Practical No. 5. HANDLING MULTIPLE WINDOWS & FRAMES USING SELENIUM

Date: _____

Aim:

To learn how to handle multiple windows, frames and modals using selenium.

Theory:

How to handle Selenium multiple window using Webdriver

In automation, when we have multiple windows in any web application, the activity may need to switch control among several windows from one to other in order to complete the operation.

After completion of the operation, it has to return to the main window i.e. parent window in Selenium.

WindowHandle:

It is a unique identifier that holds the address of all the windows. Think of it as a pointer to a window, which returns the string value. It is assumed that each browser will have a unique window handle. This window handle function helps to retrieve the handles of all windows.

In Selenium web driver there are methods through which we can handle multiple windows.

Driver.getWindowHandles();

To handle all opened windows by web driver, we can use "Driver.getWindowHandles()". This method helps to get the handles of all the windows opened and then we can switch window from one window to another in a web application. Its return type is Iterator<String>.

Driver.getWindowHandle();

When the site opens, we need to handle the main window by driver.getWindowHandle(). This method helps to get the window handle of the current window. Its return type is String.

set:

This method helps to set the window handles in the form of a string.

set<string> set= driver.get.windowhandles()

driver.switchTo()

This method helps to switch between the windows **action:**

This method helps to perform certain actions on the windows

How to handle frames in Selenium Webdriver

Frames in HTML can be used to divide a web-page vertically or horizontally. iFrames is mainly used for displaying external content on a target web page, for example, an advertisement for any online programming course on a web page. An iframe is also known as the inline frame. It is a tag used in HTML5 to embed an HTML document within a parent HTML document. An iframe tag is defined using <iframe></iframe> tags.

It is possible to identify the iframes on a web page in two ways:

- Right-click on the specific element and check all the options. If you find an option like This Frame, view Frame source or Reload Frame, it means the page includes frames.
- Similar to the first step, right-click on the page and click on View Page Source. On the page source, search for "iframe-tags". If you find any iframe tags, it means the page includes iframes.

To interact with any web element present within any frame, one needs to switch to that particular frame. This allows the user to identify elements present on that page and write tests accordingly.

QAs can switch between frames using the **Switch.frame()** function. The switch function can be implemented using three different locators: By.index, By.id, By WebElement. Refer to the commands below:

By Index

```
driver.switchTo().frame(1);
```

Switches to the frame with index number 1

By Id or Name

```
driver.switchTo().frame("resultframe");
```

Switches the frame where the value of id attribute is resultframe

By Web Element

```
WebElement iframeElement = driver.findElement(By.id("resultframe"));
driver.switchTo().frame(iframeElement);
```

The WebElement command above identifies the web element and then passes it through the iframe element object.

How to handle modals in Selenium WebDriver?

A Modal Dialog Box (also referred to as Bootstrap Modal Window) is built in Bootstrap Framework, due to which it gets displayed on top of your current page. Due to this, modal boxes need to be interacted with first before moving to the current webpage. Switching is not necessary for modals.

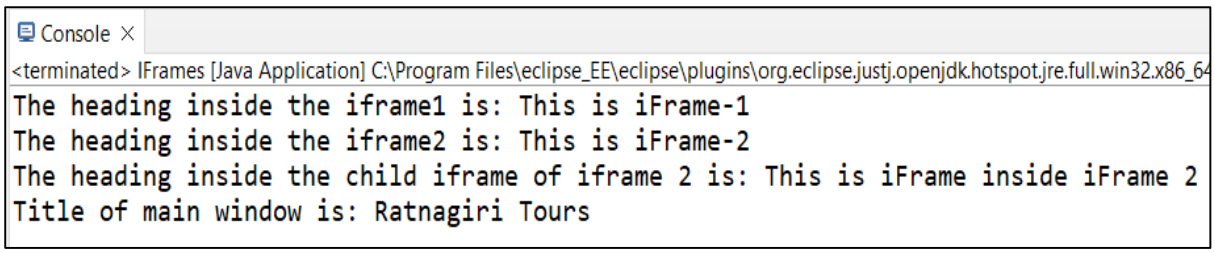
Implementation

1. **Open "flight_reservation.html" and write a script to handle frames and print heading of page in each frame.**
   Program:

```
package seleniumScriptsA1;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.firefox.FirefoxDriver;
public class IFrames {
	public static void main(String[] args) {
		//setting path
			System.setProperty("webdriver.gecko.driver","C:\\Selenium_Drivers\\geckodriv
		er.exe");
		//creating instance of web driver
		WebDriver driver=new FirefoxDriver();
		//navigate to web page
	driver.get("file:///C:/Users/parth/OneDrive/Desktop/SQA_HTML%20FILES/flight_reservation
.html");
		//switch to frame1 using Index
	driver.switchTo().frame(0);    //switch focus from main window to 1
		//locate and print heading in iframe1
	WebElement heading_frame1 = driver.findElement(By.id("heading1"));
		System.out.println("The heading inside the iframe1 is: "+heading_frame1.getText());
		driver.switchTo().defaultContent();		//switch to main menu
		//swtiching to iframe2 using id
		driver.switchTo().frame("IF2");
		//locate and print the heading in iframe2
	WebElement heading_frame2 = driver.findElement(By.id("heading2"));
		System.out.println("The heading inside the iframe2 is: "+heading_frame2.getText());
		driver.switchTo().frame(0);
		//locate and print heading inside the child iframe of iframe2
		WebElement heading_childframe1 = driver.findElement(By.id("heading3"));
		System.out.println("The heading inside the child iframe of iframe 2 is:
"+heading_childframe1.getText());
```

```
                driver.switchTo().defaultContent();
                System.out.println("Title of main window is: "+driver.getTitle());
        }       }
```

