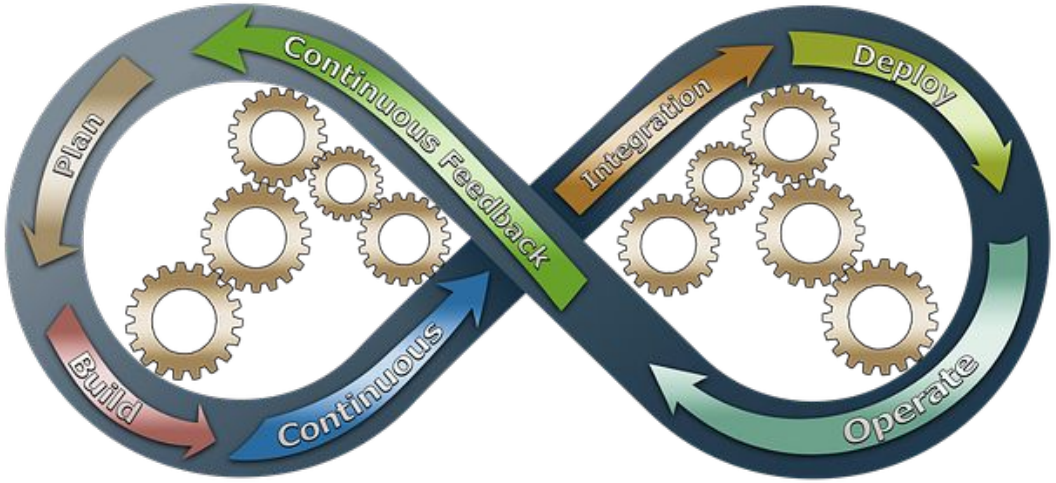


Session 4.6

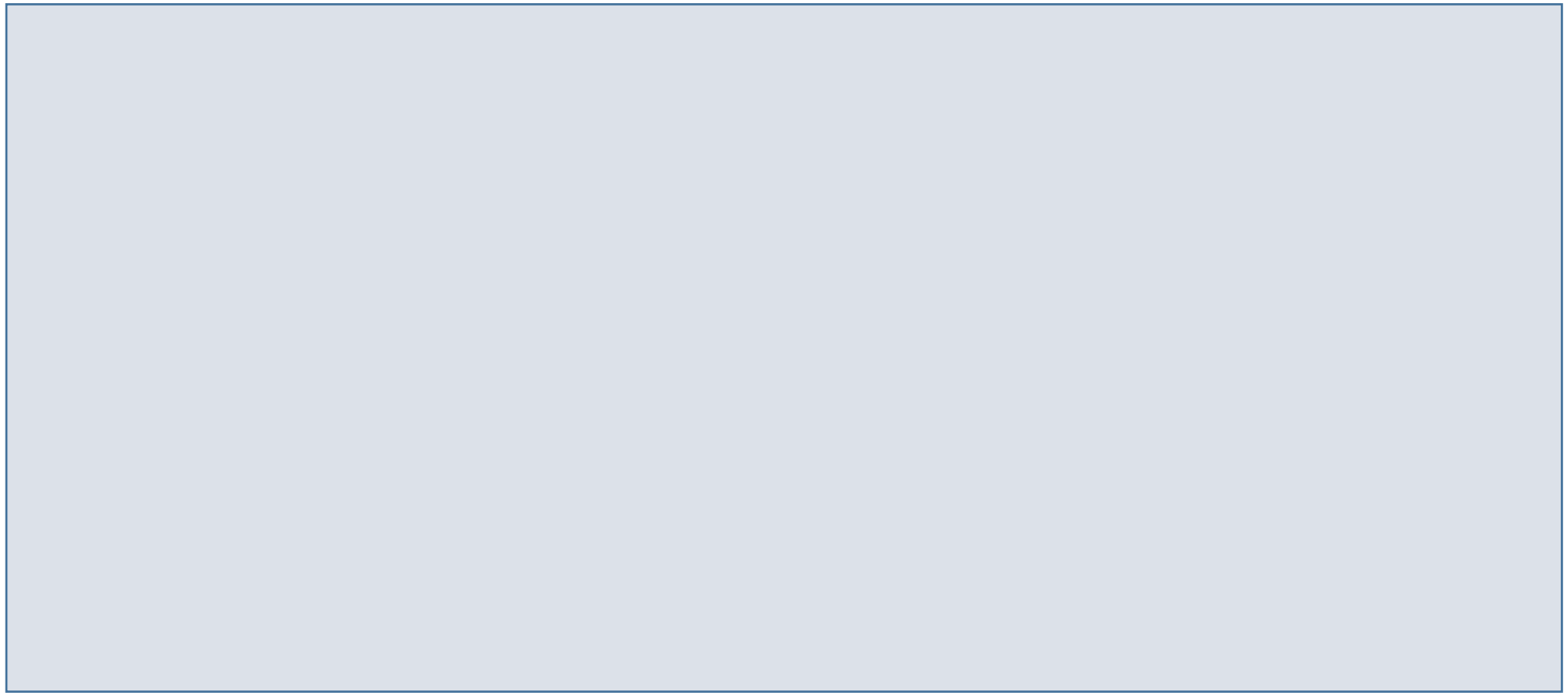
Alerts,Frames, Windows

AN INITIATIVE BY

UNICAL ACADEMY

Introduction

UNICAL ACADEMY



Let’s go!!!

Alerts, Frames, Windows

UNICAL ACADEMY

Alerts:

- The alerts are pop-ups that appear on clicking a link or a button. Immediately after it appears on the page, the focus is shifted from the current page to the alert.
- To work with an alert pop up we need to switch the driver focus from the parent page to the pop up generated. his is achieved with the help of the **switch_to.alert ()** method.
  -

Syntax:

```
a = driver.switcht_to.alert ();
```

- After switching the focus on the alert, we can accept, dismiss, or get the text displayed in the alert.
- For accepting the alert or clicking on the OK button, we have to use the method as a.accept ().
- Next to dismiss or to click on the X symbol, the alert method to be used is a.dismiss ().
- If we want to extract text on the pop-up, the method to be used is a.getText().
- Also, we can input text on the alert with the a.send_keys () method.

UNICAL ACADEMY

Code Implementation with Alert:

```
#   import the webdriver
from selenium import webdriver
# import the Keys class
from selenium.webdriver.common import keys
driver = webdriver.Chrome (executable_path="C:\\chromedriver.exe")
