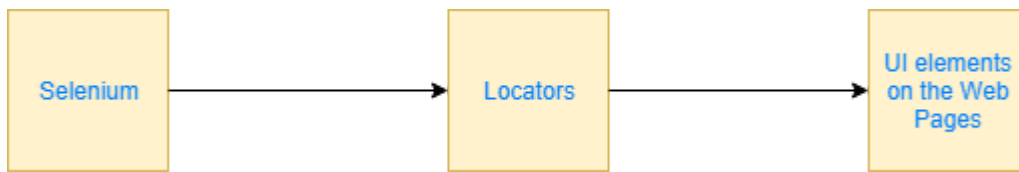# Selenium Java Training - Session 2 - Locators copy

Updated Feb 18th, 2022

Locators

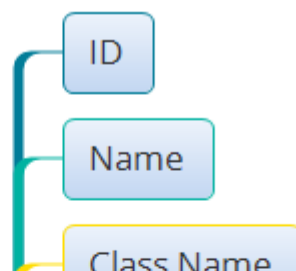### Introduction to Locators

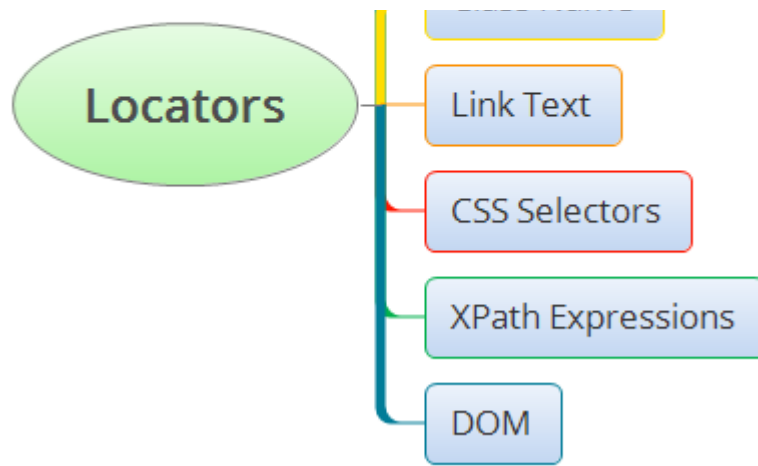- Locators help Selenium in finding the UI elements on Web pages.

- We can use these locators in our test automation code to locate the UI elements on the Web pages.
- Demonstrate the usage of Locators in locating the UI Elements on the Web Pages using Selenium IDE

Different types of Locators

The below are the different types of locators which can be used for locating the UI elements on the Web Pages:

HTML Basics for Locators

Having knowledge of below HTML basics will help you in understanding the locators better:

1. Take any simple web page say http://compendiumdev.co.uk/selenium/basic_web_page.html and inspect it to find its html code
2. Let's decode the above to create our own web page similar to above
    1. Title
    2. Paragraphs
3. Now, lets learn the HTML code for displaying different types of UI elements:
    1. Adding Rulers using <hr/>
    2. Adding Hyperlink (Requires href attribute with value and target attribute with value '_blank')
    3. Adding a Button (Using button tags and id attribute)
4. Inspect different UI elements on Omayo to identify the HTML code behind it
    1. Radio options having name attributes
    2. Drop down option having class attribute
    3. Image having src attribute
    4. Text Area field having some random attributes

Demonstrating Different types of Locators using Selenium IDE

Install Selenium IDE in any supported browser.

