



Contextual Selectors

LOCATOR	EXPLANATION
<code>//img</code>	image element
<code>//img/*[1]</code>	first child of element img
<code>//ul/child::li</code>	first child 'li' of 'ul'
<code>//img[1]</code>	first img child
<code>//img/*[last()]</code>	last child of element img
<code>//img[last()]</code>	last img child
<code>//img[last()-1]</code>	second last img child
<code>//ul[*]</code>	ul' that has children

Math Methods

LOCATOR	EXPLANATION
<code>ceiling(number)</code>	evaluates a decimal number and returns the smallest integer greater than or equal to the decimal number
<code>floor(number)</code>	evaluates a decimal number and returns the largest integer less than or equal to the decimal number
<code>round(decimal)</code>	returns a number that is the nearest integer to the given number
<code>sum(node-set)</code>	returns a number that is the sum of the numeric values of each node in a given node-set

Attribute Selectors

LOCATOR	EXPLANATION
<code>//img[@id='myId']</code>	image element with @id= 'myId'
<code>//img[@id!='myId']</code>	image elements with @id not equal to 'myId'
<code>//img[@name]</code>	image elements that have name attribute
<code>//*[contains(@id, 'Id')]</code>	element with @id containing
<code>//*[starts-with(@id, 'Id')]</code>	element with @id starting with
<code>//*[ends-with(@id, 'Id')]</code>	element with @id ending with
<code>//*[matches(@id, 'r')]</code>	element with @id matching regex 'r'
<code>//*[@name='myName']</code>	image element with @name= 'myName'
<code>//*[@id='X' or @name='X']</code>	element with @id X or a name X
<code>//*[@name="N"][@value="v"]</code>	element with @name N & specified @value 'v'
<code>//*[@name="N" and @value="v"]</code>	element with @name N & specified @value 'v'
<code>//*[@name="N" and not(@value="v")]</code>	element with @name N & not specified @value 'v'
<code>//input[@type="submit"]</code>	input of type submit
<code>//a[@href="url"]</code>	anchor with target link 'url'
<code>//section[//h1[@id='hi']]</code>	returns <section> if it has an <h1> descendant with @id= 'hi'
<code>//*[@id="TestTable"]//tr[3]//td[2]</code>	cell by row and column
<code>//input[@checked]</code>	checkbox (or radio button) that is checked
<code>//a[@disabled]</code>	all 'a' elements that are disabled
<code>//a[@price > 2.50]</code>	'a' with price > 2.5

XPath Methods

LOCATOR	EXPLANATION
<code>//table[count(tr) > 1]</code>	return table with more than 1 row
<code>//*[.="t"]</code>	element containing text 't' exactly
<code>//a[contains(text(), "Log Out")]</code>	anchor with inner text containing 'Log Out'
<code>//a[not(contains(text(), "Log Out"))]</code>	anchor with inner text not containing 'Log Out'
<code>//a[not(@disabled)]</code>	all 'a' elements that are not disabled

Axis Navigation

LOCATOR	EXPLANATION
<code>//td[preceding-sibling::td="t"]</code>	cell immediately following cell containing 't' exactly
<code>//td[preceding-sibling::td[contains(., "t")]]</code>	cell immediately following cell containing 't'
<code>//input/following-sibling::a</code>	'a' following some sibling 'input'
<code>//a/following-sibling::*</code>	sibling element immediately following 'a'
<code>//input/preceding-sibling::a</code>	'a' preceding some sibling 'input'
<code>//input/preceding-sibling::*[1]</code>	sibling element immediately preceding 'input'
<code>//img[@id='MyId']::parent/*</code>	the parent of image with id

String Methods

LOCATOR	EXPLANATION
<code>contains(space-string, planet-string)</code>	determines whether the first argument string contains the second argument string and returns boolean true or false
<code>concat(string1, string2 [stringn]*)</code>	concatenates two or more strings and returns the resulting string
<code>normalize-space(string)</code>	strips leading and trailing white-space from a string, replaces sequences of whitespace characters by a single space, and returns the resulting string
<code>starts-with(spacetrack, space)</code>	checks whether the first string starts with the second string and returns true or false
<code>string-length([string])</code>	returns a number equal to the number of characters in a given string
<code>substring(string, start [length])</code>	returns a part of a given string
<code>substring-after(spacetrack, track)</code>	returns a string that is the rest of a given string after a given substring
<code>substring-before(spacetrack, tra)</code>	returns a string that is the rest of a given string before a given substring
<code>translate(string, ghj, GHJ)</code>	evaluates a string and a set of characters to translate and returns the translated string