SPATH • CSS • DOM • SEMENDOM 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
	Whole web page	xpath=/html	css=html	document.documentElement	NA
General	Whole web page body	xpath=/html/body	css=body	document.body	NA
General	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>	<mark>//E</mark>	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'] @	NA .	NA
	Element <e> with attribute A containing text 't'</e>	//E[contains(@A,'t')]	css=E[A*='t'] @	NA .	NA
Tag	Element <e> whose attribute A begins with 't'</e>	//E[starts-with(@A, 't')]	css=E[A^='t']	NA .	NA NA
Tag					
	Element <e> whose attribute A ends with 't'</e>	//E[ends-with(@A, 't')]	css=E[A\$='t'] 2	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'] @	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
			,	{Se: document.gEBTN('E')[0]@A }	
ttribute [€]	Attribute A of any element	//*/@A 🔀 {Se: //*@A }	NA {Se: css=*@A }	NA NA	NA
ittibute -	Attribute A1 of element <e> where attribute A2 is 't' exactly</e>	///e/		NA NA	NA NA
			NA {Se: css=E[A2='t']@A1 }		
	Attribute A of element <e> where A contains 't'</e>	//E[contains(@A,'t')]/@A Se: //E[contains(@A,'t')]@A (⟨E	NA {Se: css=E[A*='t']@A }	NA	NA
	Element <e> with id I</e>	//E[@id='l']	css=E#I	NA	NA
Id	Element with id I	//*[@id='l']	CSS=#I	document.gEBI('I')	id=I
-	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
	Element with id X or, failing that, a name X	//*[@id='X' or @name='X']	NA NA	NA	X ◀ OR▶ identifie
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 o-based index v	//*[@name='N'][@value='v']	css=[name=N][value='v']	NA NA	name=N value=v
Lang	Element <e> is explicitly in language L or subcode</e>	/ <mark>/E[@lang='L' or star</mark> ts-with(@lang, concat('L', '-'))]	css=E[lang =L]	NA	NA
&		NA	css=E:lang(L)	NA	NA
Class	Element with a class C	//*[contains(concat('@', @class, '@'), '@C@')]	css=.C	document.getElementsByClassName('C')[0]	NA
	Element <e> with a class C</e>	//E[contains(concat('⊕', @class, '⊕'), '⊕C⊕')]	css=E.C	NA	NA
	Element containing text 't' exactly	//*[.='t']	NA	NA	NA
T	Element <e> containing text 't'</e>	//E[contains(text(),'t')]	css=E:contains('t') •	NA NA	NA
Text	Link element	//a	CSS=a	document.links[0]	NA
0	<a> containing text 't' exactly	//a[.='t']	NA	NA NA	link=t
&					
Link	<a> containing text 't'	//a[contains(text(),'t')]	css=a:contains('t') 4	NA	NA
	<a> with target link 'url'	//a[@href='url']	css=a[href='url']	NA	NA
Parent	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>	//E[last()]	css=E:last-of-type 🖾	document.gEBTN(E)[document.gEBTN(E).length-1]	NA
	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>		css=E:nth-child(2)	NA	NA NA
		//*[2][name()='E']			
	Second-to-last <e> child</e>	//E[last()-1]	css=E:nth-last-of-type(2)	document.gEBTN(E)[document.gEBTN(E).length-2]	
i ai CIIL					NA
	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>		css=E:nth-last-child(2) NA	NA NA	NA NA
&	Element <e1> with only <e2> children Parent of element <e></e></e2></e1>	//*[last()-1][name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E/	css=E:nth-last-child(2)	10000	NA
	Element <e1> with only <e2> children</e2></e1>	//*[last()-1][name()='E']	css=E:nth-last-child(2) NA	NA NA	NA NA
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path</e></e></e2></e1>	//*(last()-1](name()='E') //E1/[EZ and not(*[not(self::E2)])] //E/ //*[@id='l']// /E	css=E:nth-last-child(2) NA NA	NA document.gEBTN('E')[0].parentNode document.gEBI('I')gEBTN('E')[0]	NA NA NA
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path</e></e></e></e2></e1>	//*(last()-1](name()='E') //E1/[E2 and not(*[not(self::E2)])] //E/ //*(@id='l']///E //*(@id='l']//E	css=E:nth-last-child(2)	NA document.gEBTN('E')[0].parentNode	NA NA NA NA
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children</e></e></e></e></e2></e1>	//*[last()-1][name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E1. //*@id='1]///E //*@id='1']/E //E[count(*)=0]	css=E:nth-last-child(2) NA NA (SS=#1 > > > E CSS=#1 E css=E:empty	NA document.gEBTN('E')[0].parentNode document.gEBI('I')gEBTN('E')[0] document.gEBI('I').gEBTN('E')[0] NA	NA NA NA NA NA NA
