

XPATH 빠른 참조

축, 조건자, 함수, 연산자, 노드 선택

구문	
경로 표현식	
/	루트 노드 (절대 경로 시작)
/bookstore/book	직접 자식 선택
//book	어디서든 모든 book 노드 선택
.	현재 컨텍스트 노드
..	현재 노드의 부모
@lang	lang이라는 이름의 속성
node()	모든 타입의 노드
*	모든 요소 노드
@*	모든 속성

기본 예제	
/html/body/div	# absolute path to <div>
//input[@type='text']	# all text inputs
//div[@class='main']*	# children of div.main
//a/@href	# all href attributes

경로 결합	
//book/title //book/price	# union of two paths
//h1 //h2 //h3	# multiple element types

축	
축 방향	
child::	직접 자식 (기본 축)
parent::	직접 부모
ancestor::	루트까지 모든 조상
ancestor-or-self::	조상 + 현재 노드
descendant::	모든 자손
descendant-or-self::	자손 + 현재 노드
following::	문서에서 현재 이후의 모든 노드
following-sibling::	현재 이후의 형제
preceding::	문서에서 현재 이전의 모든 노드
preceding-sibling::	현재 이전의 형제
self::	현재 노드만
attribute::	현재 노드의 속성
namespace::	네임스페이스 노드

축 예제	
//div/child::p	# <p> children of <div>
//td/parent::tr	# <tr> parent of <td>
//h2/preceding-sibling::p	# <p> after <h2>
//li/ancestor::ul	# containing

조건자	
조건자로 필터링	
//book[1]	# first book element
//book[last()]	# last book element
//book[position() < 3]	# first two books
//book[@lang='en']	# books with lang="en"
//book[price > 30]	# books with price > 30

조건자 패턴	
[n]	n번째 위치의 요소 (1부터 시작)
[last()]	마지막 요소
[last()-1]	끝에서 두 번째
[@attr]	속성 있음
[@attr='val']	속성이 값과 같음
[element]	자식 요소 있음
[element='text']	자식 요소가 텍스트를 포함
[not(@attr)]	속성 없음

연결된 조건자	
//div[@class='list']/a[1]	# first <a> in div.list
//input[@type='text'][@name='q']	# AND condition
//book[price>30][@lang='en']	# multiple conditions

함수	
문자열 함수	
contains(s, sub)	s가 sub를 포함하면 참
starts-with(s, pre)	s가 pre로 시작하면 참
string-length(s)	문자열 길이
normalize-space(s)	문자열 백제거 및 내부 공백 정리
concat(a, b, ...)	문자열 결합
substring(s, pos, len)	문자열 부분 추출 (1부터 시작)
translate(s, from, to)	문자열 치환

숫자 함수	
sum(node-set)	숫자 값의 합
count(node-set)	노드 수
floor(n)	내림
ceiling(n)	올림
round(n)	가장 가까운 정수로 반올림
number(val)	숫자로 변환

함수 예제	
//div[contains(@class, 'active')]	
//a[starts-with(@href, 'https ')]	
//p[string-length(text()) > 100]	
//ul[count(li) > 5]	

연산자	
비교 연산자	
=	같음
!=	같지 않음
<	보다 작음
<=	보다 작거나 같음
>	보다 큼
>=	보다 크거나 같음
논리 및 산술	
and	논리 AND
or	논리 OR

not()	논리 NOT (함수)
+	덧셈
-	빼셈
*	곱셈
div	나눗셈
mod	나머지
	노드 집합의 합집합

연산자 예제	
//book[price > 20 and price < 50]	
//item[@type='a' or @type='b']	
//span[not(contains(@class, 'hidden'))]	

노드 테스트	
노드 타입	
node()	모든 노드 (요소, 텍스트, 주석, PI)
text()	텍스트 노드만
comment()	주석 노드만
processing-instruction()	처리 명령 노드
*	모든 요소 노드
@*	모든 속성 노드
element-name	특정 이름의 요소

노드 테스트 예제	
//p/text()	# text content of <p>
//div/comment()	# comments inside <div>
//body/node()	# all child nodes of <body>
//div*	# all element children of <div>

불리언 함수	
true()	불리언 참
false()	불리언 거짓
boolean(expr)	불리언으로 변환
not(expr)	불리언 반전
lang(code)	노드의 언어가 일치하면 참

약어	
약식 vs 전체 형식	
(없음) child::(기본 축)	
@ attribute::	
// /descendant-or-self::node()	
.	self::node()
..	parent::node()
[n]	position()=n

약어 예제	
# These pairs are equivalent:	
child::div	→ div
attribute::href	→ @href
/descendant-or-self::node()/p	→ //p
self::node()	→ .
parent::node()	→ ..

일반적인 약식 패턴	
//div[@id='main']	# div with id="main"
//table/td	# all <td> in any <table>
../sibling	# sibling via parent
//span	# span descendants of context

자주 쓰는 패턴	
웹스크래핑 / 테스트	
//input[@name='username']	# form input by name
//button[text()='Submit']	# button by text
//div[contains(@class, 'error')]	# element by partial class
//a[contains(@href, 'login')]	# link by partial href

조건부 선택	
//div[@class='item'][./span[@class='price']]	
//tr[td[1]='Active']	# row where 1st cell = Active
//*[contains(text(), 'Warning')]	# any element with text

Python에서 XPath (lxml)	
from lxml import html	
tree = html.fromstring(page_content)	
links = tree.xpath('//a/@href')	
titles = tree.xpath('//h2/text()')	

Selenium에서 XPath	
driver.find_element(By.XPATH, "//input[@id='search']")	
driver.find_elements(By.XPATH, "//li[@class='result']")	