Selenium 4 Beginners

Rex Jones II

Target Elements With XPath Axes

Table of Contents

13 XPath Axes

Next, is XPath Axes. An axis represents a relationship. That relationship depends on the current node and it is used to find other nodes like a parent node, sibling node, or child node. There's a total of 13 XPath Axes. In alphabetical order, the XPath axes are Number 1 ancestor, Number 2 ancestor-or-self, Number 3 attribute, Number 4 child, Number 5 descendant, Number 6 descendant-or-self, Number 7 following, Number 8 following-sibling, Number 9 namespace, Number 10 parent, Number 11 preceding, Number 12 preceding-sibling, and Number 13 self.

XPath Axes Syntax

Here is the syntax for an XPath Axes. Axis specifies the tree-relationship between current node and selected nodes. The 2 colons separate an axis and node. The node identifies the Tag Name and index is not required. It specifies the node sequence.

Following & Preceding XPath Axes

We are going to cover the following and preceding XPath Axes.

Syntax Cheat Sheet

The preceding axis select all Tag Names before the current node and following selects all Tag Names after the current node.

Demo XPath Axes

Let's pick up where we left off after using XPath Functions. Right now, we have already found the First name label. To use the preceding XPath Axes. We write 2 forward slashes, next is the preceding axis. 2 colons, then the tagName. The tagName depends on the element we are looking for. How about the link element which has an a for anchor as the tag? This query returned 177. The index is optional but let's narrow our search down to Link Text by writing 1. 1 represents the first anchor tag that precedes the First name label. To select partial link text, we write 2 because it is 2 elements before the First name label.

The next Axis is following. Let's select all of the input tags after Years of Experience. Inspect Years of Experience, we see a Label tagName. Let's write 2 forward slashes, label, brackets, at symbol. There are 2 attributes: class and for. If we use class, ='control-label' then the query returns 3 elements. 1, 2, 3. Years of Experience, Profession, Profile Picture. Therefore, let's use the for attribute: for='exp'. Now, we see 1 of 1 for Years of Experience then write 2 forward slashes, following, 2 colons, input. There are 20 input elements following Years of Experience. Search, Search, …. Some of the elements are the radio buttons for Years of Experience, Date Picker, Manual Tester, Automation Tester. That's how we select WebElements using XPath Axes. You can download the Cheat Sheet and Transcript at https://tinyurl.com/SeleniumLocatorsForWebElements and from github at RexJonesII/Selenium4Beginners. Next, is Creating Our Own Customized CSS Selector Values For Selenium.