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child</e></e></e></e></e></e2></e1>	//*[last()-1][name()='E'] //E1/[EZ and not(*[not(self::E2)])] //E/ //*[@id='I']///E //*[@id='I']//E //E[count(*)=0] //E[count(*)=1]	css=E:nth-last-child(2) ISO NA VA CSS=#1>>> E CSS=#1 E CSS=E:empty NA	MA document.gEBTN('E')[0].parentNode document.gEB!('I')gEBTN('E')[0] document.gEB!('I').gEBTN('E')[0] NA NA	NA NA NA NA NA NA
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with a only child Element <e> that is an only child</e></e></e></e></e></e></e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E/ //*[@id='1']////E //*[@id='1']//E //E[count(*)=0] //E[count(*)=1] //E[count(preceding-sibling::*)+count(following-sibling::*)=0]	css=E:nth-last-child(2) NA NA css=#1 > > > E css=#1 E css=E:empty NA css=E:only-child	MA document.gEBTN("E')[0].parentNode document.gEBI("1")gEBTN("E')[0] document.gEBI("1").gEBTN("E')[0] NA NA NA	NA
&	Element <e1> with only <e2> children Parent of element <<>> Descendant <<>> of element with id I using specific path Descendant <<>> of element with id I using unspecified path Element <<>> with on children Element <<>> with on children Element <<>> with an only child Element <>> with at is an only child Element <>> with a children Element <>> with on <<>> siblings</e2></e1>	//*[last()-1][name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E/ //*(@id='T)///E //*(@id='T)/-E //E[count(*]=0] //E[count(p=ceding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1]	css=E:nth-last-child(2) \(\bar{\text{SS}} \) NA (SS=#1 > > > E CSS=#1 = CSS=E:empty NA (SS=E:only-child CSS=E:only-of-type \(\bar{\text{SS}} \)	NA document.gEBTN("E")[0].parentNode document.gEBI("I")gEBTN("E")[0] document.gEBI("I").gEBTN("E")[0] NA NA NA NA NA	NA
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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></e></e></e2></e1>	//*[last()-1][name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E/ //*[@id='I']///E //*[@id='I']//E //E[count(*]-0] //E[count(*)-1] //E[count(./E) = 1] //E[position() mod N = M + 1]	css=E:nth-last-child(2)	MA document.gEBTN['E'][0].parentNode document.gEBI('I')gEBTN['E'][0] document.gEBI('I')gEBTN['E'][0] NA NA NA NA NA NA	NA N
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not (*[not(self::E2)])] //E1. //*[@id='t]///.E //*[@id='t]///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	css=E:nth-last-child(2)	MA document.gEBTN('E')[0].parentNode document.gEBI('I')gEBTN('E')[0] document.gEBI('I')gEBTN('E')[0] NA NA NA NA NA NA	NA N
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with or <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></e></e></e2></e1>	//*[last()-1][name()='E'] //E1/[EZ and not(*[not(self::E2])]) //E/ //*[@id='I']///E //*[@id='I']//E //E[count(*]-0] //E[count(*)-1] //E[count(./E) = 1] //E[position() mod N = M + 1]	css=E:nth-last-child(2)	MA document.gEBTN['E'][0].parentNode document.gEBI('I')gEBTN['E'][0] document.gEBI('I')gEBTN['E'][0] NA NA NA NA NA NA	NA N
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with or <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></e></e></e2></e1>	//*[last()-1][name()='E'] //E1/[E2 and not (*[not(self::E2)])] //E/ //*(@id='I)///E //*(@id='I)/ //E[count(*]=0] //E[count(*]=1] //E[count(p=ceding-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::E1	css=E:nth-last-child(2)	MA document.gEBTN('E')[0].parentNode document.gEBI('I')gEBTN('E')[0] document.gEBI('I')gEBTN('E')[0] NA NA NA NA NA NA	NA N
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with o<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> Element <e1> following sibling <e2> Element <e1> following sibling <e2> Element <e3> following sibling <e2> Element <e3> following sibling <e2></e2></e3></e2></e3></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1][name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E/ //*[@id='']///E //*[@id='']//E //E[count(*]=0] //E[count(*]=0] //E[count(:/E)=1] //E[count(./E)=1] //E[position() mod N = M + 1] //E2/following-sibling::1*	css=E:nth-last-child(2)	MA document.gEBIN('E')[0].parentNode document.gEBI('I')gEBTN('E')[0] document.gEBI('I')gEBTN('E')[0] NA	NA N
