EPATH • GSS • DOM • SELENDOM Rosetta Stone and Cookbook

Sprinkled with Selenium usage tips, this is both a general-purpose set of recipes for each technology as well as a cross-reference to map from one to another. The validation suite for this reference chart (http://bit.ly/gTd5oc) provides example usage for each recipe supported by Selenium (the majority of them).

Category	Recipe	XPath (1.0 – 2.0)	CSS (CSS1 – 3)	DOM	Selenium
General	Whole web page	xpath=/html	css=html	document.documentElement	NA
	Whole web page body	xpath=/html/body	css=body	document.body	NA
	All text nodes of web page	//text() 🖾	NA	NA	NA
	Element <e> by absolute reference</e>	xpath=/html/body///E	css=body>>>E	document.body.childNodes[i]childNodes[j]	NA
	Element <e> by relative reference</e>	//E	css=E	document.gEBTN('E')[0]	NA
	Second <e> element anywhere on page</e>	xpath=(//E)[2]	NA	document.gEBTN('E')[1]	NA
	Image element	//img	css=img	document.images[0]	NA
	Element <e> with attribute A</e>	//E[@A]	css=E[A]	dom=for each (e in document.gEBTN('E')) if (e.A) e	NA
	Element <e> with attribute A containing text 't' exactly</e>	//E[@A='t']	css=E[A='t'] ②	NA	NA
	Element <e> with attribute A containing text 't'</e>	//E[contains(@A,'t')]	css=E[A*='t']	NA .	NA NA
Tag					
	Element <e> whose attribute A begins with 't'</e>	//E[starts-with(@A, 't')]	css=E[A^='t']	NA	NA
	Element <e> whose attribute A ends with 't'</e>	//E[ends-with(@A, 't')] ☑ ◀OR▶	css=E[A\$='t'] @	NA	NA
		//E[substring(@A, string-length(@A) - string-length('t')+1)='t']			
	Element <e> with attribute A containing word 'w'</e>	//E[contains(concat('◉', @A, '◉'), '◉w◉')	css=E[A~='w'] 2	NA	NA
	Element <e> with attribute A matching regex 'r'</e>	//E[matches(@A, 'r')] 🔯	NA .	NA	NA
	Element <e1> with id I1 or element <e2> with id I2</e2></e1>	//E1[@id=I1] //E2[@id=I2]	css=E1#I1,E2#I2	NA	NA
	Element <e1> with id I1 or id I2</e1>	//E1[@id=I1 or @id=I2]	css=E1#I1,E1#I2	NA	NA
	Attribute A of element <e></e>	//E/@A 🔀 {Se: //E@A }	NA {Se: css=E@A }	document.gEBTN('E')[0].getAttribute('A')	NA
6 ـ ـ ـ ـ ا: ـ ـ ـ ـ 6	The state of the s	majer is become and	101 (SC. CSS E.G. 1.)	{Se: document.gEBTN('E')[0]@A }	,
	Attailment A of any alamant	//*/@A P [C //*@A]	NA (C * @ A)		NA
Attribute [€]	Attribute A of any element	//*/@A 🐼 {Se: //*@A }	NA {Se: css=*@A }	NA NA	NA NA
	Attribute A1 of element <e> where attribute A2 is it exactly</e>	//E[@A2='t']/@A1	NA {Se: css=E[A2='t']@A1 }	NA	7471
	Attribute A of element <e> where A contains 't'</e>	//E[contains(@A,'t')]/@A Se: //E[contains(@A,'t')]@A	NA {Se: css=E[A*='t']@A }	NA	NA
	Element <e> with id I</e>	//E[@id='I']	css=E#I	NA	NA
Id	Element with id I	//*[@id='l']	CSS=#I	document.gEBI('I')	id=I
ıu	Element <e> with name N</e>	//E[@name='N']	css=E[name=N]	NA	NA
&	Element with name N	//*[@name='N']	css=[name=N]	document.getElementsByName('N')[0]	name=N
α	Element with id X or, failing that, a name X	//*[@id='X' or @name='X']	NA	NA	X ◀ OR▶ identifier
Name					
Name	Element with name N & specified 0-based index 'v'	//*[@name='N'][v+1]	css=[name=N]:nth-child(v+1)	NA	name=N index=v
	Element with name N & specified value 'v'	//*[@name='N'][@value='v']	css=[name=N][value='v']	NA	name=N value=v
Lang	Element <e> is explicitly in language L or subcode</e>	//E[@lang='L' or starts-with(@lang, concat('L', '-'))]	css=E[lang =L]	NA	NA
&	Element <e> is in language L or subcode (possibly inherited)</e>	NA	css=E:lang(L)	NA	NA
	Element with a class C	//*[contains(concat('@', @class, '@'), '@C@')]	css=.C	document.getElementsByClassName('C')[0]	NA
Class	Element <e> with a class C</e>	//E[contains(concat('⊕', @class, '⊕'), '⊕C⊕')]	css=F.C	NA , , , , , , , , , , , , , , , , , , ,	NA
	Element containing text 't' exactly	//*[.₹'t']	NA NA	NA .	NA
	Element <e> containing text 't'</e>	//E[contains(text(),'t')]	css=E:contains('t') ①	NA .	NA
Text				document.links[0]	NA NA
•	Link element	//a	css=a		
