1]/div[2]/label")).getText());
```

**OUTPUT-** You have selected yes

TC24- To click on “No” radio button

```
w.findElement(By.xpath("//*[@id=\"app\"]/div/div/div[2]/div[2]/div[1]/div[4]/label")).click();
System.out.println("you select
"+w.findElement(By.xpath("//*[@id=\"app\"]/div/div/div[2]/div[2]/div[1]/div[4]/label")).getText());
```



**OUTPUT/DEFECT**- Enable to click on “NO” radio button

TC25- To scroll down

```
JavascriptExecutor js2 = (JavascriptExecutor) w;  
js2.executeScript("window.scrollTo(0,200);")
```

TC26- Click on web tabs

```
w.findElement(By.id("item-3")).click();
```

TC27- Click on add button

```
w.findElement(By.id("addNewRecordButton")).click();
```

TC28- To enter the name in first name text field

```
w.findElement(By.id("firstName")).sendKeys("Sharayu");
```

TC29- To enter the last name in last name text field

```
w.findElement(By.id("lastName")).sendKeys("Durugkar");
```

TC30- To enter the user Email in user Email text field

```
w.findElement(By.id("userEmail")).sendKeys("surabhi7591@gmail.com");
```

TC31- To enter the age in age text field

```
w.findElement(By.id("age")).sendKeys("56");
```

TC32- To enter the salary in salary text field

```
w.findElement(By.id("salary")).sendKeys("50000000");
```

TC33- To enter the data in department text field

```
w.findElement(By.id("department")).sendKeys("IT");
```

TC34- Click on submit button and print on console

```
w.findElement(By.id("submit")).click();
System.out.println("Added successfully!!!");
```

**OUTPUT-** Added successfully!!!

TC35- Click on Search button

```
w.findElement(By.id("searchBox")).sendKeys("Sharayu");
w.findElement(By.id("basic-addon2")).click();
```

## TC36- Click on buttons field

```
w.findElement(By.id("item-4")).click();
```

## TC37- To double click on button and print on console

```
Actions A=new Actions(w);
WebElement elementLocator =
w.findElement(By.id("doubleClickBtn"));
A.doubleClick(elementLocator).perform();
System.out.println("Double clicked");
```

**OUTPUT-** Double clicked

## TC38- To right click on button and print on console

```
A.contextClick(w.findElement(By.id("rightClickBtn"))).perform();
System.out.println("Right clicked");
```

**OUTPUT-** Right clicked

## TC39- To click on click button and print on console

```
A.click(w.findElement(By.id("Iv2dU"))).perform();
System.out.println("clicked successfully!!!!");
```



**OUTPUT/DEFECT-** Enable to click on click button

## TC40- To scroll down

```
JavascriptExecutor js3 = (JavascriptExecutor) w;  
js3.executeScript("window.scrollTo(0,1000);")
```

## TC41- Click on links field

```
w.findElement(By.id("item-5")).click();
```

## TC42- Click on “Home” link, opens new window and back to current window and get output on console

```
w.findElement(By.LinkText("Home")).click();  
Set <String> windows = w.getWindowHandles();  
Iterator <String> IT = windows.iterator();  
String parent_window = IT.next();  
String child_window = IT.next();  
w.switchTo().window(child_window);  
Thread.sleep(10000);  
w.switchTo().window(parent_window);  
System.out.println(" Window open successfully!!!!");
```

**OUTPUT-** Window open successfully!!!!

## TC43- Click on “Created” link, scroll down and print output on console

```
w.findElement(By.LinkText("Created")).click();  
JavascriptExecutor js4 = (JavascriptExecutor) w;  
js4.executeScript("window.scrollTo(0,200);")  
System.out.println("Link has responded with status 201 and  
status text Created");
```

**OUTPUT-** Link has responded with status 201 and status text Created

TC44- Click on “No content” link and print the output

```
w.findElement(By.linkText("No Content")).click();  
System.out.println("Link has responded with status 204 and  
status text No Content");
```

**OUTPUT-** Link has responded with status 204 and status text No Content

TC45- Click on “Moved” link and print the output

```
w.findElement(By.linkText("Moved")).click();  
System.out.println("Link has responded with status 204 and  
status text moved permanently");
```

**OUTPUT-** Link has responded with status 204 and status text moved permanently

TC45- Click on “Bad Request” link and print the output

```
w.findElement(By.linkText("Bad Request")).click();  
System.out.println("Link has responded with status 400 and  
status text Bad Request");
```

**OUTPUT-** Link has responded with status 400 and status text Bad Request

## TC46- Click on “Unauthorized” link and print the output

```
w.findElement(By.linkText("Unauthorized")).click();
System.out.println("Link has responded with status 401 and
status text Unauthorized");
```

**OUTPUT-** Link has responded with status 401 and status text Unauthorized

## TC46- Click on “Forbidden” link and print the output

```
w.findElement(By.linkText("Forbidden")).click();
System.out.println("Link has responded with status 403 and
status text Forbidden");
```

**OUTPUT-** Link has responded with status 403 and status text Forbidden

## TC46- Click on “Not Found” link and print the output

```
w.findElement(By.linkText("Not Found")).click();
System.out.println("Link has responded with status 404 and
status text Not Found");
```

**OUTPUT-** Link has responded with status 404 and status text Not Found

## TC47- To scroll down

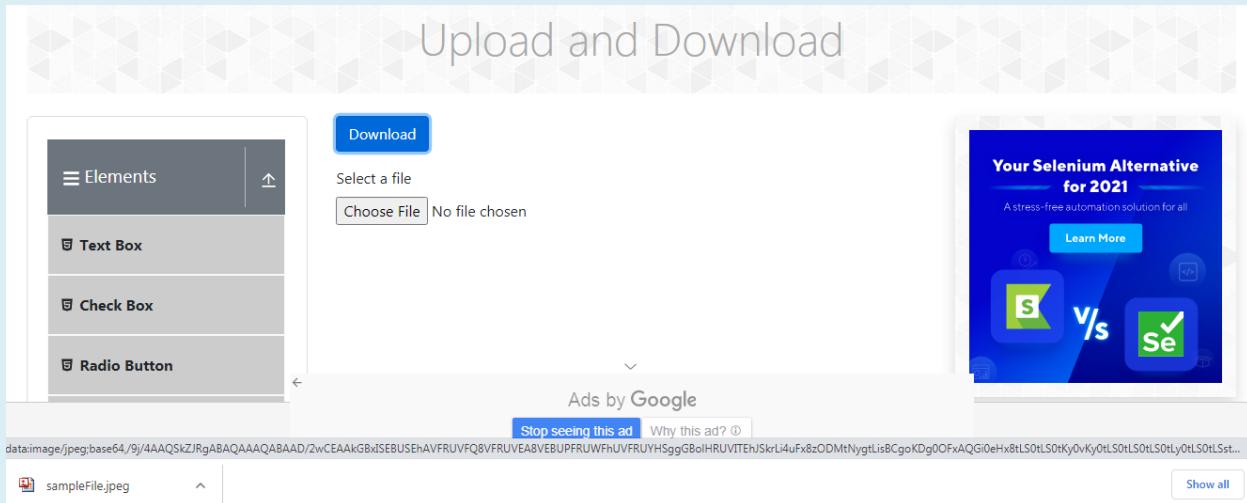
```
JavascriptExecutor js5 = (JavascriptExecutor) w;
js5.executeScript("window.scrollTo(0,1000);")
```

## TC48-Click on “Upload Download” field

```
w.findElement(By.xpath("//span[contains(text(),'Upload and Download')])).click();
```

## TC49-Click on “Download” field and print on output

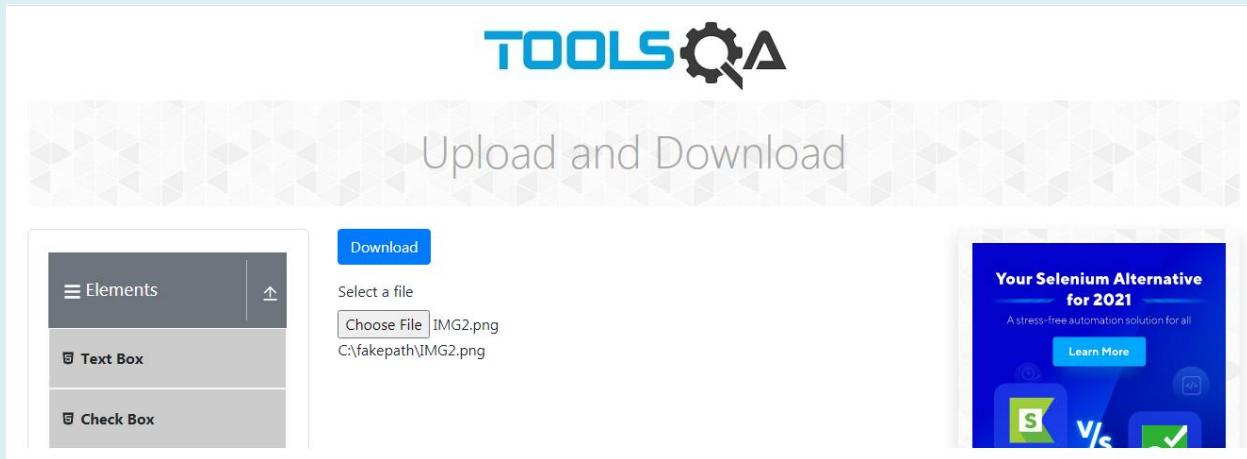
```
w.findElement(By.id("downloadButton")).click();
System.out.println("Downloaded successfully");
```



**OUTPUT-** Downloaded successfully

## TC50-Click on “Upload file” field and print on output

```
w.findElement(By.xpath("//input[@id='uploadFile']")).sendKeys("C:\\\\Users\\\\Surabhi\\\\Desktop\\\\project\\\\IMG2.png ");
System.out.println("File successfully uploaded!!!");
```



**OUTPUT- File successfully uploaded!!!**

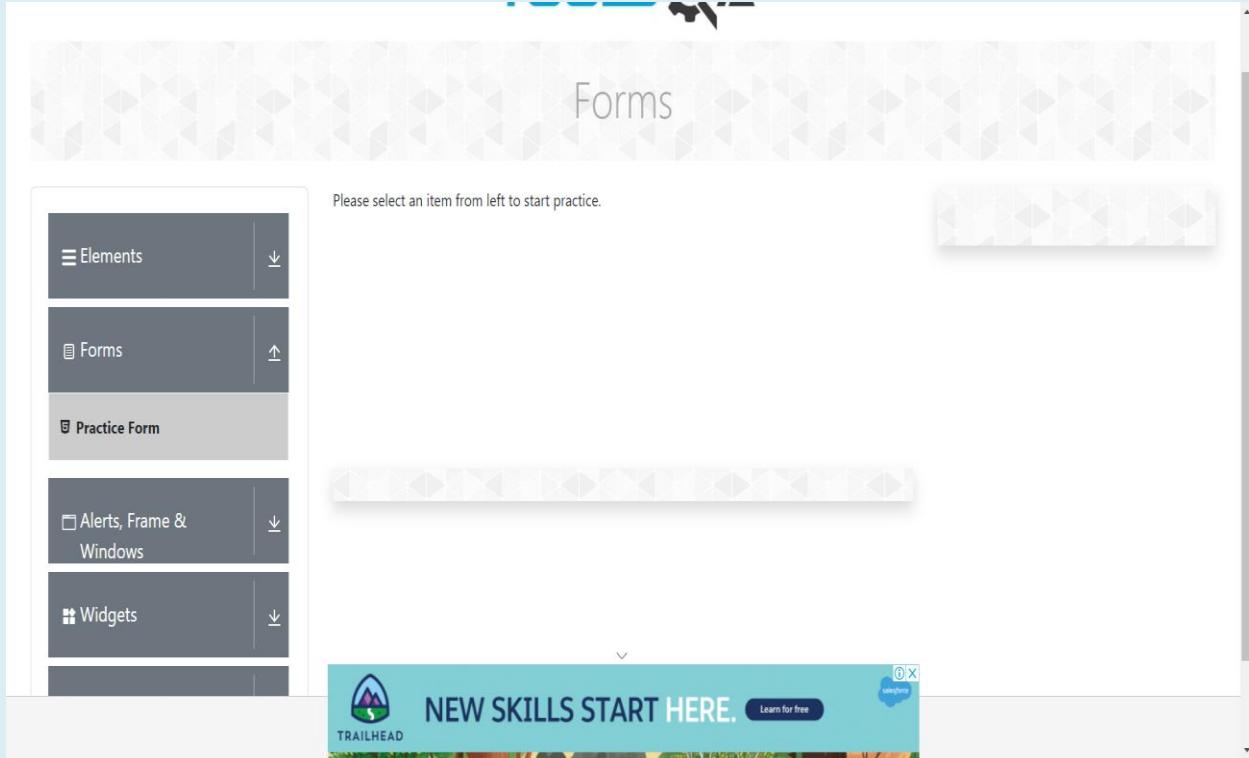
## **Module No.2 - Forms**

**TC51: To Click on Forms field Vertically Scroll Down**

```
JavascriptExecutor js = (JavascriptExecutor) w;  
  
js.executeScript("window.scrollBy(0,500)");  
  
WebElement E=w.findElement(By.xpath("/html[1]/body  
[1]/div[2]/div[1]/dxiv[1]/div[2]/div  
[1]/div[2]/div[1]/div[3]/h5[1]"));
```

**TC52: To take screenshot for forms option**

```
TakesScreenshot t1= ( TakesScreenshot) w;  
  
File src1=t1.getScreenshotAs(OutputType.FILE)  
  
Files.copy(src1, new File  
("D:\\Selenium_Screenshot\\Forms.png"));
```



TC53: To click on Practice Form.

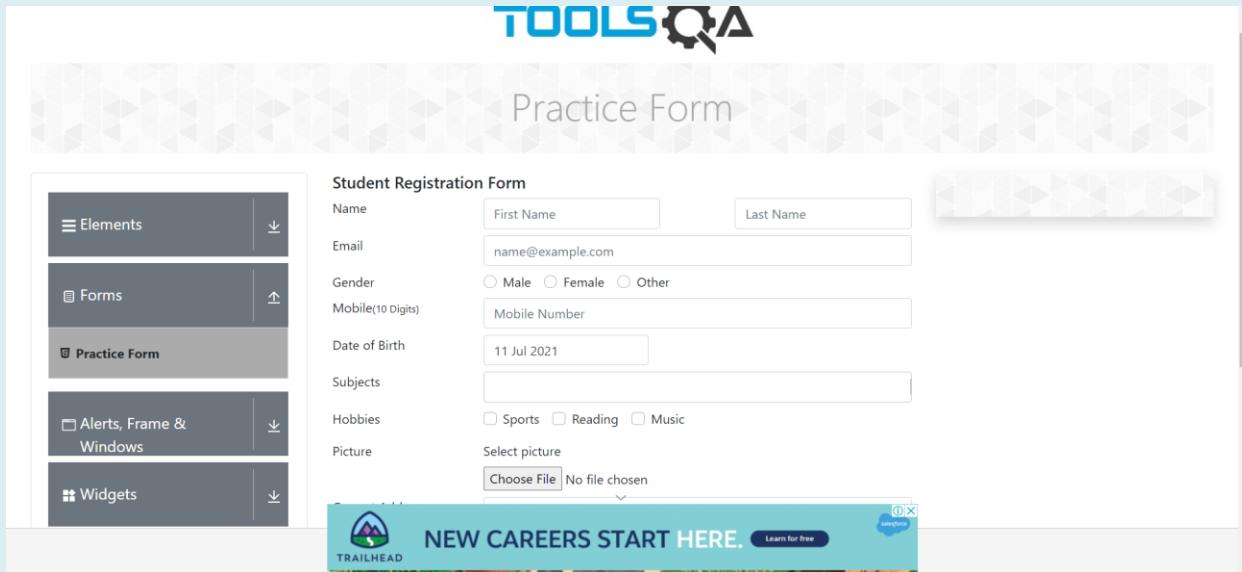
```
w.findElement(By.cssSelector("div.body-height:nth-child(2) div.container.playgound-body div.row div.col-12.mt-4.col-md-3:nth-child(1) div.left-pannel div.accordion div.element-group:nth-child(2) div.element-list.collapse.show ul.menu-list li.btn.btn-light > span.text:nth-child(2)").click();
```

TC54: Maximize the window

```
w.manage().window().maximize();
```

TC55:To take screenshot for Practice form option

```
TakesScreenshot t2= ( TakesScreenshot ) w;  
File src2=t2.getScreenshotAs(OutputType.FILE);  
Files.copy(src2, new  
File("D:\\Selenium_Screenshot\\Practice.png"));
```



TC56 : To Check the functionality of student registration form with valid First Name, Last Name, Email Id

```
w.findElement(By.cssSelector("input#firstName")).sendKeys("Shaurya");  
w.findElement(By.cssSelector("input#lastName")).sendKeys("Gejage");  
w.findElement(By.cssSelector("input#userEmail")).sendKeys("shambhugejage@gmail.com");
```

TC57: To Check the functionality of student registration form with selecting valid Gender

```
w.findElement(By.xpath("/html/body/div/div/div/div[2]/div[2]/div[1]/form/div[3]/div[2]/div[1]/label")).click(); //Gender
```

TC58: To Check the functionality of student registration form by entering valid mobile number

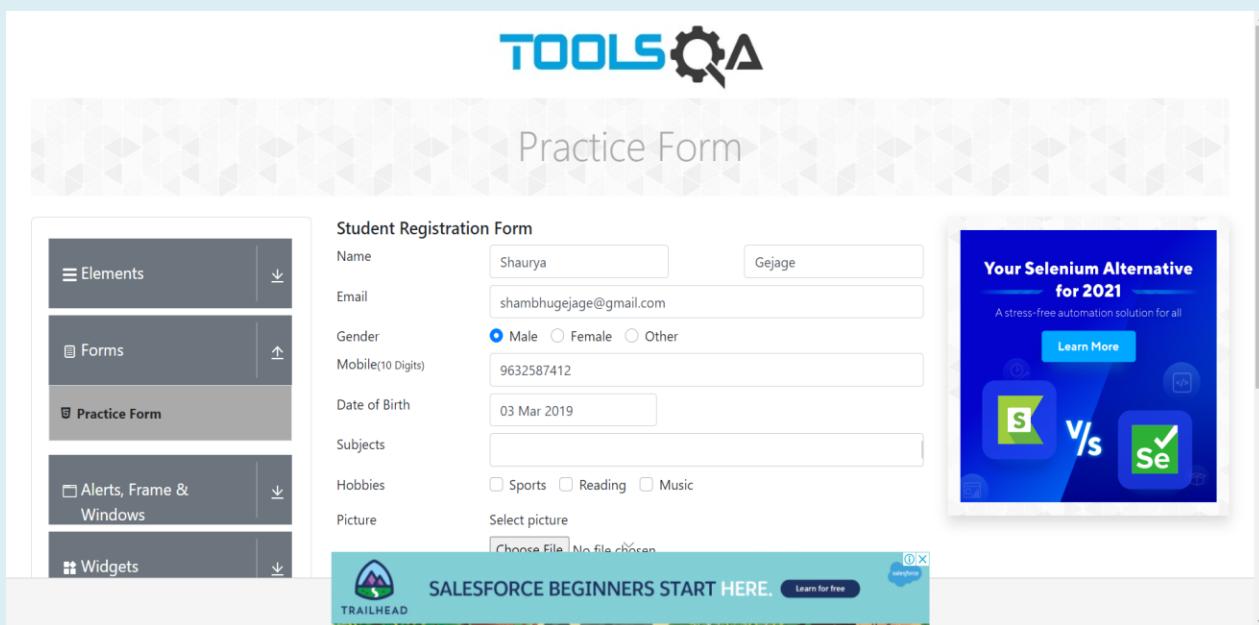
```
w.findElement(By.cssSelector("input#userNumber")).sendKeys(  
"9632587412"); //MobileNo
```

TC59: To Check the functionality of student registration form with selecting valid Data of birth.

```
w.findElement(By.id("dateOfBirthInput")).click(); //Select  
Date box  
  
Select s=new  
Select(w.findElement(By.xpath("/html/body/div/div/div/div[2]  
/div[2]/div[1]/form/div[5]/div[2]/div[2]/div[2]/div/div/div[  
2]/div[1]/div[2]/div[1]/select")));  
s.selectByVisibleText("March");//month  
  
Select s1=new  
Select(w.findElement(By.xpath("/html/body/div/div/div/div[2]  
/div[2]/div[1]/form/div[5]/div[2]/div[2]/div[2]/div/div/div[  
2]/div[1]/div[2]/div[2]/select")));  
s1.selectByVisibleText("2019");//year  
  
w.findElement(By.xpath("/html/body/div/div/div/div[2]/div[2]  
/div[1]/form/div[5]/div[2]/div[2]/div[2]/div/div/div[2]/div[  
2]/div[2]/div[1]")).click();//date
```

TC60: To take screenshot for Student registration form with valid First Name, Last Name, Email Id, with selecting Gender by entering valid mobile number and selecting valid Data of birth.

```
TakesScreenshot t3= (TakesScreenshot) w;  
File src3=t3.getScreenshotAs(OutputType.FILE);  
Files.copy(src3, new File  
("D:\\Selenium_Screenshot\\BeforSubjectExecution.png"));
```



TC61 : To Check the functionality of student registration form with selecting valid Subject

```
w.findElement(By.cssSelector("div.body-height:nth-child(2)  
div.container.playgound-body div.row div.col-12.mt-4.col-md-  
6:nth-child(2) div.practice-form-wrapper div.mt-2.row:nth-  
child(6) div.col-md-9.col-sm-12 div.css-2b097c-container  
div.subjects-auto-complete__control.css-yk16xz-  
control>div.subjects-auto-complete__value-  
container.subjects-auto-complete__value-container--is-  
multi.css-1hwfws3")).sendKeys("Commerce"); //subject
```

**OUTPUT:-** Subject is not entering properly therefore it is Defect

**TC62 : To Raise defect for Subject in Jira Tool**

The screenshot shows the Jira Software interface. On the left, there's a sidebar with 'First\_batch\_Jira' selected under 'Software project'. The main area is titled 'Issues' and shows a single issue titled 'SK101\_5334 : When selecting a subject the subject is not selected properly'. The issue details pane shows the reporter is 'testereEDU' and the assignee is also 'testereEDU'. The description field contains steps to reproduce the bug: '1.Launch Browser. 2.Open URL - https://demona.com/'. A pinned field 'Reporter' is also visible.

**TC63 : To Check the functionality of student registration form click on hobbies Checkbox button with valid hobby name**

```
w.findElement(By.cssSelector("div.body-height:nth-child(2) div.container.playgound-body div.row div.col-12.mt-4.col-md-6:nth-child(2) div.practice-form-wrapper form:nth-child(2) div.mt-2.row:nth-child(7) div.col-md-9.col-sm-12 div.custom-control.custom-checkbox.custom-control-inline:nth-child(1) > label.custom-control-label")).click(); //Sports

w.findElement(By.cssSelector("div.body-height:nth-child(2) div.container.playgound-body div.row div.col-12.mt-4.col-md-6:nth-child(2) div.practice-form-wrapper form:nth-child(2) div.mt-2.row:nth-child(7) div.col-md-9.col-sm-12 div.customcontrol.customcheckbox.custom-control-inline:nth-child(2)>label.custom-controllabel")).click(); //reading
```

TC64: To Check the functionality of student registration form with uploading picture

```
w.findElement(By.cssSelector("#uploadPicture")).sendKeys("C:\\\\Users\\\\Rohan\\\\OneDrive\\\\Desktop\\\\Peacock.jpg");//UploadPicture
```

TC65: To Check the functionality of student registration form enter valid Current address

```
w.findElement(By.cssSelector("#currentAddress")).sendKeys("Oasis City,Narhe Pune"); //Address
```

TC66:To Click on Submit button Scroll Down Vertically

```
JavascriptExecutor js1 = (JavascriptExecutor) w;  
js1.executeScript("window.scrollBy(0,1000);");
```

```
w.findElement(By.id("submit")).click();
```

TC67: To Check the functionality of student registration form with selecting valid State and City.

```
Select S1=new Select(w.findElement(By.cssSelector("div.body-height:nth-child(2) div.container.playgound-body div.row div.col-12.mt-4.col-md-6:nth-child(2) div.practice-form-wrapper div.mt-2.row:nth-child(10) div.col-md-4.col-sm-12:nth-child(2) div.css-2b097c-container div.css-yk16xz-control > div.css-1hwfws3"))); //State  
  
S1.selectByVisibleText("Uttar Pradesh");
```

```

Select S2=new Select(w.findElement(By.cssSelector("div.body-height:nth-child(2) div.container.playgound-body div.row div.col-12.mt-4.col-md-6:nth-child(2) div.practice-form-wrapper div.mt-2.row:nth-child(10) div.col-md-4.col-sm-12:nth-child(3) div.css-2b097c-container div.css-yk16xz-control > div.css-1hwfws3"))); //City

S2.deselectByVisibleText("Agra");

```

**OUTPUT:-** State and City is selecting properly therefore it is Defect

## TC68: To Raise defect for State and City in Jira Tool

The screenshot shows a Jira software interface. On the left, there's a sidebar with project navigation: First\_batch\_Jira Software project, Board, Reports, Issues (which is selected), Components, Releases, Project pages, Add-ons, Add item, and Project settings. Below this, a message says "You're in a company-managed project" and "Learn more". The main content area displays an issue titled "SK102\_5334 : When we selecting state and city at that time State and City is not selected properly". The issue details include:

- Description: Steps-  
1.Launch Browser.  
2.Open URL : <https://demoqa.com/>  
3.Scroll down upto Forms option and then click on that  
4.After that Click on Practice form  
5.In that form when we select the State as "Uttar Pradesh" and City name "Agra" then it does not show in that text field  
6.Click on sumit Button
- Environment: Window 10  
Google Chrome Version: 93.4.4177.130
- Buttons: Attach, Create subtask, Link issue, qTest: Test Execution
- Right panel (Issue Details): Open, Reporter: testereEDU, Assignee: testereEDU, Labels: None, Due date: Jul 11, 2021, Priority: High, Show 6 more fields (Original estimate, Time tracking, Epic Link, Components, Fix ...)
- Bottom right: Configure, Created 14 seconds ago, Updated 14 seconds ago

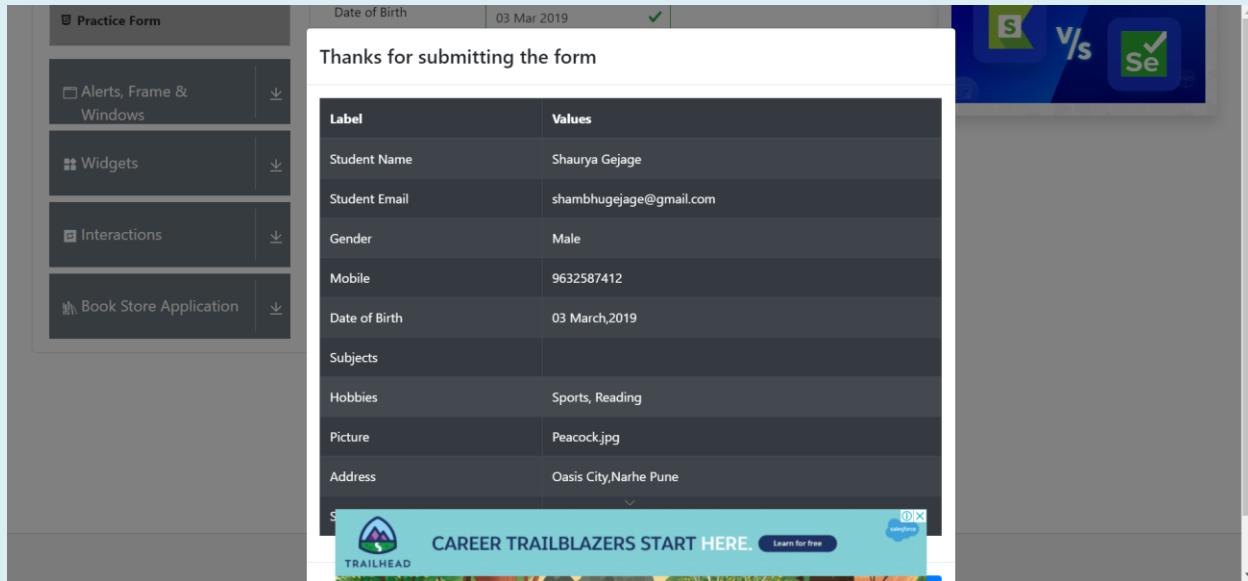
## TC69:-To Take Screenshot For information of the student

```

TakesScreenshot t4= (TakesScreenshot) w;
File src4=t4.getScreenshotAs(OutputType.FILE);
Files.copy(src4, new File
("D:\\Selenium_Screenshot\\Registation.png"));
Thread.sleep(2000);

```

This form will look like this after filling all the



## Module No.2 - Alerts, Frame and Windows

TC70: To click on Alerts, Frame and Windows option.  
Scroll down vertically

```

JavascriptExecutor js = (JavascriptExecutor) w;
js.executeScript("window.scrollBy(0,500)");
WebElement E= w.findElement(By.xpath("/html[1]/body[1]/
div[2]/div[1]/div[1]/div[2]/div[1]/div[3]/div[1]/div[3]/h5[1]"));

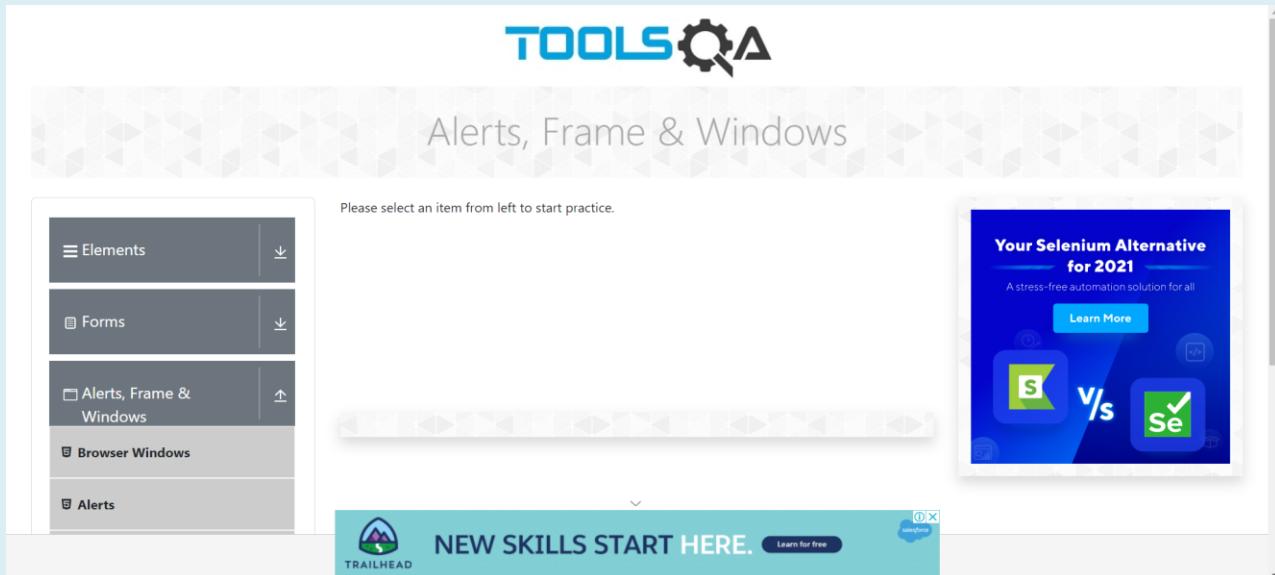
js.executeScript("arguments[0].scrollIntoView()", E);

E.click();

```

## TC71:-To Take Screenshot For Alerts, Frame and Windows

```
TakesScreenshot t5= (TakesScreenshot) w;  
File src5=t5.getScreenshotAs(OutputType.FILE);  
Files.copy(src5, new File  
("D:\\Selenium_Screenshot\\Main_Option.png"));
```



### A . Browser Window

## TC72: To Check the functionality of Browser Windows option.

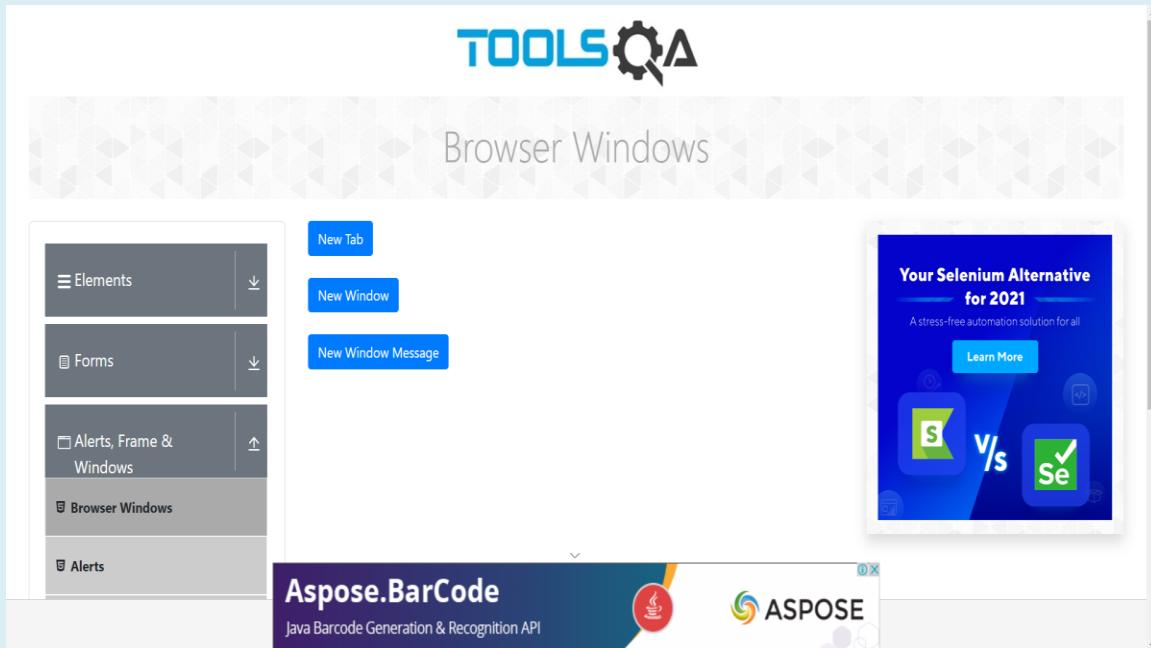
```
w.findElement(By.xpath("/html[1]/body[1]/div[2]/div[1]/div[1]/div[2]/div[1]  
/div[1]/div[1]/div[3]/div[1]/ul[1]/li[1]/span[1]")).click();  
  
Thread.sleep(2000);
```

## TC73: To Maximize the window

```
w.manage().window().maximize();
```

## TC74: To take Screenshot for browser window

```
TakesScreenshot t6= (TakesScreenshot) w;  
File src6=t6.getScreenshotAs(OutputType.FILE);  
Files.copy(src6, new File  
("D:\\Selenium_Screenshot\\windows.png"));  
Thread.sleep(2000);
```



## TC75: To Check the functionality of New Tab Button .

