Last updated: May 14, 2021

Selenium Java Training - Session 16 - XPath Expressions - Cheatsheet (Part 2)

XPath Expressions - Part 2

- Finding 'input' tags having name attribute as 'gender' - //input[@name='gender']
- Finding the first 'input' tags having name attribute as 'gender' - //input[@name='gender'][1]
- Finding the second 'input' tags having name attribute as 'gender' - //input[@name='gender'][2]
- Finding any tags having name attribute as 'gender' - //*[@name='gender']
- Finding 'input' tags having any attribute value as 'gender' - //input[@*='gender']
- Finding any tags having any attribute value as 'gender' - //*[@*='gender']
- Finding 'input' tags having name attribute value as anything - //input[@name]
- Finding any tags having any attribute value as anything - //*[@*]havi
- Finding any elements having id attribute value as 'radio1' and name attribute value as 'gender' - //*[@id='radio1'][@name='gender']
- Finding the first 'input' tags having name attribute as 'gender' and is the first element - //input[@name='gender'][1]
- Finding the second 'input' tags having name attribute as 'gender' and is the second element - //input[@name='gender'][2]
- Finding any elements having id attribute value as 'radio1' or name attribute value as 'gender' - //*[@id='radio1' or @name='gender']
- XPath Expressions - Part 2 - Demonstrate at http://omayo.blogspot.in/
    - Find all the hyper links in the page - //a
    - Find all the hyper links having URL 'http://www.Selenium143.blogspot.com' - //a[@href='http://www.Selenium143.blogspot.com']
    - Find the first hyper link having URL 'http://www.Selenium143.blogspot.com' - (//a[@href='http://www.Selenium143.blogspot.com'])[1]
    - Find the third hyper link having URL 'http://www.Selenium143.blogspot.com' - (//a[@href='http://www.Selenium143.blogspot.com'])[3]
    - Difference between (//a[@href='http://www.Selenium143.blogspot.com'])[3] and //a[@href='http://www.Selenium143.blogspot.com'][3]
        - Second XPath searches for the third element at tag level
        - First XPath searches for the third element at page level
    - Find first child of 'html' tag - //html/*[1]
    - Find second child of 'html' tag - //html/*[2]
    - Find first child of 'body' tag - //body/*[1]
    - Find second child of 'body' tag - //body/*[2]

XPath Functions

- XPath functions: Part1
  - **text()** - Demonstrate at http://omayo.blogspot.in/
    - Find the p tags having the exact text 'PracticeAutomationHere' - //p[text()='PracticeAutomationHere']
    - Use . instead of text() - Find the p tag having the exact text 'PracticeAutomationHere' - //p[.='PracticeAutomationHere']
  - **contains()** - Demonstrate at http://omayo.blogspot.in/
    - Purpose:
      - It is used when the value of any attribute changes dynamically.
      - Has the ability to find the elements with partial text
      - If part of the attribute value is changing dynamically i.e. id='123main123' to id='456main456', we can use //tagName[contains(@id,'main')] to locate such dynamically changing attribute values.
    - Find the input tag having the text 'ra' inside its value attribute text - //input[contains(@value,'ra')]
    - Find the p tag containing the text 'Automation' - //p[contains(text(),'Automation')]
    - Find the p tag containing the text 'Automation' using.- //p[contains(.,'Automation')]
  - **starts-with()**
    - Purpose:
      - It is used when the value of any attribute changes dynamically.
      - Has the ability to find the elements with partial text i.e. initial partial text
      - If part of the attribute value is changing dynamically i.e. id='main123' to id='main456', we can use //tagName[starts-with(@id,'main')] to locate such dynamically changing attribute values.
    - Find the input tag having the value attribute text starting with letter 'o' - //input[starts-with(@value,'o')]
    - Find the p tag starting with text 'Practice' - //*[starts-with(text(),'Practice')]
    - Find the p tag starting with text 'Practice' using . - //*[starts-with(.,'Practice')]
- XPath functions: Part2 (Demonstrate at http://compendiumdev.co.uk/selenium/basic_web_page.html)
  - Find the first child of 'body' tag - //body/*[1]
  - **last()** - Find the last child of 'body' tag - //body/*[last()]
  - Find the first 'p' tag - //p[1]
  - last() - Find the last 'p' tag - //p[last()]
  - Find the last but one 'p' tag - //p[last()-1]
  - Locate the last but 2 input tag - (//input)[last()-2] (Demonstrate at http://omayo.blogspot.in/)
  - Find second 'p' tag having class 'sub' - //p[2][@class='sub']
  - Find the last 'p' tag having class 'sub' - //p[last()][@class='sub']

