**Dynamic Web Elements**

Generally the location of elements on web page is achieved by using ID, NAME, CLASS, etc. however the mentioned attributes might be missing or randomly changing the values in the DOM. In these kind of situations, we need to use smart locators to identify the elements on the page and below are the few different techniques used.

**XPath Selenium Selectors**

Basically, the tag names are HTML tags [label, a, Input, button, iframe, video and others]

/: from root [absolute xpath]

//: from current node [relative xpath].

Tag name: html tags.

@: to select html tag attributes.

Attribute name: attribute name.

Value: attribute vale.

**Absolute XPath**

It is a direct way to locate an element from the root [meaning start of HTML tag].

Starts with single slash “/” that means starting to search from the root.

Example: /html/body/div[1]/div[3]/div[2]/div[1]/div[2]/form/div/input

**Relative XPath**

Starts from the middle of the HTML DOM.

Starts with a double slash “//” that means it can start to search anywhere in the DOM structure.

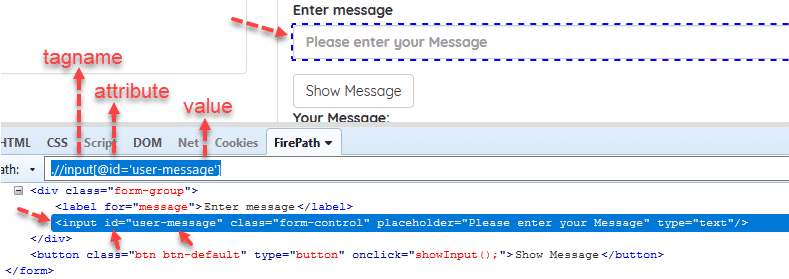
Shorter than Absolute XPath.

Example: //div[@class=list-of-items’]//div[2]//input[@id=’sub-items]

**Tag Attribute and Value**

**Syntax : //tagname[@attribute=’Value‘]**

**Example: //input[@id=’user-message‘]**



//input[@type='send text']

//label[@id='clkBtn']

//input[@value='SEND']

**Contains**

Partial matching attribute values.

Example:

name = ‘btnK’

//\*[contains(@name,'btn')]

--> It searches "btn" for all name attributes in the DOM.

//\*[contains(text(),'Offered')]

--> It searches the text "Offered" in the DOM.

**Starts-with**

//input[starts-with(@id, ‘user’)]

**Chained XPaths**

/div[@class=’form-group’]//input[@id=’user-message’]

**XPath with “or” Statement**

//\*[@id=’user-message’ or @class=’form-control’]

**XPath with “and” Statement**

//\*[@id=’user-message’ and @class=’form-control’]

**XPath Text**

//label[text()=’Enter message’]

Locating an element based on the text displayed.

Example:

'Search' button [on yahoo or google]

//\*[text()='Google Search'] ( '\*' represents any tag with text Google Search)

**Ancestor**

The ancestor axis selects all ancestors element (grandparent, parent, etc.) of the current node.

Xpath=//\*[text()='Enterprise Testing']//ancestor::div[1]

**Following**

Selects all elements in the document of the current node( )

Xpath=//\*[@type='text']//following::input

Xpath=//\*[@type='text']//following::input[1]

**Child**

Selects all children elements of the current node

Xpath=//ul[@id='list-of-items']/child::li

**Preceding**

This method selects all the nodes that come before the current node. Below is an example.

Xpath=//\*[@type=’text’]//preceding::input

Xpath=//\*[@type=’text’]//preceding::input[3]

**Following-sibling**

This method Select the following siblings from the context node. Siblings are located at the same level of the current. Below is an example.

xpath=//\*[@type=’text’]//following-sibling::input

**Descendant**

This method selects the descendants of the current. Below is an example.

Xpath=//\*[@id=’soft-test-class’]//descendant::a

Xpath=//\*[@id=’soft-test-class’]//descendant::a[1]

**Parent**

This method selects the parent of the current node. Below is an example.

Xpath=//\*[@id=’soft-test-class’]//parent::div

Xpath=//\*[@id=’soft-test-class’]//parent::div[1]

|  |  |
| --- | --- |
| XPATH axes | Result |
| **ancestor** | Selects all ancestors (parent, grandparent, etc.) of the current node |
| **ancestor-or-self** | Selects all ancestors (parent, grandparent, etc.) of the current node and the current node itself |
| **attribute** | Selects all attributes of the current node |
| **child** | Selects all children of the current node |
| **descendant** | Selects all descendants (children, grandchildren, etc.) of the current node |
| **descendant-or-self** | Selects all descendants (children, grandchildren, etc.) of the current node and the current node itself |
| **following** | Selects everything in the document after the closing tag of the current node |
| **following-sibling** | Selects all siblings after the current node |
| **namespace** | Selects all namespace nodes of the current node |
| **parent** | Selects the parent of the current node |
| **preceding** | Selects all nodes that appear before the current node in the document, except ancestors, attribute nodes and namespace nodes |
| **preceding-sibling** | Selects all siblings before the current node |
| **self** | Selects the current node |
|  |  |

**References:**

<https://www.w3schools.com/xml/xpath_nodes.asp>

<https://www.guru99.com/xpath-selenium.html>

<https://www.w3schools.com/xml/xpath_intro.asp>

<https://www.w3.org/TR/xpath/all/>

<http://www.zvon.org/comp/r/tut-XPath_1.html>