```
w.findElement(By.cssSelector("#tabButton")).click();
```

```
Set <String> windows= w.getWindowHandles();  
Iterator <String> it= windows.iterator();
```

```
String p_window=it.next();
String c_window=it.next();
w.switchTo().window(p_window);
    Thread.sleep(3000);
w.switchTo().window(c_window);
    Thread.sleep(3000);
System.out.println(w.findElement(By.id("sampleHeading"))
.getText());
```

**OUTPUT on Console:-** This is a sample page

**TC76: To Check the functionality of New Window Button .**

```
w.findElement(By.id("windowButton")).click();

Set <String> S=w.getWindowHandles();
Iterator <String> it =S.iterator();

String p_win=it.next();
String c_win=it.next();
w.switchTo().window(p_win);
Thread.sleep(3000);
w.switchTo().window(c_win);
Thread.sleep(3000);
System.out.println(w.findElement(By.xpath("//*[@id=\"sampleH
eading\"]")).getText());
```

**OUTPUT on Console:-** This is a sample page

**TC78: To Check the functionality of New Window Message Button .**

```
w.findElement(By.id("messageWindowButton")).click();

Set <String> windows= w.getWindowHandles();
Iterator <String> it= windows.iterator();

String p_window=it.next();
String c_window=it.next();

w.switchTo().window(p_window);
Thread.sleep(3000);
w.switchTo().window(c_window);
Thread.sleep(3000);
w.manage().window().maximize();
System.out.println(w.findElement(By.xpath("/html[1]/body[1]"))
").getText());
```

**OUTPUT on Console:-** Knowledge increases by sharing but not by saving. Please share this website with your friends and in your organization.

### **B. Alert**

TC79: To Check the functionality of Alerts option.

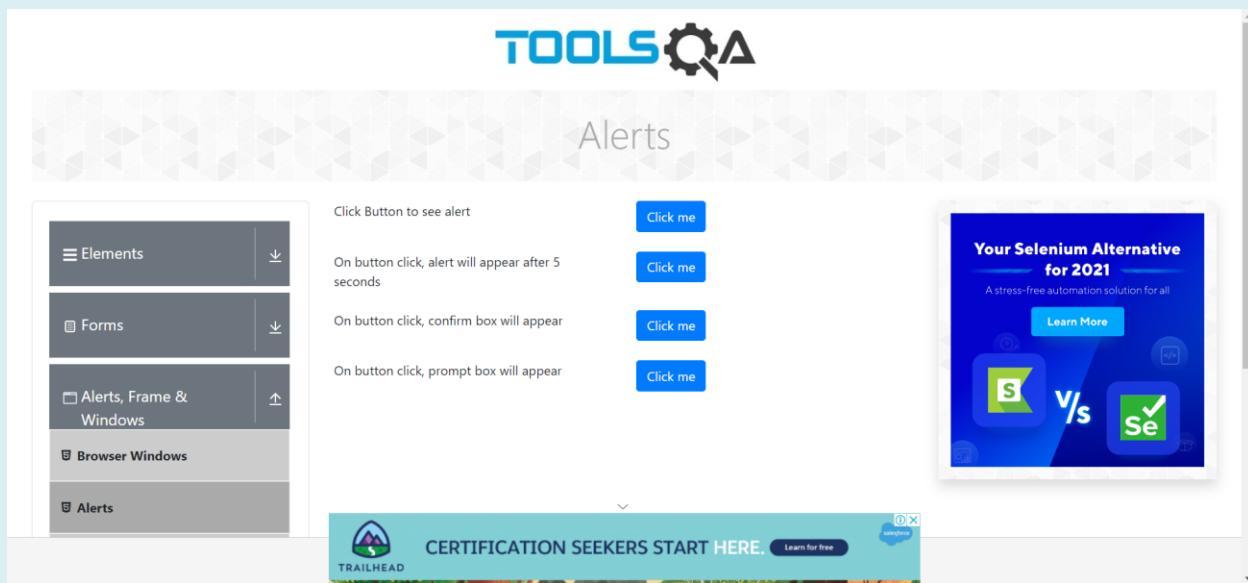
```
JavascriptExecutor js = (JavascriptExecutor) w;
js.executeScript("window.scrollBy(0,500)");

WebElement E=
w.findElement(By.xpath("//span[contains(text(),'Alerts')]"));
js.executeScript("arguments[0].scrollIntoView();", E);
E.click();

js.executeScript("window.scrollBy(0, -500)");
```

How to take screenshot of alerts window

```
TakesScreenshot T1= (TakesScreenshot) w;  
File Src1=T1.getScreenshotAs(OutputType.FILE);  
Files.copy(Src1, new File ("D:\\Selenium_Screenshot\\Alert.png"));
```



**TC81:** To Check the functionality of Click Button to see Alert.

```
w.findElement(By.xpath("//*[@id='alertButton']")).click();  
Thread.sleep(2000);  
  
Alert A1 = w.switchTo().alert();  
System.out.println(A1.getText());  
Thread.sleep(2000);  
A1.accept(); //to click OK button  
w.quit();
```

**OUTPUT on Console:-** You clicked a button

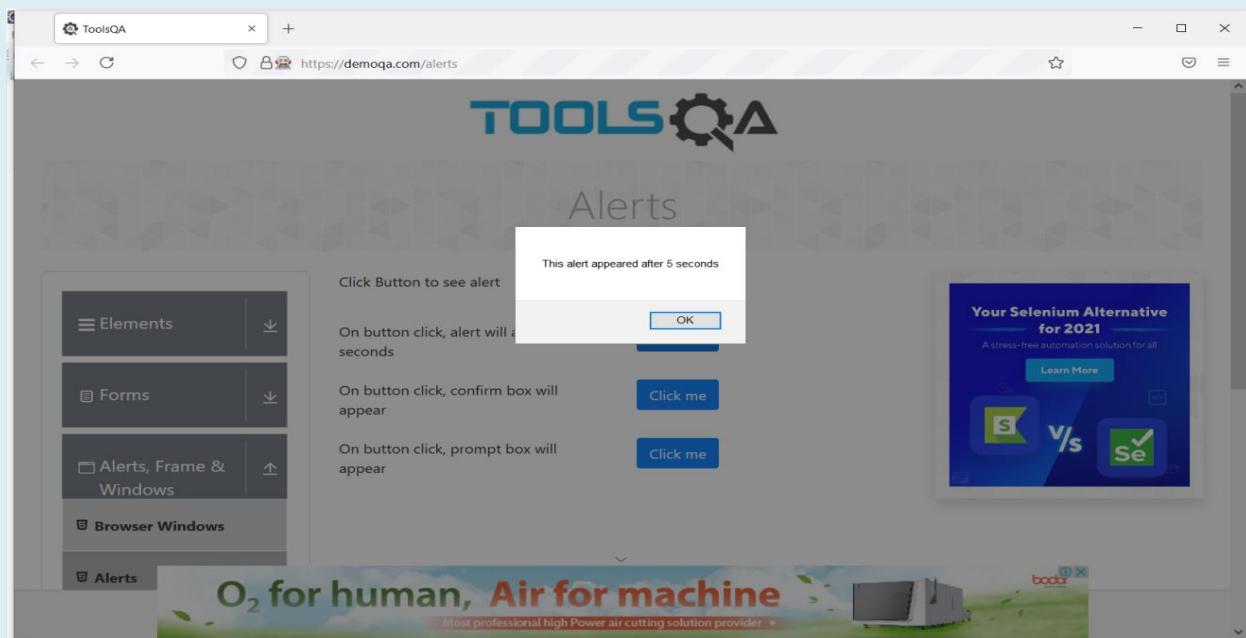
**TC82:** Take screenshot for single Alert button

```
TakesScreenshot T2= (TakesScreenshot) w;  
File Src2=T2.getScreenshotAs(OutputType.FILE);  
Files.copy(Src2, new File ("D:\\Selenium_Screenshot\\Single.png"));
```

TC83: To Check the functionality of On Button Click, Alert will appear after 5 second.

```
w.findElement(By.id("timerAlertButton")).click();  
Thread.sleep(2000);  
  
Alert A2=w.switchTo().alert();  
System.out.println("output is: " + A2.getText());  
Thread.sleep(2000);  
A2.accept();  
w.quit();
```

TC84: Take screenshot for second Alert button



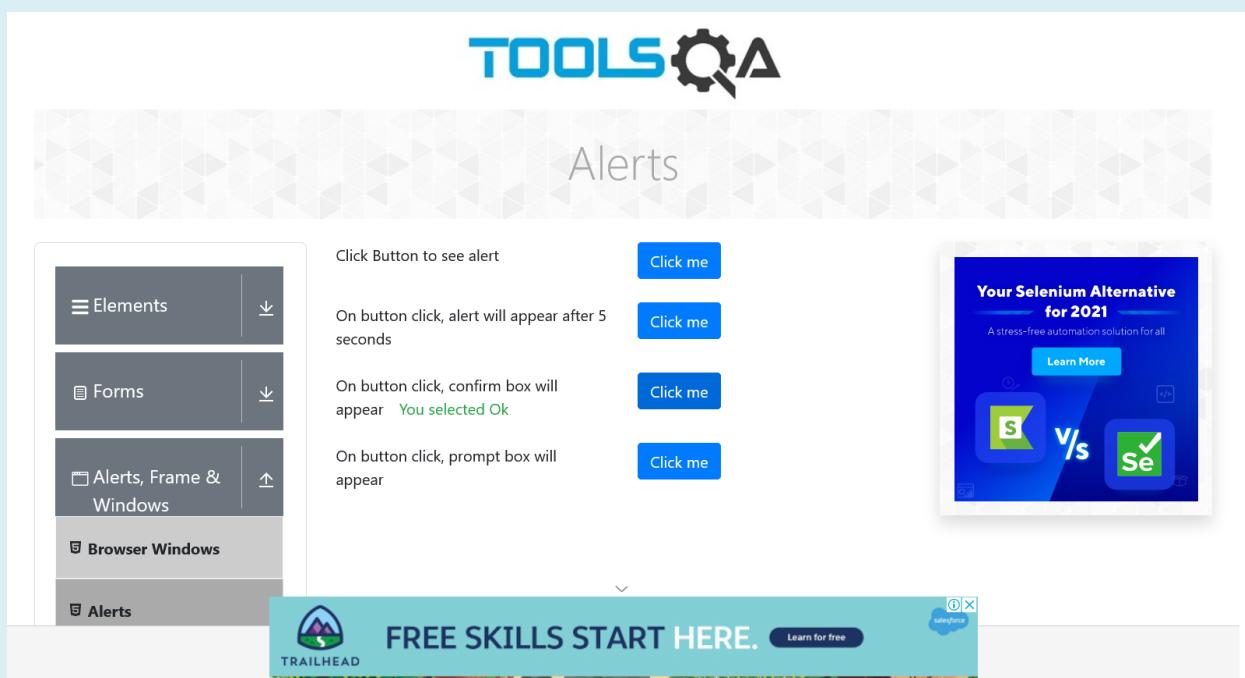
TC85: To Check the functionality of On Button Click, confirm box will appear

```
w.findElement(By.id("confirmButton")).click();
Alert A3=w.switchTo().alert();
Thread.sleep(2000);
System.out.println(A3.getText());
A3.accept();
```

**OUTPUT on Consle:-** Do you confirm action?

TC86: Take screenshot for Confimed Alert button

```
TakesScreenshot T3= (TakesScreenshot) w;
File Src3=T3.getScreenshotAs(OutputType.FILE);
Files.copy(Src3, new File ("D:\\Selenium_Screenshot\\\
confirm.png"));
```



TC87: To Check the functionality of On Button Click, prompt box will appear with valid data

```
w.findElement(By.id("promptButton")).click();
Thread.sleep(2000);

Alert a=w.switchTo().alert();
a.sendKeys("Sandhya");
Thread.sleep(2000);

System.out.println(a.getText());
Thread.sleep(2000);
a.accept();
```

**OUTPUT on console:-**Please enter your name

**TC88: Take screenshot for prompt Alert button**

```
TakesScreenshot T4= (TakesScreenshot) w;
File Src4=T4.getScreenshotAs(OutputType.FILE);
Files.copy(Src4,new File("D:\\Selenium_Screenshot\\\
prompt.png"));
```

TOOLSAQA

## Alerts

- Click Button to see alert
- Click me
- On button click, alert will appear after 5 seconds
- Click me
- On button click, confirm box will appear
- Click me
- On button click, prompt box will appear You entered Sandhya
- Click me

Your Selenium Alternative for 2021  
A stress-free automation solution for all  
[Learn More](#)

REMOTE CAREERS START HERE. [Learn for free](#)

TRAILHEAD

## TC89: To Check the functionality of On Button Click, prompt box will appear with blank data

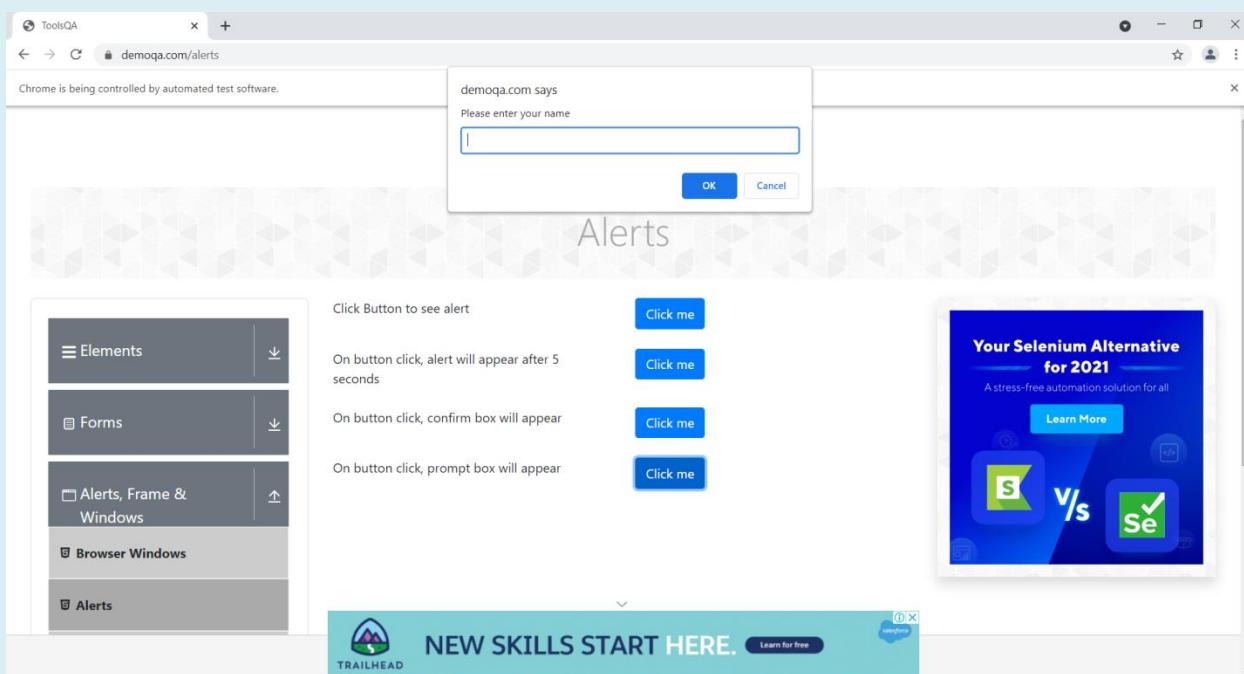
```
w.findElement(By.id("promptButton")).click();
Thread.sleep(2000);

Alert a=w.switchTo().alert();
System.out.println(a.getText());
a.sendKeys("");

Thread.sleep(2000);
a.accept();
```

## TC88: Take screenshot for prompt Alert button with blank data

```
TakesScreenshot T5= (TakesScreenshot) w;
File Src5=T5.getScreenshotAs(OutputType.FILE);
Files.copy(Src5,new File("D:\\Selenium_Screenshot\\\
prompt_blank.png"));
```



## C.Frames

TC89: To get Frames option scroll down

```
JavascriptExecutor js = (JavascriptExecutor) w;
js.executeScript("window.scrollBy(0,500)");
WebElement E=
w.findElement(By.xpath("/html[1]/body[1]/div[2]/div[1]/div[1]/div[2]/div[1]/div[3]/div[1]/div[3]/h5[1]"));
js.executeScript("arguments[0].scrollIntoView()", E);
E.click();
Thread.sleep(2000);
```

TC90: To click on Frames option

```
JavascriptExecutor js1 = (JavascriptExecutor) w;
js1.executeScript("window.scrollBy(0,500)");
WebElement E1=
w.findElement(By.xpath("/html[1]/body[1]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]/div[1]/div[3]/div[1]/ul[1]/li[3]/span[1]"));

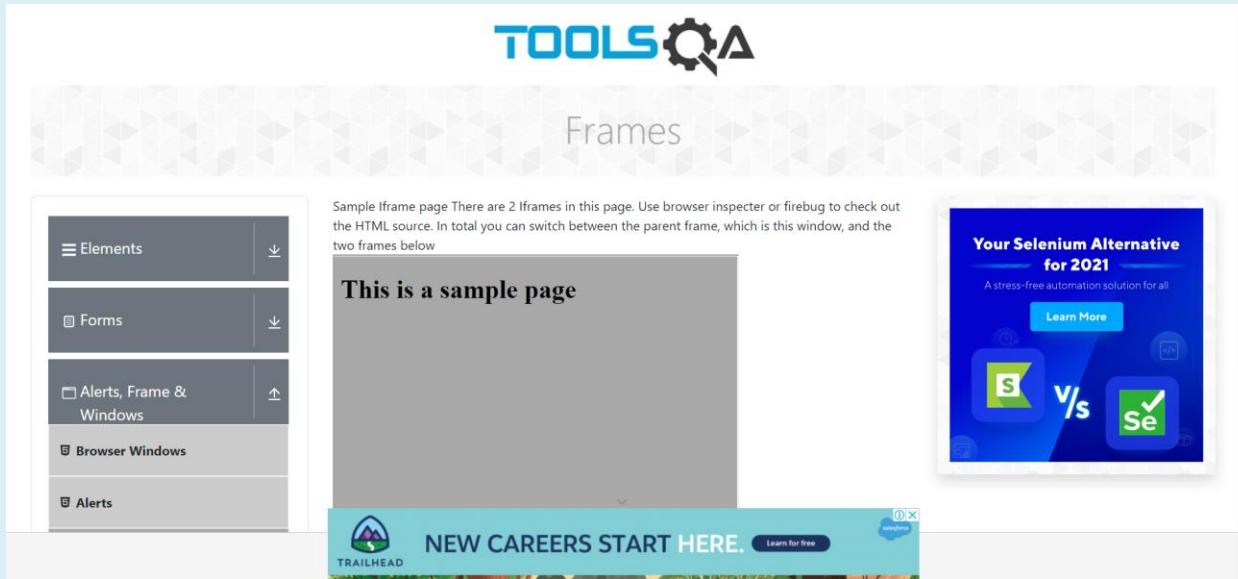
js.executeScript("arguments[0].scrollIntoView()", E1);
E1.click();
Thread.sleep(2000);
```

TC91: Page up

```
js.executeScript("window.scrollBy(0,-500)");
```

## TC92: To take screenshot for Frames option

```
TakesScreenshot t1= (TakesScreenshot) w;  
File src1=t1.getScreenshotAs(OutputType.FILE);  
Files.copy(src1, new File  
("D:\\Selenium_Screenshot\\Frames.png"));
```



## TC93: To Check Total no of iframes in that page

```
List<WebElement>  
list=w.findElements(By.tagName("iframe"));  
System.out.println("Total no of iframs are = " +  
list.size());
```

**OUTPUT on Console:-** Total no of iframs are = 6

## TC94: To Display the message of frame on console

```
w.switchTo().frame("frame2");  
System.out.println(w.findElement(By.id("sampleHeading"))  
.getText());  
w.switchTo().defaultContent();
```

```
w.findElement(By.xpath("//*[@id=\"sampleHeading\"]")).getText();
w.close();
```

**OUTPUT on console:-**This is a sample page

### **D.Nasted Frames**

TC95: To get Nasted Frames option scroll down

```
JavascriptExecutor js = (JavascriptExecutor) w;
js.executeScript("window.scrollBy(0,500)");

WebElement E=
w.findElement(By.xpath("/html[1]/body[1]/div[2]/div[1]/div[1]/div[2]/div[1]/div[3]/div[1]/div[3]/h5[1]"));

js.executeScript("arguments[0].scrollIntoView()", E);
E.click();
Thread.sleep(2000);
```

TC96: To click on Nasted Frames option

```
JavascriptExecutor js1 = (JavascriptExecutor) w;
js1.executeScript("window.scrollBy(0,500)");

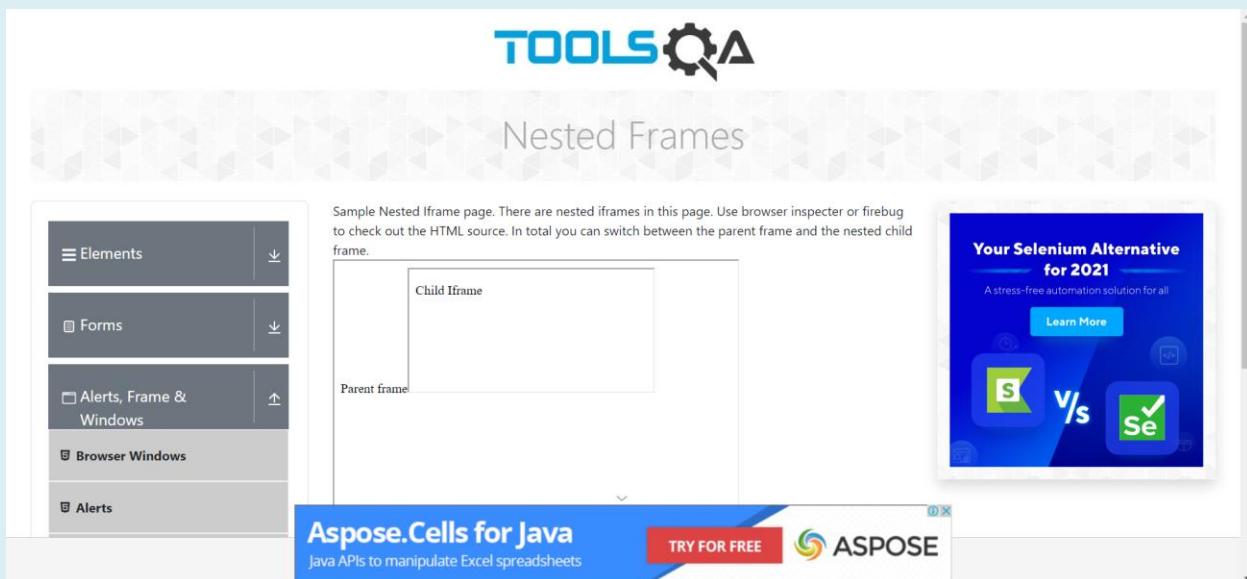
WebElement E1=
w.findElement(By.xpath("//span[contains(text(),'Nested
Frames')]"));
js.executeScript("arguments[0].scrollIntoView()", E1);
E1.click();
Thread.sleep(2000);
```

## TC97: Scroll up

```
js.executeScript("window.scrollBy(0,-500)");
```

## TC98:- To take screenshot for Nested Frames option

```
TakesScreenshot t1= (TakesScreenshot) w;  
File src1=t1.getScreenshotAs(OutputType.FILE);  
Files.copy(src1, new File  
("D:\\Selenium_Screenshot\\NestedFrames.png"));
```



## TC99: To Check Total no of iframes in that page

```
List<WebElement>  
list=w.findElements(By.tagName("iframe"));  
System.out.println("Total no of iframs are = " +  
list.size());
```

**OUTPUT on Console:-**Total no of iframs are = 4

## D.Modal Dialogs

TC100: To get Modal Dialogs option scroll down page

```
JavascriptExecutor js = (JavascriptExecutor) w;
js.executeScript("window.scrollBy(0,500)");

WebElement E=
w.findElement(By.xpath("/html[1]/body[1]/div[2]/div[1]/div[1]/
div[2]/div[1]/div[3]/div[1]/div[3]/h5[1]"));

js.executeScript("arguments[0].scrollIntoView()", E);
E.click();
Thread.sleep(2000);
```

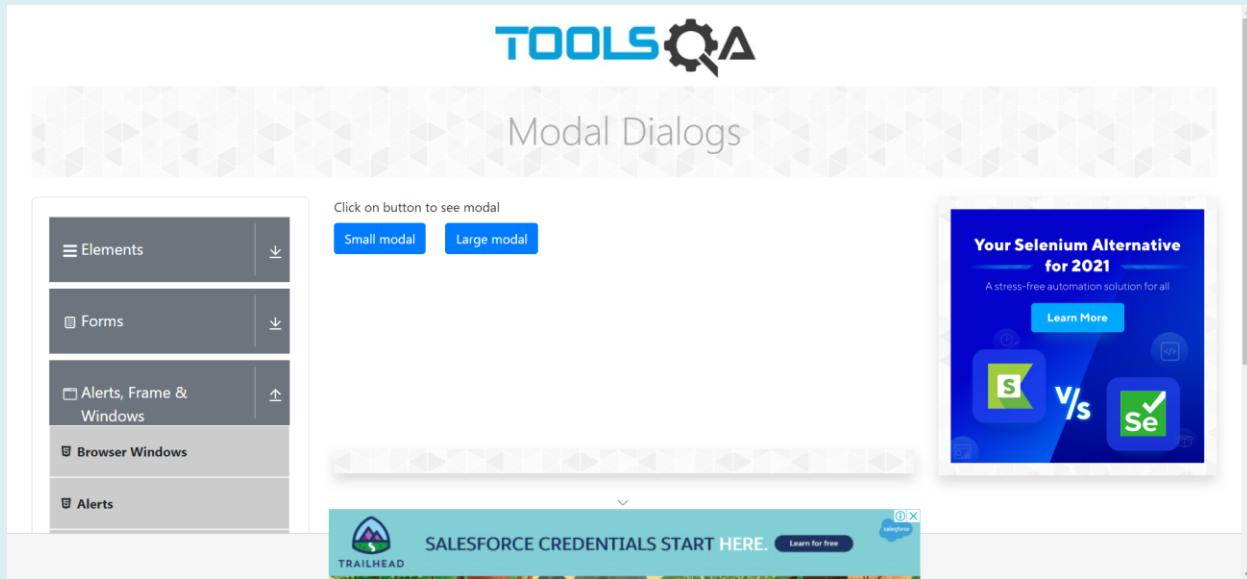
TC101: To Click on Modal Dialogs option

```
JavascriptExecutor js1 = (JavascriptExecutor) w;
js1.executeScript("window.scrollBy(0,500)");

WebElement E1=w.findElement(By.Xpath("//span
[contains(text(),'Modal Dialogs')]"));
js.executeScript("arguments[0].scrollIntoView()", E1);
E1.click();
Thread.sleep(2000);
js.executeScript("window.scrollBy(0,-500)");
Thread.sleep(2000);
```

TC102: To Take screenshot for Modal Dialogs Page

```
TakesScreenshot t1= (TakesScreenshot) w;  
File src1=t1.getScreenshotAs(OutputType.FILE);  
Files.copy(src1, new File  
("D:\\Selenium_Screenshot\\Modal_Dialogs.png"));
```

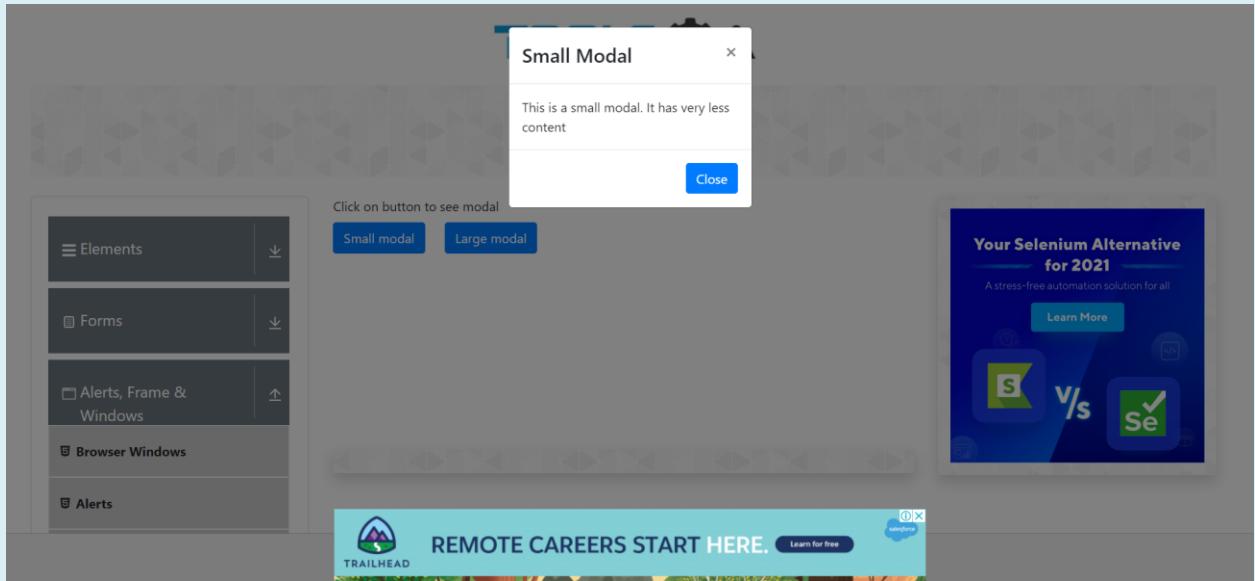


## TC103: To Click on Small Model button

```
w.findElement(By.id("showSmallModal")).click();  
Thread.sleep(2000);
```

## TC104: To Take screenshot for Small Modal

```
TakesScreenshot t2= (TakesScreenshot) w;  
File src2=t2.getScreenshotAs(OutputType.FILE);  
Files.copy(src2, new File  
("D:\\Selenium_Screenshot\\SmallModal.png"));
```



## TC105: To Print small modal message on console

```
WebElement  
S=w.findElement(By.xpath("/html/body/div[4]/div/div/div[2]"));  
System.out.println(S.getText());
```

**Output on Console:-**This is a small modal. It has very less content

## TC106:To click on Close Option i.e 'x'sign