&	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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> following sibling <e2> with one intermediary Sibling element immediately following <e></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not (*[not(self::E2)])] //E1. //* [@id='t] / / E //* [@id='t] / / E //* [@id='t] / E //* [@id=t'] / E // [@id=t'] / E //* [@id=t'] / E // [@id=t'	css=E:nth-last-child(2) \(\begin{align*} NA \\ MA \\ \css=#1 \\ \css=\text{E} : \text{E} : \text{E} : \css=\text{E} : \text{E} : \t	MA document.gEBTN('E')[0].parentNode document.gEBI('I')gEBTN('E')[0] document.gEBI('I')gEBTN('E')[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> buth an only child Element <e> with an only child Element <e> with on <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> preceding some sibling <e2></e2></e1></e2></e1></e></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1][name()='E'] //*[lE2 and not (*[not(self::E2)])] //E[//*(@id='I)///E //*(@id='I)/ //E[count(*)=0] //E[count(*)=0] //E[count(p=ceding-sibling::*)+count(following-sibling::*)=0] //E[count(/E) = 1] //E[position() mod N = M + 1] //E2/following-sibling::E1 //E2/following-sibling::*[1][name()='E1'] //E2/following-sibling::*[2][name()='E1'] //E2/foceding-sibling::* //E2/proceding-sibling::E1	css=E:nth-last-child(2) \(\bar{S} \) NA	MA document_gEBTN('E')[0].parentNode document_gEBI('I')gEBTN('E')[0] document_gEBI('I')gEBTN('E')[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with on <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> Element <e1> following sibling <e2> Element <e1> prollowing sibling <e2> Element <e1> minmediately following sibling <e2> Element <e1> immediately following <e> Element <e1> immediately following sibling <e2></e2></e1></e></e1></e></e1></e></e1></e></e1></e></e1></e></e1></e></e1></e></e1></e></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E1. //*[wid='1']//E //*[wid='1']//E //E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E2/following-sibling::E1 //E2/following-sibling::E1 //E2/following-sibling::f1[name()='E1'] //E2/following-sibling::f2[]name()='E1'] //E1/following-sibling::f2[]name()='E1'] //E2/following-sibling::f1[]name()='E1']	css=E:nth-last-child(2)	MA document_gEBTN['E'][0].parentNode document_gEBI('I')gEBTN['E'][0] document_gEBI('I')gEBTN['E'][0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <<>> Descendant <<>> of element with id I using specific path Descendant <<>> of element with id I using unspecified path Descendant <<>> of element with id I using unspecified path Element <<>> with on children Element <<>> with an only child Element <<>> with a children Element <<>> with or children Element <<<>> with or children Element <<>> following some sibling <<>> Element <<>> with or children Element <<>> with or children Sibling element immediately following <<>> Element < Element < Element < Element < Element < Element < Element preceding some sibling < Element < Element < Element < Element preceding some sibling < Element < Element with one intermediary</e2></e1>	//*[last()-1[name()='E'] //*[le2 and not (*[not(self::E2)])] //E/. //*(@id='I)///E //*(@id='I']/.E //*[count(*)-0] //E[count(*)-1] //E[css=E:nth-last-child(2) \(\infty \) NA	MA document.gEBTN('E')[0].parentNode document.gEBI('I')gEBTN('E')[0] document.gEBI('I')gEBTN('E')[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with or <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 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> 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> with one intermediary Sibling element immediately preceding <e> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding <e></e></e2></e1></e></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	/* last()-1 name()='E'] /E1/[E2 and not (*[not(self::E2)])] /EL. /*(@id='I]///E /*(@id='I]///E /*(@id='I]///E /*[count(*]=0] /E[count(*]=0] /E[count(p=ceding-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::*[1][name()='E1'] /E2/preceding-sibling::E1] /E2/preceding-sibling::E1] /E2/preceding-sibling::E1] /E2/preceding-sibling::*[2][name()='E1'] /E2/preceding-sibling::*[2][name()='E1'] /E2/preceding-sibling::*[2][name()='E1']	css=E:nth-last-child(2) \(\bar{S} \) NA	MA document_gEBTN("E")[0].parentNode document_gEBI("I")gEBTN("E")[0] document_gEBI("I")gEBTN("E")[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <<>> Descendant <<>> of element with id I using specific path Descendant <<>> of element with id I using unspecified path Descendant <<>> of element with id I using unspecified path Element <<>> with on children Element <<>> with an only child Element <<>> with a children Element <<>> with or children Element <<<>> with or children Element <<>> following some sibling <<>> Element <<>> with or children Element <<>> with or children Sibling element immediately following <<>> Element < Element < Element < Element < Element < Element < Element preceding some sibling < Element < Element < Element < Element preceding some sibling < Element < Element with one intermediary</e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E1. //* @id='1'] / / . /E //* @id='1'] / / . /E //* [count(*)=0] //E[count(*)=0] //E[count(r)=0] //E[count(r)=0] //E[count(r)=0] //E[count(r)=0] //E[count(r)=0] //E[count(r)=0] //E[count(r)=0] //E[count(r)=0] //E2/following-sibling::1[][name()='E1'] //E2/following-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1']	css=E:nth-last-child(2) \(\begin{align*} NA \\ NA \\ SS=#I > > > E \\ Css=#I = E \\ Css=E:empty \\ NA \\ Css=#i:only-child \\ Css=E:only-of-type \begin{align*} Css=E:empty \\ NA \\ Css=E:enth-child(Nn + M) \\ Css=E:enth-child(Nn + M) \\ Css=E + E I \\ Css=E + * E I \\ Css=E + * E I \\ Css=E + * E I \\ NA \\	MA document.gEBTN("E")[0].parentNode document.gEB!("I")gEBTN("E")[0] document.gEB!("I")gEBTN("E")[0] NA NA NA NA NA NA NA NA A document.gEBTN("E")[0].nextSibling NA NA NA document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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> following sibling <e2> with no <e> siblings 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 some 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)</e></e2></e1></e2></e1></e2></e1></e2></e1></e></e2></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1][name()='E'] //*[lE2 and not (*[not(self::E2)])] //E1. //*[wid='1]///E //*[wid='1]/ //*[wid='1]/ //*[count(*)-0] //E[count(*)-0] //E[count(preceding-sibling::*)+count(following-sibling::*)-0] //E[count(r-)-1] //E[count(r-)-1] //E[count(r-)-1] //E2/following-sibling::1] //E2/following-sibling::1] //E2/following-sibling::1[][name()='E1'] //E2/following-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[] //*[wid='TestTable]//t13]	css=E:nth-last-child(2) \(\bar{S} \) NA	MA document_gEBTN("E")[0].parentNode document_gEBI("I")gEBTN("E")[0] document_gEBI("I")gEBTN("E")[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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> following sibling <e2> with no <e> siblings 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 some 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)</e></e2></e1></e2></e1></e2></e1></e2></e1></e></e2></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1][name()='E'] //*[lE2 and not (*[not(self::E2)])] //E1. //*[wid='1]///E //*[wid='1]/ //*[wid='1]/ //*[count(*)-0] //E[count(*)-0] //E[count(preceding-sibling::*)+count(following-sibling::*)-0] //E[count(r-)-1] //E[count(r-)-1] //E[count(r-)-1] //E2/following-sibling::1] //E2/following-sibling::1] //E2/following-sibling::1[][name()='E1'] //E2/following-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[][name()='E1'] //E2/preceding-sibling::1[] //*[wid='TestTable]//t13]	css=E:nth-last-child(2) \(\begin{align*} NA \\ NA \\ SS=#I > > > E \\ Css=#I = E \\ Css=E:empty \\ NA \\ Css=#i:only-child \\ Css=E:only-of-type \begin{align*} Css=E:empty \\ NA \\ Css=E:enth-child(Nn + M) \\ Css=E:enth-child(Nn + M) \\ Css=E + E I \\ Css=E + * E I \\ Css=E + * E I \\ Css=E + * E I \\ NA \\	MA document.gEBTN("E")[0].parentNode document.gEB!("I")gEBTN("E")[0] document.gEB!