&	<a> containing text 't' exactly	//a[.='t']	NA	NA	link=t
بامنا	<a> containing text 't'	//a[contains(text(),'t')]	css=a:contains('t') @	NA	NA
Link	<a> with target link 'url'	//a[@href='url']	css=a[href='url']	NA .	NA
	Link URL labeled with text 't' exactly	//a[.='t']/@href	NA	NA	NA
	First child of element <e></e>	//E/*[1]	css=E > *:first-child { Se: css=E > * }	document.gEBTN('E')[0].firstChild	NA
	First <e> child</e>	//E[1]	css=E:first-of-type \(\Se: \css=E \)	document.getEBTN('E')[0]	NA
	Last child of element E	//E/*[last()]	css=E *:last-child	document.gEBTN('E')[0].lastChild	NA
	Last <e> child</e>		css=E:last-of-type 🖾	document.gEBTN(E)[document.gEBTN(E).length-1]	NA
		//E[last()]			
	Second <e> child</e>	//E[2] ◀OR► //E/following-sibling::E	css=E:nth-of-type(2)	document.getEBTN('E')[1]	NA
	Second child that is an <e> element</e>	//*[2][name()='E']	css=E:nth-child(2)	NA	NA
Parent	Second-to-last <e> child</e>	//E[last()-1]	css=E:nth-last-of-type(2)	document.gEBTN(E)[document.gEBTN(E).length-2]	NA
1 di Ciit	Second-to-last child that is an <e> element</e>	//*[last()-1][name()='E']	css=E:nth-last-child(2)	NA	NA
&	Element <e1> with only <e2> children</e2></e1>	//E1/[E2 and not(*[not(self::E2)])]	NA	NA	NA
	Parent of element <e></e>	//E/	NA	document.gEBTN('E')[0].parentNode	NA
Child	Descendant <e> of element with id I using specific path</e>	//*[@id='l']///E	CSS=#I > > > E	document.gEBI('I')gEBTN('E')[0]	NA
					1
		//*[@id='l']//F			NA
	Descendant <e> of element with id I using unspecified path</e>	//*[@id='l']//E	css=#I E	document.gEBI('I').gEBTN('E')[0]	NA NA
	Element <e> with no children</e>	//E[count(*)=0]	css=E:empty	NA	NA
	Element <e> with no children Element <e> with an only child</e></e>	//E[count(*)=0] //E[count(*)=1]	css=E:empty NA	NA NA	NA NA
	Element <e> with no children Element <e> with an only child Element <e> that is an only child</e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(preceding-sibling::*)+count(following-sibling::*)=0]	css=E:empty NA css=E:only-child	NA NA NA	NA NA NA
	Element <e> with no children Element <e> with an only child Element <e> that is an only child Element <e> with no <e> siblings</e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1 //E[count(preceding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1]	css=E:empty NA css=E:only-child css=E:only-of-type \[\begin{align*} \text{CSS} & CSS	NA NA NA NA	NA NA NA NA
	Element <e> with no children Element <e> with an only child Element <e> that is an only child</e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(preceding-sibling::*)+count(following-sibling::*)=0]	css=E:empty NA css=E:only-child	NA NA NA	NA NA NA
	Element <e> with no children Element <e> with an only child Element <e> that is an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th</e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(preceding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E[position() mod N = M + 1]	css=E:empty NA css=E:only-child css=E:only-of-type \[\begin{align*} \text{CSS} & CSS	NA NA NA NA	NA NA NA NA
	Element <e> with no children Element <e> with an only child Element <e> that is an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e> following some sibling <e>></e></e></e></e></e></e></e>	//E[count(*)=0] //E[count(preceding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::£1	css=E:empty AA css=E:only-child css=E:only-of-type css=E:nth-child(Nn + M) css=E2 ~ E1	NA NA NA NA NA	NA NA NA NA NA
	Element <e> with no children Element <e> with an only child Element <e> that is an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> immediately following sibling <e2></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(preceding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1] //E[count(/E) = 1] //E2/following-sibling::£1 //E2/following-sibling::£1 //E2/following-sibling::£1][name()='E1']	css=E:empty NA css=E:only-child css=E:only-of-type css=E:nth-child(Nn + M) css=E2 ~ E1 css=E2 + E1	NA	NA NA NA NA NA NA NA