Output:

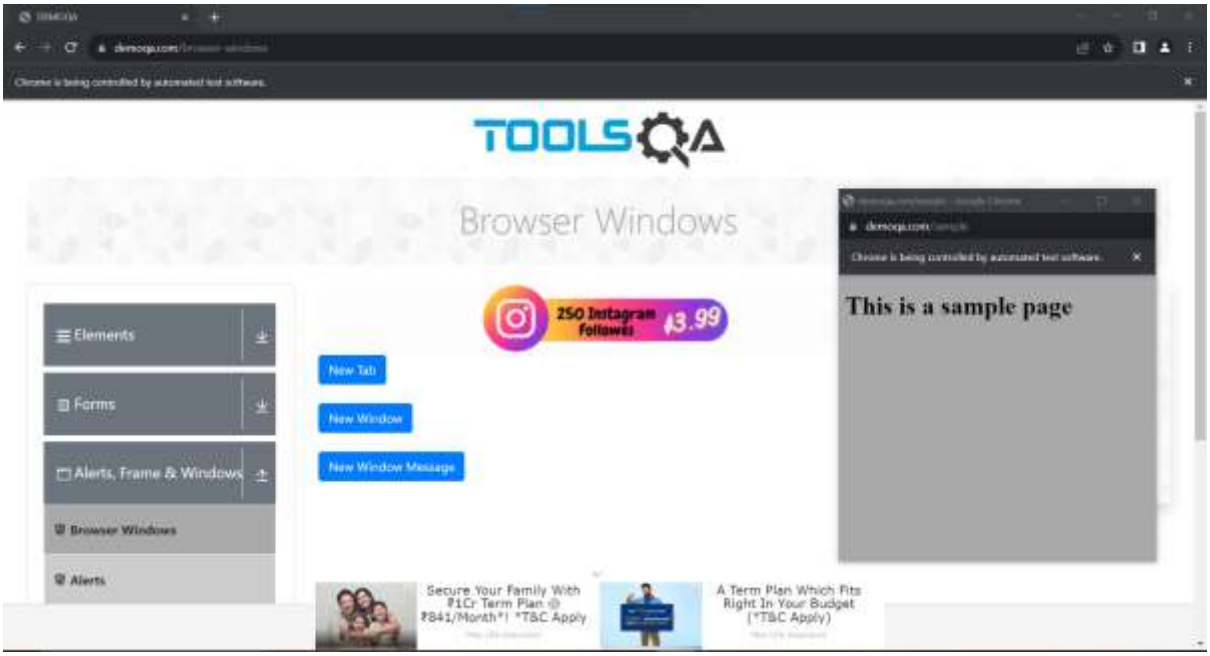


2. Open https://demoqa.com/browser-windows and write a script to handle multiple windows that opens after clicking button "New Window" and print the heading on that new window.

Program:

```
package webDriver;
import java.util.Set;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
public class Question_02 {
        public static void main(String[] args) throws InterruptedException {
                System.setProperty("webdriver.chrome.driver",
"E:\\selenium_setup\\chromedriver.exe");
                WebDriver driver = new ChromeDriver();
                driver.get("https://demoqa.com/browser-windows");
                String mainWindow = driver.getWindowHandle();
                WebElement newWindow = driver.findElement(By.id("windowButton"));
                newWindow.click();
                Set<String> handles = driver.getWindowHandles();
                for (String windowHandle : handles) {
                        if (!windowHandle.equals(mainWindow)) {
                                driver.switchTo().window(windowHandle);
                                WebElement heading = driver.findElement(By.id("sampleHeading"));
                                System.out.println(heading.getText());
                                }
                        }               }       }
```

Output:

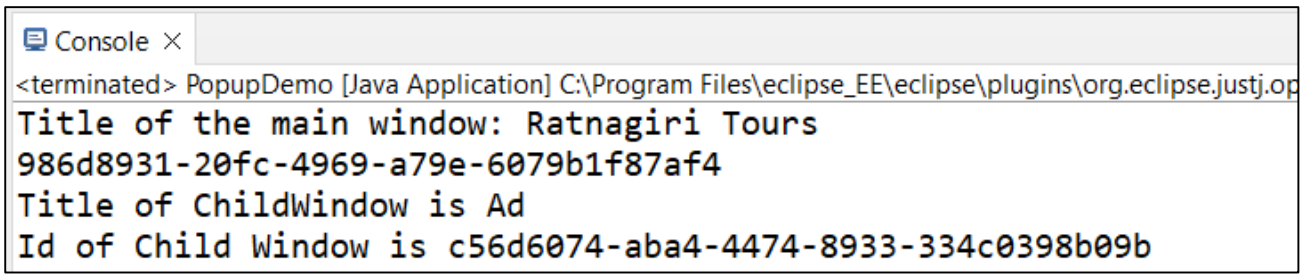


3. Write a selenium script to login to http://demo.guru99.com/test/login.html .
Use name and ID locator strategy to locate elements.
Program:

```
package seleniumScriptsA1;
import java.util.Set;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.chrome.ChromeOptions;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.firefox.FirefoxOptions;
public class PopupDemo {
	public static void main(String[] args) throws InterruptedException {
		System.setProperty("webdriver.gecko.driver",
"C:\\Selenium_Drivers\\geckodriver.exe");
		//setting the options for google chrome browser
		FirefoxOptions options = new FirefoxOptions();
		options.addArguments("--remote-allow-origins=*");
		//creating instance of web driver
		WebDriver driver=new FirefoxDriver();
		//navigate to web page
	driver.get("file:///C:/Users/parth/OneDrive/Desktop/SQA_HTML%20FILES/flight_reservation
.html");
```

```
System.out.println("Title of the main window: "+driver.getTitle());
//print windows ID or windows handle
String mainWindow = driver.getWindowHandle();          System.out.println(mainWindow);
//locate and click on the hotels link
WebElement hotel_link = driver.findElement(By.linkText("Hotels"));
hotel_link.click();
Set<String> handles = driver.getWindowHandles();
for (String windowHandle : handles)
{
        if(!windowHandle.equals(mainWindow))
        {
                driver.switchTo().window(windowHandle);
                System.out.println("Title of ChildWindow is "+driver.getTitle());
                System.out.println("Id of Child Window is "+driver.getWindowHandle());
//locate and print heading in the child window
WebElement heading = driver.findElement(By.id("sampleHeading"));
System.out.println("heading inside child window is "+heading.getText());
                        Thread.sleep(5000);
                        driver.close();                 }               }       }       }
```

Output:



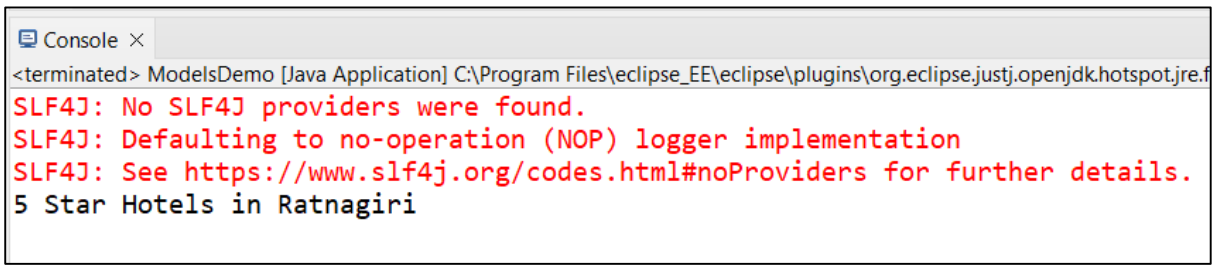
4. Open "hotel_search.html" page and handle modals that appear after clicking on buttons on that webpage.

Program:

```
package seleniumScriptsA1;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.firefox.FirefoxDriver;
public class ModelsDemo {
        public static void main(String[] args) throws InterruptedException {
                //setting path
                System.setProperty("webdriver.gecko.driver",
"C:\\Selenium_Drivers\\geckodriver.exe");
                //creating instance of web driver
```

```
        WebDriver driver=new FirefoxDriver();
        //navigate to web page
    driver.get("file:///C:/Users/parth/OneDrive/Desktop/SQA_HTML%20FILES/hotel_search.html
");
        WebElement hotels = driver.findElement(By.cssSelector("button[data-
target='#myModal']"));
        hotels.click();
        //locate modal header and print it
        WebElement modal_header_1 = driver.findElement(By.className("modal-title"));
        System.out.println(modal_header_1.getText());
        Thread.sleep(5000);
        WebElement btn_ok = driver.findElement(By.xpath("//button[text()='Ok']"));
        btn_ok.click();
        Thread.sleep(2000);
        WebElement room = driver.findElement(By.id("myBtn1"));
        room.click();
        //locate and type in text box a modal
        WebElement hotel_name= driver.findElement(By.id("htlname"));
        hotel_name.sendKeys("Saptagiri");
        Thread.sleep(3000);
        WebElement cancel_button=
driver.findElement(By.xpath("//button[@type='submit']//following::button"));
        cancel_button.click();          }       }
```

Output:



Conclusion: Learnt to handle multiple windows, frames and modals in Selenium.