Mastering XPath in Selenium

Xpath is one of the essential identifier available in Selenium WebDriver. Out of those eight identifiers, we studied earlier tutorials Id, Classname, and Name is available as attributes in html code and is preferred if they are unique. LinkText and Partial LinkText are used if a web element is a link. But if no other identifier is available then we are left with either to go with CSS Selector or XPath.

# Definition of XPath:

XPath is the path written using html tags(or XML tags) and their attributes to reach to a particular node (or web element) in an HTML page or XML page.

# Types of XPath:

* Absolute XPath
* Relative XPath

# Absolute XPath:

Absolute xpath starts from the beginning of the page. As in an html page, the first tag is HTML. So all absolute xpaths always start with html tag and then accessing immediate child to reach to a node.

“/” is used to access an immediate child of the parent tag.

Example: html/body/table/tbody/tr[2]/td/input

Here, we are navigating to an input type of node which exists in a table in the second row and first cell

.

**Relative XPath:**

Relative xpath starts from anywhere on the page.

“//” is used to access any child of the parent tag.

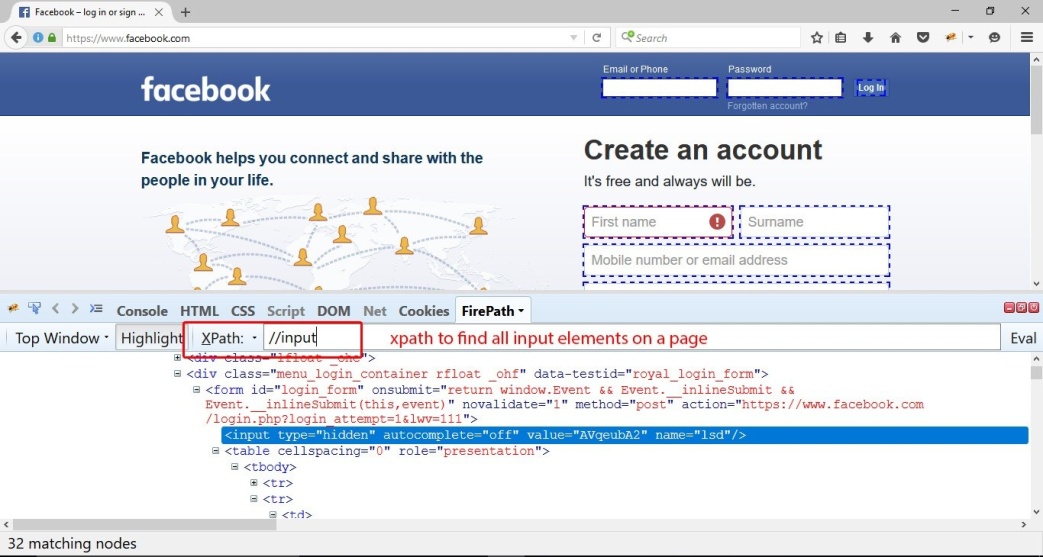
**Basic Syntax:** //htmlTagname[@attribute=’value’]

**Example:** //input[@type=’text’] – It represents xpath of a WebElement which is represented by an input tag and also has an attribute type = ‘text’.

Absolute xpath are fast, but they have one disadvantage that if there is an addition or deletion of some nodes in between,  then they fail to work.

In such a situation relative xpath are much better, and one should be good at writing dynamic xpath. There are some methods, operators, and axes which are available in xpath which can help to locate elements uniquely.

Some Examples to write Relative XPath:



**Relative XPath example** to search all input types of elements on a page.Firepath also shows us the number of identifying elements based on the criteria

# How to Identify Dynamic Web Elements?

Dynamic Web Elements are one which changes dynamically like – their attributes such as Id or Classname are changing, or text associated is changing.

Best possible way to identify these elements is to first search for another web element which stable and also can be identified uniquely, then on this element using methods, operators, or axes, one can reach to the desired element.

# Methods in Xpaths:

# Contains():

Contains is a method used when the value of any attribute changes dynamically. It can search an element with partial information.

Example: To search a button xPath used can be:

**//input[contains(@type=’submit’)]**

Example 2: Xpath of a below displayed web element using contains method can be written like:

**//input[contains(@name=’email’)] or //input[contains(@name=’ema’)]**

Both above examples will work because contains uses partial matching.

Example 3: Using another method text() as an argument to the contains() method in XPath.

**//a[contains(text(), ‘Mobile & Accessories’)]**

Above xpath will search for the links which has a text Mobile & Accessories as the linktext.

# Starts-with()–

This method can be used when we are searching for web elements matching the start of the text of the attributes passed. The text() method can also be used which will match the starting of the text.

Example: For above example, xpath can be written using starts-with as

**//a[starts-with(text(),’Mobile’)]**

Text() –

This method is used when we are searching for elements matching the exact text.

**//a[text()=’Mobile & Accessories’]**

# Operators in xpath:

AND and OR is two operators available in xpath.

AND operator, when applied to multiple attributes, identifies a web element only when all the attributes are pointing to that element.

Example: //input[@type=’text’  and @name=’uid’]

This xpath will identify an input web element which is of type text and name as uid.

OR operator, when applied to multiple attributes, defines a web element only when any one of the attributes points to that element:

Example: //input[@type=’text’  or @name=’uid’]

This xpath will identify an input web element which is either of type text or name is “uid.”

# AXES in Selenium WebDriver:

**Axis** in xpath points the xpath processor to the direction in which it should navigate in the hierarchy of the html nodes.

**Basic Syntax:**

//html\_tag[@attribute=’value’]//axes::html\_tag

Frequntly used Axes in XPath:

1. Ancestor – contains ancestor of the context node.
2. Child – includes the child of the context node.
3. Following – select the elements which follow after the context node.
4. Preceding – choose the elements which precede before the context node.
5. Following-Sibling – choose the sibling coming after the context node.
6. Preceding-Sibling – choose the sibling coming before the context node.
7. Parent – contains the parent of the context node.