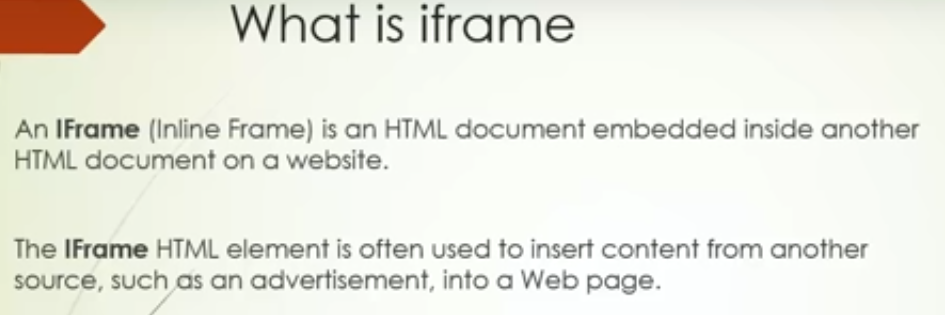
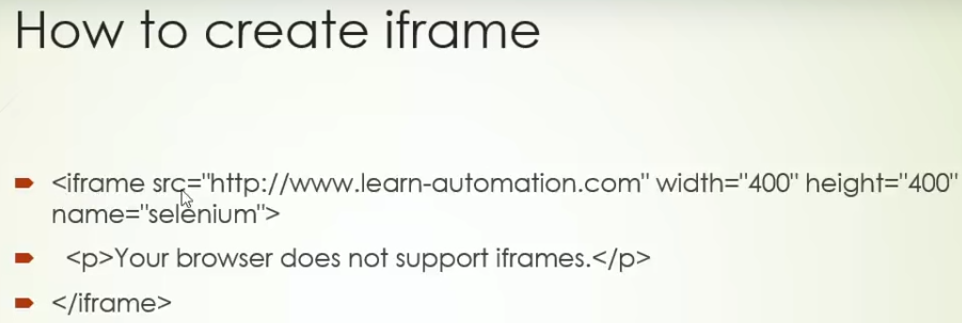
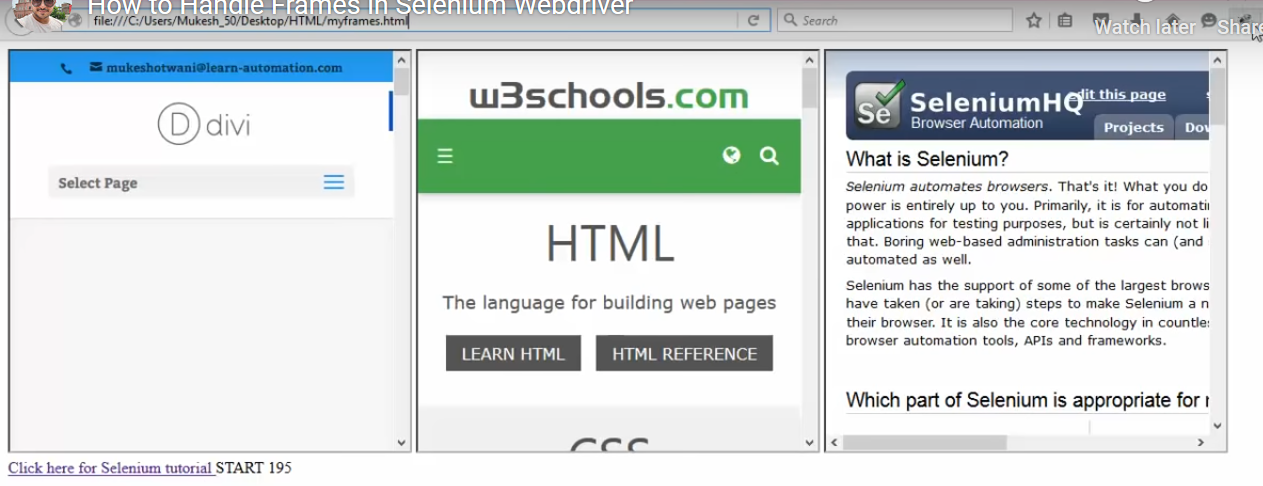
## Programs to handle frames in selenium