# get method to launch the URL
driver.get("http://183.82.4.93:5887/moss/javascript-alert-test-page.html")
# to identify the element
I = driver.find_element_by_xpath ("//*[@id = 'content']/button")
# to click the button
I.click()
# to switch to the alert
a = driver.switch_to.alert ();
# to get the alert text
print ( a.text)
# to accept the alert
a.accept ()
# again producing the alert
I = driver.find_element_by_xpath ("//*[@id = 'content']/button")
# to click the button
I.click ()
# to dismiss the alert
a.dismiss ()
# to close the browser
driver.close ()
```

Handling Windows

UNICAL ACADEMY

Handling Windows

A window in any browser is the main webpage on which the user is landed after hitting a link/URL. Such a window in Selenium is referred to as the parent window also known as the main window

Syntax:

- **get.windowhandle():** This method helps to get the window handle of the current window
- **get.windowhandles():** This method helps to get the handles of all the windows opened
- **set:** This method helps to set the window handles in the form of a string.

  set<string> set= driver.get.windowhandles()
- **switch to:** This method helps to switch between the windows
- **action:** This method helps to perform certain actions on the windows

Alert interface methods

UNICAL ACADEMY

Handling Windows

Selenium WebDriver provides three methods to accept and reject the Alert depending on the Alert types.