```
w.findElement(By.xpath("/html/body/div[4]/div/div/div[1]/button/span[1]")).click();
```

## TC107:To click on Close Button

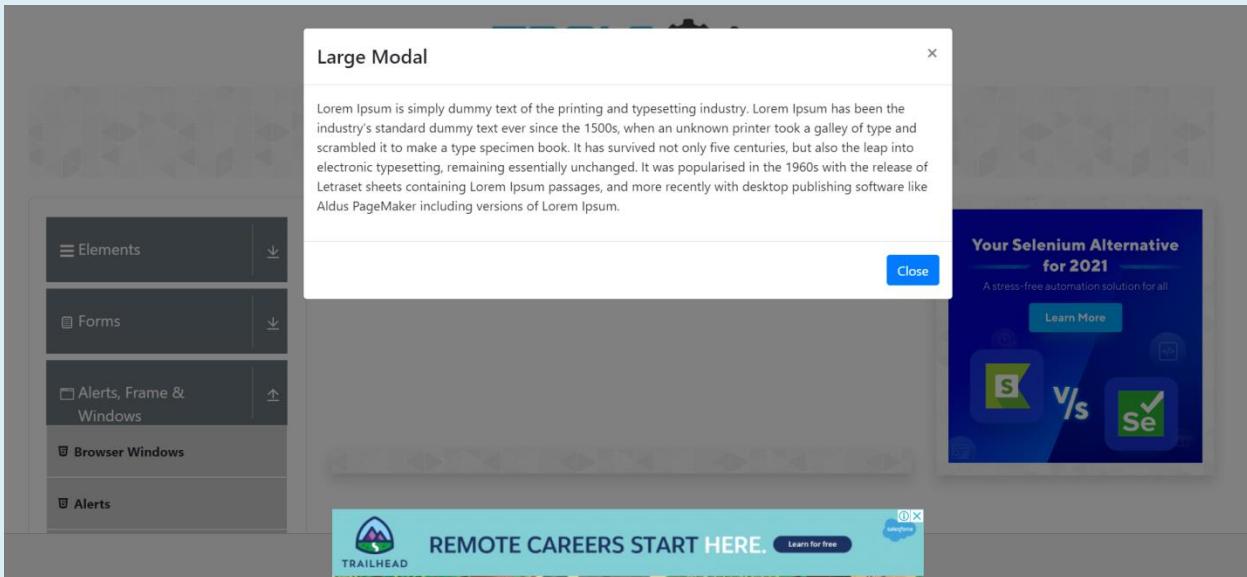
```
w.findElement(By.id("closeSmallModal")).click();
```

## TC108: To Click on Large Model button

```
w.findElement(By.id("showLargeModal")).click();
Thread.sleep(2000);
```

## TC109: To Take screenshot for Large Modal

```
TakesScreenshot t3= (TakesScreenshot) w;
File src3=t3.getScreenshotAs(OutputType.FILE);
Files.copy(src3, new File
("D:\\Selenium_Screenshot\\LargeModal.png"));
```



## TC110:To Print large modal message on console

```
WebElement
L=w.findElement(By.xpath("/html/body/div[4]/div/div/div[2]/p"));
System.out.println(L.getText());
```

**Output on Console:-** Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

TC111:To click on Close Option i.e 'x'sign

```
w.findElement(By.xpath("//span[contains(text(), 'x')]")).click();
```

TC112:To click on Close Button

```
w.findElement(By.id("closeLargeModal")).click();
```

## Project Script

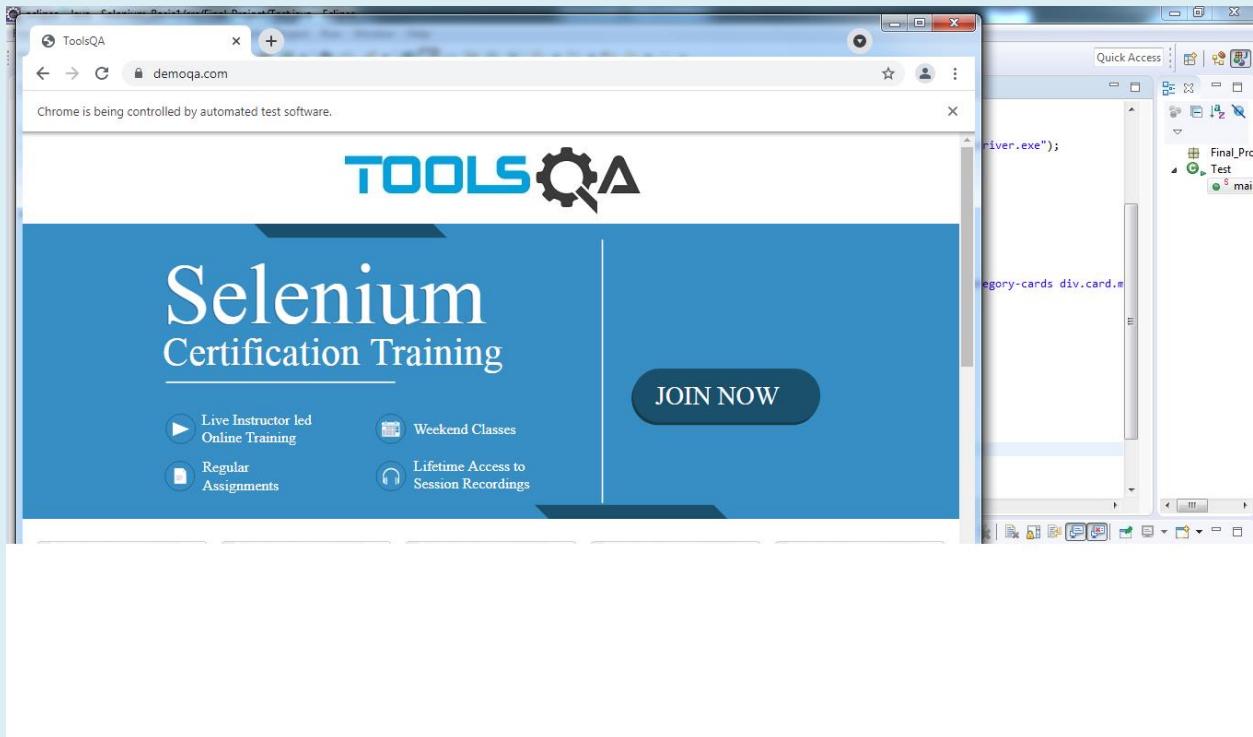
### Module No.4 – Widgets

TC113 - Launch the chrome browser

```
System.setProperty("webdriver.chrome.driver",
"C:\\\\Users\\\\Admin\\\\Desktop\\\\Selenium Jar\\\\chromedriver.exe");
WebDriver w=new ChromeDriver();
```

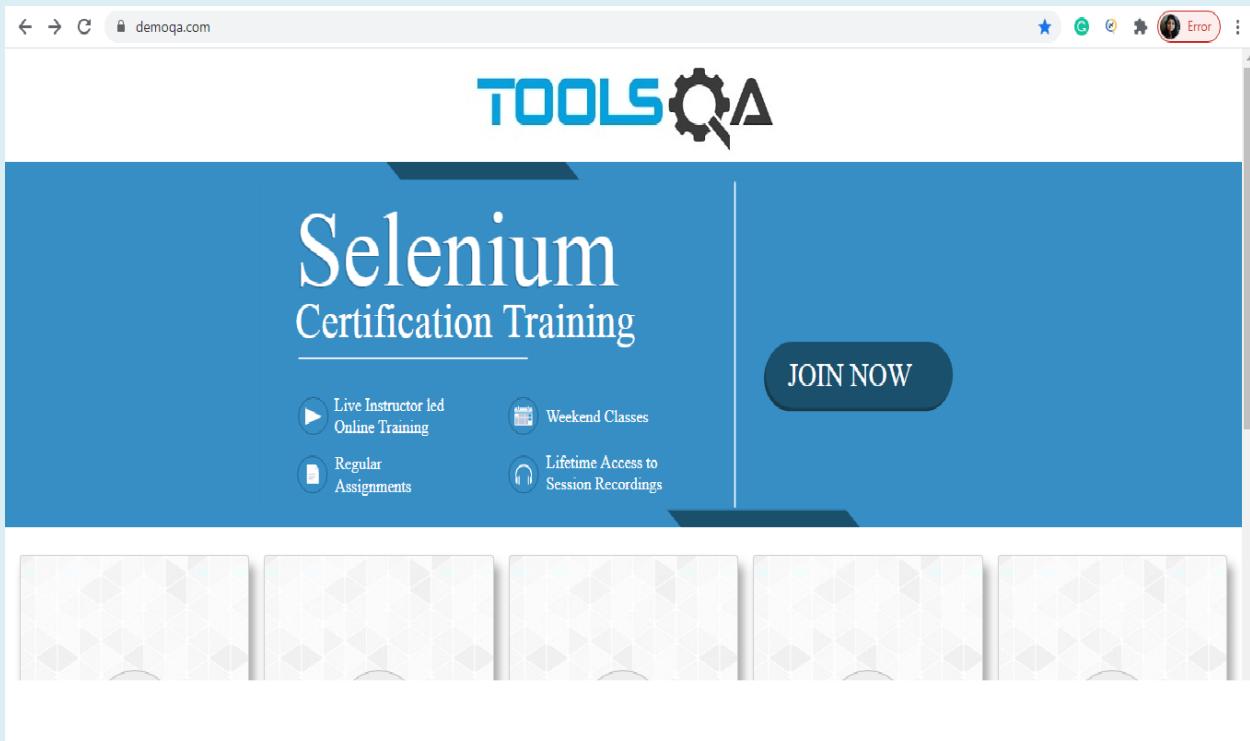
TC114 - Open toolsQA URL

```
w.get ("https://www.demoqa.com/");
```



## TC115 - To Maximize the window

```
w.manage().window().maximize(); //for maximize the window
```



## TC116 - To Get current page URL

```
System.out.println(w.getCurrentUrl());
```

**OUTPUT-** https://demoqa.com/

## TC117 - To scroll down

```
JavascriptExecutor js = (JavascriptExecutor) w;  
js.executeScript ("window.scrollTo (0, 1000)");
```

## TC118 - To click on Widgets field

```
w.findElement(By.cssSelector("div.body-height:nth-child(2) div.home-content  
div.home-body div.category-cards > div.card.mt-4.top-card:nth-  
child(4)").click();
```

## TC119 - To get page title

```
System.out.println (w.getTitle ());
```

**OUTPUT-** ToolsQA

## TC120 - To click on Accordian

```
w.findElement(By.xpath("/html[1]/body[1]/div[2]/div[1]/div[1]/div[2]/div[1]/di  
v[1]/div[1]/div[4]/div[1]/ul[1]/li[1]/span[1]").click();
```

## TC121 – In Accordian to click on First element

```
w.findElement(By.xpath("//*[@id=\"section1Heading\"]")).click();
```

## TC122 – In Accordian to click on Second element

```
w.findElement(By.xpath("//*[@id=\"section2Heading\"]")).click();
```

## TC123 – In Accordian to click on Third element

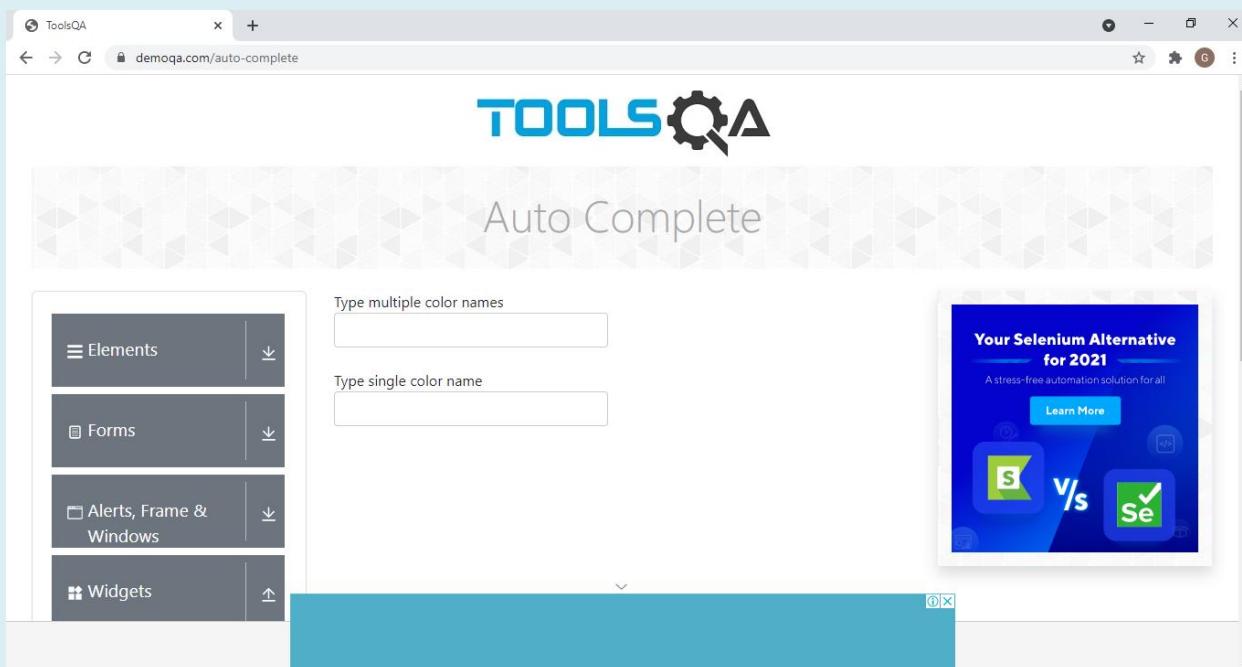
```
w.findElement(By.xpath("//*[@id='section3Heading']").click());
```

## TC124 - To scroll down

```
JavascriptExecutor js1 = (JavascriptExecutor) w;  
js1.executeScript("window.scrollTo(0,200);");
```

## TC125 - To click on Auto Complete

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[1]/di  
v[1]/div[1]/div[4]/div[1]/ul[1]/li[2]").click());
```



**OUTPUT/DEFECT** – Enable to enter the text in that text fields.

## TC126 - To scroll down

```
JavascriptExecutor js2 = (JavascriptExecutor) w;  
js2.executeScript("window.scrollTo(0,000);")
```

## TC127 - To click on Date Picker

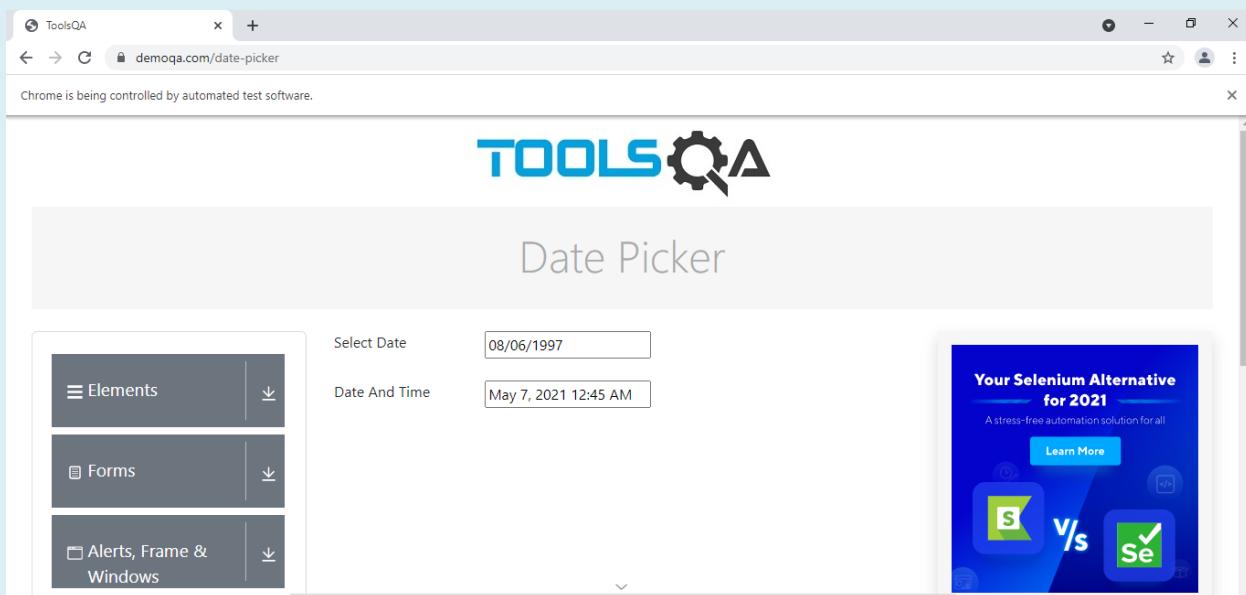
```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[1]/div[1]/div[1]/div[4]/div[1]/ul[1]/li[3]")).click();
```

## TC128 – In Date Picker to Select Date,Month,Year,Time

```
w.findElement(By.xpath("//*[@id=\"datePickerMonthYearInput\"]")).clear();  
  
Select s1=new Select(w.findElement(By.className("react-datepicker__month-select")));  
  
s1.selectByIndex(7);  
  
Select s2=new Select(w.findElement(By.className("react-datepicker__year-select")));  
  
s2.selectByValue("1997");  
  
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[2]/div[2]/div[1]/div[1]/div[2]/div[2]/div[2]/div[4]")).click();
```

## TC129 – Date and Time

```
w.findElement(By.xpath("//*[@id=\"dateAndTimePickerInput\"]")).clear();  
  
w.findElement(By.className("react-datepicker__month-read-view--down-arrow")).click();  
  
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[2]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[2]/div[1]/div[1]/div[5]")).click();  
  
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[2]/div[2]/div[2]/div[1]/div[1]/div[2]/div[2]/div[2]/div[6]")).click();  
  
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[2]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[2]/div[1]/div[1]/span[1]")).click();  
  
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[2]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[2]/div[1]/div[1]/div[6]")).click();
```



**OUTPUT/DEFECT** – Enable to select Year in Date and Time Section.

TC130 - To scroll down

```
JavascriptExecutor js2 = (JavascriptExecutor) w;  
js2.executeScript("window.scrollTo(0,000);")
```

TC131 - To click on Slider

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[1]/di  
v[1]/div[1]/div[4]/div[1]/ul[1]/li[4]")).click();
```

TC132 – Move to Backword and Forward Slider

```
WebElement slider =  
w.findElement(By.xpath("//*[@id=\"sliderContainer\"]/div[1]/span/input"));  
  
Actions act = new Actions(w);  
  
act.dragAndDropBy(slider, -150, 100).perform();  
  
Thread.sleep(2000);  
  
act.dragAndDropBy(slider, 200, 100).perform();
```

TC133 - To scroll down

```
JavascriptExecutor js2 = (JavascriptExecutor) w;  
js2.executeScript("window.scrollTo(0,000);")
```

## TC134 - To click on Progress Bar

```
w.findElement(By.xpath("//span[contains(text(),'Progress Bar')]")).click();
```

## TC135 - To click on Start button

```
w.findElement(By.id("startStopButton")).click();
```

## TC136 - To click on Stop button

```
w.findElement(By.xpath("//button[@id='startStopButton']")).click();
```

## TC137 - To scroll down

```
JavascriptExecutor js2 = (JavascriptExecutor) w;  
js2.executeScript("window.scrollTo(0,000);");
```

## TC138- To click on Tabs

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[1]/div[1]/div[1]/div[4]/div[1]/ul[1]/li[6]")).click();
```

## TC139- To click on What

```
w.findElement(By.id("demo-tab-what")).click();
```

## TC140 - To click on Origin

```
w.findElement(By.id("demo-tab-origin")).click();
```

## TC141 - To click on Use

```
w.findElement(By.id("demo-tab-use")).click();
```

## TC142 - To scroll down

```
JavascriptExecutor js2 = (JavascriptExecutor) w;  
js2.executeScript("window.scrollTo(0,000);");
```

## TC143 - To click on Tool Tips

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[1]/div[1]/div[1]/div[4]/div[1]/ul[1]/li[7]").click();  
  
w.findElement(By.id("toolTipButton")).click();  
  
w.findElement(By.id("toolTipTextField")).sendKeys("Hello");
```

## TC144 - To scroll down

```
JavascriptExecutor js2 = (JavascriptExecutor) w;  
js2.executeScript("window.scrollTo(0,000);");
```

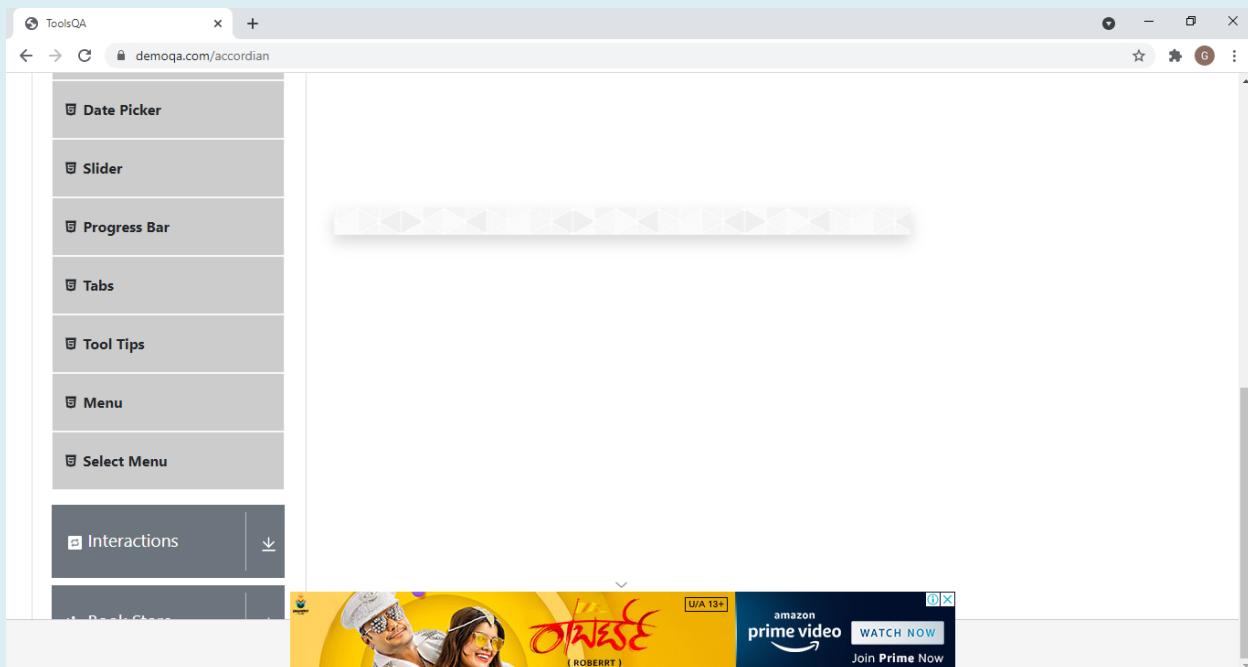
## TC145 - To click on Menu

```
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[1]/div[1]/div[1]/div[4]/div[1]/ul[1]/li[8]").click());
```

## TC146 – In Menu Actions on Main Items

```
Actions A=new Actions(w);
A.moveToElement(w.findElement(By.xpath("//*[@id=\"nav\"]/li[1]/a"))).perform();
A.moveToElement(w.findElement(By.xpath("//*[@id=\"nav\"]/li[2]/a"))).perform();
w.findElement(By.xpath("//*[@id=\"nav\"]/li[2]/ul/li[3]/a")).click();
A.moveToElement(w.findElement(By.xpath("//*[@id=\"nav\"]/li[3]/a"))).perform();
```

## TC147 - To click on Select Menu



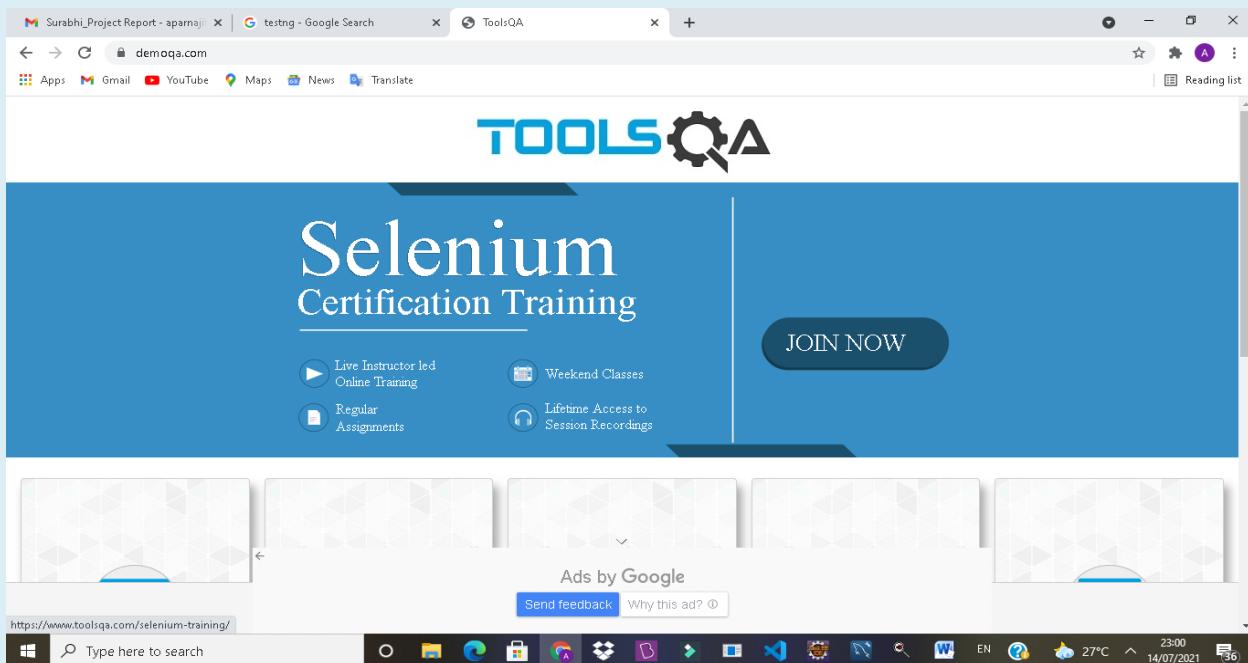
**OUTPUT/DEFECT**- Select Menu button is not working.

## Module No.5 - Interactions

TC148- Launch the Firefox browser and open ToolsQA Website

TestNG Test Script

```
WebDriver W;  
@BeforeTest  
public void Set() {  
    System.setProperty("webdriver.gecko.driver",  
    "C:\\\\Users\\\\hp\\\\Desktop\\\\Seleniumjar\\\\geckodriver.exe");  
    W = new FirefoxDriver();  
  
    W.get("https://demoqa.com/");
```



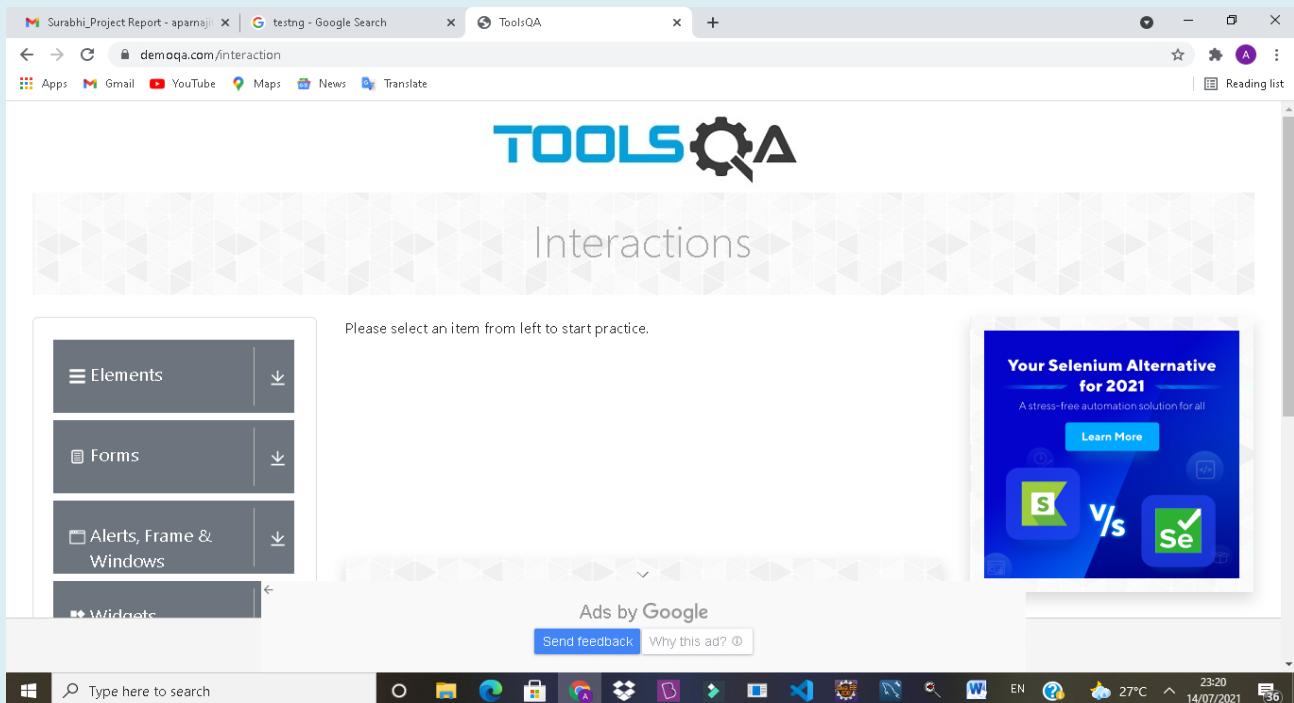
# TC149- To verify the functionality of Interactions module of ToolsQA Website

## TestNG Test Script

```
@Test
    public void mainpage1() throws InterruptedException
{
    WebElement apps1=
W.findElement(By.xpath("//h5[contains(text(),'Interact
ions')]"));
JavascriptExecutor js1 = (JavascriptExecutor) W;

js1.executeScript("arguments[0].scrollIntoView();",
apps1);

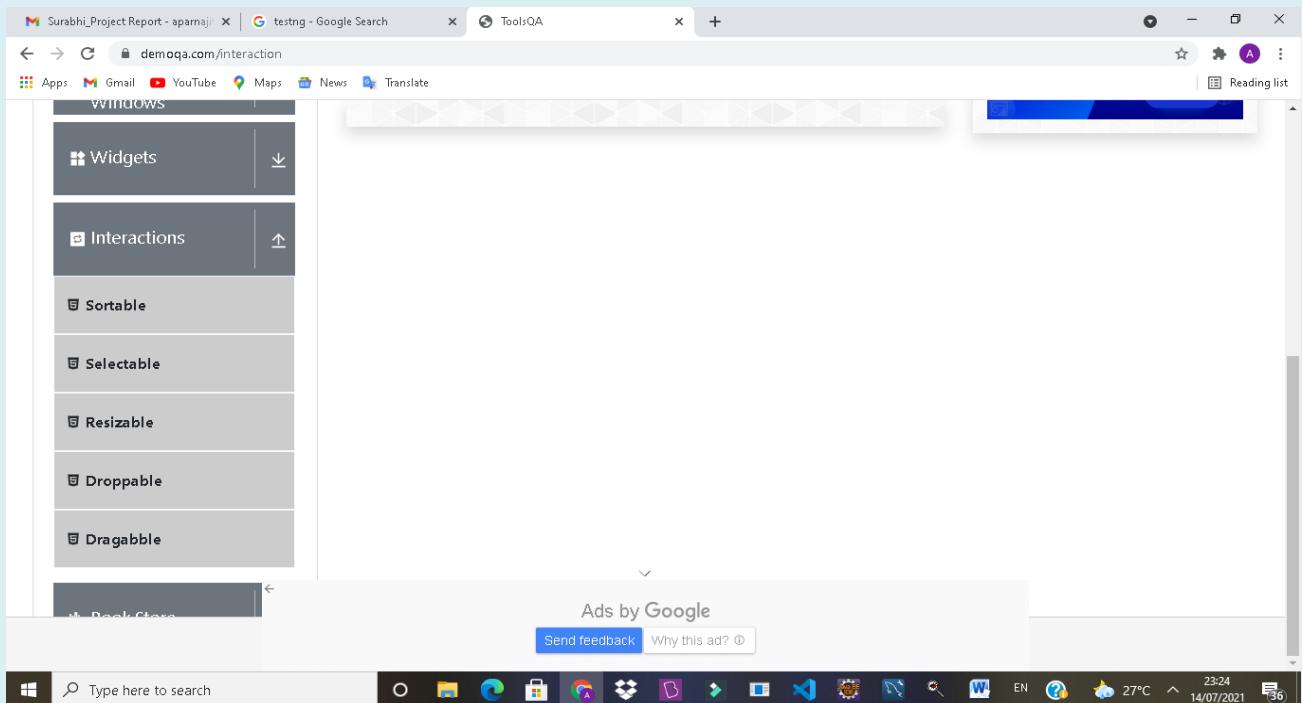
apps1.click();
```



# TC150- To verify the functionality of scrolling in ToolsQA Website

## TestNG Test Script

```
JavascriptExecutor js = (JavascriptExecutor) w;  
  
js.executeScript("window.scrollTo(0,document.body.scrollHeight)");  
  
System.out.println("page end");
```



### OUTPUT/DEFECT:

In interactions module arrow functionality is Not working by scripting way and also without clicking an arrow, it automatically opens the subsection of interaction

TC151- To verify and display the Current URL as well as Title of Current page

TestNG Test Script