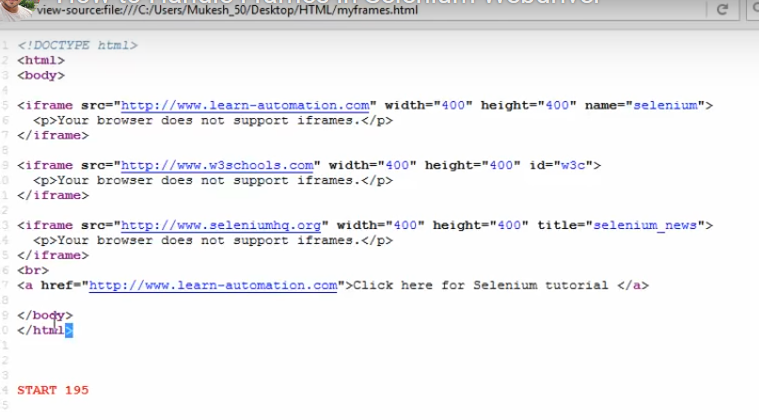


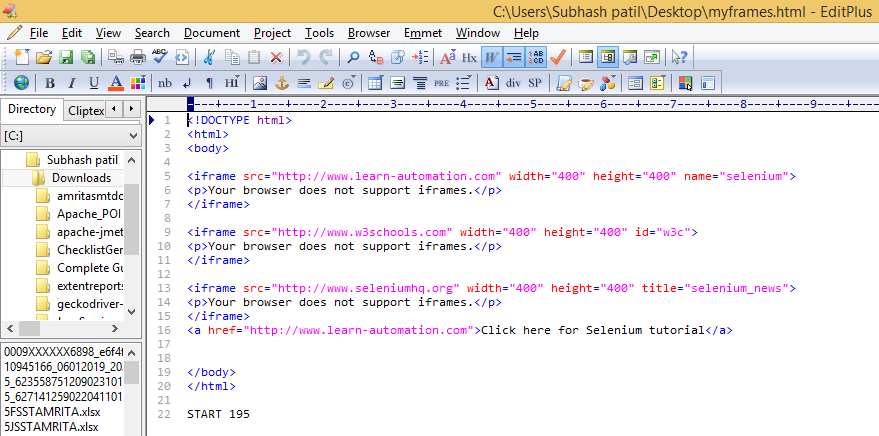


Using iFrame below 3 websites are embedded in one single site

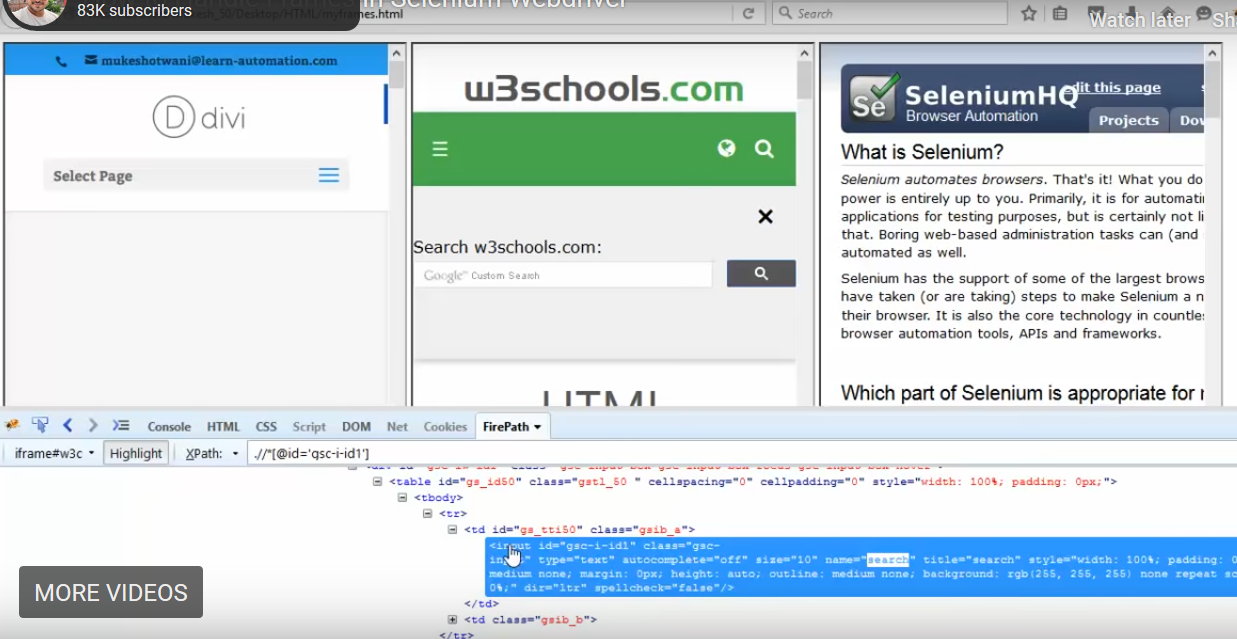


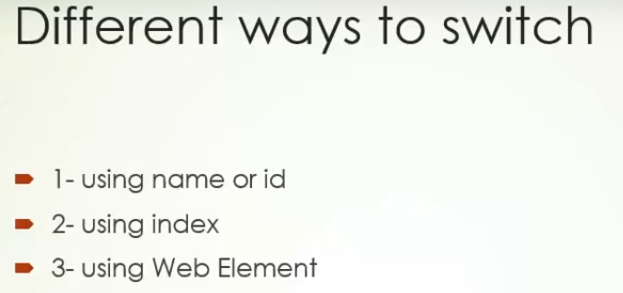
Right click and page source

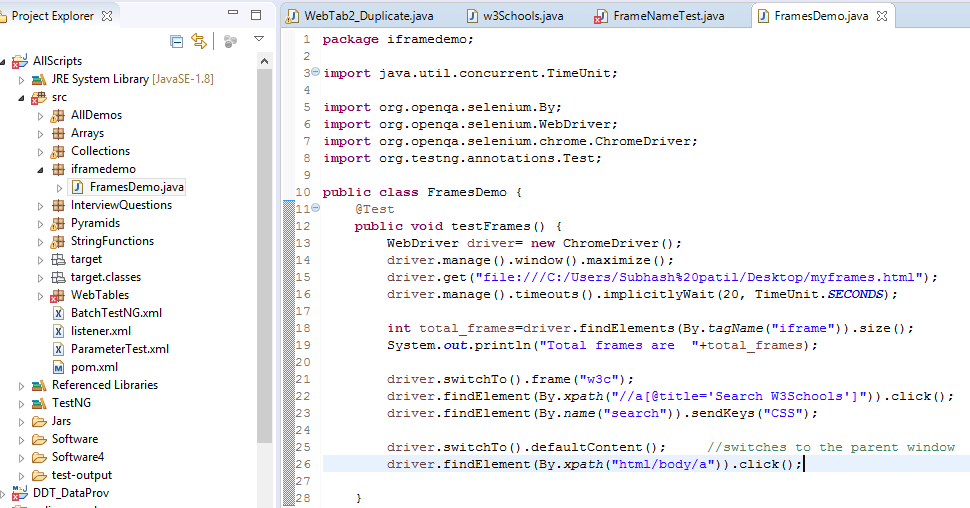


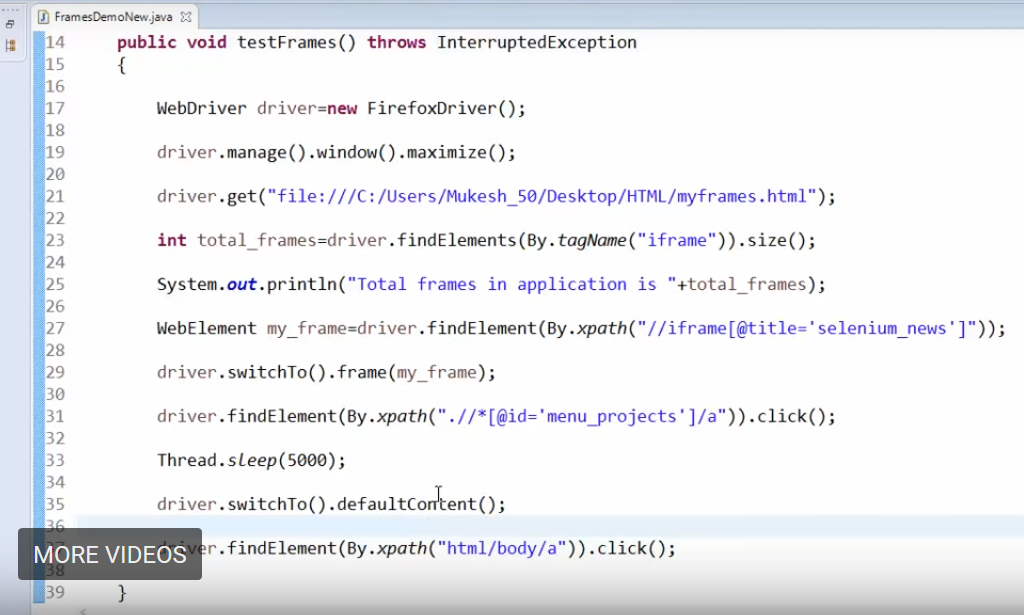


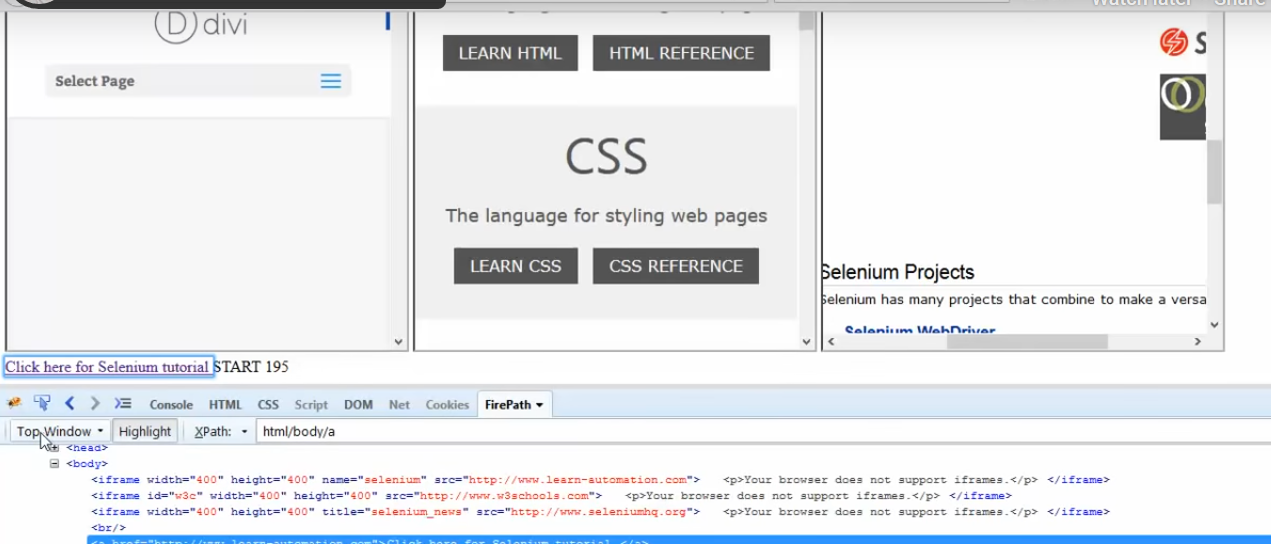
After spying we get below 3 frames





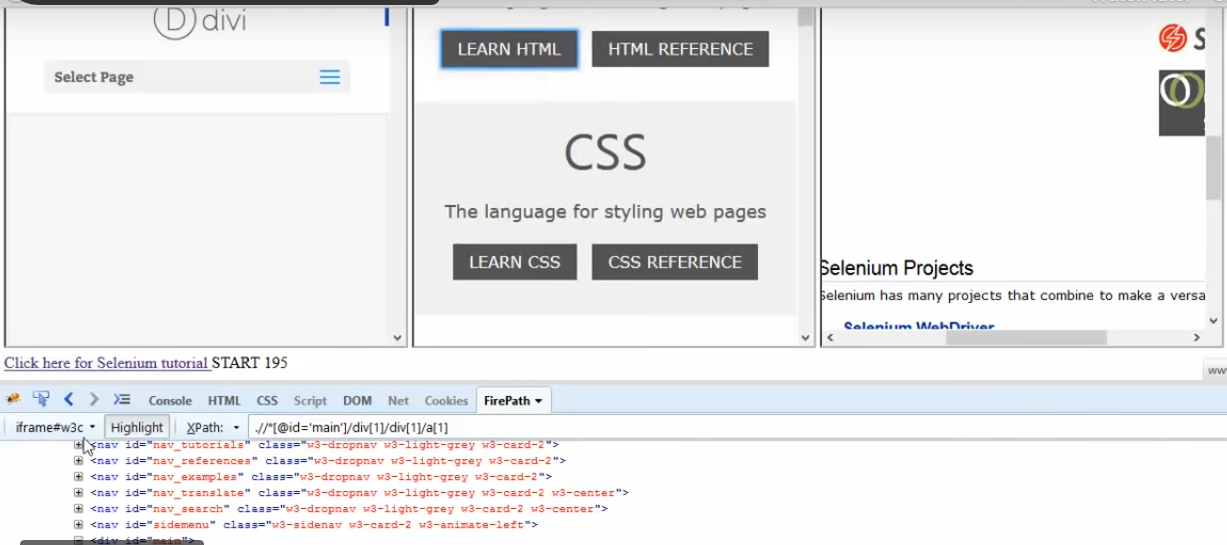