("I")gEBTN("E")[0] NA NA NA NA NA NA NA NA A document.gEBTN("E")[0].nextSibling NA NA NA document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling document.gEBTN("E")[0].previousSibling	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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 some sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately following <e3 <e1="" element=""> immediately preceding sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> immediately preceding sibling <e2> Cell by row and column (e.g. 3rd row, 2nd column) Cell immediately following cell containing 't' exactly</e2></e1></e2></e1></e2></e3></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	/*[last()-1[name()='E] /E1/[E2 and not(*[not(self::E2)])] /E1/ /*[@id='1]///E /*[@id='1]/ /*[@id='1]/ /E[count(*]=0] /E[count(*]=1] /E[count(*]=1] /E[count(,E]=1] /E[count(,E]=1] /E[count(,E]=1] /E2/following-sibling::*1=[name()='E1'] /E2/following-sibling::*1[]name()='E1'] /E2/following-sibling::*1[]name()='E1'] /E2/following-sibling::*1[]name()='E1'] /E2/following-sibling::*1[]name()='E1'] /E2/preceding-sibling::*1[]name()='E1'] /E2/preceding-sibling::*1[]name()='E1'] /E2/preceding-sibling::*1[]name()='E1'] /E2/preceding-sibling::*1[]name()='E1'] /E2/preceding-sibling::*1[] /*[@id=TestTable']/tr(3]/td[2] Se: /*(@id=TestTable'].2.1	css=E:nth-last-child(2)	MA document_gEBTN("E)[0].parentNode document_gEBI("I")gEBTN("E)[0] document_gEBI("I")gEBTN("E)[0] MA NA NA NA NA NA NA NA NA NA	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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> following sibling <e2> with one intermediary Sibling element immediately following <e> Element <e1> minediately following <e> Element <e1> immediately following <e> Element <e1> minediately preceding sibling <e2> Element <e1> minediately preceding sibling <e2> Element <e3 <e2="" preceding="" sibling=""> with one intermediary Sibling element immediately preceding sibling <e2> Element <e3 <e2="" preceding="" sibling=""> with one intermediary Sibling element immediately preceding <c 't'="" 't'<="" cell="" containing="" ell="" exactly="" following="" immediately="" td=""><td>//*[last()-1[name()='E'] //E1/[E2 and not (*[not(self::E2)])] //E1. //* (wid='1'] / . / . / . / . / . / . /</td><td> css=E:nth-last-child(2) S</td><td>MA document.gEBIN("E")[0].parentNode document.gEBI("I")gEBTN("E")[0] document.gEBI("I")gEBTN("E")[0] NA NA NA NA NA NA NA document.gEBTN("E")[0].nextSibling NA NA NA document.gEBTN("E")[0].nextSibling NA NA NA NA NA NA NA NA NA N</td><td> NA</td></c></e3></e2></e3></e2></e1></e2></e1></e></e1></e></e1></e></e1></e></e1></e></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not (*[not(self::E2)])] //E1. //* (wid='1'] / . / . / . / . / . / . /	css=E:nth-last-child(2) S	MA document.gEBIN("E")[0].parentNode document.gEBI("I")gEBTN("E")[0] document.gEBI("I")gEBTN("E")[0] NA NA NA NA NA NA NA document.gEBTN("E")[0].nextSibling NA NA NA document.gEBTN("E")[0].nextSibling NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with no (E) sibling Every Nth element starting with the (M+1)th Element <e1> inlowing some sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately 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' exactly User interface element <e> that is disabled</e></e2></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></e></e2></e1>	/* last()-1 name()='E'] /*E1/E2 and not (*[not(self::E2)])] /EL. /*(@id='I]///E /*(@id='I]/.E /*[Count(*)-I] /E[count(*)-I] /E[count(preceding-sibling::*)+count(following-sibling::*)+0] /E[count(preceding-sibling::*)+count(following-sibling::*)+0] /E[count(/E) = 1] /E2/following-sibling::1] /E2/following-sibling::1] /E2/following-sibling::1[][name()='E1'] /E2/following-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[] /*[@id='TestTable]//t1] /t0[preceding-sibling::d-I'] /td[preceding-sibling::d-I'] /td[preceding-sibling::d-I'] /td[preceding-sibling::d-I'] /td[preceding-sibling::d-I'] /td[preceding-sibling::d-I'] /td[preceding-sibling::d-I']	css=E:nth-last-child(2) S	MA document_gEBTN("E")[0].parentNode