	Element <e> with no children Element <e> with an only child Element <e> that is an only child Element <e> that is an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> with one intermediately following sibling <e2> Element <e1> following sibling <e2> with one intermediary</e2></e1></e2></e2></e1></e2></e1></e></e></e></e></e></e>	//E[count(*)=0] //E[count(r)=1] //E[count(r)=cding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::*1] //E2/following-sibling::*1] //E2/following-sibling::*2[] //E2/following-sibling::*2[] //E2/following-sibling::*2[] //E2/following-sibling::*2[] //E2/following-sibling::*2[] //E2/following-sibling::*2[]	css=E:empty NA css=E:only-child css=E:only-of-type css=E:nth-child(Nn + M) css=E2 ~ E1 css=E2 + E1 css=E2 + * E1	NA N	NA
Sibling	Element <e> with no children Element <e> with an only child Element <e> that is an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following <e></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(preceding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::*1][name()='E1'] //E2/following-sibling::*{2}[name()='E1'] //E2/following-sibling::*4	css=E:empty AA css=E:only-child css=E:only-of-type css=E:only-of-type css=E:only-of-type css=E2 × E1 css=E2 + E1 css=E2 + * E1 css=E4 * * *	NA N	NA N
Sibling	Element <e> with no children Element <e> with an only child Element <e> that is an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following <e> Element <e1> preceding some sibling <e2></e2></e1></e></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(preceding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1] //E[count(/E) = 1] //E2/following-sibling::E1 //E2/following-sibling::E1 //E2/following-sibling::*[1][name()='E1'] //E2/following-sibling::*[2][name()='E1'] //E2/preceding-sibling::E1	css=E:empty NA css=E:only-child css=E:only-of-type css=E:nth-child(Nn + M) css=E2 = E1 css=E2 = E1 css=E2 + E1 css=E2 + * E1 NA	NA N	NA N
Sibling	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e> with one intermediarely following sibling <e> Element <e1> following sibling <e> with one intermediary Sibling element immediately following <e> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding sibling <e> Element <e1> immediately preceding sibling <e> Element <e1> immediately preceding sibling <e></e></e1></e></e1></e></e1></e2></e1></e></e></e1></e></e></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[position() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::E1 //E2/following-sibling::F1[]name()='E1'] //E2/following-sibling::F2[]name()='E1'] //E2/following-sibling::E1 //E2/foceding-sibling::E1 //E2/preceding-sibling::E1 //E2/preceding-sibling::E1 [name()='E1']	css=E:empty NA css=E:only-child css=E:only-of-type css=E:onth-child(Nn + M) css=E2 ~ E1 css=E2 + E1 css=E2 + * E1 css=E + * NA NA	NA N	NA N
Sibling	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding sibling <e2> with one intermediary</e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[position() mod N = M + 1] //E2/following-sibling::1 //E2/following-sibling::1 //E2/following-sibling::1 //E2/following-sibling::4 //E2/preceding-sibling::1	css=E:empty NA css=E:only-child css=E:only-of-type (X) css=E:only-of-type (X) css=E2 = E1 css=E2 + E1 css=E2 + E1 css=E2 + * + E1 css=E4 + * E1 NA NA NA NA	NA N	NA N