NOTE: When u inspect the text on main window , it will be displayed under Top Window tab as seen below.

Now when u inspect the LEARN HTML in the middle it will be shown under iframe#w3c

This feature is available only in Firefox



**http://mundrisoft.com/tech-bytes/code-to-handle-alert-window-and-iframe-in-selenium/**

**Code to Handle alert, window and iframe in Selenium**

This post was most recently updated on June 14th, 2019

SwitchTo() Functions are used to handle alert, browser window and iframe.

**driver.switchTo()** functions are used to handle alert window, new tab or new window in browser and iframe within html document.

Refer below table for switchTo functions and their uses:

|  |  |
| --- | --- |
| **switchTo functions** | **Uses** |
| driver.switchTo().frame(index); | Switch focus to iframe element using index of iframe in html document. |
| driver.switchTo().frame(“iframe\_name”); | Switch focus to iframe element using name of iframe |
| driver.switchTo().frame(webElement); | Switch focus to iframe element using name of WebElement object. |
| driver.switchTo().parentFrame(); | Switch focus to parent iframe element. |
| driver.switchTo().defaultContent(); | Switch focus to main html document. |
| driver.switchTo().window(windowHandle); | Switch focus to new window using window handle string |
| driver.switchTo().alert(); | Switch focus to alert window or message box. |
| driver.switchTo().activeElement(); | Switch to currently focused web element within html document |

**ActiveElement**

Using this we can perform actions on currently focused element.