- **void dismiss()**
- This method is used to click on the 'Cancel' button of the alert.
  - **Syntax:**
  - driver.switchTo().alert().dismiss();
- **void accept()**
- This method is used to click on the 'Ok' button of the alert.
  - **Syntax:**
  - driver.switchTo().alert().accept();
- **String getText()**
- This method is used to capture the alert message.
  - **Syntax:**
  - driver.switchTo().alert().getText();
- **void sendKeys(String stringToSend)**
- This method is used to send some data to the alert box.
  - **Syntax:**
  - driver.switchTo().alert().sendKeys("Text");

Handling Frames /IFrames

UNICAL ACADEMY

iFrame in Selenium Webdriver is a web page or an inline frame which is embedded in another web page or an HTML document embedded inside another HTML document.

1. Switch to the frame by index:

- Index is one of the attributes for frame handling in Selenium through which we can switch to it.
- Index of the iframe starts with '0'.
- Suppose if there are 100 frames in page, we can switch to frame in Selenium by using index.

Syntax:

```
driver.switchTo().frame(0);
driver.switchTo().frame(1);
```

2. Switch to the frame by Name or ID:

- Name and ID are attributes for handling frames in Selenium through which we can switch to the iframe.

Syntax:

```
driver.switchTo().frame("iframe1");
driver.switchTo().frame("id of the element");
```

UNICAL ACADEMY

3. Switch to the frame by Name or ID:

Syntax:

```
driver.switchTo().frame(WebElement frameElement);
```

Code Implementation with frames:

```
#   import the webdriver
from selenium import webdriver
# import the Keys class
from selenium.webdriver.common import keys
driver = webdriver.Chrome (executable_path="C:\\chromedriver.exe")
# get method to launch the URL
driver.get ("http://183.82.4.93:5887/moss/nested_frames ")
# to switch to a frame with name attribute
driver.switch_to.frame ("frame-bottom")
# to identify an element inside that frame and get text
print ( driver.find_element_by_css_selector ("body").text)
# to switch to the main page content
driver.switch_to.default_content ()
# to close the browser
driver.close ()
```

Handling Nested Frames

UNICAL ACADEMY

Nested frames:

- At first we must switch to the outer frame by either Index or ID of the iframe.
- Once we switch to the outer frame we can find the total number of iframes inside the outer frame, and.
- We can switch to the inner frame by any of the known methods.

Syntax to find frame count:

```
driver.findElement(By.tagName("iframe")).getSize();
```

Syntax to print frame count in develop console:

```
System.out.println(driver.findElement(By.tagName("iframe")).getSize());
```

Handling multiple Browsers or Tabs

Now let's take an example scenario to understand how it works. The scenario here is as follows:

- Open the Loan Management System URL.
- Search for "Branch" location.
- Save the URL of Branch.
- Open a new tab.
- Switch to the new tab and launch the stored URL.

To open the URL, use the sendKeys command as shown below:

```
driver.findElement(By.cssSelector("body")).sendKeys(Keys.CONTROL+ "t");
```

To switch back to the original tab:

```
driver.navigate().to("URL of the tab");
```

To close the Tab:

```
driver.close()
```

TargetLocator interface methods

UNICAL ACADEMY

Interface WebDriver.TargetLocator:

Modifier and Type	Method	Description
WebElement	**activeElement()**	Switches to the element that currently has focus within the document currently "switched to", or the body element if this cannot be detected.
Alert	**alert()**	Switches to the currently active modal dialog for this particular driver instance.
WebDriver	**defaultContent()**	Selects either the first frame on the page, or the main document when a page contains iframes.
WebDriver	**frame(int index)**	Select a frame by its (zero-based) index.
WebDriver	**frame(java.lang.String nameOrId)**	Select a frame by its name or ID.
WebDriver	**frame(WebElement frameElement)**	Select a frame using its previously located WebElement.
WebDriver	**newWindow(WindowType typeHint)**	Creates a new browser window and switches the focus for future commands of this driver to the new window.
WebDriver	**parentFrame()**	Change focus to the parent context.
WebDriver	**window(java.lang.String nameOrHandle)**	Switch the focus of future commands for this driver to the window with the given name/handle.

UNICAL ACADEMY

Session Recap