Sibling	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e> with one intermediarely following sibling <e> Element <e1> following sibling <e> with one intermediary Sibling element immediately following <e> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding sibling <e> Element <e1> immediately preceding sibling <e> Element <e1> immediately preceding sibling <e></e></e1></e></e1></e></e1></e2></e1></e></e></e1></e></e></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1] //E[count(/E) = 1] //E2/following-sibling::1 //E2/following-sibling::1] //E2/following-sibling::*[1][name()='E1'] //E2/following-sibling::*[2][name()='E1'] //E2/preceding-sibling::1 //E2/preceding-sibling::1 //E2/preceding-sibling::*[1][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1']	css=E:empty NA css=E:only-child css=E:only-of-type css=E:onth-child(Nn + M) css=E2 ~ E1 css=E2 + E1 css=E2 + * E1 css=E + * NA NA	NA N	NA N
Sibling	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following <e> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1] //E[count(/E) = 1] //E2/following-sibling::1 //E2/following-sibling::1] //E2/following-sibling::*[1][name()='E1'] //E2/following-sibling::*[2][name()='E1'] //E2/preceding-sibling::1 //E2/preceding-sibling::1 //E2/preceding-sibling::*[1][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1']	css=E:empty NA css=E:only-child css=E:only-of-type css=E:nth-child(Nn + M) css=E2 = E1 css=E2 + E1 css=E2 + * + E1 css=E + * NA NA NA NA	NA N	NA N
	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding sibling <e2> with one intermediary</e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count[following-sibling::*)=0] //E[count[./E] = 1] //E[position() mod N = M + 1] //E2/following-sibling::f1 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/preceding-sibling::f1	css=E:empty NA	NA N	NA N
	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding sibling <e2> Element <e1> preceding sibling <e2> Element <e1> greceding sibling <e2> with one intermediary Sibling element immediately preceding <e> Cell by row and column (e.g. 3rd row, 2nd column)</e></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1 //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::f1][name()='E1'] //E2/following-sibling::f2[name()='E1'] //E2/following-sibling::f2[name()='E1'] //E2/preceding-sibling::f1[name()='E1'] //E2/preceding-sibling::f1[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E/preceding-sibling::f2[name()='E1'] //E/preceding-sibling::f2[name()='E1'] //E/preceding-sibling::f2[name()='E1'] //E/preceding-sibling::f2[name()='E1']	css=E:empty NA css=E:only-child css=E:only-of-type IX css=E:only-of-type IX css=E2 = 1 css=E2 + E1 css=E2 + E1 css=E2 + * E1 NA NA NA NA NA NA NA NA NA N	NA N	NA
	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e2> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following sibling <e> Element <e1> immediately following <e> Element <e1> immediately following <e> Element <e1> immediately preceding sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding sibling <e> Cell by row and column (e.g. 3rd row, 2nd column) Cell immediately following cell containing 't' exactly</e></e2></e1></e2></e1></e2></e1></e></e1></e></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::E1 //E2/following-sibling::E2[[name()='E1'] //E2/following-sibling::* //E2/preceding-sibling::E2[[name()='E1'] //E2/preceding-sibling::E1[[name()='E1'] //E2/preceding-sibling::E1[[name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::E1[] //E[preceding-sibling::E1] //*[@id='TestTable'].2.1] //*[@id='TestTable'].2.1]	css=E:empty NA css=E:only-child css=E:only-of-type IX css=E:only-of-type IX css=E:only-of-type IX css=E2 + E1 css=E2 + E1 css=E2 + * E1 css=E + * NA NA NA NA NA NA NA NA NA N	NA document.gEBTN('E')[0].nextSibling NA NA NA document.gEBTN('E2')[0].previousSibling document.gEBTN('E2')[0].previousSibling NA	NA
