**XPATH:** (XML Path) is a syntax for finding any element on the web page using HTML DOM structure.

**Syntax of XPath :**

//tagName[@attribute='value']

**Types of XPath: Two types of XPath**

**1.Absolute Xpath(/) :** Direct way to find element.

**Disadvantage :** If there are any changes made in the path of element then XPath gets failed. //

**EX :** html/body/div/div[1]/span/div/b

**2.Relative Xpath(//) :** This will starts from middle of the HTML DOM structure which starts with double forward slash(), which means it can search the element anywhere at the webpage.

**EX:** //<tagName>[@<attribute>=’<value’]

**Different XPath axes:** XPath axes are the methods used t find dynamic elements,which a rent possible by normal XPath method having noID,ClassName,Name,type….

1.Contains

2.OR & AND

3.starts-with

4.text()

**5.XPath axes methods :**

**5.1.following**

**5.2.Ancestor**

**5.3.child**

**5.4.preceding**

**5.5.following-sibling**

**5.6.parent**

**5.7.self**

|  |  |
| --- | --- |
| **S.No.** | **Axis & Description** |
| 1 | **ancestor**  Represents the ancestors of the current node which include the parents up to the root node. |
| 2 | **ancestor-or-self**  Represents the current node and it's ancestors. |
| 3 | **child**  Represents the children of the current node. |
| 4 | **descendant**  Represents the descendants of the current node. Descendants include the node's children upto the leaf node(no more children). |
| 5 | **descendant-or-self**  Represents the current node and it's descendants. |
| 6 | **following**  Represents all nodes that come after the current node. |
| 7 | **following-sibling**  Represents the following siblings of the context node. Siblings are at the same level as the current node and share it's parent. |
| 8 | **parent**  Represents the parent of the current node. |
| 9 | **preceding**  Represents all nodes that come before the current node (i.e. before it's opening tag). |
| 10 | **self**  Represents the current node. |