```
System.out.println("Current URL is "
+W.getCurrentUrl());
System.out.println(("Title of page is "
+W.getTitle()));
```

**OUTPUT-** CurrentURL is https://demoqa.com/interaction  
Title of page is ToolsQA

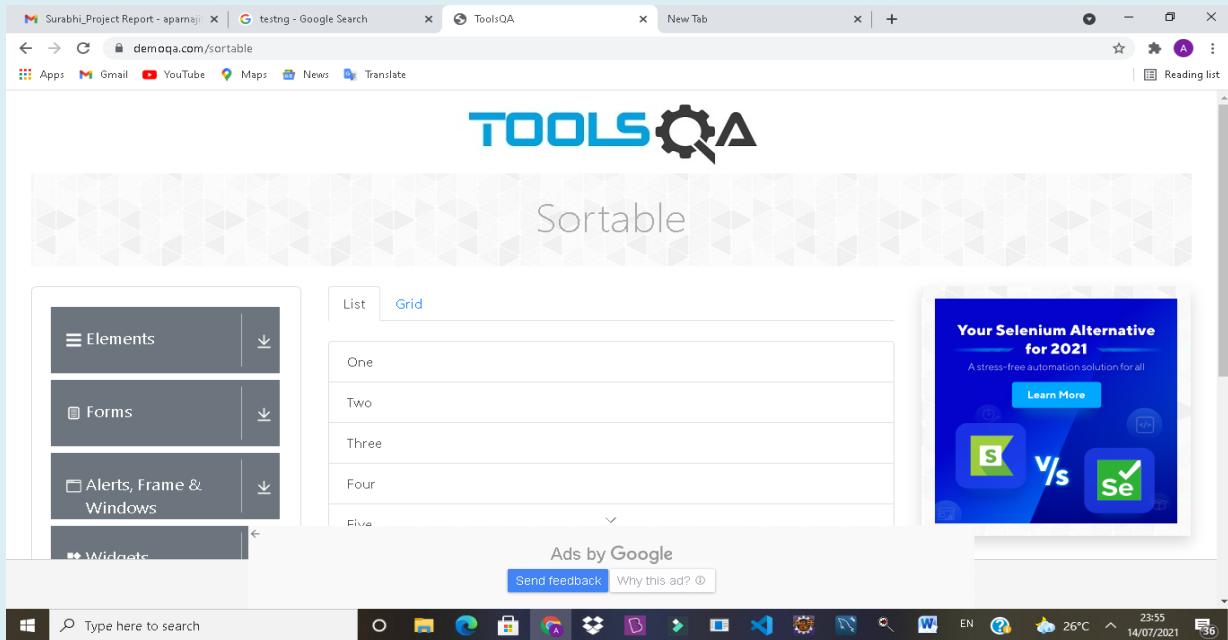
TC152- To verify the functionality of submodule (sortable section) of Interactions module

TestNG Test Script

```
@Test(dependsOnMethods = {"mainpage1"})
public void sortablepage() throws
InterruptedException {
    W.navigate().to("https://demoqa.com/sortable");
```

**OUTPUT/DEFECT:**

Interaction submodule unable to click by scripting method



TC153- To verify the functionality of List in sortable section

## TestNG Test Script

```
//*****ListSection*****
//*****Drag*****
    WebElement draglist1 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]"));
    WebElement draglist2 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[2]"));
    WebElement draglist3 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[3]"));
```

```
WebElement draglist4 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[4]"));
 WebElement draglist5 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[5]"));
 WebElement draglist6 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[1]/div[6]"));
//*****Drop*****
WebElement droplist6 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[1]/div[6]"));
 WebElement droplist5 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[5]"));
 WebElement droplist4 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[1]/div[4]"));
 WebElement droplist3 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[1]/div[3]"));
 WebElement droplist2 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[1]/div[2]"));
 WebElement droplist1 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/div[1]/div[1]/div[1]"));
```

```

Actions act = new Actions(W);
act.dragAndDrop(draglist1,dropList6).build().perform()
;
Thread.sleep(3000);
act.dragAndDrop(draglist6,dropList1).build().perform()
;
Thread.sleep(3000);
act.dragAndDrop(draglist2,dropList5).build().perform()
;
Thread.sleep(3000);
act.dragAndDrop(draglist5,dropList2).build().perform()
;
Thread.sleep(3000);
act.dragAndDrop(draglist3,dropList4).build().perform()
;
Thread.sleep(3000);
act.dragAndDrop(draglist4,dropList3).build().perform()
;
Thread.sleep(3000);

```

TC154- To verify the functionality of Grid in sortable section

## TestNG Test Script

```

//*****Grid section*****
W.findElement(By.linkText("Grid")).click();

WebElement draggrid1 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
WebElement dropgrid9 =
W.findElement(By.xpath("//div[contains(text(),'Nine')]"));

```

```
    WebElement draggrid2 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
    WebElement dropgrid8 =
w.findElement(By.xpath("//div[contains(text(),'Eight')]"));
    WebElement draggrid3 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
    WebElement dropgrid7 =
w.findElement(By.xpath("//div[contains(text(),'Seven')]"));
    WebElement draggrid4 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
    WebElement dropgrid6 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
    WebElement draggrid5 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
    WebElement dropgrid4 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
    WebElement draggrid6 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
    WebElement dropgrid5 =
w.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]"));
```

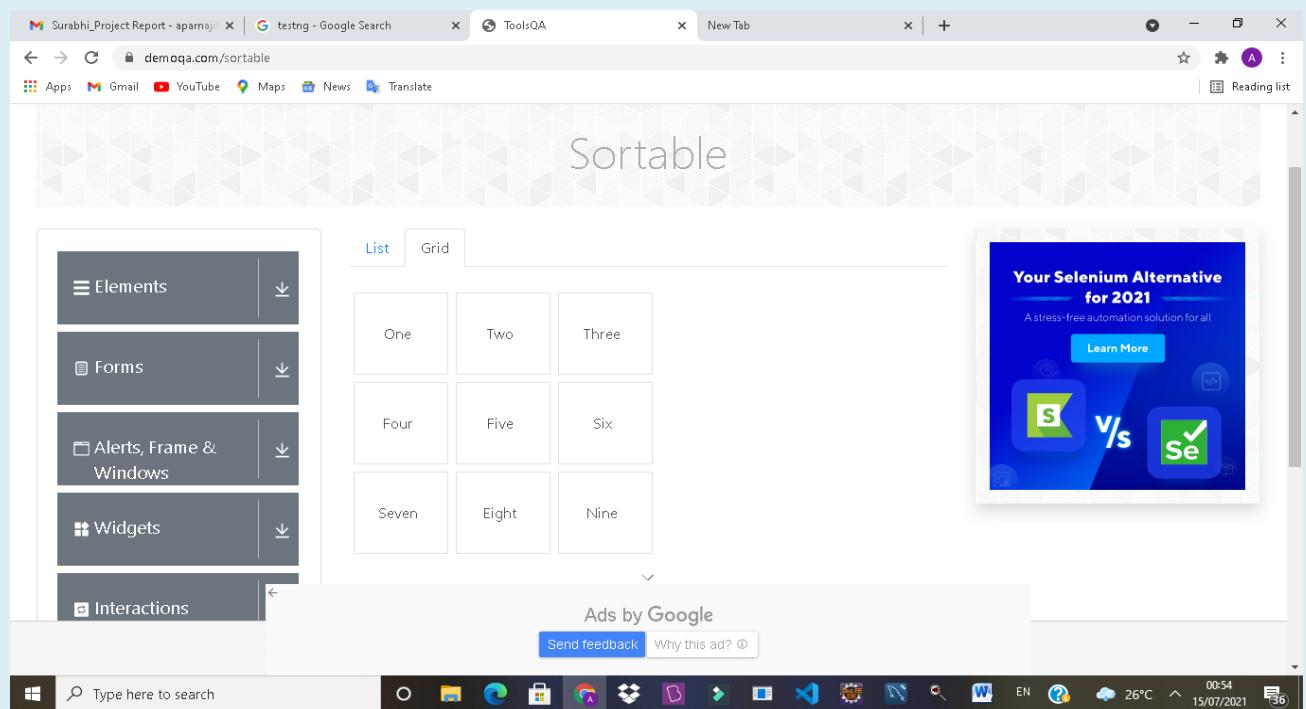
```
    WebElement draggrid7 =
W.findElement(By.xpath("//div[contains(text(),'Seven')]"))
);
    WebElement dropgrid3 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]/di
v[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]/d
iv[5]"));
    WebElement draggrid8 =
W.findElement(By.xpath("//div[contains(text(),'Eight')]")
);
    WebElement dropgrid2 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]/di
v[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]/d
iv[7]"));
    WebElement draggrid9 =
W.findElement(By.xpath("//div[contains(text(),'Nine')]")
);
    WebElement dropgrid1 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]/di
v[1]/div[2]/div[2]/div[1]/div[1]/div[2]/div[1]/div[1]/d
iv[9]"));

Actions act1 = new Actions(W);
act1.dragAndDrop(draggrid1,dropgrid9).build().perform()
;
Thread.sleep(5000);
act1.dragAndDrop(draggrid2,dropgrid8).build().perform()
;
Thread.sleep(5000);
act1.dragAndDrop(draggrid3,dropgrid7).build().perform()
;
Thread.sleep(5000);
act1.dragAndDrop(draggrid4,dropgrid6).build().perform()
;
Thread.sleep(5000);
act1.dragAndDrop(draggrid5,dropgrid4).build().perform()
;
Thread.sleep(5000);
```

```

act1.dragAndDrop(draggrid6,dropgrid5).build().perform()
;
Thread.sleep(5000);
act1.dragAndDrop(draggrid7,dropgrid3).build().perform()
;
Thread.sleep(5000);
act1.dragAndDrop(draggrid8,dropgrid2).build().perform()
;
Thread.sleep(5000);
act1.dragAndDrop(draggrid9,dropgrid1).build().perform()
;
Thread.sleep(5000);
}

```



TC155- To verify the functionality of navigating submodule (selectable section) page from sortable section

TestNG Test Script

```

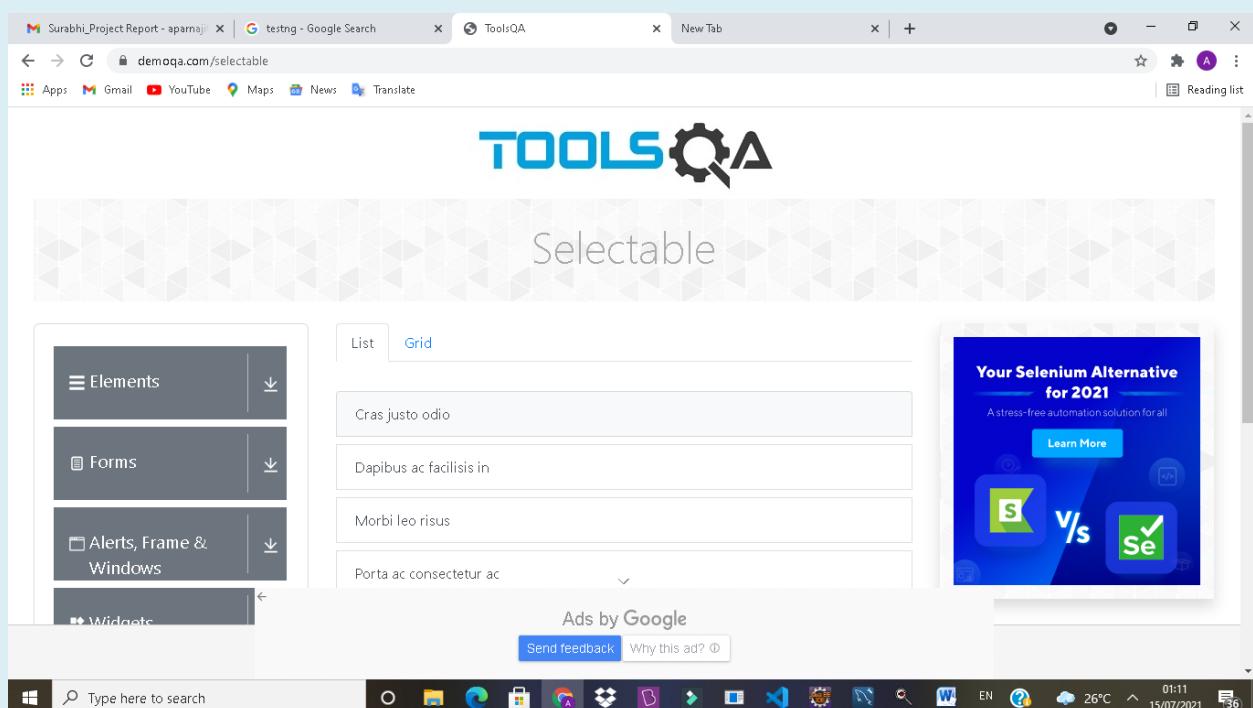
@Test(dependsOnMethods = {"sortablepage"})
public void selectablepage() throws
InterruptedException {

W.navigate().to("https://demoqa.com/selectable");

```

## OUTPUT/DEFECT:

Interaction submodule unable to click by scripting method



TC156- To verify the functionality of List in selectable section whether it selecting all elements or not

## TestNG Test Script

```

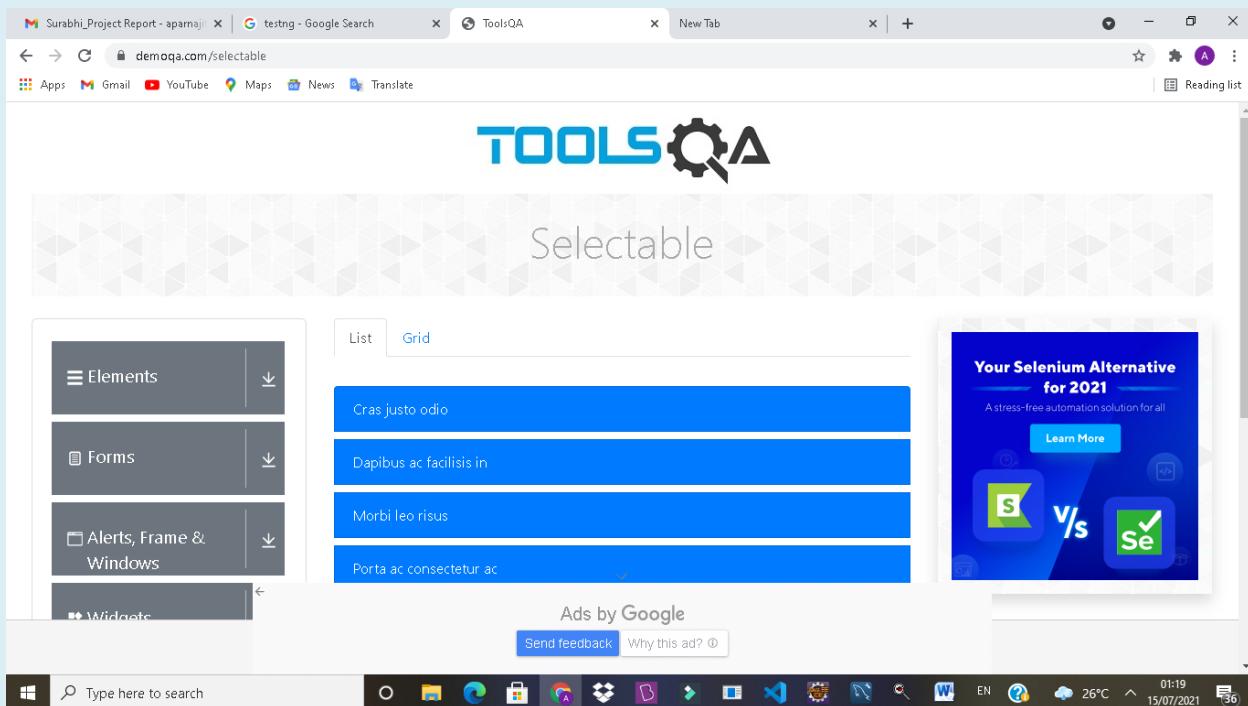
//*****Listsection*****
//*****select*****
W.findElement(By.xpath("//li[contains(text(),'Cras
justo odio')]")).click();
W.findElement(By.xpath("//li[contains(text(),'Dapibus
ac facilisis in')]")).click();

```

```

W.findElement(By.xpath("//li[contains(text(),'Morbi leo risus')]")).click();
W.findElement(By.xpath("//li[contains(text(),'Porta ac consectetur ac')]")).click();
Thread.sleep(5000);

```



TC157- To verify the functionality of List in selectable section whether it removes all elements from selection or not

## TestNG Test Script

```

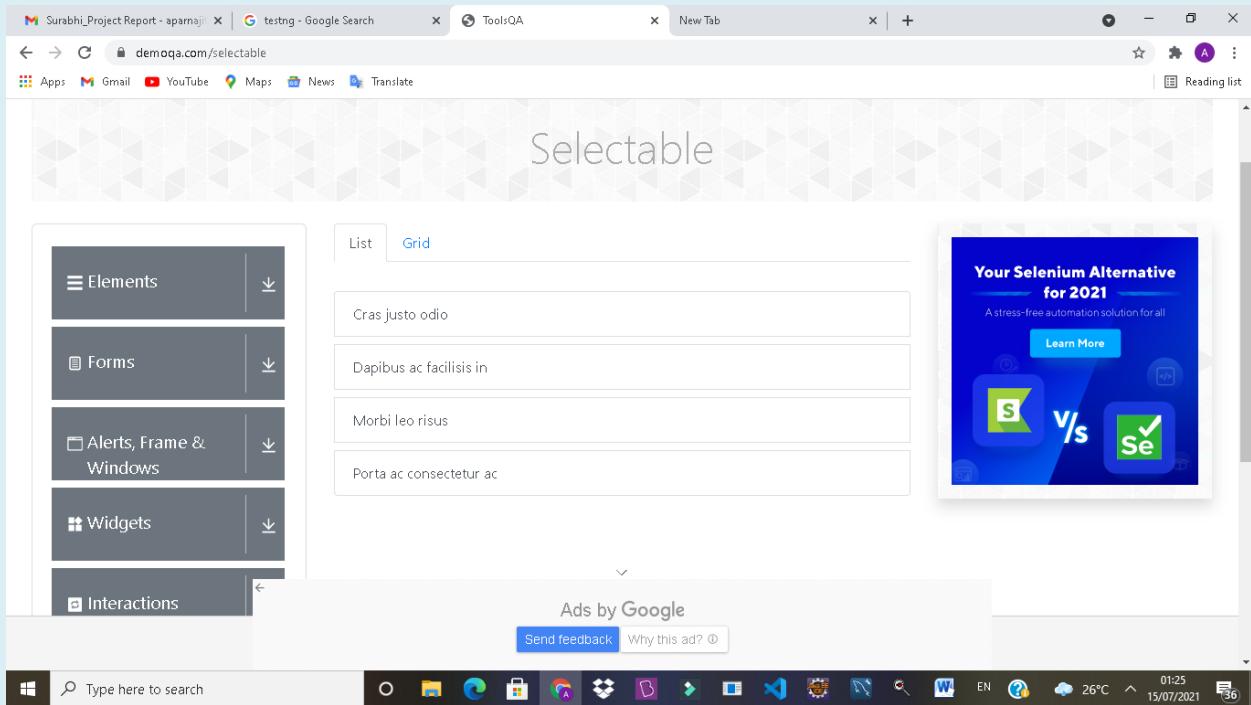
//*****Listsection*****
//*****Remove_selection*****
W.findElement(By.xpath("//li[contains(text(),'Dapibus ac facilisis in')]")).click();
W.findElement(By.xpath("//li[contains(text(),'Morbi leo risus')]")).click();

```

```

W.findElement(By.xpath("//li[contains(text(),'Cras
justo odio')])).click();
W.findElement(By.xpath("//li[contains(text(),'Porta ac
consectetur ac')])).click();

```



TC158- To verify the functionality of Grid in selectable section whether it selecting all elements or not

## TestNG Test Script

```

//*****Gridsection*****
//*****select*****
W.findElement(By.linkText("Grid")).click();
W.findElement(By.xpath("//li[contains(text(),'One')]"))
.click();
W.findElement(By.xpath("//li[contains(text(),'Two')]"))
.click();
W.findElement(By.xpath("//li[contains(text(),'Three')]"))
.click();

```

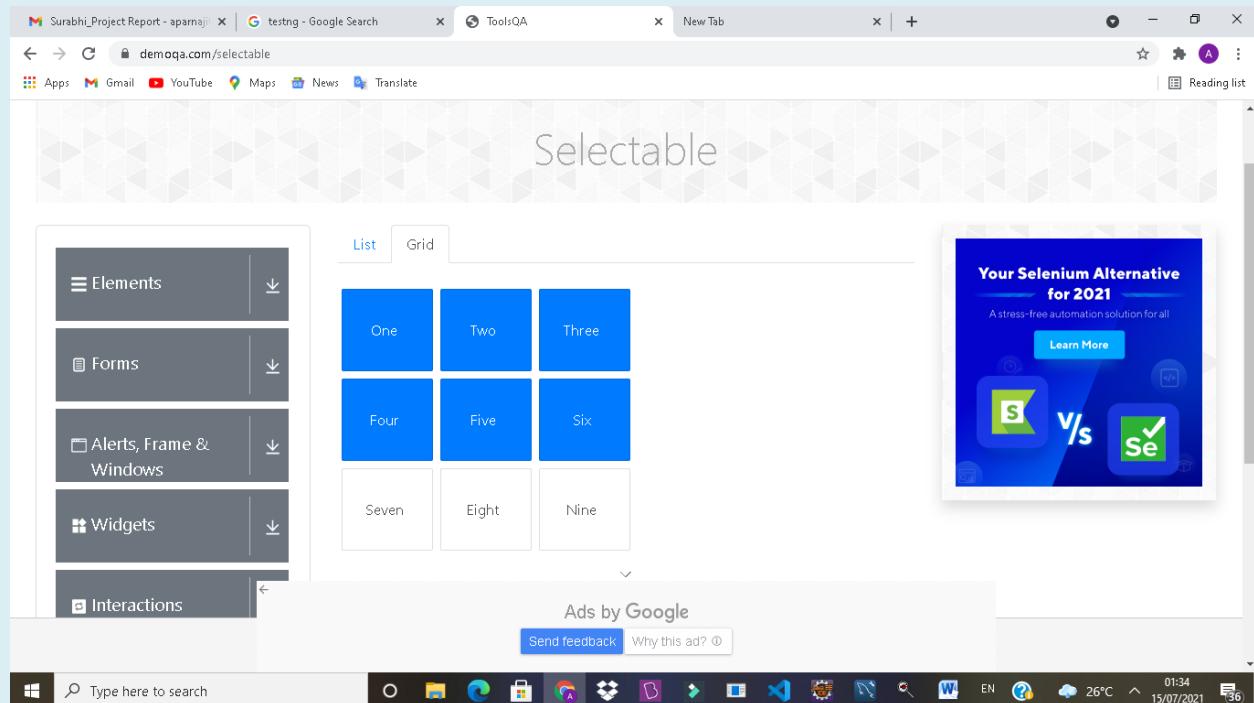
```

W.findElement(By.xpath("//li[contains(text(),'Four')]"))
).click();
W.findElement(By.xpath("//li[contains(text(),'Five')]"))
).click();
W.findElement(By.xpath("//li[contains(text(),'Six')]"))
).click();

```

## OUTPUT/DEFECT:

Unable to select 7,8,9 by scripting method in grid selection



TC159- To verify the functionality of Grid in selectable section whether it removes all elements from selection or not

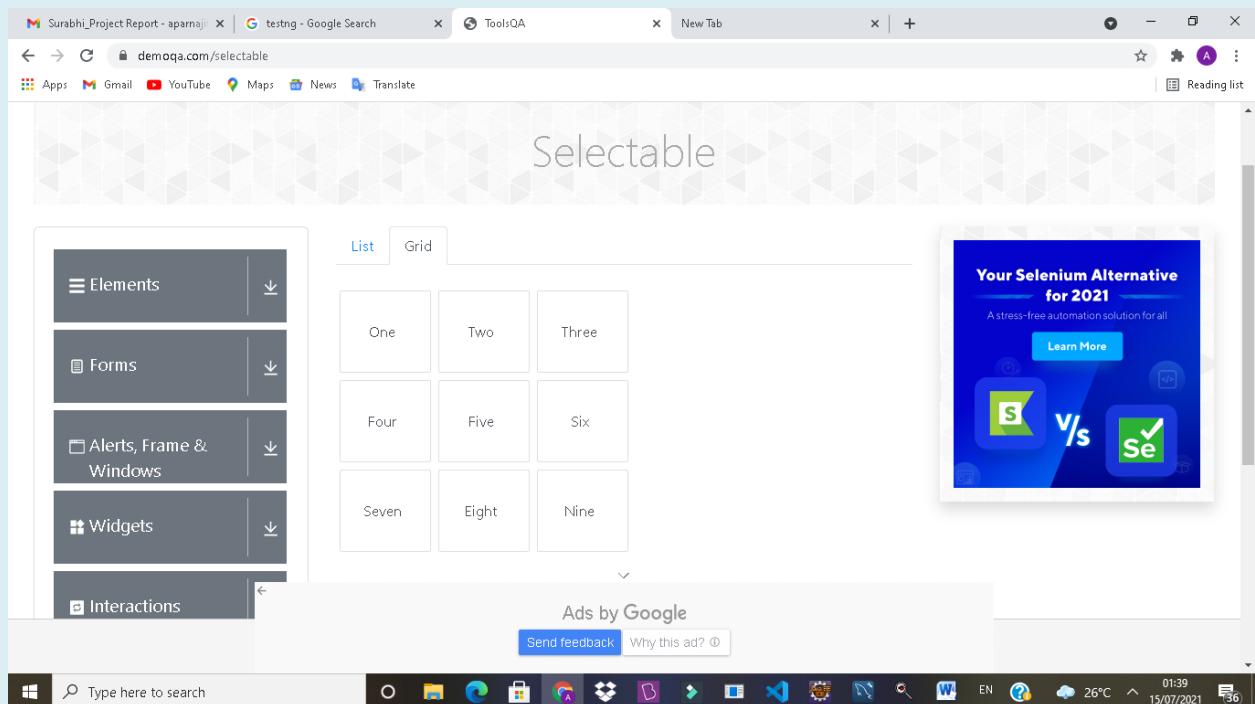
## TestNG Test Script

```

//*****Gridsection*****
//*****Remove_selection*****
Thread.sleep(5000);
W.findElement(By.xpath("//li[contains(text(),'One')]"))
.click();
W.findElement(By.xpath("//li[contains(text(),'Two')]"))
.click();

```

```
W.findElement(By.xpath("//li[contains(text(),'Three')]"))
.click();
W.findElement(By.xpath("//li[contains(text(),'Four')]"))
.click();
W.findElement(By.xpath("//li[contains(text(),'Five')]"))
.click();
```

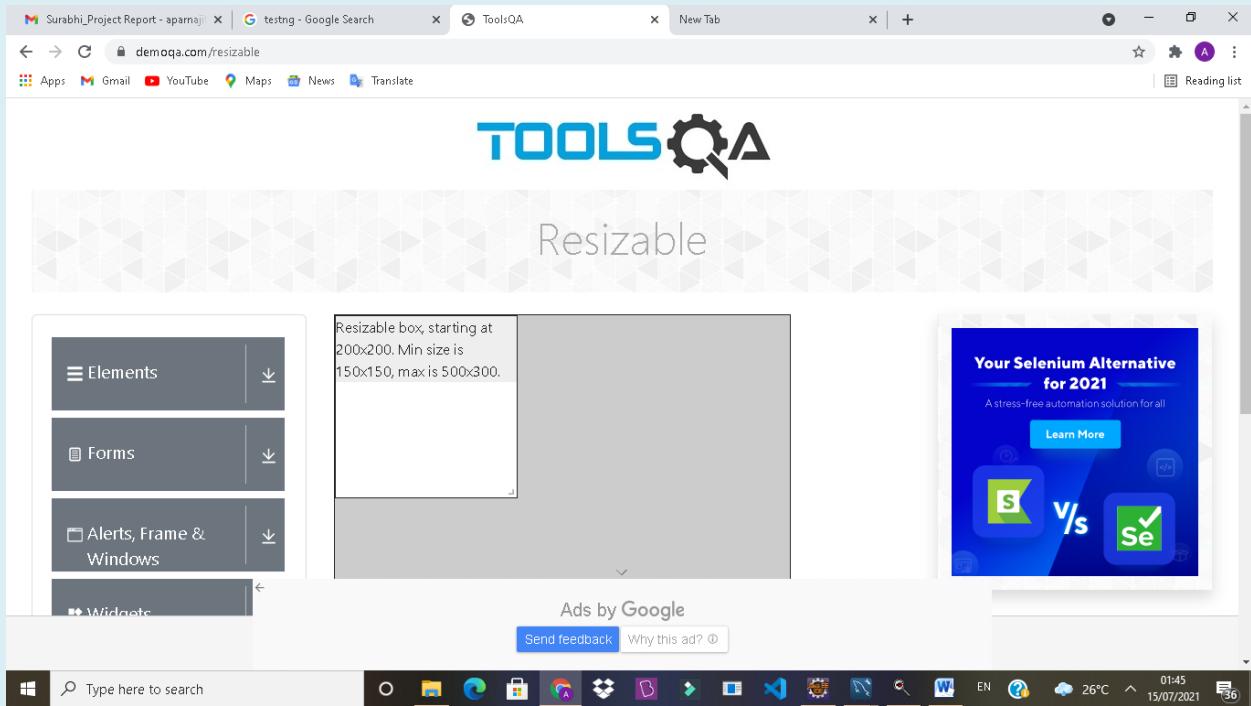


TC160- To verify the functionality of navigating submodule (Resizable section) page from selection page

## TestNG Test Script

```
@Test(dependsOnMethods = {"selectablepage"})
public void resizablepage() throws
InterruptedException {

    W.navigate().to("https://demoqa.com/resizable");
```



TC161- To verify the functionality of resizable to check whether it resizing according to script or not

## TestNG Test Script

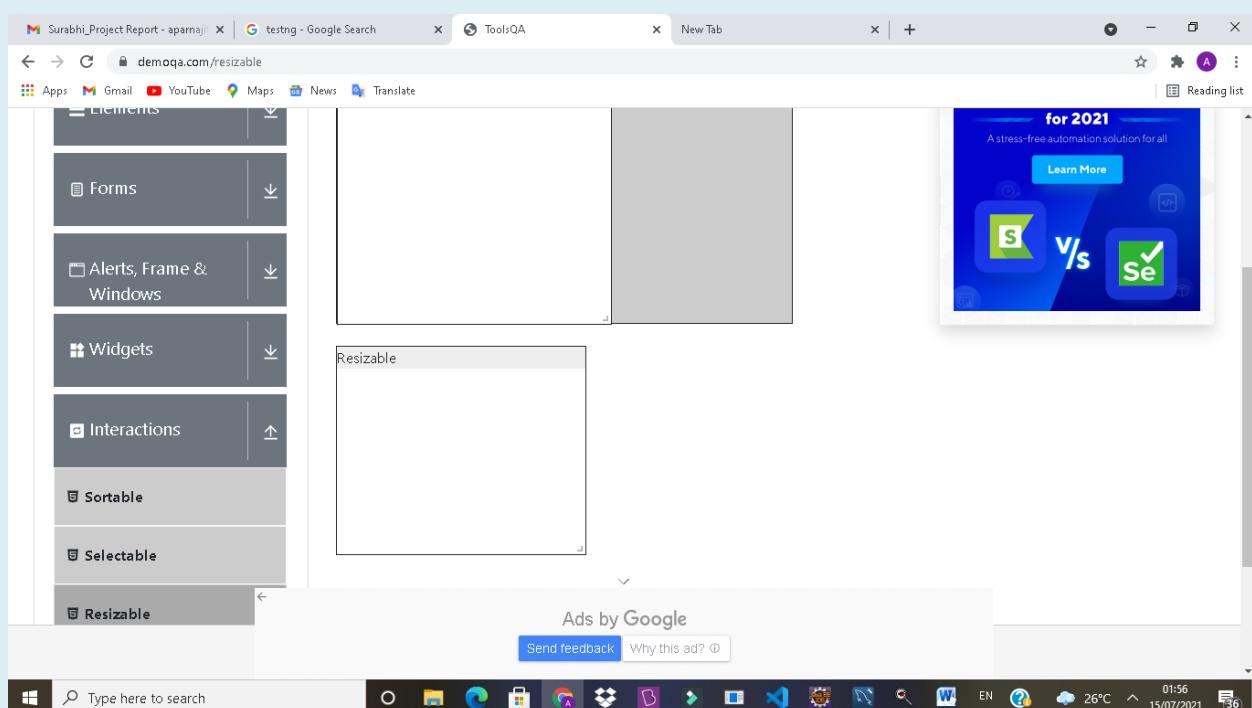
```
WebElement resizeElement =
W.findElement(By.xpath("//body/div[@id='app']/div[1]
/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/span[1]"))
;
WebElement resizeElement1 =
W.findElement(By.xpath("//body/div[@id='app']/div[1]
/div[1]/div[2]/div[2]/div[1]/div[2]/div[1]/span[1]"))
;
Actions actionsResize = new Actions(W);
//provide the x and y offset values
actionsResize.dragAndDropBy(resizeElement,
150,170).perform();
Thread.sleep(3000);