	Element <e> with no children Element <e> with an only child Element <e> with no se siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding <e3 't'="" 't'<="" (e.g.="" 2nd="" 3rd="" and="" by="" cell="" column="" column)="" containing="" exactly="" following="" immediately="" row="" row,="" td=""><td>//E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count[following-sibling::*)=0] //E[count[./E] = 1] //E[position() mod N = M + 1] //E2/following-sibling::f1 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/preceding-sibling::f2 //E2/preceding-sibling::f1 //E2/preceding-sibling::f1</td><td> css=E:empty NA</td><td>NA NA N</td><td> NA</td></e3></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count[following-sibling::*)=0] //E[count[./E] = 1] //E[position() mod N = M + 1] //E2/following-sibling::f1 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/preceding-sibling::f2 //E2/preceding-sibling::f1	css=E:empty NA	NA N	NA
	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding <e> Cell by row and column (e.g. 3rd row, 2nd column) Cell immediately following cell containing 't' exactly Cell immediately following cell containing 't' User interface element <e> that is disabled</e></e></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1 //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::f1][name()='E1'] //E2/following-sibling::f2[name()='E1'] //E2/following-sibling::f2[name()='E1'] //E2/preceding-sibling::f1[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E[gdis-TestTable']/tr[3]/td[2] Se: //*[gdis-TestTable].2.1 //td[preceding-sibling::d2[contains(.,'t')]] //E[@disabled]	css=E:empty NA css=E:only-child css=E:only-of-type IX css=E:only-of-type IX css=E2 = E1 css=E2 + E1 css=E2 + E1 css=E4 + E1 NA NA NA NA NA NA NA NA Css=#TestTable tr:nth-child(3) td:nth-child(2) fse: css=#TestTable.2.1 NA css=E:disabled	NA	NA
	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e2> Element <e1> following sibling <e2> with one intermediarty Sibling element immediately following sibling <e> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following sibling <e> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e3> preceding some sibling <e2> Element <e3> preceding some sibling <e3> Element <e3> preceding some sibling <e2> Element <e4> preceding some sibling <e4> Element <e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e3></e3></e3></e2></e3></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e2></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(r)=1] //E[count(r)=1	css=E:empty NA css=E:only-child css=E:only-of-type css=E:only-of-type css=E:only-of-type css=E2 + E1 css=E2 + E1 css=E2 + E1 css=E + * NA NA NA NA NA NA NA NA Css=#TestTable tr:nth-child(3) td:nth-child(2) Se:css=#TestTable tr:nth-child(3) td:nth-child(2) Se:css=#IestTable tr:nth-child(3) td:nth-child(2) Se:css=#IestTable tr:nth-child(3) td:nth-child(2) Se:css=#IestTable tr:nth-child(3) td:nth-child(3) NA css=#Id:contains("t") ~ td css=E:disabled css=":enabled"	NA document.gEBTN('E')[0].nextSibling NA NA NA A A A NA NA NA NA NA NA NA NA	NA
	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e2> Element <e1> following sibling <e2> with one intermediarty Sibling element immediately following sibling <e> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following sibling <e> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e3> preceding some sibling <e2> Element <e3> preceding some sibling <e3> Element <e3> preceding some sibling <e2> Element <e4> preceding some sibling <e4> Element <e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e4></e2></e3></e3></e3></e2></e3></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e2></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(r)=1] //E[count(r)=1	css=E:empty NA css=E:only-child css=E:only-of-type IX css=E:only-of-type IX css=E2 = E1 css=E2 + E1 css=E2 + E1 css=E4 + E1 NA NA NA NA NA NA NA NA Css=#TestTable tr:nth-child(3) td:nth-child(2) fse: css=#TestTable.2.1 NA css=E:disabled	NA	NA