*driver.switchTo().activeElement() → Used to find and perform action on current focused element.*

Code:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33 | import org.openqa.selenium.By;  import org.openqa.selenium.Keys;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.WebElement;  import org.openqa.selenium.chrome.ChromeDriver;  import org.openqa.selenium.chrome.ChromeOptions;    public class SeleniumMain {  public static void main(String[] args) throws InterruptedException {  System.setProperty("webdriver.chrome.driver", "D:/chromedriver.exe");  WebDriver driver;  ChromeOptions cOptions=new ChromeOptions();  cOptions.addArguments("--start-maximized");  driver=new ChromeDriver();  //Navigate to www.google.com  driver.get("https://www.google.com");  // use driver.switchTo().activeElement() to switch focus on search input box.  driver.switchTo().activeElement().sendKeys("Search new text");  //Perform TAB press to change focus. Now it will be on Voice search.  driver.switchTo().activeElement().sendKeys(Keys.TAB);  //Perform TAB press to change focus. Now it will be on google search button.  driver.switchTo().activeElement().sendKeys(Keys.TAB);  //Perform click on google search button.  driver.switchTo().activeElement().click();  //Find web element which contains search result count  WebElement searchResults = driver.findElement(By.id("resultStats"));  //Print search result count  System.out.println(searchResults.getText());  //Close chrome browser window  driver.quit();  }  } |

**Alert Window**

*driver.switchTo().alert() → Used to change focus from current active window to alert message box.*

Code:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48 | import org.openqa.selenium.Alert;  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;    public class SeleniumMain {  public static void main(String[] args) throws InterruptedException {  System.setProperty("webdriver.chrome.driver", "D:/chromedriver.exe");  WebDriver driver;  ChromeOptions cOptions=new ChromeOptions();  cOptions.addArguments("--start-maximized");  driver=new ChromeDriver();  //Navigate to https://www.seleniumeasy.com/test/  driver.get("https://www.seleniumeasy.com/test/");  //find Alert & Model Link  WebElement alertAndModelLinks=driver.findElement(By.xpath("//ul[@id='treemenu']//li[a[.='Alerts & Modals']]/a"));  alertAndModelLinks.click(); //click Alert & Model Link  //find Javascript Link  WebElement javascriptLinks=driver.findElement(By.xpath("//ul[@id='treemenu']//li[a[.='Javascript Alerts']]/a"));  javascriptLinks.click();//click javascript Link  //find button which opens alert box  WebElement simpleAlertBox=driver.findElement(By.xpath("//div[@class='panel-body']/button[@onclick='myAlertFunction()']"));  simpleAlertBox.click(); //click on button  //use driver.switchTo().alert();  Alert alert=driver.switchTo().alert();  //use below four functions based alert  alert.accept(); // -> accept(i.e click on button which is used for confirmation like OK)  //alert.dismiss(); -> to dismiss  //alert.getText(); -> read text from alert message  //alert.sendKeys("What is your Name?"); -> Pass data to alert message  //perform get text for other elment after accepting alert message  WebElement alertType=driver.findElement(By.tagName("h3"));  //Print text from web page  System.out.println(alertType.getText());  //Close chrome browser window  driver.quit();  }  } |

**Browser window**

*driver.switchTo().window(windowHandle) → Used to change focus from current active window to new opened window.*

Code:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37 | 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;    public class SeleniumMain {  public static void main(String[] args) throws InterruptedException {  System.setProperty("webdriver.chrome.driver", "D:/chromedriver.exe");  WebDriver driver;  ChromeOptions cOptions=new ChromeOptions();  cOptions.addArguments("--start-maximized");  driver=new ChromeDriver();  //Navigate to http://demo.automationtesting.in/Windows.html  driver.get("http://demo.automationtesting.in/Windows.html");  String mainWH=driver.getWindowHandle(); //Store current window handle into a mainWH variable  //find button  WebElement clickBtn=driver.findElement(By.xpath("//a/button[contains(text(),'click')]"));  clickBtn.click(); //click on button to open new window  Set<String> windowHandles=driver.getWindowHandles();  for (String windowH : windowHandles) {//iterate all window handles  if(!windowH.matches(mainWH)) { //if not matches then set that window handle as current window handle  driver.switchTo().window(windowH);//perform switch  }  }  System.out.println("After Switch: "+driver.getTitle());  driver.close();// close popup window  driver.switchTo().window(mainWH);//change foucus to main window again and continue further.  System.out.println("MAIN WINDOW Title After Switch To original: "+driver.getTitle());  //Close chrome browser window  driver.quit();  }  } |