document_gEBI("I")gEBTN("E")[0] document_gEBI("I")gEBTN("E")[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding 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 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</e></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E1. //* [@id='T]///E //* [@id='T]/E //E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E2/following-sibling::f1][name()='E1'] //E2/following-sibling::f2][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[] //E3/preceding-sibling::f1[] //E3/preceding-sibli	css=E:nth-last-child(2)	MA document.gEBTN("E)[0].parentNode document.gEBI("I")gEBTN("E)[0] document.gEBI("I")gEBTN("E)[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <<>> Descendant <<>> of element with id I using specific path Descendant <<>> of element with id I using unspecified path Element <<>> with on children Element <<>> with on children Element <<>> with an only child Element <<>> with a children Element <<>> with a children Element <<>> with on children Element <<>> with on children Element <<>> with on children Element <<>> with one children Element < Element < *E1> following some sibling < *E2> Element < Element < *E1> following some sibling <<<>> Element < *E2> Element < Element < Sibling element immediately following sibling <<<>> Element < Element < El> preceding some sibling < Element < El> preceding some sibling < Element < El> preceding sibling < Element < Sibling element immediately preceding sibling < Element < 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 <<>> that is disabled User interface element that is enabled Checkbox (or radio button) that is checked</e2></e1>	/*[last()-1[name()='E'] /E1/[E2 and not (*[not(self::E2)])] /E1. /*[wid='t]///E /*[wid='t]///E /*[wid='t]//E /*[count(*]-0] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E2/following-sibling::1] /E2/following-sibling::1] /E2/following-sibling::1[][name()='E1'] /E2/following-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[] /E2/preced	css=E:nth-last-child(2) \(\begin{align*} NA \\ MA \\ \css=#1 \> \> \> E \\ \css=E:empty \\ NA \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:anth-child(Nn + M) \\ \css=E2 \tau E1 \\ \css=E2 + F1 \\ \css=E3 + F2 \\ \css=E3 + F3 \\ \css=E3 \\ \cs	MA document.gEBIN("E")[0].parentNode document.gEBI("I")gEBTN("E")[0] document.gEBI("I")gEBTN("E")[0] NA NA NA NA NA NA NA NA document.gEBTN("E")[0].nextSibling NA NA NA NA NA NA NA NA NA N	NA
& Child Sibling	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path 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> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately preceding 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 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</e></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e2></e1>	//*[last()-1[name()='E'] //E1/[E2 and not(*[not(self::E2)])] //E1. //* [@id='T]///E //* [@id='T]/E //E[count(*)=0] //E[count(*)=1] //E[count(receding-sibling::*)+count(following-sibling::*)=0] //E[count(./E) = 1] //E2/following-sibling::f1][name()='E1'] //E2/following-sibling::f2][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[] //E3/preceding-sibling::f1[] //E3/preceding-sibli	css=E:nth-last-child(2)	MA document.gEBTN("E)[0].parentNode document.gEBI("I")gEBTN("E)[0] document.gEBI("I")gEBTN("E)[0] NA NA NA NA NA NA NA NA NA N	NA
& Child	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with an only child Element <e> with one <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 <e2> with one intermediary Sibling element immediately following <e> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately following cell containing 't' wasctly Cell immediately following cell containing 't' exactly Cell immediately following cell containing 't' wasctly 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></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e></e2></e1>	/*[last()-1[name()='E'] /E1/[E2 and not (*[not(self::E2)])] /E1. /*[wid='t]///E /*[wid='t]///E /*[wid='t]//E /*[count(*]-0] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E[count(*]-1] /E2/following-sibling::1] /E2/following-sibling::1] /E2/following-sibling::1[][name()='E1'] /E2/following-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[][name()='E1'] /E2/preceding-sibling::1[] /E2/preced	css=E:nth-last-child(2) \(\begin{align*} NA \\ MA \\ \css=#1 \> \> \> E \\ \css=E:empty \\ NA \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:conly-child \\ \css=E:anth-child(Nn + M) \\ \css=E2 \tau E1 \\ \css=E2 + F1 \\ \css=E3 + F2 \\ \css=E3 + F3 \\ \css=E3 \\ \cs	MA document.gEBIN("E")[0].parentNode document.gEBI("I")gEBTN("E")[0] document.gEBI("I")gEBTN("E")[0] NA NA NA NA NA NA NA NA document.gEBTN("E")[0].nextSibling NA NA NA NA NA NA NA NA NA N	NA
& Child Sibling Table Cell	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child 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> following 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> 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 Element being designated by a pointing device</e></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e></e2></e1>	/*[last()-1][name()='E'] /E1/[E2 and not (*[not(self::E2)])] /EL. /*(@id='I]///E /*(@id='I]/.E /*[Count(*)-0] /E[Count(*)-0] /E[Count(*)-1] /E[Count(preceding-sibling::*)+count(following-sibling::*)+0] /E[Count(/E) = 1] /E2/following-sibling::*1] /E2/following-sibling::*1] /E2/following-sibling::*1] /E2/following-sibling::*1] /E2/following-sibling::*1] /E2/preceding-sibling::*1] /E2/preceding-sibling::*1] /E2/preceding-sibling::*1] /E2/preceding-sibling::*1] /E2/preceding-sibling::*2][name()='E1'] /E2/preceding-sibling::*2] /*E2/preceding-sibling::*2] /*E2/preceding-sibling::*1] /*Epreceding-sibling::*1]	css=E:nth-last-child(2)	MA document_gEBTN("E)[0].parentNode document_gEBI("I")gEBTN("E)[0] document_gEBI("I")gEBTN("E)[0] MA NA NA NA NA NA NA NA NA NA	NA
& Child Sibling	Element <e1> with only <e2> children Parent of element <e> Descendant <e> of element with id I using specific path Descendant <e> of element with id I using unspecified path Element <e> with no children Element <e> with an only child Element <e> with an only child Element <e> with an only child Element <e> with one <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 <e2> with one intermediary Sibling element immediately following <e> Element <e1> preceding some sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> immediately following sibling <e2> Element <e1> preceding some sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately preceding sibling <e2> Element <e1> preceding sibling <e2> with one intermediary Sibling element immediately following cell containing 't' wasctly Cell immediately following cell containing 't' exactly Cell immediately following cell containing 't' wasctly 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></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e2></e1></e></e2></e1></e2></e1></e2></e1></e></e></e></e></e></e></e></e></e></e2></e1>	/*[last()-1[name()='E'] /E1/[E2 and not(*[not(self::E2)])] /E1. /*[w]de'"]///E /*[w]de'"]//E //E[count(*)=0] /E[count(preceding-sibling::*)+count(following-sibling::*)=0] /E[count(preceding-sibling::*)+count(following-sibling::*)=0] /E[count(./E) = 1] /E2/following-sibling::f1] /E2/following-sibling::f1] /E2/following-sibling::f1] /E2/following-sibling::f1] /E2/following-sibling::f1] /E2/following-sibling::f1] /E2/preceding-sibling::f1] /E2/preceding-sibling::f2] /E2/preceding-sibling::f2] /E2/preceding-sibling::f2 /E2/preceding-	css=E:nth-last-child(2)	MA document_gEBTN("E')[0].parentNode document_gEBI("I")gEBTN("E')[0] document_gEBI("I")gEBTN("E')[0] MA NA NA NA NA NA NA NA MA documentgEBTN("E')[0].nextSibling MA NA NA NA NA NA NA NA NA NA	NA

LEGEND				
XPath				
CSS				
DOM				
Selenium				
{Se:} Selenium-only variation				
Not supported by Selenium				
Space character				
expression CSS3 or XPath 2.0				
DOM abbreviations: gEBI getElementById 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): path required unless expression starts with // • dom required unless expression starts with "document." • css= 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