JavascriptExecutor js23 = (JavascriptExecutor) W;
```

```

js23.executeScript("window.scrollTo(0,document.body.scrollHeight)");
System.out.println("page end");
actionsResize.dragAndDropBy(resizeElement1, 100,
50).perform();
Thread.sleep(3000);
System.out.println("Done Resizing");
}

```



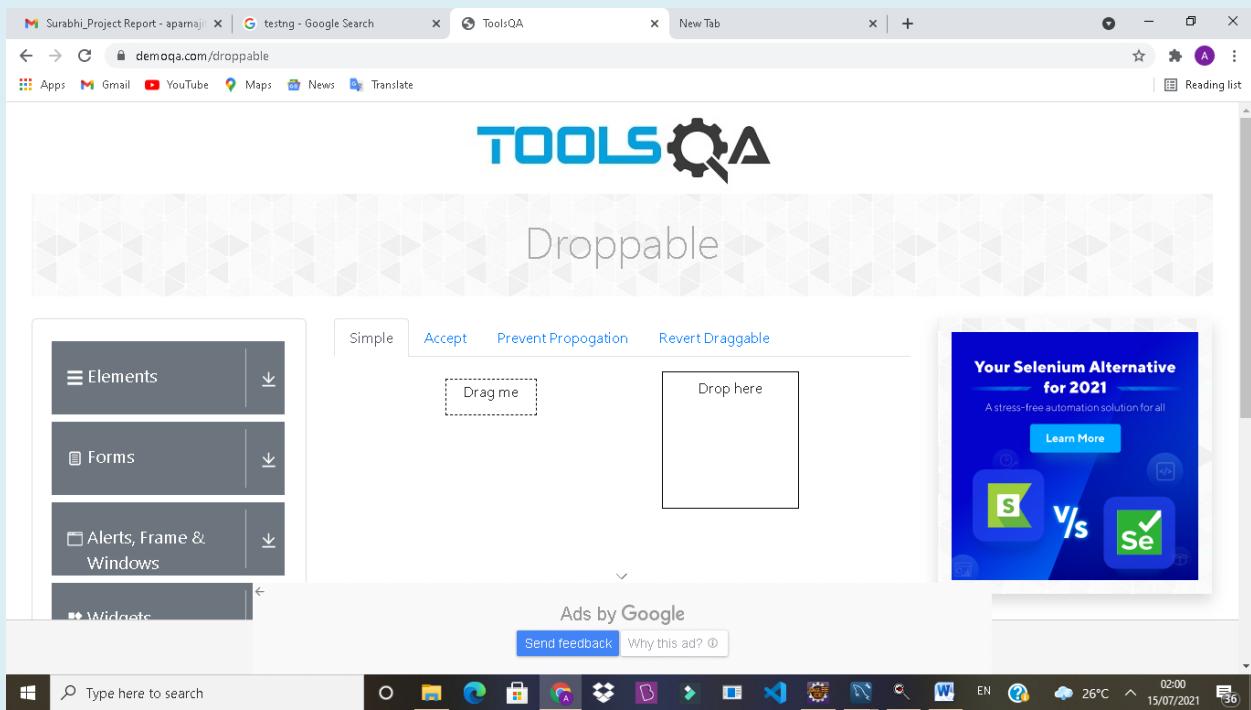
TC162- To verify the functionality of navigating submodule (Droppable section) page from Resizing page

## TestNG Test Script

```

@Test(dependsOnMethods = {"resizablepage"})
public void dropablepage() throws
InterruptedException {
W.navigate().to("https://demoqa.com/droppable");
}

```



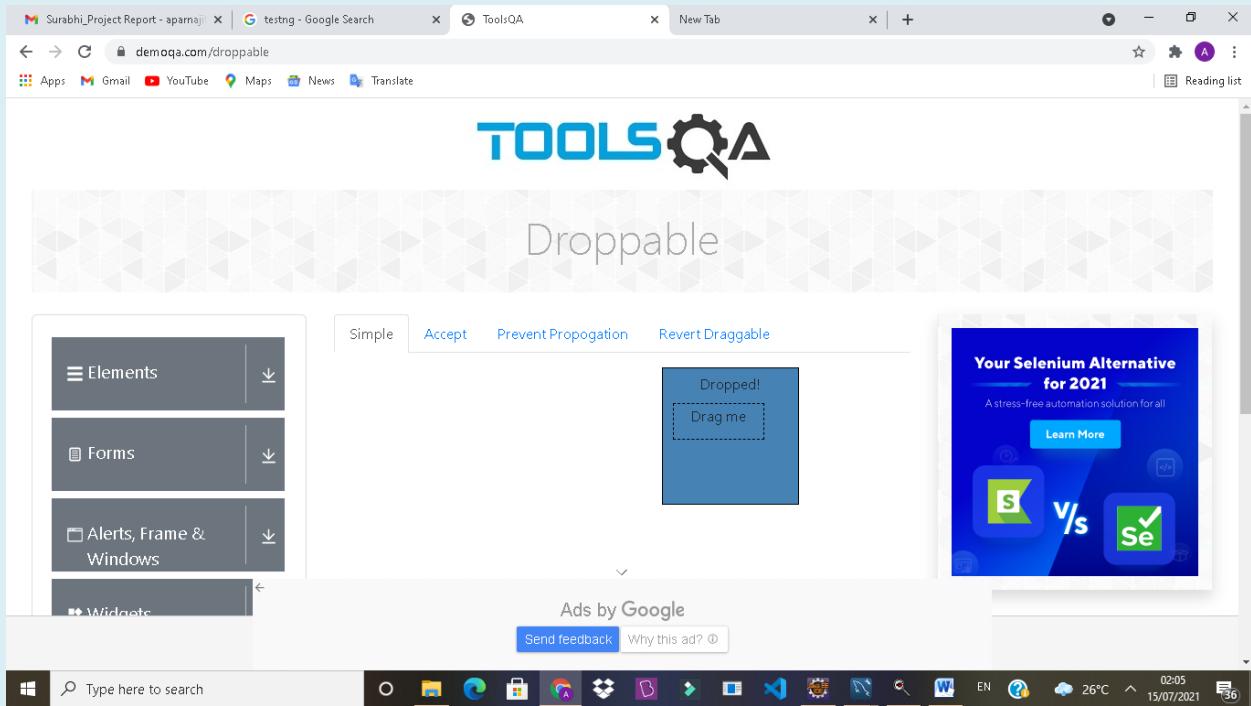
TC163- To verify the functionality of simple subsection in droppable, to check whether it dropping according to script or not

## TestNG Test Script

```
WebElement dragsimple =
W.findElement(By.id("draggable"));

WebElement dropsimple =
W.findElement(By.id("droppable"));

Actions actsimple = new Actions(W);
actsimple.dragAndDrop(dragsimple,dropsimple).build()
.perform();
Thread.sleep(3000);
```

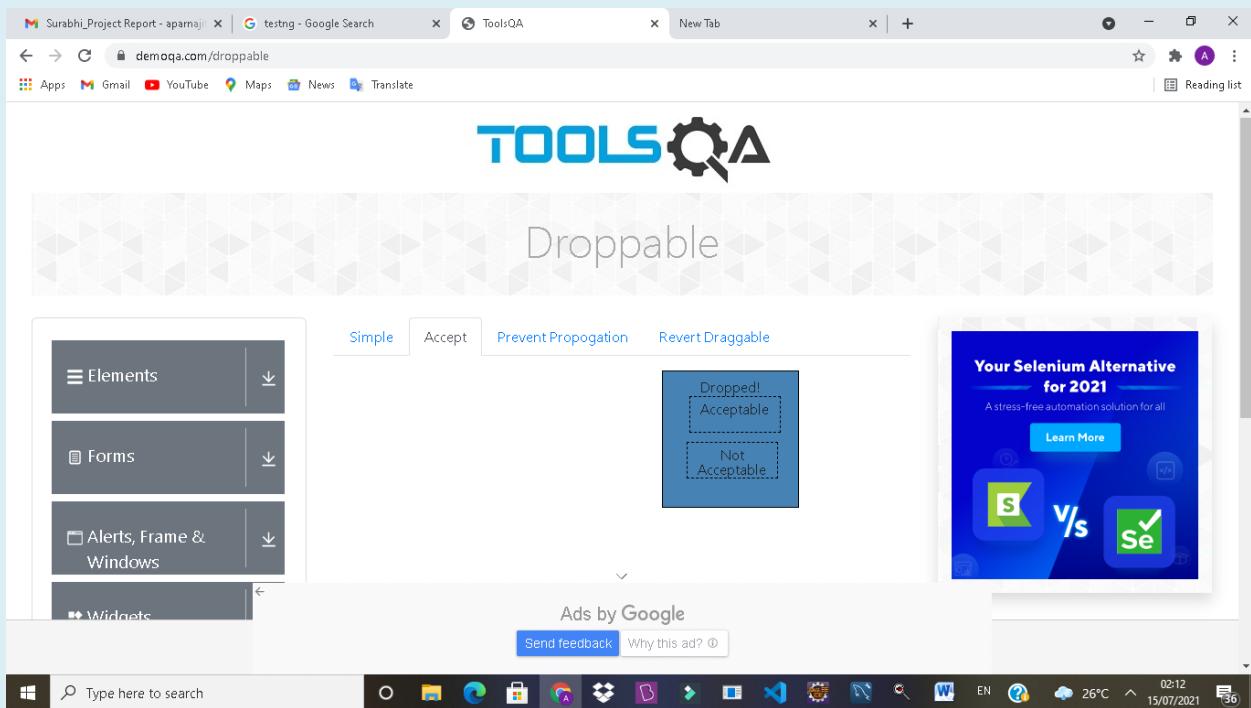


TC164- To verify the functionality of accept subsection in droppable, to check whether it dropping according to script or not

## TestNG Test Script

```
W.findElement(By.linkText("Accept")).click();
WebElement dragaccept1 =
W.findElement(By.id("acceptable"));
WebElement dragaccept2 =
W.findElement(By.id("notAcceptable"));
WebElement dropaccept =
W.findElement(By.xpath("//div[@id='acceptDropContain
er']//div[@id='droppable']"));

Actions actaccept = new Actions(W);
actaccept.dragAndDrop(dragaccept1,dropaccept).build(
).perform();
actaccept.dragAndDrop(dragaccept2,dropaccept).build(
).perform();
```



## OUTPUT/DEFECT:

In the accept section the box of Not Acceptable had been dropped

TC165- To verify the functionality of Prevent Propogation subsection in droppable, to check whether it dropping according to script or not

## TestNG Test Script

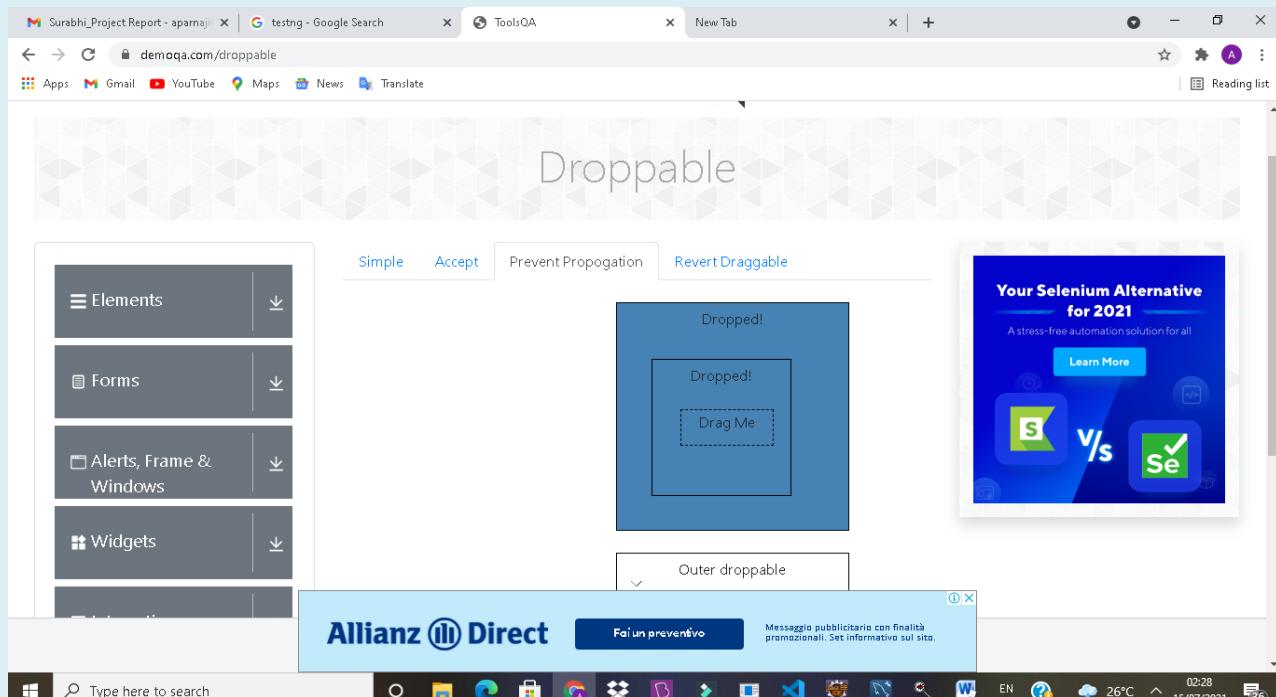
```
W.findElement(By.linkText("PreventPropogation")).click();
    WebElement dragprevent1
W.findElement(By.id("dragBox"));
    WebElement dropprevent1 =
W.findElement(By.id("notGreedyInnerDropBox"));

    WebElement dropprevent2 =
W.findElement(By.xpath("//div[@id='greedyDropBoxInner']");
Actions actprevent = new Actions(W);
```

```

actprevent.dragAndDrop(dragprevent1,dropprevent1).build().perform();
Thread.sleep(3000);
actprevent.dragAndDrop(dragprevent1,dropprevent2).build().perform();

```



### OUTPUT/DEFECT:

In prevent Propagation section the box is not dropping as per the requirement

TC166- To verify the functionality of Revert Draggable subsection in droppable, to check whether it dropping according to script or not

TestNG Test Script

```

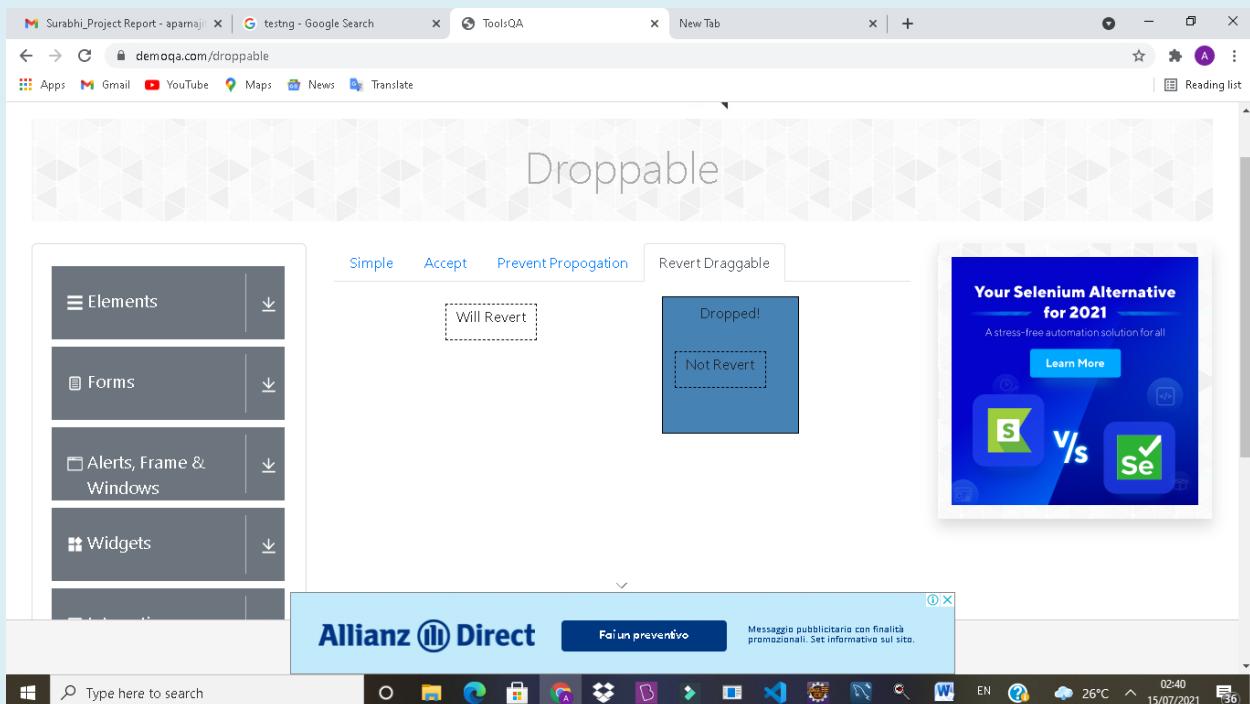
W.findElement(By.linkText("RevertDraggable")).click()
;
    WebElement dragrevert1 =
W.findElement(By.id("revertable"));
    WebElement dragrevert2 =
W.findElement(By.id("notRevertable"));

```

```

WebElement droprevert =
W.findElement(By.xpath("//div[@id='revertableDropContainer']//div[@id='droppable']"));
Actions actrevert = new Actions(W);
actrevert.dragAndDrop(dragrevert1,droprevert).build().perform();
actrevert.dragAndDrop(dragrevert2,droprevert).build().perform();

```



TC167-To verify the functionality of navigating submodule (Draggable section) page from Droppable page

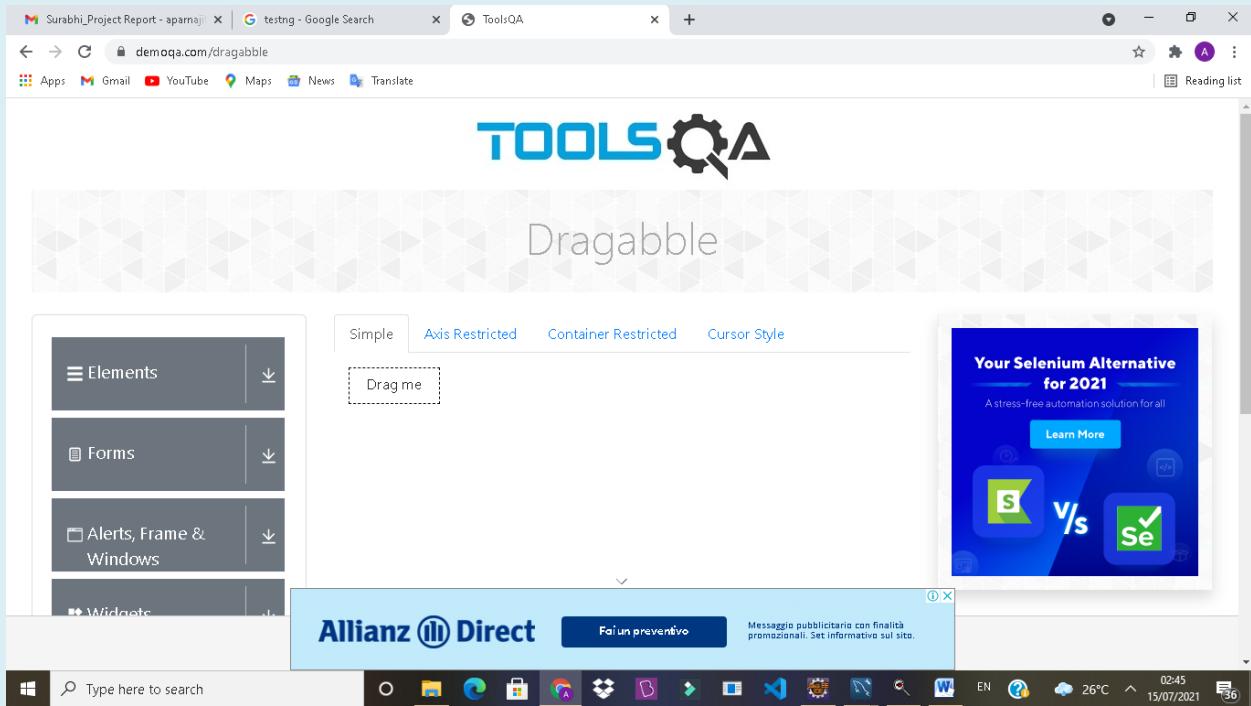
## TestNG Test Script

```

@Test(dependsOnMethods = {"dragablepage"})
public void dragablepage() throws
InterruptedException {

W.navigate().to("https://demoqa.com/dragabble");
}

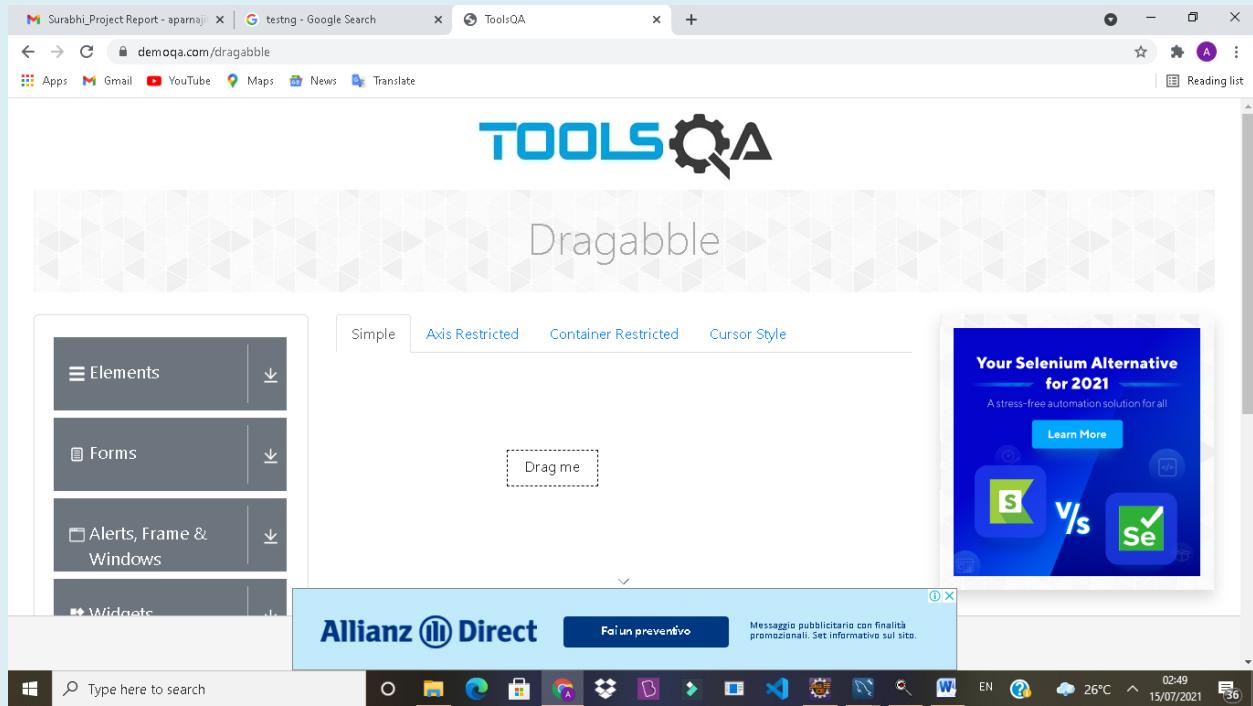
```



TC168- To verify the functionality of simple subsection in dragabble, to check whether it dragging according to script or not

## TestNG Test Script

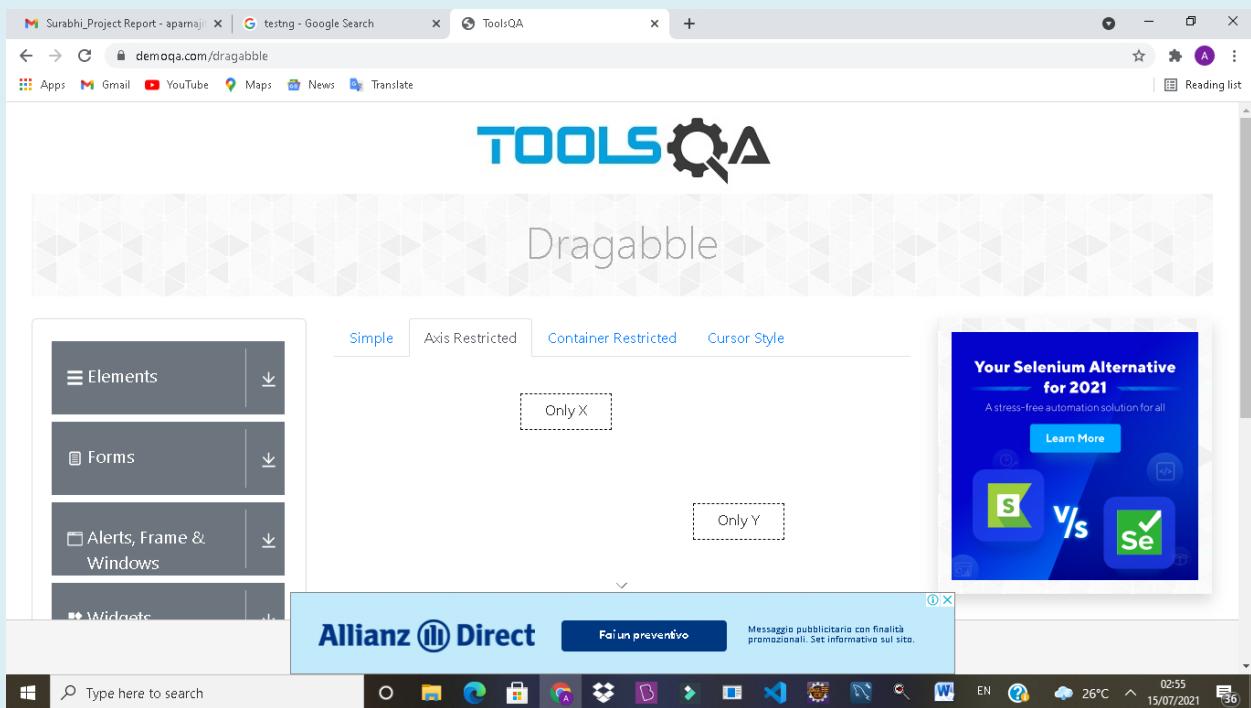
```
WebElement  
dragable=W.findElement(By.xpath("//div[@id='dragBox'  
]);  
  
Actions actdrag=new Actions(W);  
actdrag.dragAndDropBy(dragable,  
60,70).build().perform();
```



TC169- To verify the functionality of Axis restricted subsection in dragabble, to check whether it dragging according to script or not

## TestNG Test Script

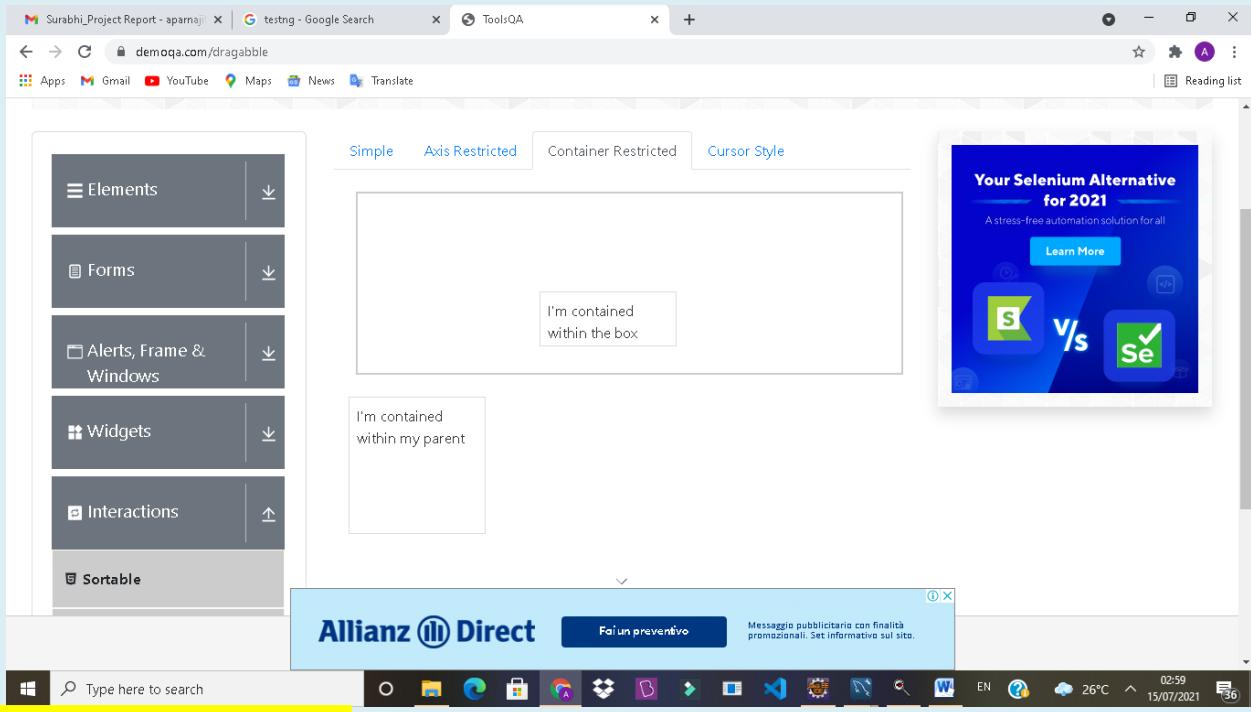
```
W.findElement(By.id("draggableExample-tab  
axisRestriction")).click();  
WebElement  
dragaxisX=W.findElement(By.id("restrictedX"));  
WebElement dragaxisY =  
W.findElement(By.id("restrictedY"));  
Actions actaxis=new Actions(W);  
  
actaxis.dragAndDropBy(dragaxisX, 80,  
0).build().perform();  
actaxis.dragAndDropBy(dragaxisY, 0,  
100).build().perform();
```



TC170- To verify the functionality of Container Restricted subsection in dragabble, to check whether it dragging according to script or not

## TestNG Test Script

```
W.findElement(By.id("draggableExample-tab-containerRestriction")).click();
WebElement
dragcontainer=W.findElement(By.xpath("//div[contains(text(),\"I'm contained within the box\")]"));
Actions actcont=new Actions(W);
actcont.dragAndDropBy(dragcontainer, 60,
90).build().perform();
JavascriptExecutor jscontainer =
(JavascriptExecutor) W;
jscontainer.executeScript("window.scrollBy(0,300)");
```



## OUTPUT/DEFECT:

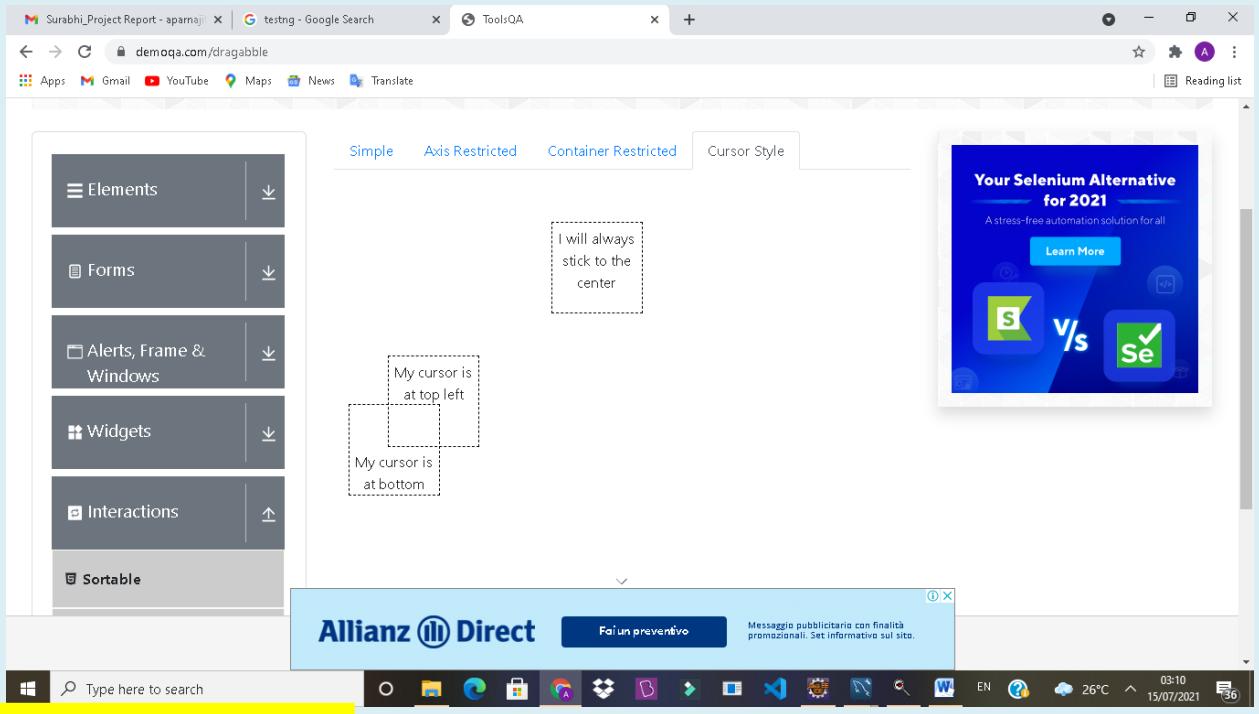
2nd box not executing as per the script

TC171- To verify the functionality of Cursor subsection in dragabble, to check whether it dragging according to script or not

## TestNG Test Script

