

# Tham Khảo Nhanh XPath

Axes, predicates, hàm, toán tử, chọn nodes

## Cú pháp

### Biểu thức đường dẫn

/	Node gốc (bắt đầu đường dẫn tuyệt đối)
/bookstore/book	Chọn con trực tiếp
//book	Chọn tất cả nodes book ở bất kỳ đâu
.	Node ngữ cảnh hiện tại
..	Cha của node hiện tại
@lang	Thuộc tính tên lang
node()	Bất kỳ node nào thuộc mọi kiểu
*	Bất kỳ node phân tử nào
@*	Bất kỳ thuộc tính nào

### Ví dụ cơ bản

/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

### Kết hợp đường dẫn

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

## Axes

### Hướng Axes

child::	Con trực tiếp (axis mặc định)
parent::	Cha trực tiếp
ancestor::	Tất cả tổ tiên đến gốc
ancestor-or-self::	Tổ tiên + node hiện tại
descendant::	Tất cả con cháu
descendant-or-self::	Con cháu + node hiện tại
following::	Tất cả nodes sau node hiện tại trong tài liệu
following-sibling::	Anh em phía sau node hiện tại
preceding::	Tất cả nodes trước node hiện tại trong tài liệu
preceding-sibling::	Anh em phía trước node hiện tại
self::	Chỉ node hiện tại
attribute::	Thuộc tính của node hiện tại
namespace::	Nodes namespace

### Ví dụ Axes

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

## Predicates

### Lọc bằng Predicates

//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

### Các mẫu Predicate

[n]	Phần tử ở vị trí n (bắt đầu từ 1)
[last()]	Phần tử cuối
[last()-1]	Thứ hai từ cuối
[@attr]	Có thuộc tính
[@attr='val']	Thuộc tính bằng giá trị
[element]	Có phần tử con
[element='text']	Phần tử con chứa văn bản
[not(@attr)]	Không có thuộc tính

## Predicates chuỗi

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

## Hàm

### Hàm String

contains(s, sub)	True nếu s chứa sub
starts-with(s, pre)	True nếu s bắt đầu bằng pre
string-length(s)	Độ dài chuỗi
normalize-space(s)	Xóa khoảng trắng đầu cuối và nén khoảng trắng nội bộ
concat(a, b, ...)	Nối các chuỗi
substring(s, pos, len)	Trích chuỗi con (bắt đầu từ 1)
translate(s, from, to)	Thay thế từng ký tự

### Hàm số

sum(node-set)	Tổng các giá trị số
count(node-set)	Số lượng nodes
floor(n)	Làm tròn xuống
ceiling(n)	Làm tròn lên
round(n)	Làm tròn đến số nguyên gần nhất
number(val)	Chuyển đổi sang số

### Ví dụ hàm

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

## Toán tử

### Toán tử so sánh

=	Bằng
!=	Không bằng
<	Nhỏ hơn
<=	Nhỏ hơn hoặc bằng
>	Lớn hơn
>=	Lớn hơn hoặc bằng

### Logic & Số học

and	Logic AND
or	Logic OR
not()	Logic NOT (hàm)
+	Cộng
-	Trừ
*	Nhân
div	Chia
mod	Lấy dư
	Hợp của tập node

### Ví dụ toán tử

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

## Node Tests

### Các loại Node

node()	Bất kỳ node nào (phần tử, văn bản, chú thích, PI)
text()	Chỉ node văn bản
comment()	Chỉ node chú thích
processing-instruction()	Node chỉ thị xử lý
*	Bất kỳ node phân tử nào
@*	Bất kỳ node thuộc tính nào
element-name	Phần tử có tên cụ thể

### Ví dụ Node Test

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

### Hàm Boolean

true()	Boolean true
false()	Boolean false
boolean(expr)	Chuyển đổi sang boolean
not(expr)	Đảo ngược boolean
lang(code)	True nếu ngôn ngữ của node khớp

## Viết tắt

### Dạng ngắn vs Dạng đầy đủ

(không có)	child:: (axis mặc định)
@	attribute::
//	/descendant-or-self::node()/
.	self::node()
..	parent::node()
[n]	[position()=n]

### Ví dụ viết tắt

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

### Các mẫu viết tắt phổ biến

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

### Các mẫu thường dùng

#### Web Scraping / Testing

//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

#### Chọn có điều kiện

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

# Tham Khảo Nhanh XPath

---

## XPath trong Python (lxml)

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

## XPath trong Selenium

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