**IFrame within HTML Document**

1. driver.switchTo().frame(index) → Used to change focus to iframe within current web page content using no of iframe  
   in page.
2. driver.switchTo().frame(name) → Used to change focus to iframe within current web page content using iframe name attribute value.
3. driver.switchTo().frame(webElementObject) → Used to change focus to iframe within current web page content using iframe webElement.
4. driver.switchTo().parentFrame() → Used to change focus to parent iframe i.e. to outer iframe.
5. driver.switchTo().defaultContent() → Used to change focus to normal web page.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56 | 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;    public class SeleniumMain {  public static void main(String[] args) throws InterruptedException {  System.setProperty("webdriver.chrome.driver", "D:/chromedriver.exe");  WebDriver driver;  ChromeOptions cOptions=new ChromeOptions();  cOptions.addArguments("--start-maximized");  driver=new ChromeDriver();  //Navigate to http://demo.automationtesting.in/Frames.html  driver.get("http://demo.automationtesting.in/Frames.html");  //find and click on button for iframe  //Main WebPage -->  Outer IFrame --> Inner IFrame --> Outer Frame(Again) --> Main WebPage  WebElement multiFrame=driver.findElement(By.xpath("//a[@href='#Multiple']"));  multiFrame.click();  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  //FIRST(Main WebPage -->  Outer IFrame) : -> Find outer iframe using index and perform operation on element within that iframe  //1. switch focus to iframe using index of iframe on webpage  driver.switchTo().frame(1);  //read any element within frame  WebElement divClass=driver.findElement(By.className("container"));  System.out.println(divClass.getAttribute("class"));  //2. switch focus to iframe using name  //driver.switchTo().frame("iframe\_name\_Attrib\_value"); not use in this script  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  //SECOND : -> Find inner iframe using WebElement and perform operation on element within that iframe  //3. switch focus to iframe using webelemnt  //used Web Element to find child iframe within iframe  WebElement iframeElement=driver.findElement(By.xpath("//iframe[@src='SingleFrame.html']"));  driver.switchTo().frame(iframeElement);  //Perform action on element in inner iframe  WebElement intput=driver.findElement(By.tagName("input"));  intput.sendKeys("Text Entered");  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  //THIRD(Inner IFrame --> Outer Frame(Again)) : -> Return to Outer FRAME using parentFrame()  driver.switchTo().parentFrame();  //getAttribute on child iframe could only after switch to parent iframe only  System.out.println(iframeElement.getAttribute("src"));  //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  //Fourth(Outer Frame(Again) --> Main WebPage) : -> switch to main web contain  driver.switchTo().defaultContent();  System.out.println("MAIN WINDOW Title After Switch To original: "+driver.getTitle());  //Close chrome browser window  driver.quit();  }  } |

### Switching back to Main Page

The main page is where all the iframes are embedded. After operating on a particular iframe,  use **switchTo().parentFrame** to move back to the page page**.** Use **switchTo().defaultContent** to shift to the primary/first parent frame. Refer to the sample code below:

WebDriver driver = new ChromeDriver();

driver.get("URL");// URL OF WEBPAGE HAVING FRAMES

//First finding the element using any of locator strategy

WebElement iframeElement = driver.findElement(By.id("iFrameResult"));

//now using the switch command to switch to main frame.

driver.switchTo().frame(0);

//Perform all the required tasks in the frame 0

//Switching back to the main window

driver.switchTo().defaultContent();

driver.quit();

The aforementioned methods will help QAs run automated test cases in [Selenium](https://www.browserstack.com/selenium) faster and with less effort. Simplify and speed up tests further by running them on real devices and browsers.