```
@Test(enabled = false)
public void cursorsection() throws
InterruptedException {
W.findElement(By.linkText("Cursor Style")).click();
WebElement dragcursor =
W.findElement(By.id("cursorCenter"));
Actions actcursor=new Actions(W);
actcursor.dragAndDropBy(dragcursor,
170,100).build().perform();
Thread.sleep(3000);

 WebElement dragcursor1=
W.findElement(By.id("cursorTopLeft"));
```



## OUTPUT/DEFECT:

Cursor section not dragging according to the requirement

TC172- To verify the functionality of closing the current module

TestNG Test Script

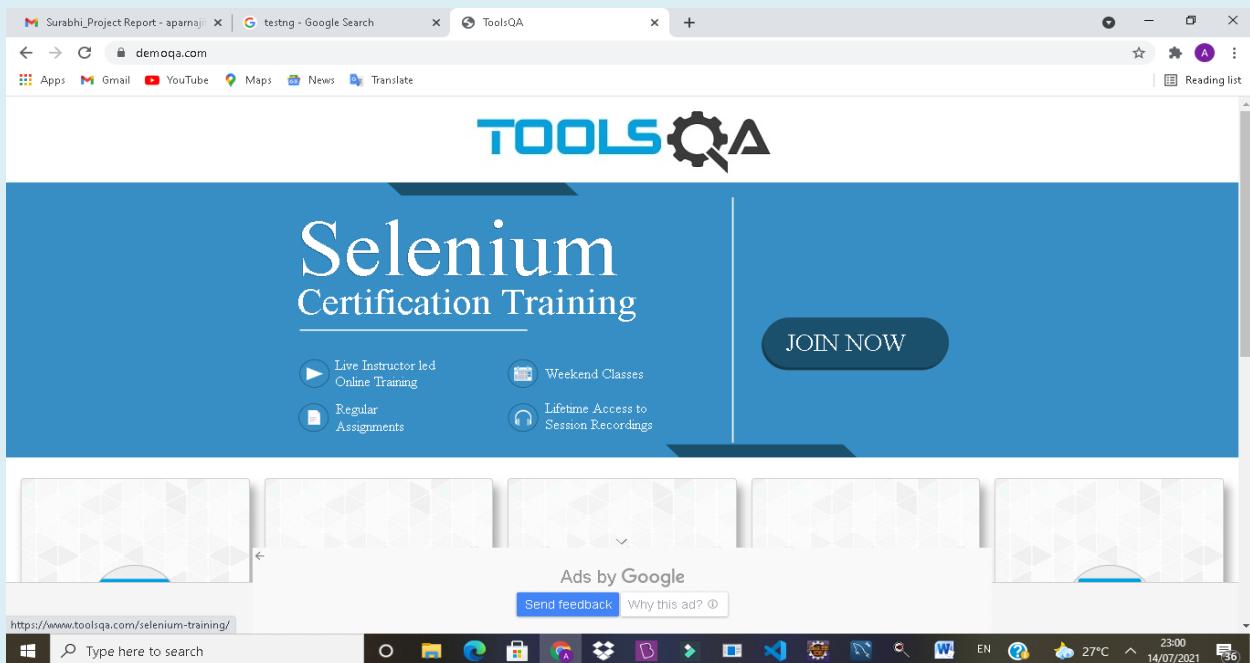
```
@AfterTest  
public void afterTest() {  
    W.close();  
}  
}
```

## Module No.6 – Book Store Application

TC173- Launch the Firefox browser and open ToolsQA Website

### TestNG Test Script

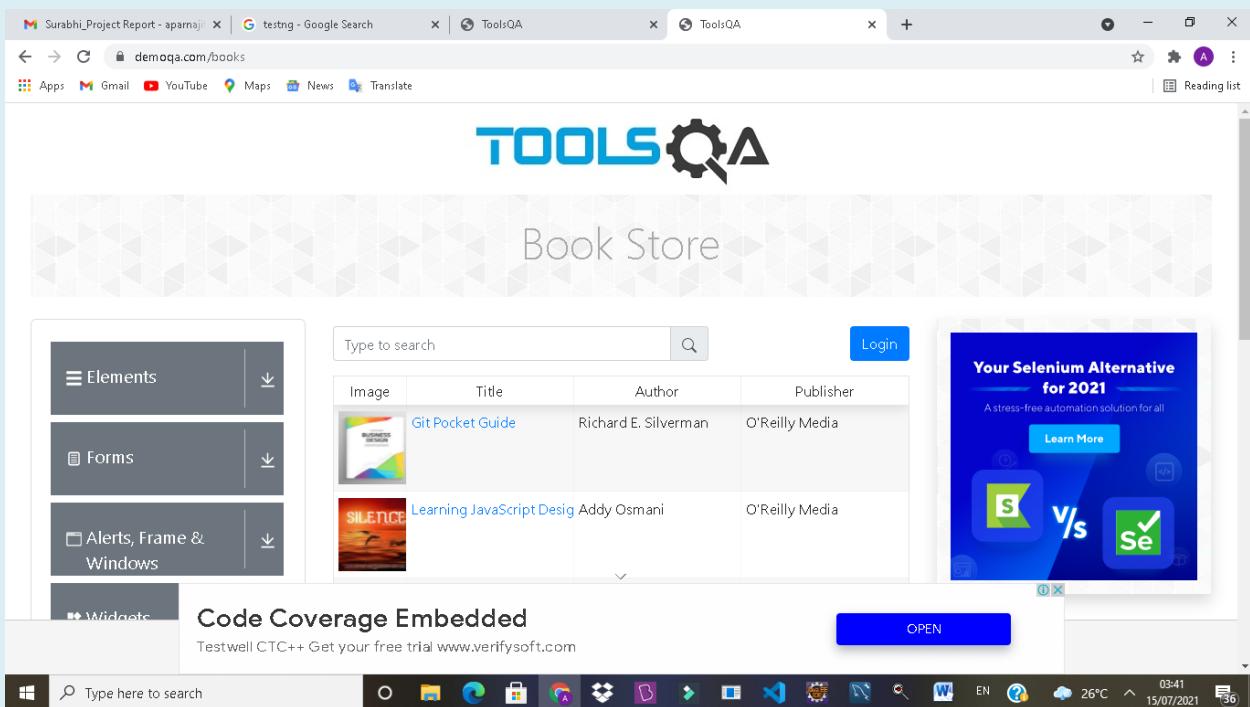
```
WebDriver W;  
@BeforeTest  
public void Setproperty() {  
    System.setProperty("webdriver.gecko.driver",  
"C:\\\\Users\\\\hp\\\\Desktop\\\\Seleniumjar\\\\geckodriver.exe");  
    W = new FirefoxDriver();  
  
    W.get("https://demoqa.com/");
```



## TC174- To verify the functionality of Book Store Application module of ToolsQA Website

## TestNG Test Script

```
@Test  
public void mainpage() throws InterruptedException {  
  
    WebElement apps1=  
W.findElement(By.xpath("//body[1]/div[2]/div[1]/div[1]  
/div[2]/div[1]/div[6]/div[1]"));  
JavascriptExecutor js1 = (JavascriptExecutor) W;  
  
js1.executeScript("arguments[0].scrollIntoView();",  
apps1);  
  
apps1.click();
```



## **OUTPUT/DEFECT:**

In Book store Application module arrow functionality is Not working by scripting way and also without clicking an arrow , it automatically open the subsection of the Book Store Application as well as display the book store subsection.

TC175- To verify and display the Current URL as well as Tittle of Current page

TestNG Test Script

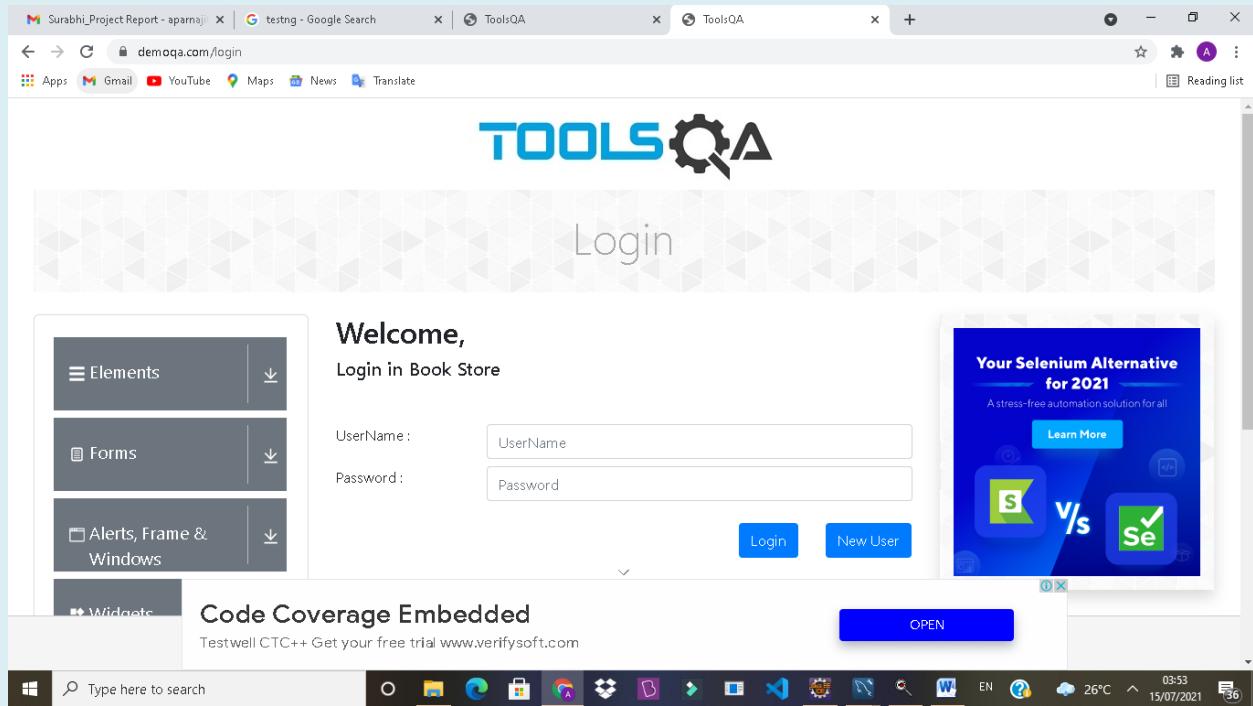
```
System.out.println("Current URL is  
+W.getCurrentUrl());  
  
System.out.println(("Tittle of page is "  
+W.getTitle()));
```

**OUTPUT:-** Current URL is https://demoqa.com/books  
Tittle of page is ToolsQA

TC176- To verify the functionality of submodule (Login section) of Book store Application module

TestNG Test Script

```
@Test(dependsOnMethods = {"mainpage"})  
public void registerpage() throws  
InterruptedException {  
  
W.navigate().to("https://demoqa.com/login");
```



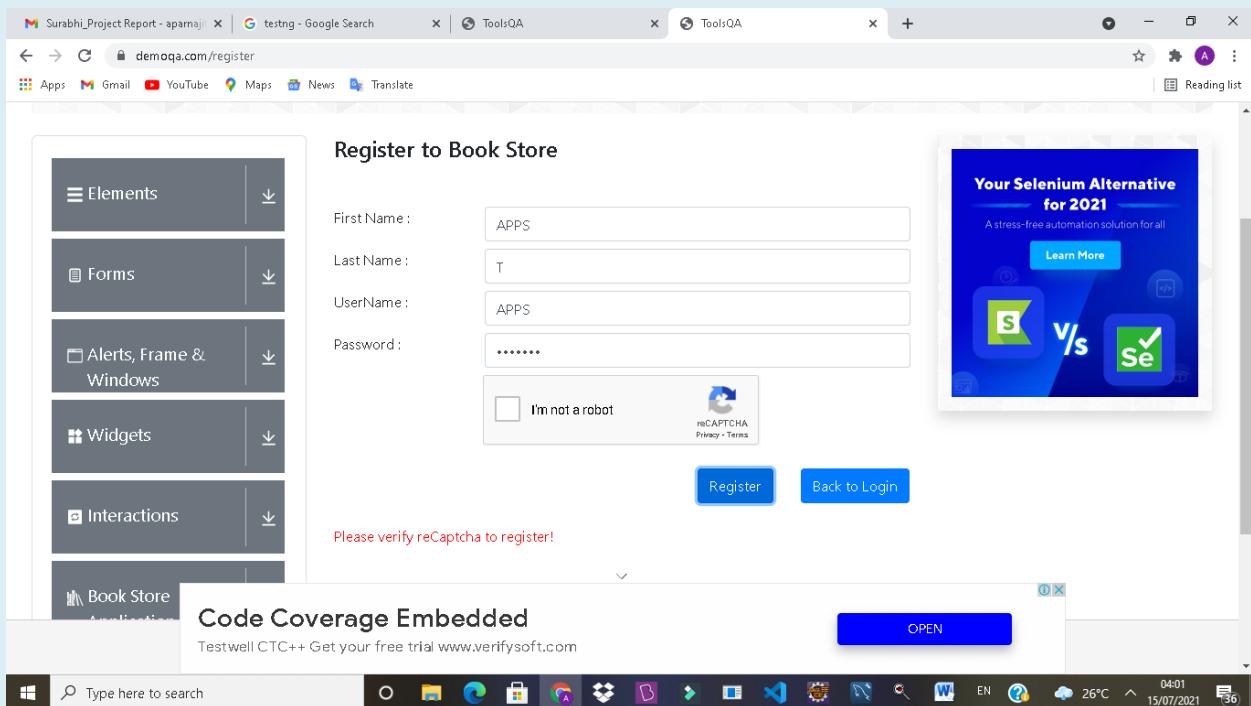
TC177- To verify the functionality of New User in Login section

## TestNG Test Script

```
W.findElement(By.id("newUser")).click();
W.findElement(By.id("firstname")).sendKeys("APPS");
W.findElement(By.id("lastname")).sendKeys("T");
W.findElement(By.id("userNmae")).sendKeys("APPS");
W.findElement(By.id("password")).sendKeys("APPS44");

//Validation is compulsory to register(i'm not robot)
WebElement store1 = W.findElement(By.id("register"));

JavascriptExecutor jsstore1 = (JavascriptExecutor) W;
jsstore1.executeScript("arguments[0].scrollIntoView();",
" , store1);
store1.click();
```



## OUTPUT/DEFECT:

Registration form don't have validation for first name and last name it accepting special characters as well as number

TC178 - To verify the functionality of go to login button from register to Login section

## TestNG Test Script

```
Thread.sleep(4000);
```

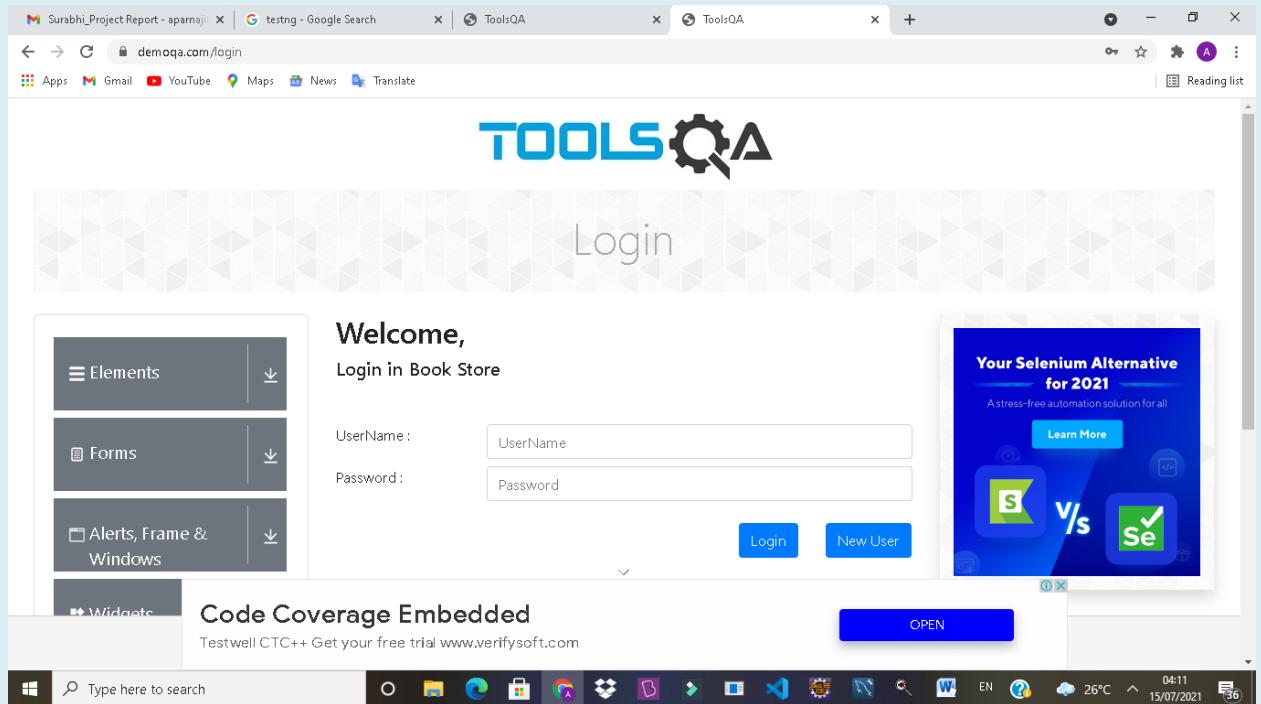
```
WebElement store=
```

```
W.findElement(By.id("gotologin"));
```

```
JavascriptExecutor jsstore = (JavascriptExecutor) W;  
jsstore.executeScript("arguments[0].scrollIntoView();"  
, store);
```

```
store.click();
```

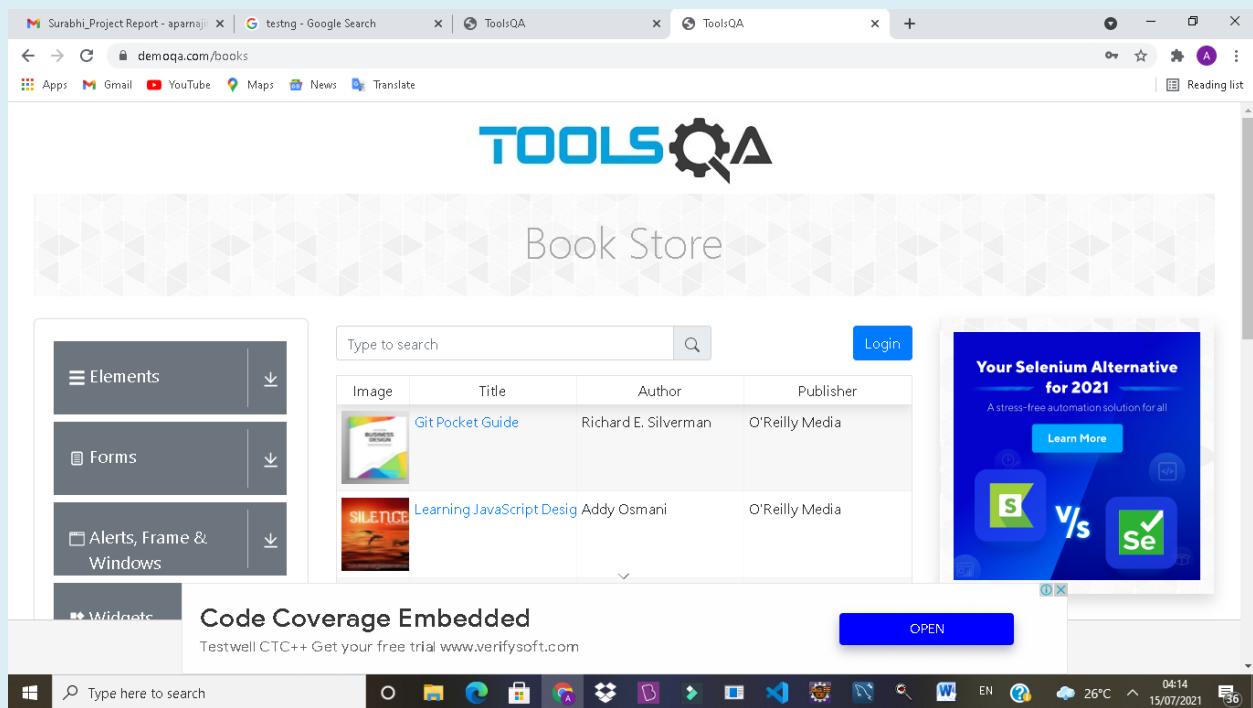
```
}
```



TC179 - To verify the functionality of navigating submodule (Book store section) page from Login section

## TestNG Test Script

```
@Test(dependsOnMethods = {"registerpage"})
public void bookpage() throws InterruptedException {
    w.navigate().to("https://demoqa.com/books");
}
```

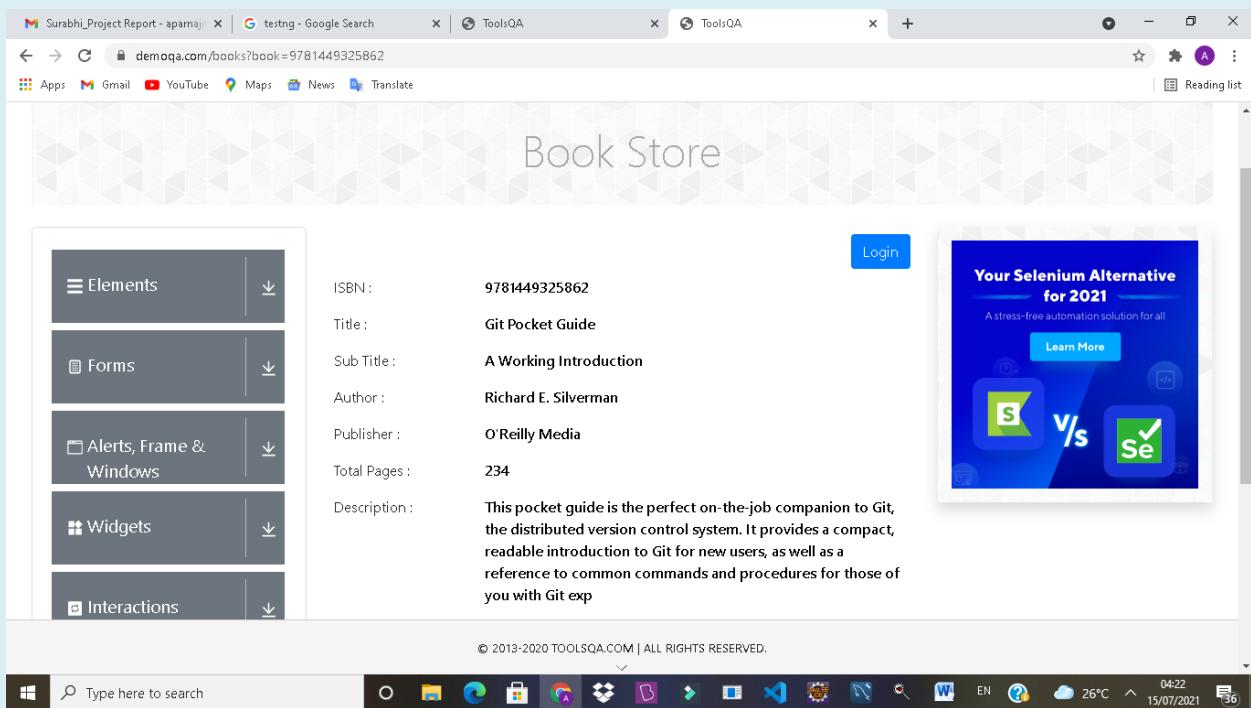


TC180- To verify the functionality of Git Pocket Guide in Book store section, to check whether it opening the details of book or not

## TestNG Test Script

```
WebElement book1=
W.findElement(By.linkText("Git Pocket Guide"));

JavascriptExecutor jsbook1 = (JavascriptExecutor) W;
jsbook1.executeScript("arguments[0].scrollIntoView();"
, book1);
book1.click();
```



TC181- To verify the functionality of Back to Book store button from Git Pocket Guide to Book store section

## TestNG Test Script

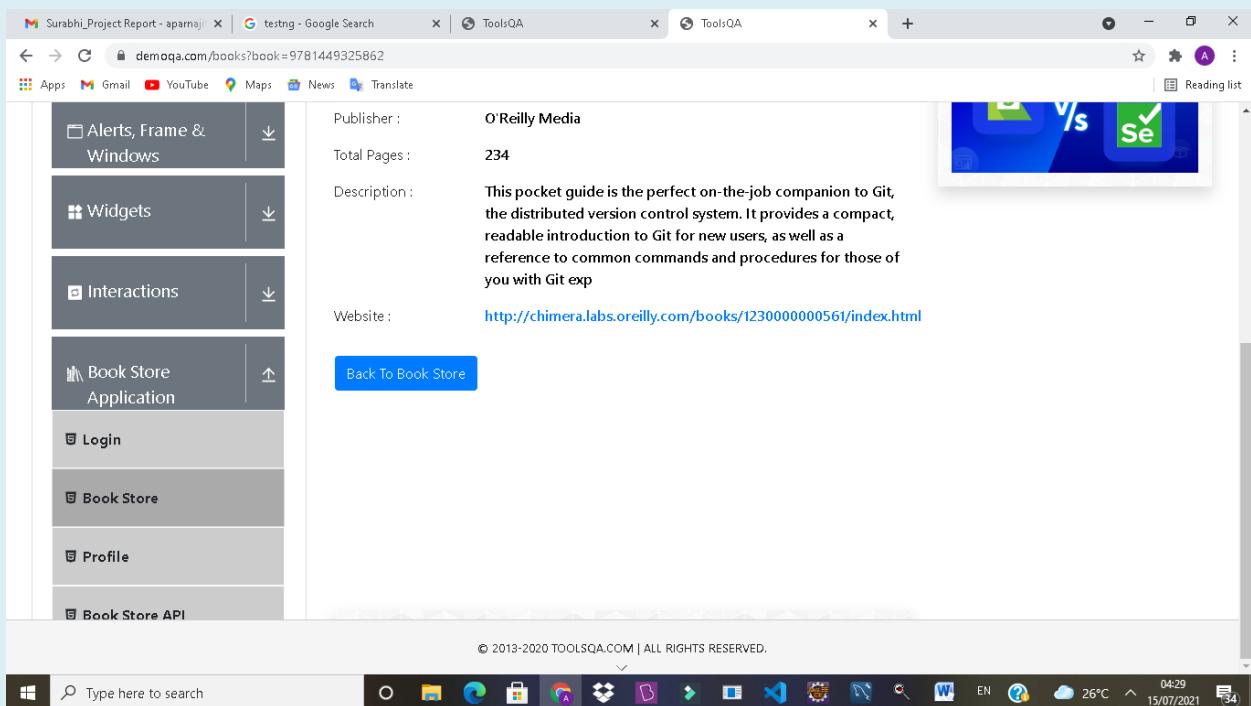
```
WebElement click1=
W.findElement(By.id("addNewRecordButton"));

JavascriptExecutor jsclick1 =
(JavascriptExecutor) W;

jsclick1.executeScript("arguments[0].scrollIntoView();
", click1);

click1.click();

Thread.sleep(4000);
```



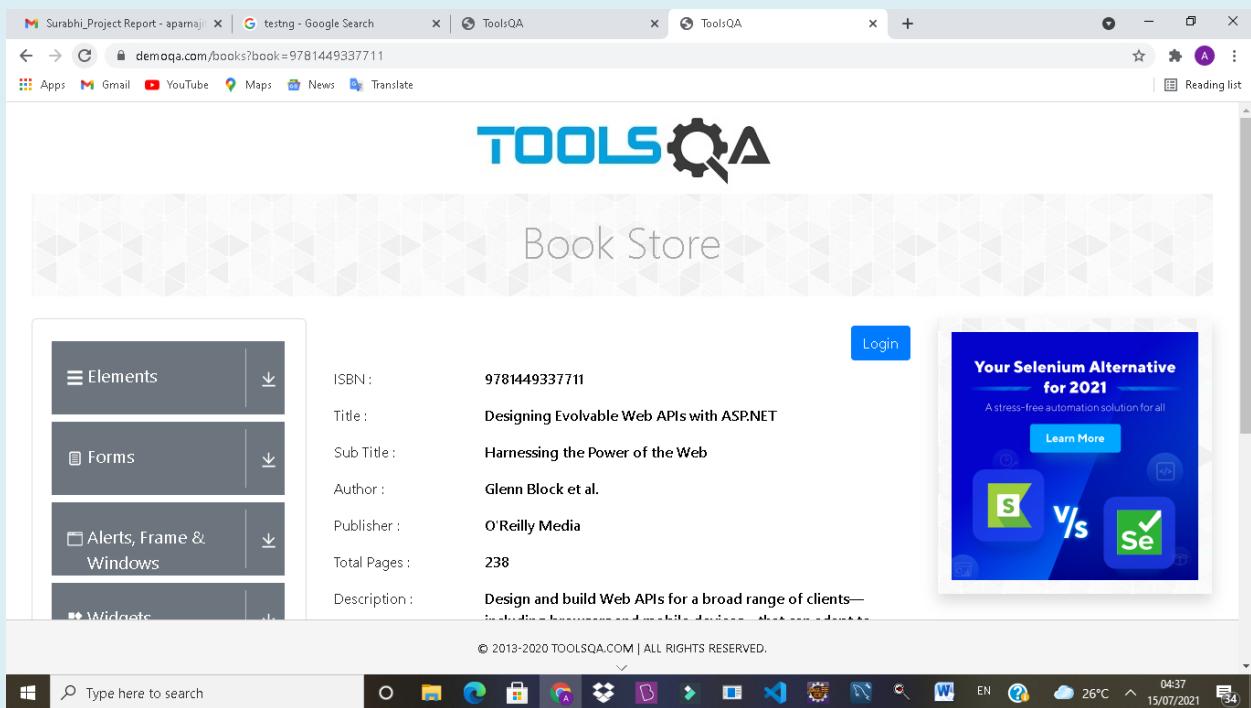
TC182- To verify the functionality of Designing Evolvable Web in Book store section, to check whether it opening the details of book or not

## TestNG Test Script

```
WebElement book3=W.findElement(By.linkText("Designing Evolvable Web APIs with ASP.NET"));