        - Find the last but one 'p' tag having class 'main' - //p[last()-1][@class='main']
    - XPath functions: Part3 (Demonstrate at http://compendiumdev.co.uk/selenium/basic_web_page.html)
        - **position()** - Find the first 'p' tag - //p[position()=1]
        - position() - Find the second 'p' tag - //p[position()=2]
        - position() - Find the 8th input tag - (//input)[position()=8] (Demonstrate at http://omayo.blogspot.in/)

XPath Axes

- XPath AXES: (Demonstrate at http://omayo.blogspot.in/)
    - Purpose:
        - If you want to locate an element which doesn't have id/name/class etc., with the help of XPath Axes we can locate such elements not having id/name/class with the help of id/name/class attributes of ancestor/descendant tags.
    - following
        - Purpose: Selects everything in the document after the closing tag of the current node
        - Find all the 'body' tags after the 'head' tag - //head/following::body
        - Find all the 'div' tags after //body/div[1]/div - //body/div[1]/div/following::div
        - Find the first 'div' after //body/div[1]/div - //body/div[1]/div/following::div[1]
        - Find all the 'input' tags after //body/div[1] - //body/div[1]/following::input
    - preceding
        - Purpose: Selects all nodes that appear before the current node in the document, except ancestors nodes
        - Find all the 'head' tags before the 'body' tag - //body/preceding::head
        - Find all the 'div' tags before //body/div[4] - //body/div[4]/preceding::div
    - following-sibling
        - Purpose: Selects all siblings after the current node
        - Find all the 'div' tag siblings after //body/div[1] - //body/div[1]/following-sibling::div
        - Find all the 'p' tag siblings after //body/p[1] - //body/p[1]/following-sibling::p (Demonstrate at http://compendiumdev.co.uk/selenium/basic_web_page.html)
    - preceding-sibling
        - Purpose: Selects all siblings before the current node
        - Find all the 'div' tag siblings before //body/div[4] - //body/div[4]/preceding-sibling::div
        - Find all the 'p' tag siblings before //body/p[2] - //body/p[2]/preceding-sibling::p (Demonstrate at http://compendiumdev.co.uk/selenium/basic_web_page.html)
    - parent
        - Purpose: Selects the parent of the current node

        - Find the parent of 'head' tag - //head/parent::html
        - Find the parent of 'body' tag - //body/parent::html
        - Find the parent of 'title' tag - //title/parent::head
        - Find the parent of first 'div' tag inside 'body' tag i.e. //div[1] - //div[1]/parent::body
    - child
        - Purpose: Selects all children of the current node
        - Find one of the child tag say 'head' of 'html' tag - //html/child::head
        - Find one of the child tag say 'body' of 'html' tag - //html/child::body
        - Find one of the child tag say 'title' of 'head' tag - //head/child::title
        - Find one of the child tag say first 'div' tag of 'body' tag - //body/child::div[1]
    - ancestor
        - Purpose: Selects all ancestors (parent, grandparent, etc.) of the current node
        - Find the ancestor 'html' tag for 'title' tag - //title/ancestor::html
        - Find the ancestor 'html' tag for 'head' tag - //head/ancestor::html
        - Find the ancestor 'html' tag for 'body' tag - //body/ancestor::html
    - descendant
        - Purpose: Selects all descendants (children, grandchildren, etc.) of the current node
        - Find the descendant 'title' tag for 'html' tag - //html/descendant::title
        - Find the descendant 'head' tag for 'html' tag - //html/descendant::head
        - Find the descendant 'body' tag for 'html' tag - //html/descendant::body

Miscellenious

- //ParentXpath//ChildXPath//GrandChildXPath
    - Child XPath will be searched in the parent XPath located section or area

---

By,
Arun Motoori