With the help of Selenium IDE's Target Text box field, demonstrate all the below different types of locators:
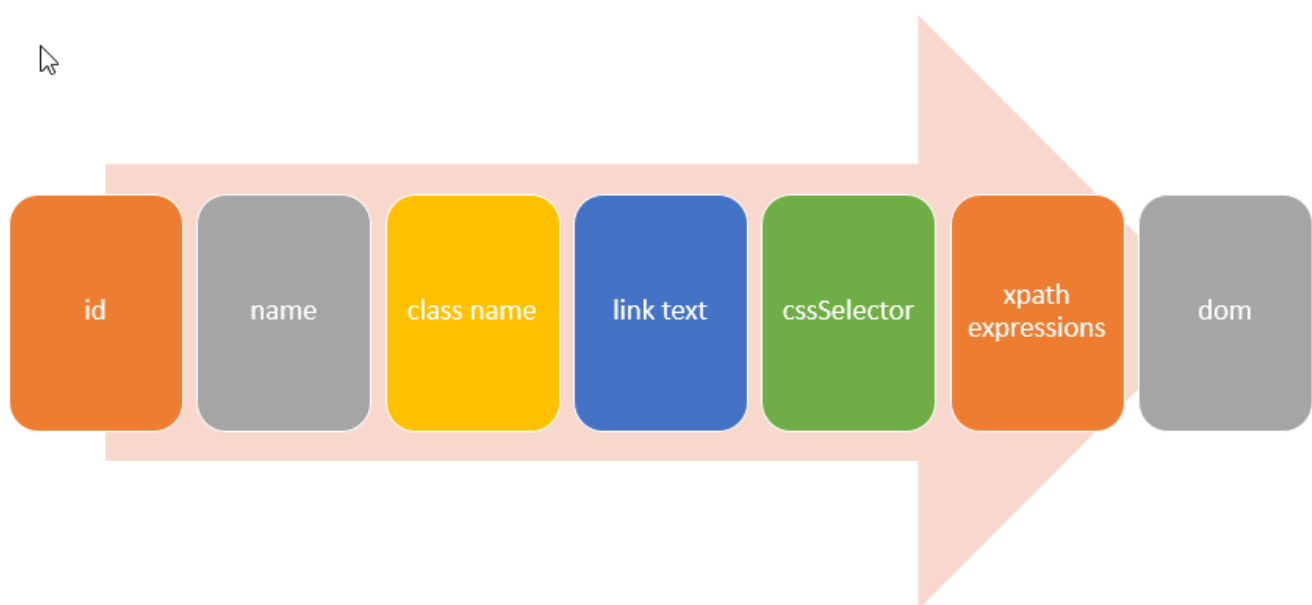
Note: class and dom locators won't work in Selenium IDE.

- **id** locator - Example: 'Button2' button in omayo blog - Syntax: id=but2
- **name** locator - Example: 'Locate using name attribute' text box field in omayo blog - Syntax: name=textboxn
- **class** locator - Example: text box "locate using class" in omayo blog - Syntax: class=classone
    - I will demonstrate using this locator during Selenium WebDriver sessions.
- **link** locator - Example: link 'compendiumdev' in omayo blog - Syntax: link=compendiumdev
- **css** locator - Example: 'Button2' button in omayo blog - Syntax: css=#but2
- **xpath** locator - Example: 'Button2' button in omayo blog - Syntax: xpath=//*[@id='but2']
- **dom** locator - Example: 'Button2' button in omayo blog - Syntax: dom=document.getElementById("but2")

Locators Priority

Though there are different types of locators available for locating the UI elements on the Web Pages, we need to use any one of them based on their priority.

The below is the priority order in which we need to select and use the locators :

- 'id' locator needs to be given the first priority. i.e. If the same UI element can be located with the help of different locators like id, name and so on, we need to choose id locator

with the help of different locators like id, name and so on, we need to choose id locator locating it.

- Similarly second priority goes for 'name' locator. i.e. If the UI element cannot be located by id locator, then we will prefer to choose 'name' locator as second priority.
- Third priority goes for 'class' locator.
- Forth priority goes for 'link text' locator.
- Fifth priority goes for 'cssSelector' locator.
- Sixth priority goes for 'xpath expressions' locator.
- Last priority goes for 'dom' locator.

ChroPath for Auto-generating the XPath Expressions and CSS Selectors

1. Installing ChroPath in any supported browser
2. Inspect any element and go to ChroPath tab to auto-generate the XPath Expressions and CSS Selectors
3. Confirm its accuracy using Selenium IDE

By,

Arun Motoori

Evernote makes it easy to remember things big and small from your everyday life using your computer, tablet, phone and the web.