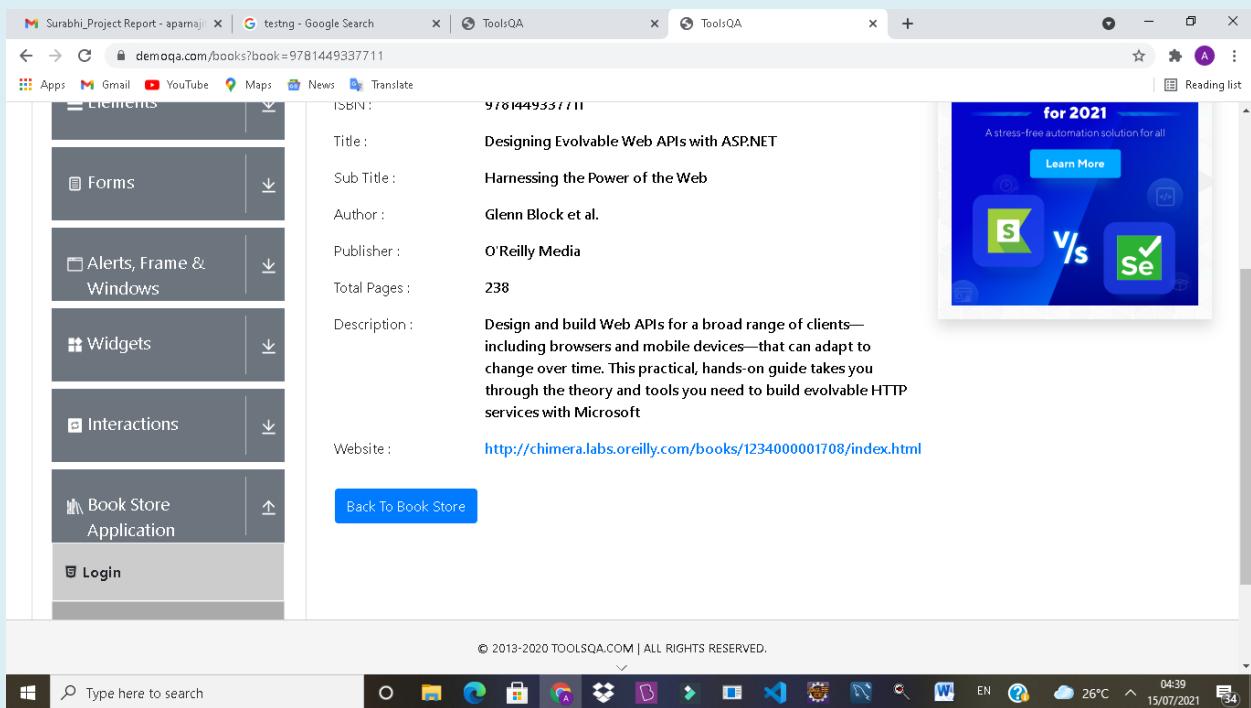
JavascriptExecutor jsbook3 = (JavascriptExecutor) W;
jsbook3.executeScript("arguments[0].scrollIntoView();"
, book3);

book3.click();
```



TC183 - To verify the functionality of Back to Book store button from Designing Evolvable Web to Book store section

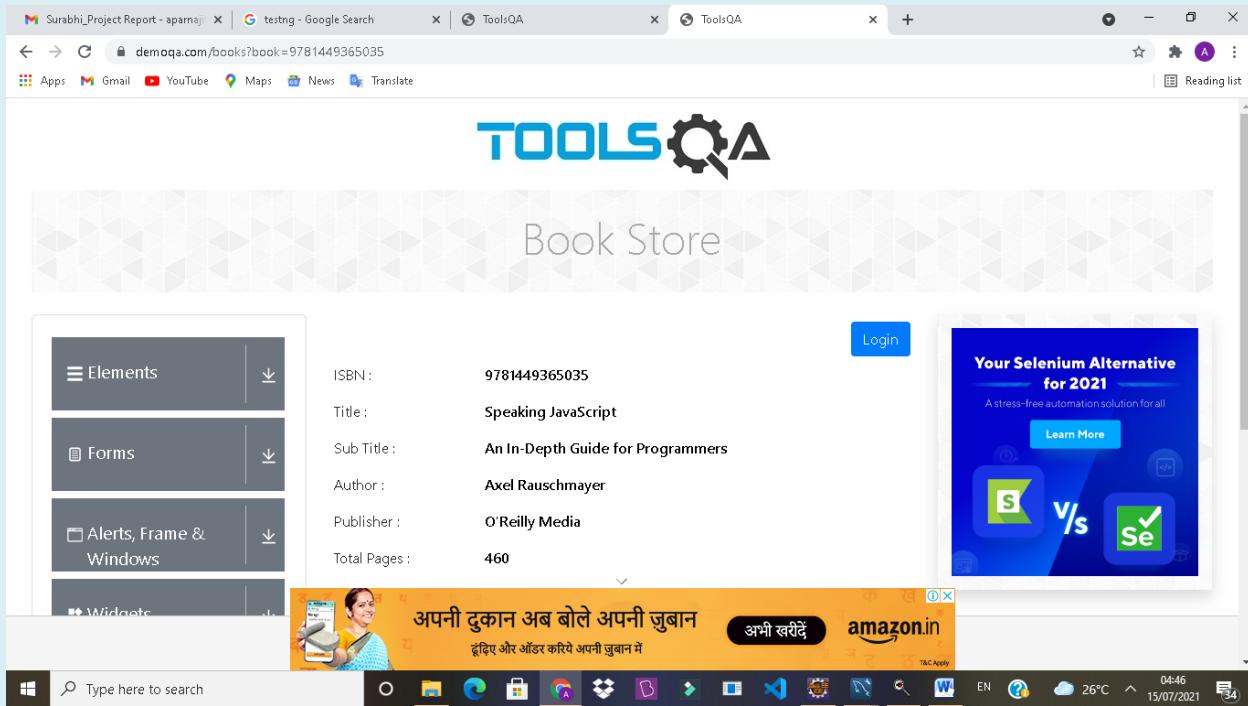
```
WebElement click3=  
W.findElement(By.id("addNewRecordButton"));  
  
JavascriptExecutor jsclick3 = (JavascriptExecutor) W;  
  
jsclick3.executeScript("arguments[0].scrollIntoView();"  
" , click3);  
  
click3.click();  
  
Thread.sleep(4000);
```



TC184- To verify the functionality of Speaking JavaScript in Book store section, to check whether it opening the details of book or not

## TestNG Test Script

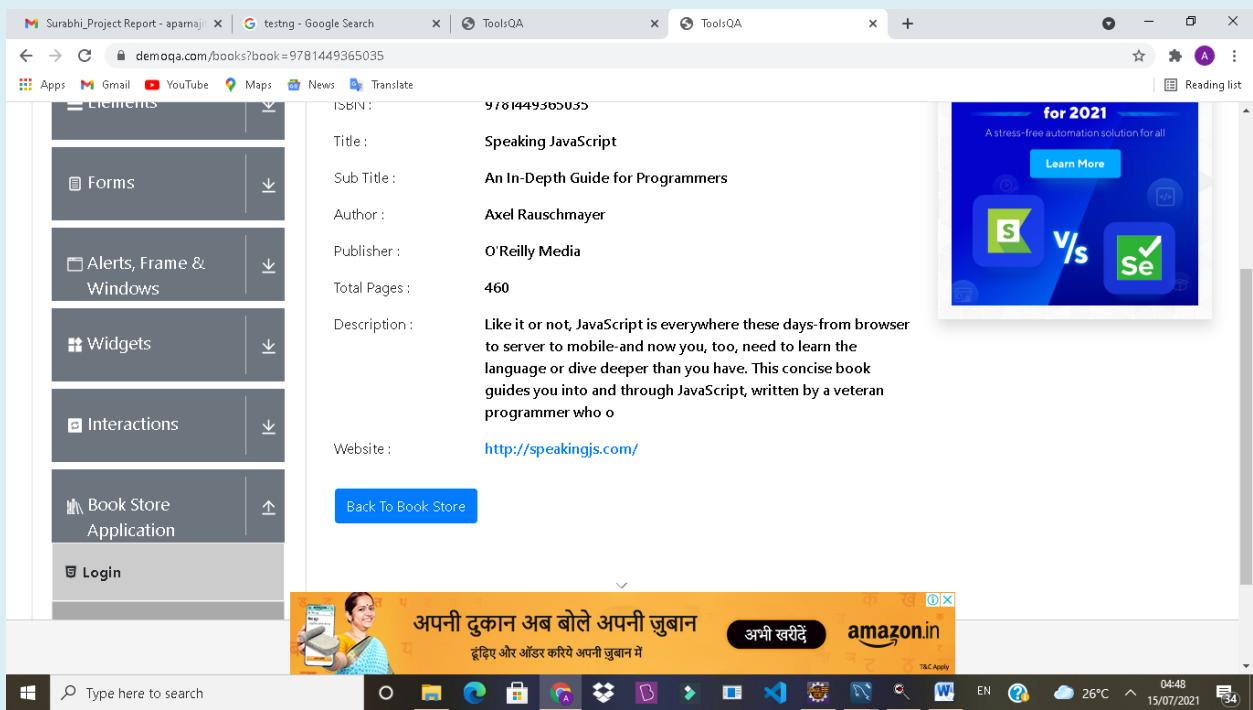
```
WebElement  
book4=W.findElement(By.linkText("Speaking  
JavaScript"));  
  
JavascriptExecutor jsbook4 = (JavascriptExecutor) W;  
  
jsbook4.executeScript("arguments[0].scrollIntoView();"  
, book4);  
  
book4.click();
```



TC185 - To verify the functionality of Back to Book store button from Speaking JavaScript to Book store section

## TestNG Test Script

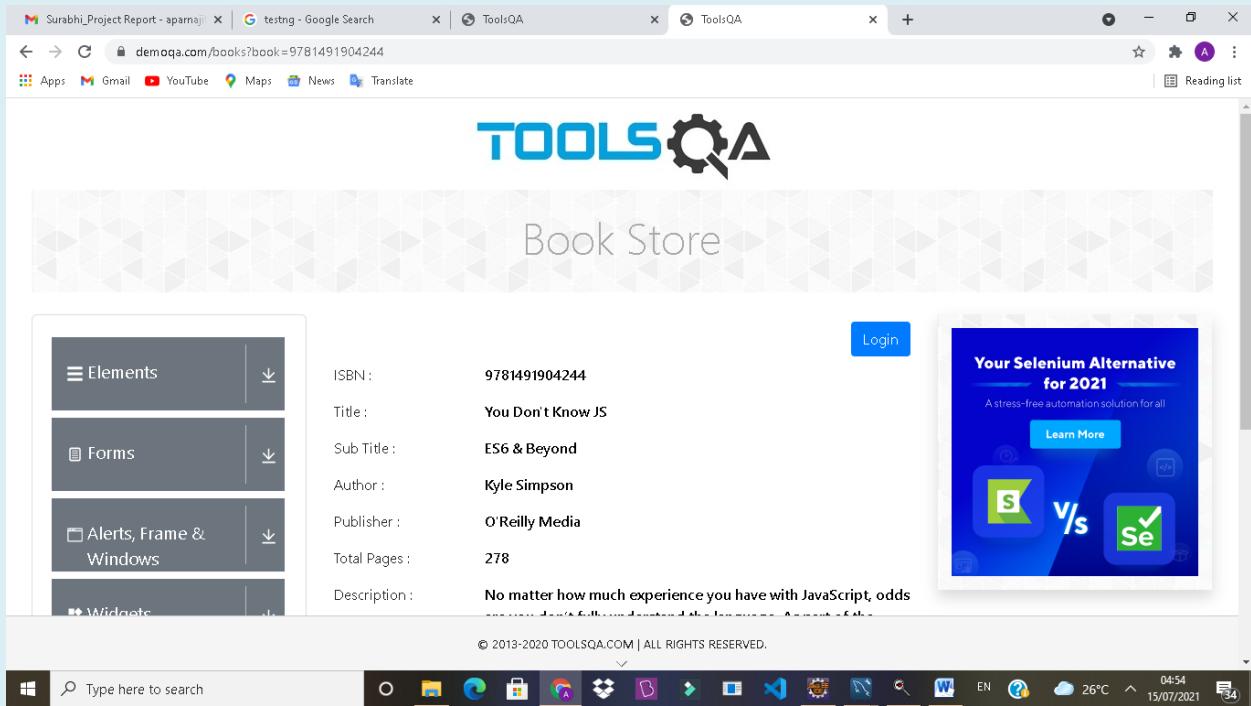
```
WebElement  
click4=W.findElement(By.id("addNewRecordButton"));  
  
JavascriptExecutor jsclick4 =  
(JavascriptExecutor) W;  
  
jsclick4.executeScript("arguments[0].scrollIntoView();"  
" , click4);  
  
click4.click();  
  
Thread.sleep(4000);
```



**TC186- To verify the functionality of You Don't Know JS in Book store section, to check whether it opening the details of book or not**

# TestNG Test Script

```
        WebElement book5=  
W.findElement(By.linkText("You Don't Know JS"));  
  
JavascriptExecutor jsbook5 = (JavascriptExecutor) W;  
  
jsbook5.executeScript("arguments[0].scrollIntoView()  
()", book5);  
  
book5.click();
```



TC187 - To verify the functionality of Back to Book store button from You Don't Know JS to Book store section

## TestNG Test Script

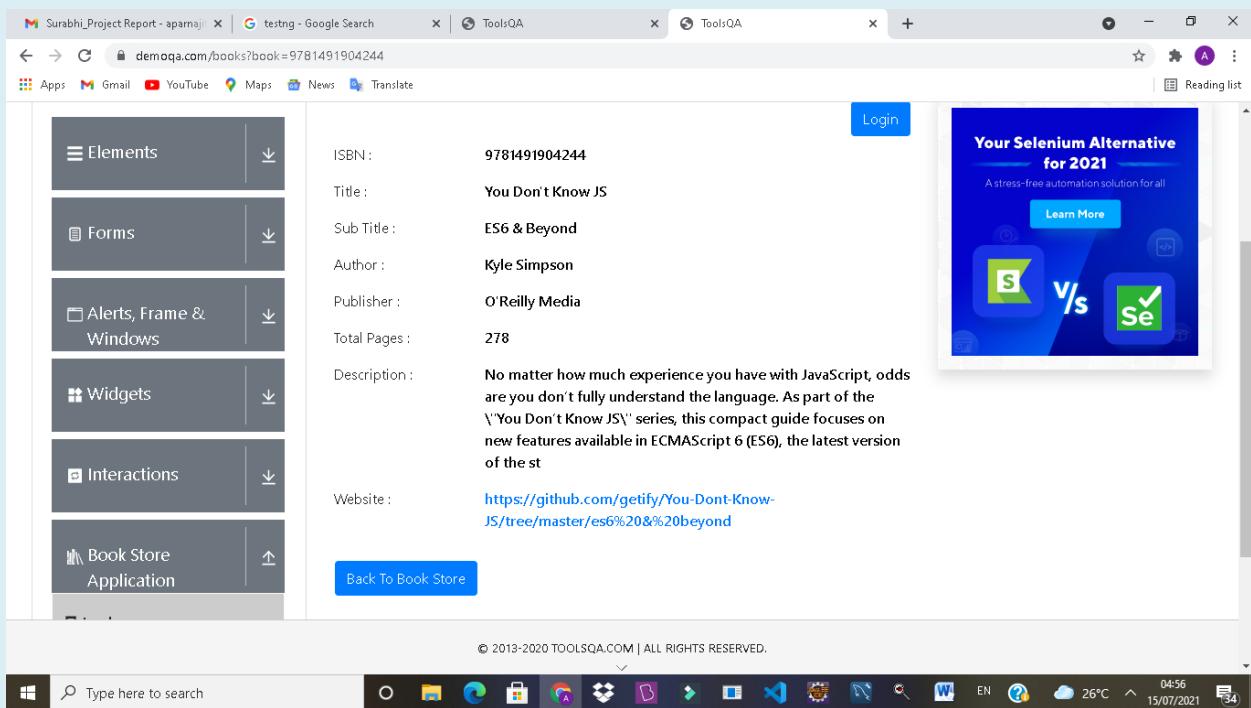
```
WebElement click5=
W.findElement(By.id("addNewRecordButton"));

JavascriptExecutor jsclick5 = (JavascriptExecutor) W;

jsclick5.executeScript("arguments[0].scrollIntoView();
", click5);

click5.click();

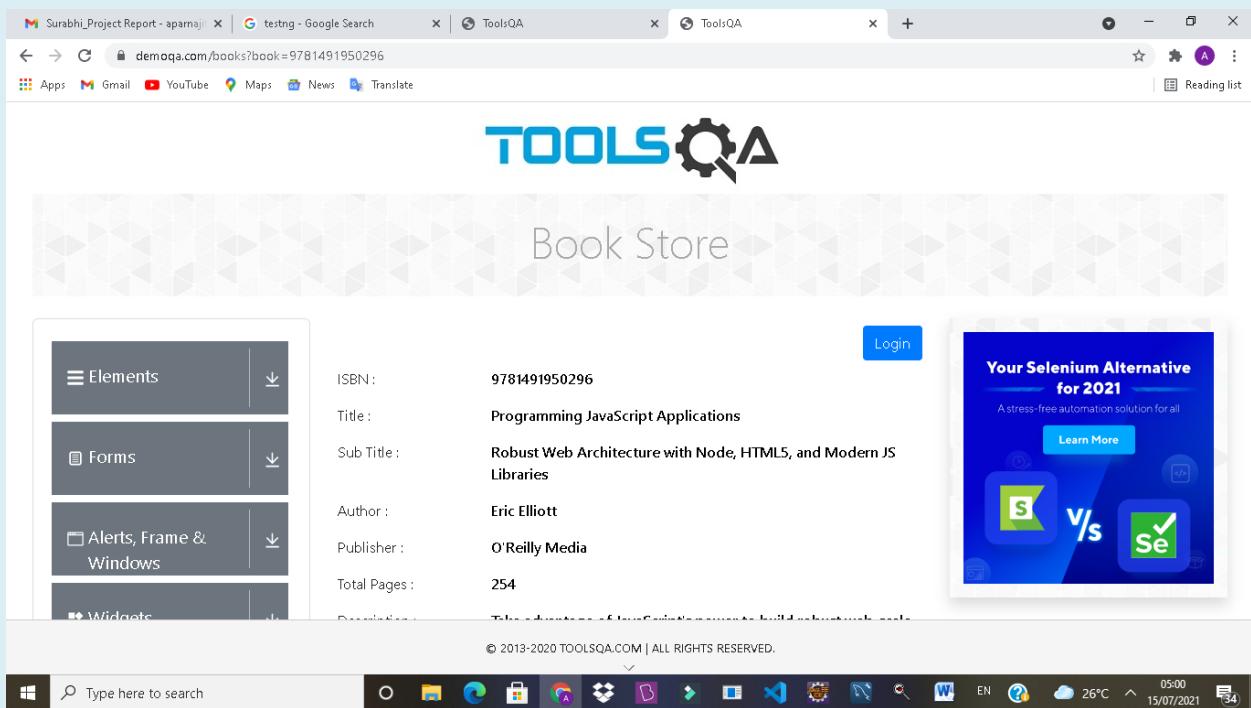
Thread.sleep(4000);
```



TC188- To verify the functionality of Programming JavaScript in Book store section, to check whether it opening the details of book or not

## TestNG Test Script

```
WebElement book6 =  
w.findElement(By.linkText("Programming JavaScript  
Applications"));  
  
JavascriptExecutor jsbook6 = (JavascriptExecutor) w;  
  
jsbook6.executeScript("arguments[0].scrollIntoView();"  
, book6);  
  
book6.click();
```



TC189 - To verify the functionality of Back to Book store button from Programming JavaScript to Book store section

## TestNG Test Script

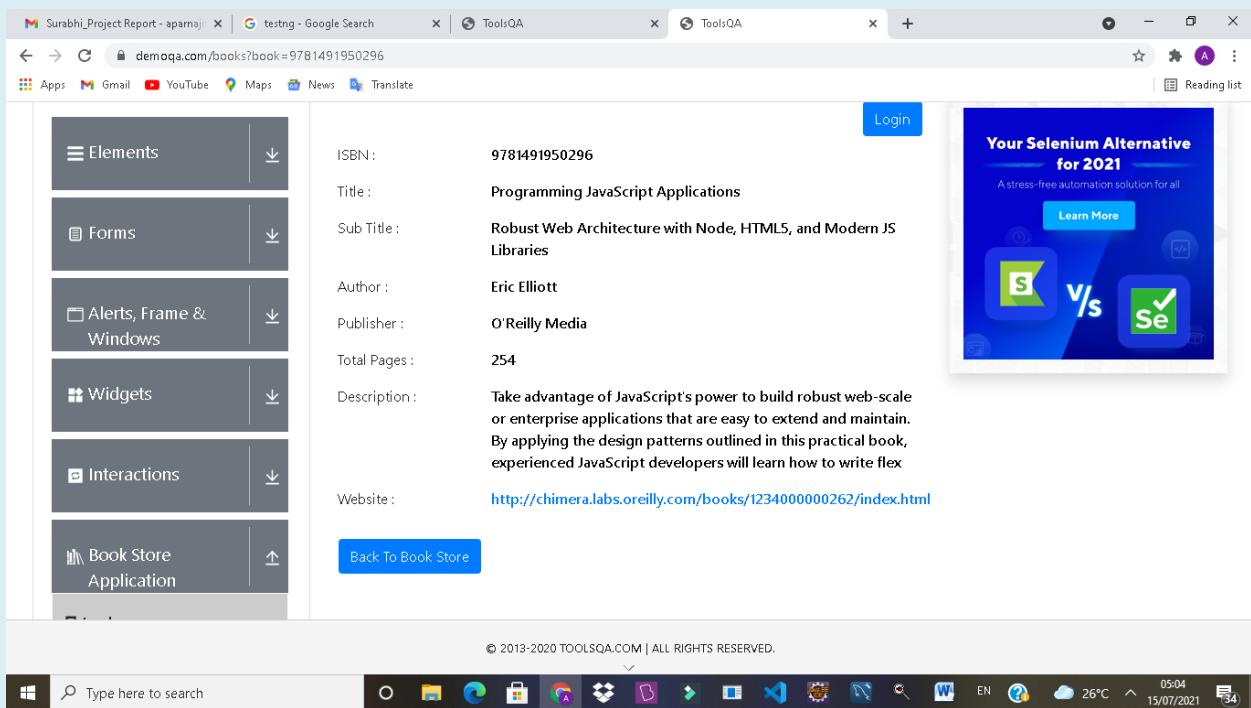
```
WebElement click6=
    W.findElement(By.id("addNewRecordButton"));

JavascriptExecutor jsclick6 = (JavascriptExecutor) W;

jsclick6.executeScript("arguments[0].scrollIntoView();
", click6);

click6.click();

Thread.sleep(4000);
```



TC190- To verify the functionality of Eloquent JavaScript in Book store section, to check whether it opening the details of book or not

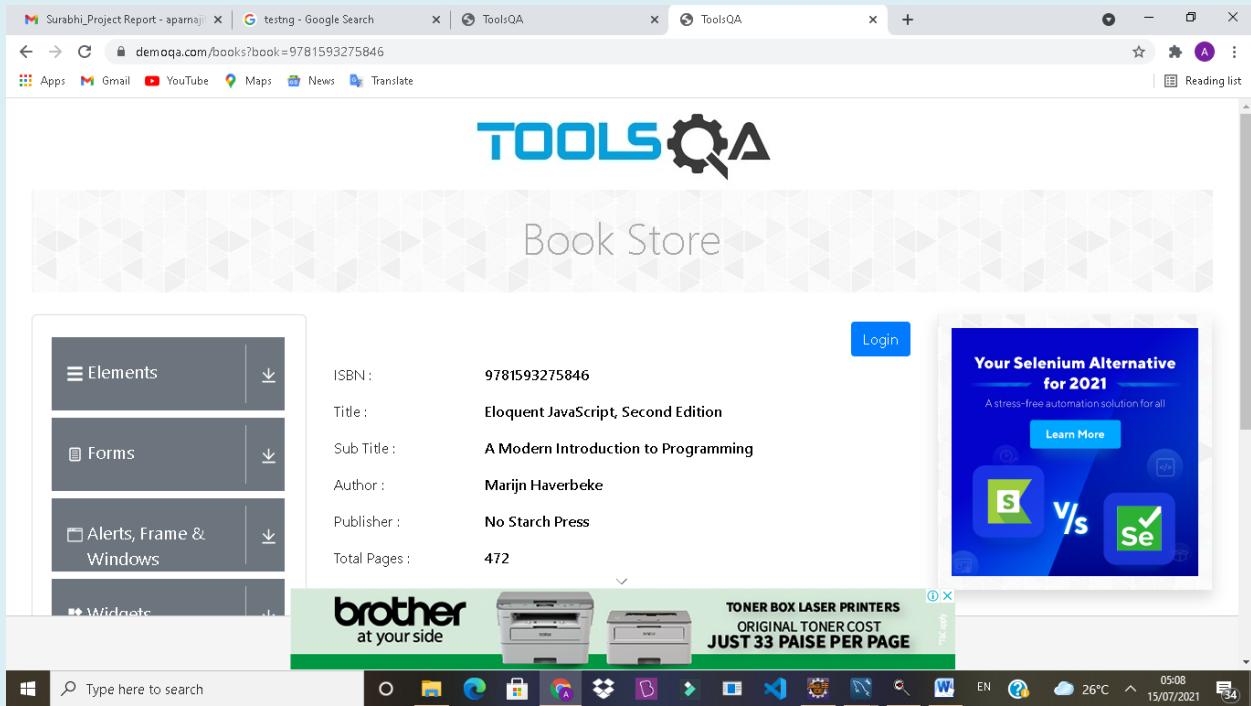
## TestNG Test Script

```
WebElement book7=
W.findElement(By.linkText("Eloquent JavaScript, Second
Edition"));

JavascriptExecutor jsbook7 = (JavascriptExecutor) W;

jsbook7.executeScript("arguments[0].scrollIntoView();"
, book7);

book7.click();
```



TC191- To verify the functionality of Back to Book store button from Programming JavaScript to Book store section

## TestNG Test Script

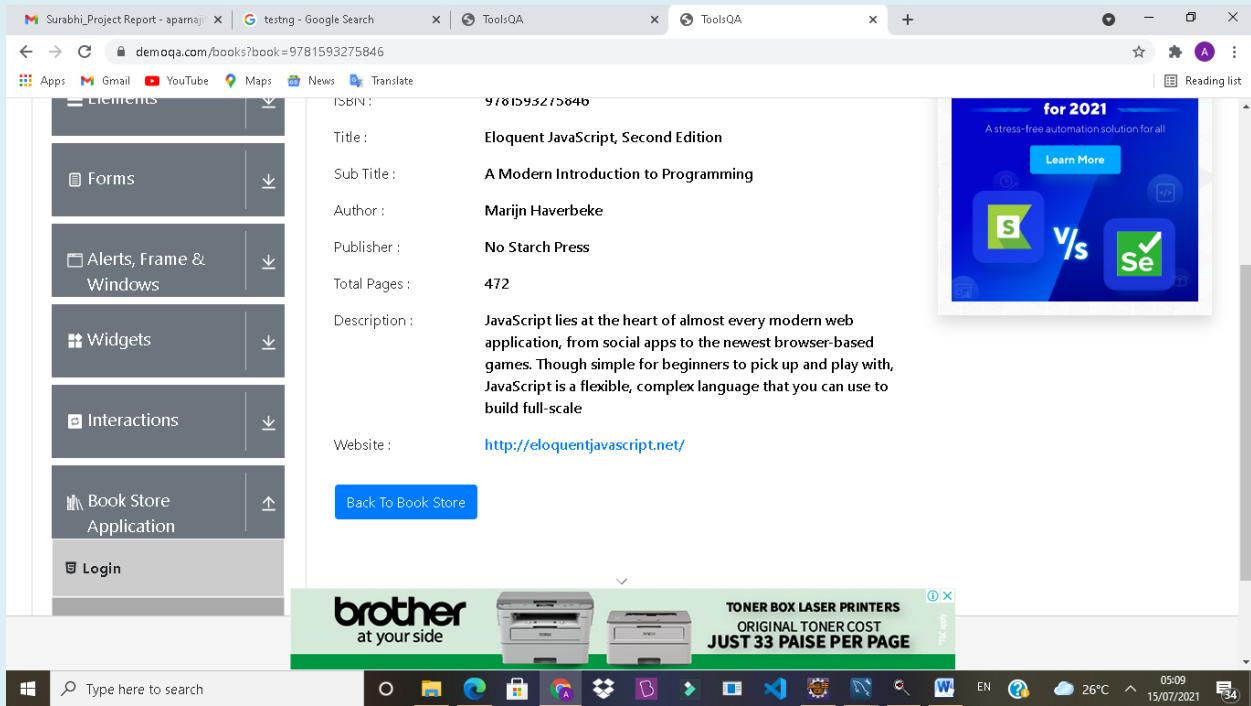
```
WebElement click7=
    W.findElement(By.id("addNewRecordButton"));

JavascriptExecutor jsclick7 = (JavascriptExecutor) W;

jsclick7.executeScript("arguments[0].scrollIntoView();
" , click7);

click7.click();

Thread.sleep(4000);
```



TC192- To verify the functionality of Understanding ECMA Scripting in Book store section, to check whether it opening the details of book or not

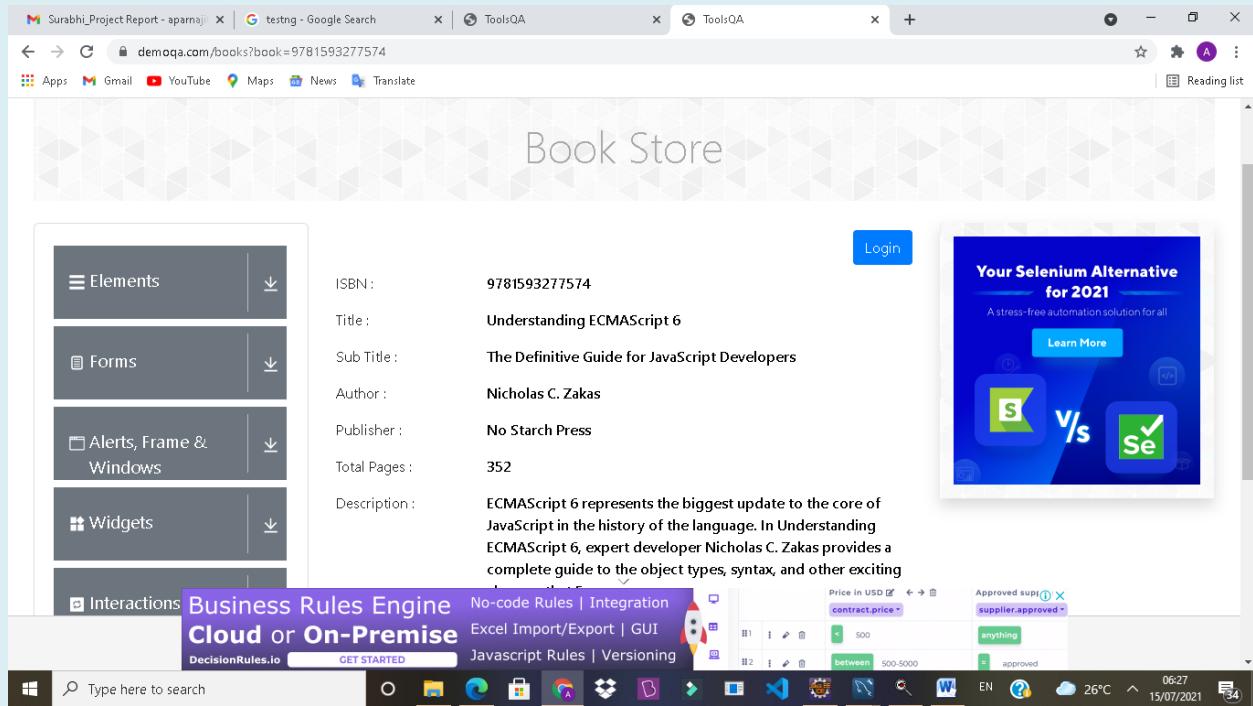
## TestNG Test Script

```
WebElement book8=
    w.findElement(By.linkText("Understanding
ECMAScript 6"));

JavascriptExecutor jsbook8 = (JavascriptExecutor) w;

jsbook8.executeScript("arguments[0].scrollIntoView();"
, book8);

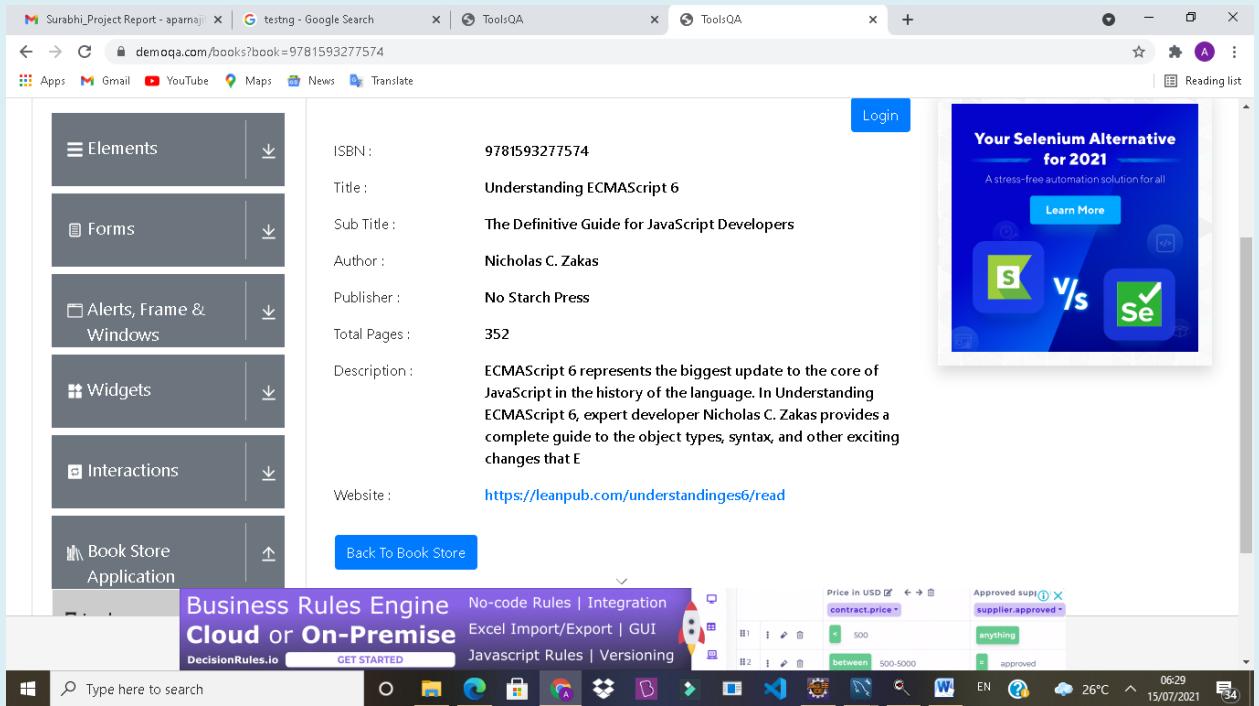
book8.click();
```



TC193 - To verify the functionality of Back to Book store button from Understanding ECMA Scripting to Book store section

## TestNG Test Script