Table Cell	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> preceding some sibling <e> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding sibling <e2> Element <e1> preceding sibling <e2> Element <e1> preceding sibling <e2> Element <e1> memdiately preceding sibling <e2> Element <e1> middiately preceding sibling <e2> Element <e1> middiately following <e1 't'="" (e.g.="" 2nd="" 3rd="" <e="" and="" by="" cell="" column="" column)="" containing="" element="" exactly="" following="" immediately="" interface="" row="" row,="" user=""> that is disabled User interface element that is enabled Checkbox (or radio button) that is checked</e1></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1 //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::f1][name()='E1'] //E2/following-sibling::f2[name()='E1'] //E2/following-sibling::f2[name()='E1'] //E2/preceding-sibling::f1[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E2/preceding-sibling::f2[name()='E1'] //E[gdis-TestTable']/tr[3]/td[2] Se: //*[gdis-TestTable].2.1 //td[preceding-sibling::d2[contains(.,'t')]] //E[@disabled]	css=E:empty NA css=E:only-child css=E:only-of-type css=E:only-of-type css=E:only-of-type css=E2 + E1 css=E2 + E1 css=E2 + E1 css=E + * NA NA NA NA NA NA NA NA Css=#TestTable tr:nth-child(3) td:nth-child(2) Se:css=#TestTable tr:nth-child(3) td:nth-child(2) Se:css=#IestTable tr:nth-child(3) td:nth-child(2) Se:css=#IestTable tr:nth-child(3) td:nth-child(2) Se:css=#IestTable tr:nth-child(3) td:nth-child(3) NA css=#Id:contains("t") ~ td css=E:disabled css=":enabled"	NA document.gEBTN('E')[0].nextSibling NA NA NA A A A NA NA NA NA NA NA NA NA	NA
	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e2> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following sibling <e2> Element <e1> immediately following <e> Element <e1> immediately following <e2> Element <e1> immediately preceding sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding sibling <e2> Element <e1> preceding sibling <e2> Element <e1> preceding sibling <e2> Cell by row and column (e.g. 3rd row, 2nd column) Cell immediately following cell containing 't' exactly Cell immediately following cell containing 't' user interface element <e> that is disabled User interface element that is enabled Checkbox (or radio button) that is checked Element being designated by a pointing device</e></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e2></e1></e></e1></e2></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[position() mod N = M + 1] //E2/following-sibling::f1 //E2/following-sibling::f1 //E2/following-sibling::f2 //E2/following-sibling::f2 //E2/preceding-sibling::f1 //f2/preceding-sibling::td[contains(.,'t')] //E2/pdisabled] //*[out-first fable /] //*[out-firs	css=E:empty NA css=E:only-child css=E:only-of-type css=E:nth-child(Nn + M) css=E2 + E1 css=E2 + E1 css=E2 + V + E1 css=E4 + V + E1 css=E4 + V + E1 css=E4 + V + E1 css=E5 + V + E1 css=E4 + V + E1 css=E5 + V + E1 css=E5 + V + V + V + V + V + V + V + V + V +	NA N	NA
Table Cell	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e2> Element <e1> following sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding <e2> Cell by row and column (e.g. 3rd row, 2nd column) Cell immediately following cell containing 't' exactly Cell immediately following cell containing 't' User interface element <e> that is disabled User interface element <e> that is disabled Checkbox (or radio button) that is checked Element being designated by a pointing device Element being designated by a pointing device Element being designated by a pointing device</e></e></e2></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[count(r)=1] //E[postion() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::E1 //E2/following-sibling::F1[]name()='E1'] //E2/following-sibling::F2[]name()='E1'] //E2/following-sibling::F2[]name()='E1'] //E2/preceding-sibling::F1[]name()='E1'] //E2/preceding-sibling::F1[]name()='E1'] //E2/preceding-sibling::F1[]name()='E1'] //E2/preceding-sibling::F1[]name()='E1'] //E2/preceding-sibling::F1[]name()='E1'] //E2/preceding-sibling::F1[] //E2/preceding-sibling::F1[] //E2/preceding-sibling::F1[] //E3/preceding-sibling::F1[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T2[] //E3/preceding-sibling::T3[] //E3/preceding-	css=E:empty MA css=E:only-child css=E:only-of-type css=E:only-of-type css=E:only-of-type css=E2 + E1 css=E2 + E1 css=E2 + * E1 css=E + * NA NA NA AA NA css=#TestTable tr:nth-child(3) td:nth-child(2) {Se:css=#TestTable.2.1} NA css=cd:contains('t') ~ td css=cd:css=ded css=*checked css=*checked css=E:hover Sess=Css=Elover Css=E:css=Elover Css=Elover Css=Elover	NA document.gEBTN("E")[0].nextSibling NA	NA