```
WebElement click8=  
    w.findElement(By.id("addNewRecordButton"));  
  
JavascriptExecutor jsclick8 = (JavascriptExecutor) w;  
  
jsclick8.executeScript("arguments[0].scrollIntoView();"  
    , click8);  
  
    click8.click();  
  
    Thread.sleep(4000);
```



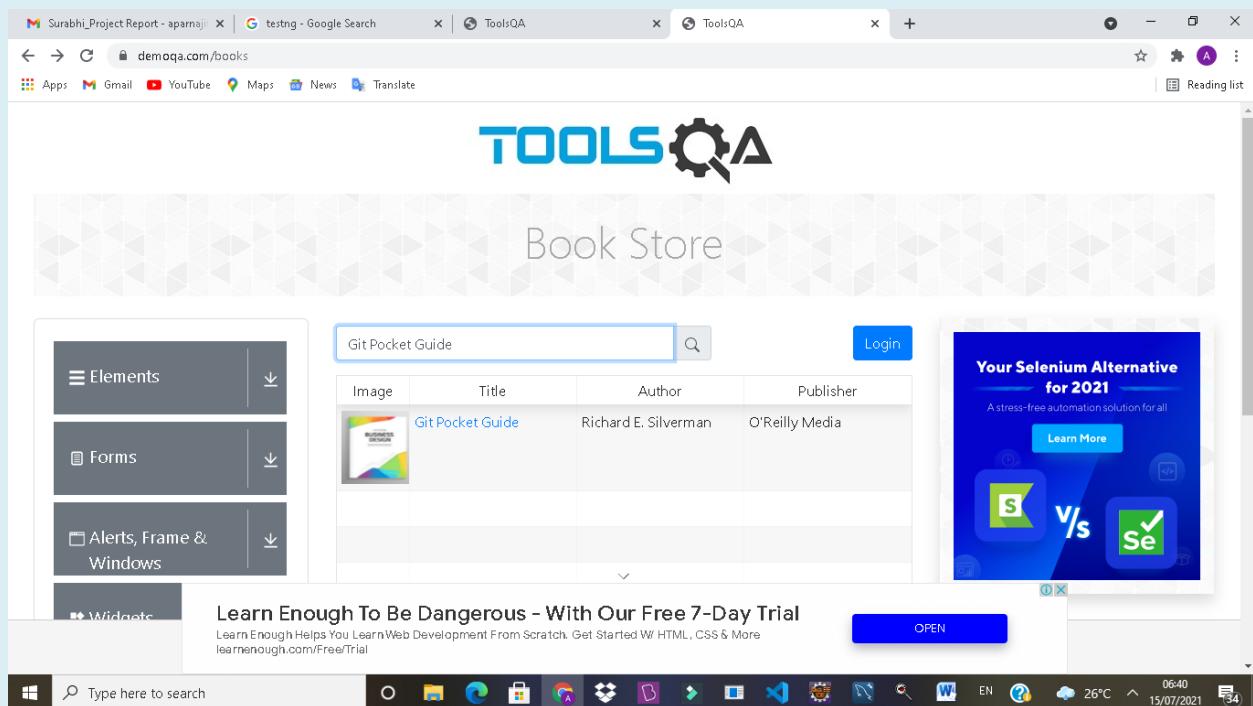
**TC194- To verify the functionality of Search button in Book store section TestNG Test Script**

```
W.findElement(By.xpath("//input[@id='searchBox']")).sendKeys("Git Pocket Guide");

W.findElement(By.xpath("//body/div[@id='app']/div[1]/div[1]/div[2]/div[2]/div[1]/div[1]/div[1]/span[1]/span[1]/*[1]")).click();
Thread.sleep(3000);
```

**TC195- To verify the functionality of login button in Book store section TestNG Test Script**

```
W.findElement(By.id("login")).click();
}
```



TC196- To verify the functionality of Learning JavaScript Design in Book store section, to check whether it opening the details of book or not

## TestNG Test Script

```
@Test(enabled = false)
public void bookpage2() throws InterruptedException {
    WebElement book2=
    W.findElement(By.LinkText("Learning JavaScript Design Patterns"));

    JavascriptExecutor jsbook2 = (JavascriptExecutor) W;

    jsbook2.executeScript("arguments[0].scrollIntoView();"
    , book2);

    book2.click();
}
```

The screenshot shows a web browser window with four tabs open. The active tab displays a book's details from a demoQA website. The sidebar on the left lists categories: Elements, Forms, Alerts, Frame & Windows, Widgets, Interactions, and Book Store. The main content area shows the following details for the book:

ISBN :	9781449331818
Title :	Learning JavaScript Design Patterns
Sub Title :	A JavaScript and jQuery Developer's Guide
Author :	Addy Osmani
Publisher :	O'Reilly Media
Total Pages :	254
Description :	With Learning JavaScript Design Patterns, you'll learn how to write beautiful, structured, and maintainable JavaScript by applying classical and modern design patterns to the language. If you want to keep your code efficient, more manageable, and up-to-date.
Website :	<a href="http://www.addyosmani.com/resources/essentialjsdesignpatterns/book/">http://www.addyosmani.com/resources/essentialjsdesignpatterns/book/</a>

A blue "Back To Book Store" button is located at the bottom of the content area. In the top right corner of the browser window, there is an advertisement for "Your Selenium Alternative for 2021". The status bar at the bottom of the screen shows system information like date, time, and battery level.

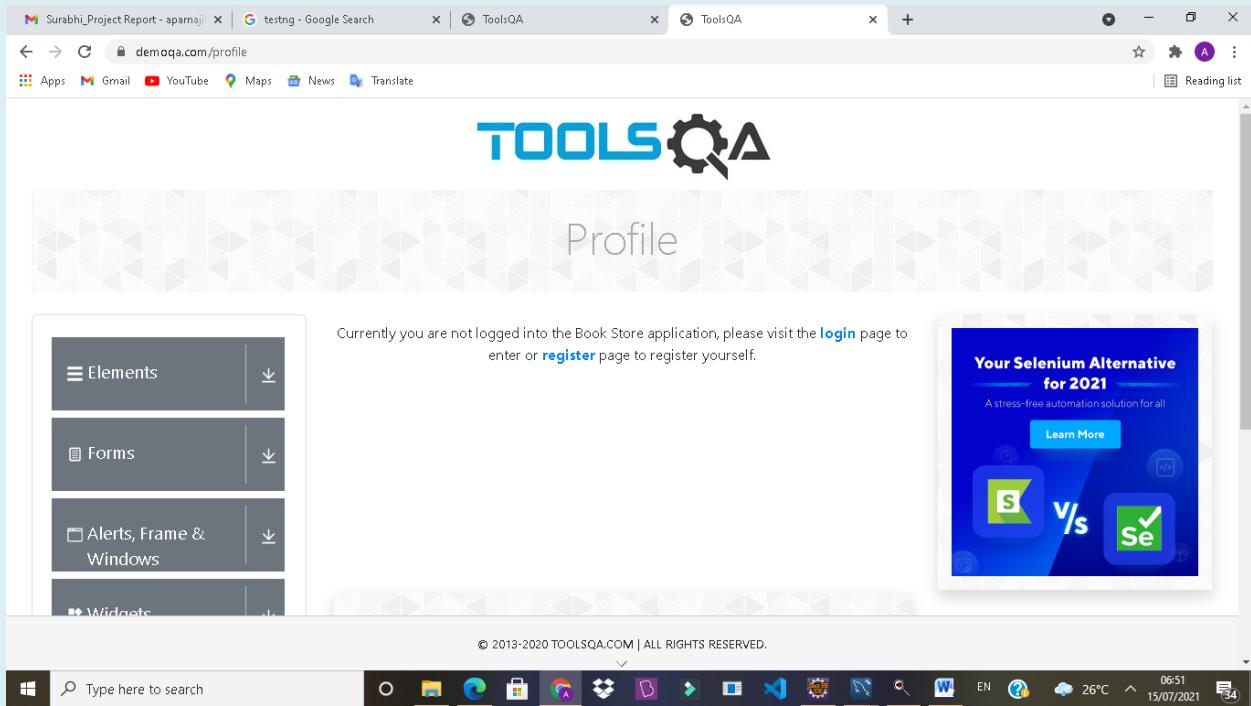
## OUTPUT/DEFECT:

Back To Book Store button is not working in Learning JavaScript Design which is in Book Store section

TC197- To verify the functionality of submodule (Profile section) of Book store Application module

## TestNG Test Script

```
@Test(dependsOnMethods = {"bookpage"})
public void profilepage() throws
InterruptedException {
    W.navigate().to("https://demoqa.com/profile");
```



TC198- To verify the functionality of login icon in Profile section  
TestNG Test Script

```
Thread.sleep(2000);
w.findElement(By.linkText("login")).click();

System.out.println("title " +w.getTitle());
System.out.println("URL " +w.getCurrentUrl());

Thread.sleep(2000) ;
```

**OUTPUT :-** Title is ToolsQA  
URL is <https://demoqa.com/login>

TC199- To verify the functionality of Register icon in Profile section  
TestNG Test Script

```
W.navigate().to("https://demoqa.com/profile");
Thread.sleep(2000);
W.findElement(By.linkText("register")).click();

System.out.println("title " +W.getTitle());
System.out.println("URL " +W.getCurrentUrl());
}
```

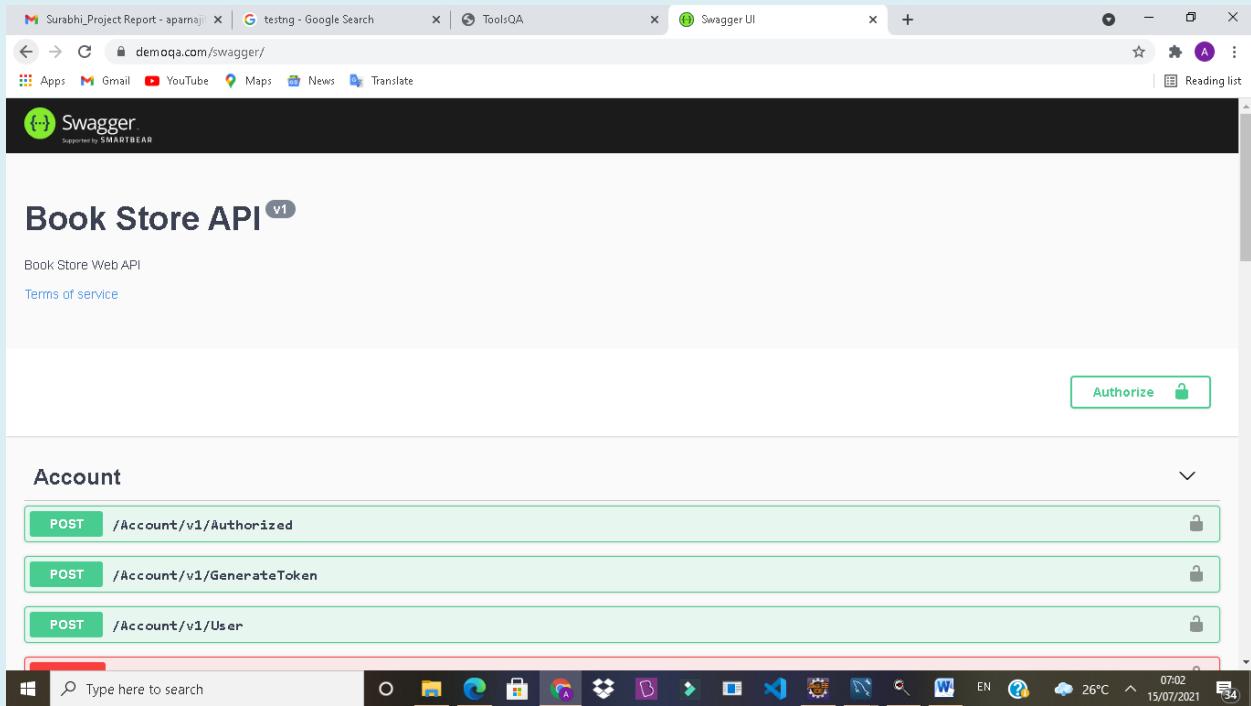
**OUTPUT:-** Title is ToolsQA  
URL is <https://demoqa.com/register>

TC200- To verify the functionality of submodule (Book Store API section) of Book store Application module

TestNG Test Script

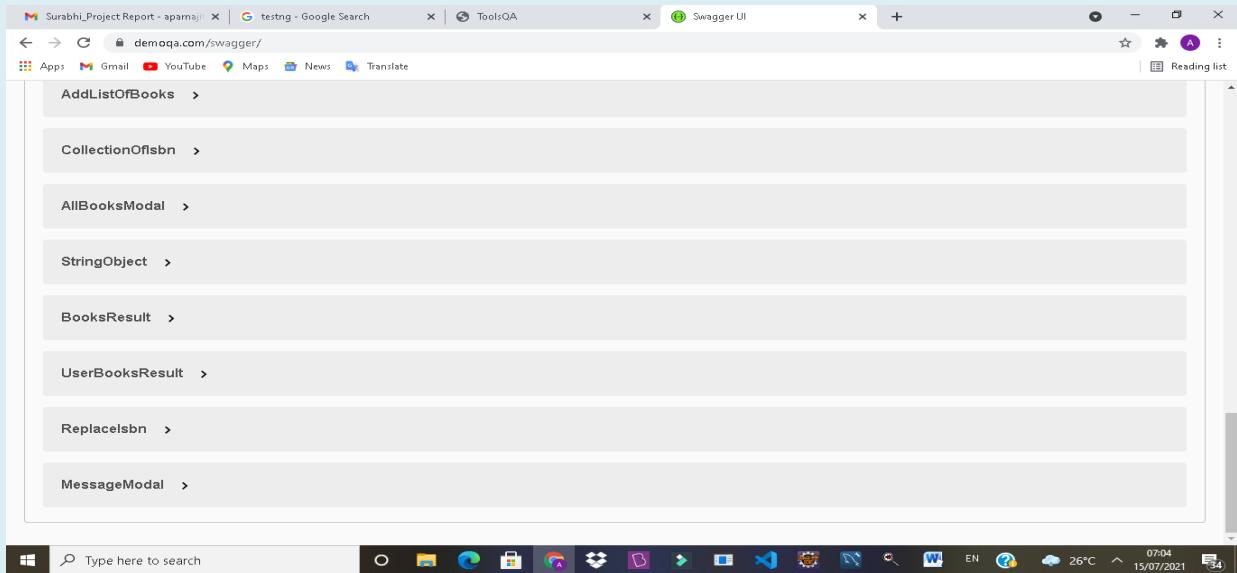
```
@Test(dependsOnMethods = {"profilepage"})
public void bookapi() throws
InterruptedException {

W.navigate().to("https://demoqa.com/swagger/");
Thread.sleep(3000);
```



TC201- To verify the functionality of scrolling page in Book store API section  
TestNG Test Script

```
JavascriptExecutor js = (JavascriptExecutor) w;  
  
js.executeScript("window.scrollTo(0,document.body.scrollHeight)");  
  
System.out.println("page end");  
  
Thread.sleep(5000);  
}
```



**TC202- To verify the functionality of closing the current module**

## Testng Test Script

```
@AfterTest  
public void afterTest() {  
    W.close();  
}  
}
```

```
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Problems @ Javadoc Declaration Console Results of running class Booksore_projectpart2
<terminated> Booksore projectpart2 [TestNG] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe (15-Jul-2021, 3:46:35 am)
console.error: Region.json: Error fetching region (new Error('TIEOUT', 'resource://gre/modules/Region.json', ''))
console.error: Region.json: "Failed to fetch region" (new Error("TIMEOUT", "resource://gre/modules/Region.json", 419))
1626301015839 Marionette INFO Listening on port 1037
1626301017842 Marionette WARN TLS certificate errors will be ignored for this session
Jul 15, 2021 3:46:57 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Web browser launch
page end
Current URL is https://demoqa.com/books
Title of page is ToolsQA
Javascript error: , line 0: uncaught exception: undefined
title ToolQA
URL https://demoqa.com/login
title ToolQA
URL https://demoqa.com/register
page end
1626301124821 Marionette INFO Stopped listening on port 1037

#### [Child][RunMessage] Error: Channel closing: too late to send/recv, messages will be lost

PASSED: profilepage
PASSED: registerpage
PASSED: bookapipage
PASSED: bookpage
PASSED: mainpage

=====
Default test
Tests run: 5, Failures: 0, Skips: 0
=====

=====
Default suite
Total tests run: 5, Passes: 5, Failures: 0, Skips: 0
=====
```

## DEFECT raised in Jira tool

## Module1:

D01: Enable to click on “No” radio Button

The screenshot shows a Jira Software interface for an issue titled "FBJ-460". The left sidebar lists "Projects / First\_batch\_Jira / FBJ-460". The main content area displays the following details:

- Description:**
  - 1) Launch the Browser
  - 2) Open URL : <https://demoqa.com/Elements>
  - 3) Click on Radio Button
  - 4) Click on "NO" Radio Button

Expected Result : "NO" Radio Button should be click  
Actual Result : "NO" Radio Button is not Clickable
- Environment:** Windows 7
- Activity:** A comment box with placeholder "Add a comment..." and a note: "Pro tip: press M to comment".

The right side panel contains pinned fields (Reporter: testereEDU), details (Assignee: Testing team leader, Labels: None, Due date: Jul 18, 2021, Priority: Medium), and more fields (Original estimate, Time tracking, Epic Link, Component...).

## D02:Enable to click on “Click” Button

SM02 - In ToolsQA website Buttons Module - Enable to click on Click Button.

Description

- 1) Launch the Browser
- 2) Open URL : <https://demoqa.com/Elements>
- 3) Click on Buttons
- 4) Click on Click Me

Expected Result : Click Me Button should be click  
Actual Result : Click me Button is not clickable

Environment

Windows 7

Activity

Add a comment...

Created 22 seconds ago  
Updated 22 seconds ago

## Module 2:

### Defect 01 : To Raise defect for Subject in Jira Tool

SK101\_5334 : When selecting a subject the subject is not selected properly

Description

Steps:-

- 1.Launch Browser.
- 2.Open URL : <https://demoqa.com/>

### Defect 02: To Raise defect for State and City in Jira Tool

Projects / First\_batch\_Jira / FBI-440

**SK102\_5334 : When we selecting state and city at that time State and City is not selected properly**

Description  
Steps:-  
1.Launch Browser.  
2.Open URL : <https://demoqa.com/>  
3.Scroll down upto Forms option and then click on that  
4.After that Click on Practice form  
5.In that form when we select the State as "Uttar Pradesh" and City name "Agra" then it does not show in that text field  
6.Click on sumit Button

Environment  
Window 10  
Chrome Version: 91.4.4472.124

Reporter: testereEDU  
Assignee: testereEDU  
Labels: None  
Due date: Jul 11, 2021  
Priority: High

Created 14 seconds ago Updated 14 seconds ago

## Module 3:

**Defect 01:-In Alert option wait alert ok Button is Not clickable by Automatic way but its work manually Properly**

Projects / First\_batch\_Jira / FBI-462

**SK103\_5334:-In ToolsQA website ,Alert option wait alert ok Button is Not clickable by Automatic way but its work manually Properly**

Description  
Steps:  
1.Launch Brower  
2.OpenURL : <https://demoqa.com/>  
3.Scroll down upto Alert, window and frames option and then click on Alert option new window is opened  
4.After that click on Second button i.e 'On' button click, alert will appear after 5 seconds"  
5.It shows the message This alert appeared after 5 seconds  
6.when click on ok button after 5 sec its clickable manually but not atomic way

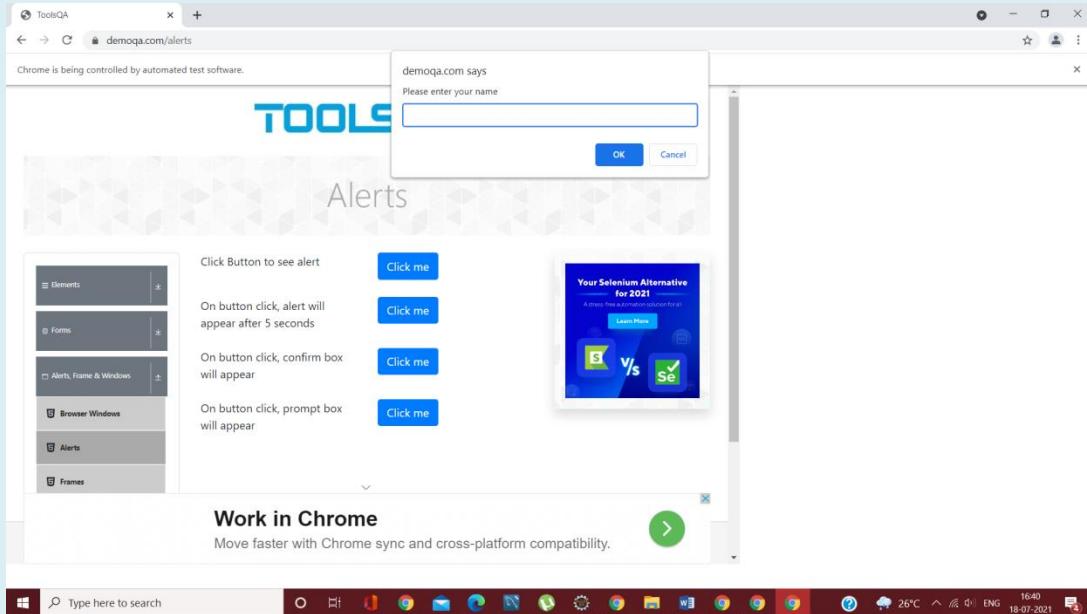
Environment  
Windows 10  
Chrome Version: 91.4.4472.124

Attachments (1)

Reporter: testereEDU  
Assignee: testereEDU  
Labels: None  
Due date: Jul 18, 2021  
Priority: Low

Created 36 seconds ago Updated 36 seconds ago

## Defect 02:-In Alert option prompt alert not showing text while running the program



The screenshot shows a Jira Software issue page for defect tracking. The issue key is FB14-466, titled "SK104\_5334:-In ToolsQA website ,In Alert option prompt alert not showing text while running the program". The issue details section contains the following text:

1.Launch Browser  
2.OpenURL: <https://demoqa.com/>  
3.Scroll up to Alert, window and frames option and then click on Alert option new window is opened  
4.After that click on "On button click, prompt box will appear" click me button  
5.It shows the popup box in that Please enter your name textbox  
6.Enter name as "Sandhya"  
7.In that text box manually "Sandhya" name is visible but when I using script for that textbox name is not visible

The issue is assigned to "testereEDU" and has a priority of "High". The environment is listed as "Windows 10" and "Chrome Version 91.0.4472.124". The attachments section shows one attachment named "Test Execution". The status of the issue is "Open". The Jira sidebar on the left shows the project navigation and settings. The browser status bar at the bottom shows the date and time as 17:06 18-07-2021.

## Module 4:

### D01: Text field is not working

The screenshot shows the Jira Software interface for issue GC02. The main content area contains the following information:

- Description:**
  - 1) Launch the Browser
  - 2) Open URL : <https://demoqa.com/widgets>
  - 3) Click on Auto Complete
  - 4) Enter the text in that text field

Expected Result : Enter the text in Auto Complete  
Actual Result : Doesn't enter the text in Auto Complete
- Environment:** Windows 10
- Activity:** A text input field with placeholder "Add a comment..." and a note "Pro tip: press M to comment".

The right sidebar displays pinned fields:

Reporter	testereEDU
Assignee	Testing team leader
Labels	None
Due date	Jul 18, 2021
Priority	Medium

Buttons at the bottom right of the sidebar include "Configure" and "More fields".

### D02: More tab button is not working

The screenshot shows the Jira Software interface for issue GC03. The main content area contains the following information:

- Description:**
  - 1) Launch the Browser
  - 2) Open URL : <https://demoqa.com/widgets>
  - 3) Click on Tabs
  - 4) Click on More Tab

Expected Result : More Tab should be click & Tab with data should be visible  
Actual Result : More Tab is not clickable& previous tab data is visible
- Environment:** Windows 10
- Activity:** A text input field with placeholder "Add a comment..." and a note "Pro tip: press M to comment".

The right sidebar displays pinned fields:

Reporter	testereEDU
Assignee	Testing team leader
Labels	None
Due date	Jul 18, 2021
Priority	Medium

Buttons at the bottom right of the sidebar include "Configure" and "More fields".

## Module 5:

### D01: Unable to click

AT0005: Interaction submodule unable to click by scripting method

Description

1. Launch the browser
2. Open URL

Expected result: Interaction submodule should able to click by scripting method  
Actual result: Interaction submodule unable to click by scripting method

Environment

Activity

Reported 6 seconds ago - Updated 6 seconds ago

### D02: Unable to select 7,8, and 9

AT0006: Unable to select 7,8,9 by scripting method in grid selection

Description

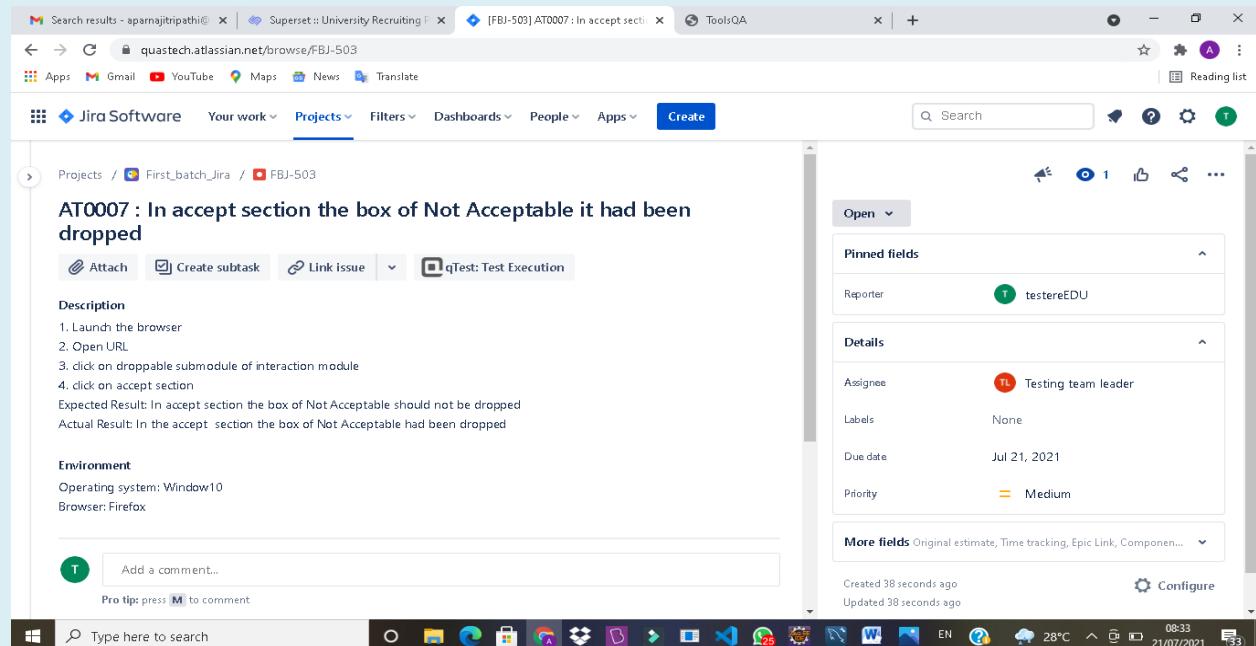
1. Launch browser
2. Open URL
3. Click on selectable submodule in interaction module
4. Go to subsection grid in selectable.

Expected Result: It should be able to select 7,8,9 by scripting method  
Actual Result: Unable to select 7,8,9 by scripting method

Environment

Reported 6 seconds ago - Updated 6 seconds ago

## D03: Not Acceptable box is dropping



AT0007 : In accept section the box of Not Acceptable it had been dropped

Description

1. Launch the browser
2. Open URL
3. click on droppable submodule of interaction module
4. click on accept section

Expected Result: In accept section the box of Not Acceptable should not be dropped  
Actual Result: In the accept section the box of Not Acceptable had been dropped

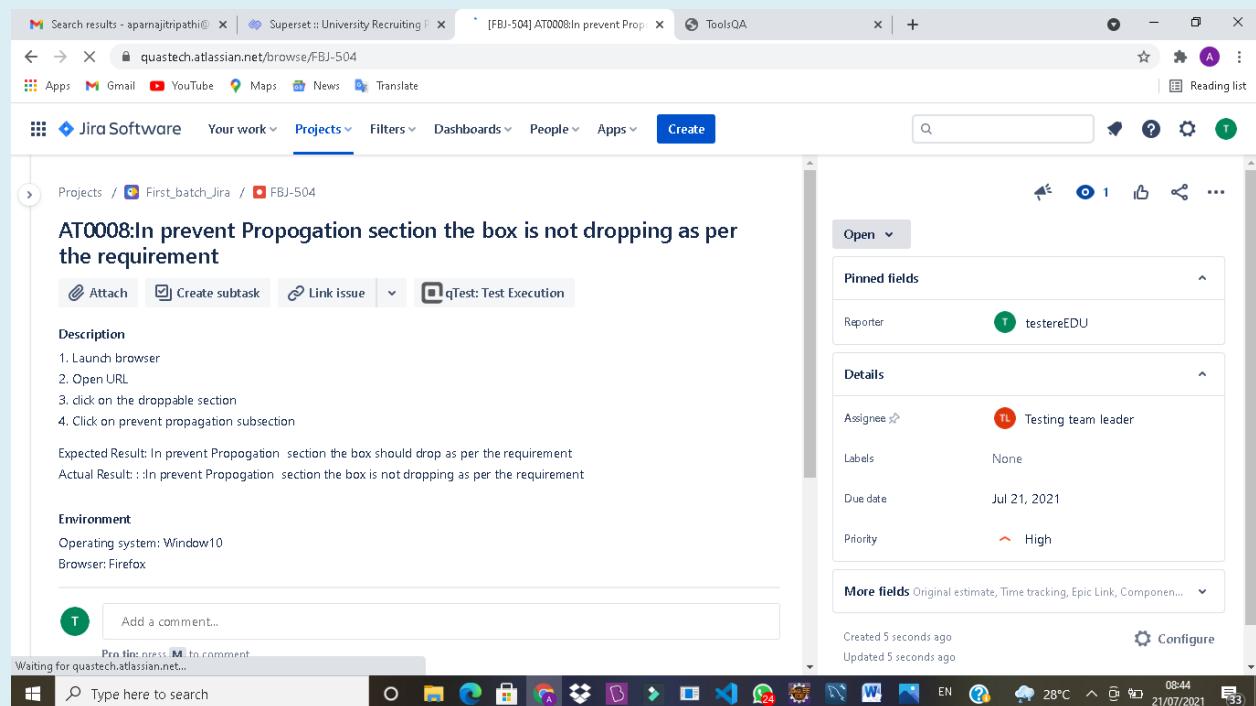
Environment

Operating system: Window10  
Browser: Firefox

Add a comment...  
Pro tip: press M to comment

Created 38 seconds ago  
Updated 38 seconds ago

## D04: Box is not dropping as per the requirement



AT0008:In prevent Propogation section the box is not dropping as per the requirement

Description

1. Launch browser
2. Open URL
3. click on the droppable section
4. Click on prevent propagation subsection

Expected Result: In prevent Propogation section the box should drop as per the requirement  
Actual Result: In prevent Propogation section the box is not dropping as per the requirement

Environment

Operating system: Window10  
Browser: Firefox

Add a comment...  
Pro tip: press M to comment

Waiting for quastech.atlassian.net...

Created 5 seconds ago  
Updated 5 seconds ago

## Module 6:

### D01: Arrow functionality is not working

In Book store Application module arrow functionality is Not working by scripting way and also without clicking an arrow, it automatically open the subsection of Book Store Application as well as it display the book store subsection.

Description • Unsaved changes

1. Launch the browser
2. Open URL

Expected result- arrow functionality should work in book store application and also after clicking on subsection of book store application it should open

Actual result- In Book store Application module arrow functionality is Not working by scripting way and also without clicking an arrow, it automatically open the subsection of Book Store Application as well as it display the book store subsection.

Add a comment...  
Pro tip: press M to comment

Project.java Project.class Project Report\_Sur....pdf

### D02: Form don't have validation

AT0001 ToolQA website, Registration form don't have validation

Description

- 1Launch the browser
- 2Open URL
- 3 Navigate to the login section
- 4 click on the New user button
- 5 enter first name- 12
- enter Lastname-e-9
- username- 9
- password- 9

Expected result- The registration form should have validation in every section

Actual result- Registration form don't have validation for first name and last name it accepting special

Add a comment...  
Pro tip: press M to comment

Project.java Project.class Project Report\_Sur....pdf

## D03: Back To Book Store button is not working