Table Cell	Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no <e> siblings Every Nth element starting with the (M+1)th Element <e1> following some sibling <e2> Element <e1> following some sibling <e2> Element <e1> following sibling <e2> with one intermediary Sibling element immediately following sibling <e2> Element <e1> immediately following <e> Element <e1> immediately following <e2> Element <e1> immediately preceding sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding sibling <e2> Element <e1> preceding sibling <e2> Element <e1> preceding sibling <e2> Cell by row and column (e.g. 3rd row, 2nd column) Cell immediately following cell containing 't' exactly Cell immediately following cell containing 't' user interface element <e> that is disabled User interface element that is enabled Checkbox (or radio button) that is checked Element being designated by a pointing device</e></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e2></e1></e></e1></e2></e2></e1></e2></e1></e2></e1></e></e></e></e></e>	//E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::E1 //E2/following-sibling::*[1][name()='E1'] //E2/following-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[1][name()='E1'] //E2/preceding-sibling::*[1][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E2/preceding-sibling::*[2][name()='E1'] //E[poseding-sibling::*[2][name()='E1'] //E[poseding-sibling::t2][name()='E1'] //E[poseding-sibling::t2][name()='E1'] //E[poseding-sibling::t2][poseding-sibling::t2] //E[poseding-sibling::t2][po	css=E:empty NA css=E:only-child css=E:only-of-type css=E:nth-child(Nn + M) css=E2 + E1 css=E2 + E1 css=E2 + V + E1 css=E4 + V + E1 css=E4 + V + E1 css=E4 + V + E1 css=E5 + V + E1 css=E4 + V + E1 css=E5 + V + E1 css=E5 + V + V + V + V + V + V + V + V + V +	NA N	NA

LEGEND XPath CSS DOM Selenium {Se: ... } Selenium-only variation Not supported by $|X\rangle$ Selenium ◉ Space character expression CSS3 or XPath 2.0 DOM abbreviations: gEBI getElementBvId gEBTN getElementsByTagName

Copyright © 2011 Michael Sorens 2011.04.05 • Version 1.0.2

Download the latest version from Simple-Talk http://bit.ly/gTd5oc.

Indexing (all): XPath and CSS use 1-based indexing; DOM and Selenium's table syntax use 0-based indexing.

Prefixes (all): weath—required unless expression starts with // • dom=required unless expression starts with "document." • cs= always required • dentifier= never required.

Cardinality (Selenium): XPath and CSS may specify a node set or a single node; DOM must specify a single node. When a node set is specified, Selenium returns just the first node.

Content (XPath): Generally should use normalize-space() when operating on display text.

- DOM has limited capability with a simple 'document...' expression; however, arbitrary JavaScript code may be used as shown in this example.
- CSS does not support qualifying elements with the style attribute, as in div[style*="border-width"].
- Selenium uses a special syntax for returning attributes; normal XPath, CSS, and DOM syntax will fail.
- CSS: The CSS2 contains function is not in CSS3; however, Selenium supports the superset of CSS1, 2, and 3.
- DOM: firstChild, lastChild, nextSibling, and previousSibling are problematic with mixed content; they will point to empty text nodes rather than desired elements depending on whitespace in web page