

Referência Rápida de XPath

Eixos, predicados, funções, operadores, seleção de nós

Sintaxe

Expressões de Caminho

/	Nó raiz (início de caminho absoluto)
/bookstore/book	Seleção de filho direto
//book	Selecionar todos os nós book em qualquer lugar
.	Nó de contexto atual
..	Pai do nó atual
@lang	Atributo chamado lang
node()	Qualquer nó de qualquer tipo
*	Qualquer nó elemento
@*	Qualquer atributo

Exemplos Básicos

```
/html/body/div # caminho absoluto para <div>
//input[@type='text'] # todos os inputs de texto
//div[@class='main']/* # filhos de div.main
//a/@href # todos os atributos href
```

Combinando Caminhos

```
//book/title | //book/price # união de dois caminhos
//h1 | //h2 | //h3 # múltiplos tipos de elemento
```

Eixos

Direções de Eixo

child::	Filhos diretos (eixo padrão)
parent::	Pai direto
ancestor::	Todos os ancestrais até a raiz
ancestor-or-self::	Ancestrais + nó atual
descendant::	Todos os descendentes
descendant-or-self::	Descendentes + nó atual
following::	Todos os nós após o atual no documento
following-sibling::	Irmãos após o atual
preceding::	Todos os nós antes do atual no documento
preceding-sibling::	Irmãos antes do atual
self::	Apenas o nó atual
attribute::	Atributos do nó atual
namespace::	Nós de namespace

Exemplos de Eixo

```
//div/child::p # filhos <p> de <div>
//td/parent::tr # pai <tr> de <td>
//h2/following-sibling::p # <p> após <h2>
//li/ancestor::ul # <ul> contendo <li>
```

Predicados

Filtrando com Predicados

```
//book[1] # primeiro elemento book
//book[last()] # último elemento book
//book[position() < 3] # primeiros dois books
//book[@lang='en'] # books com lang="en"
//book[price > 30] # books com preço > 30
```

Padrões de Predicado

[n]	Elemento na posição n (base 1)
[last()]	Último elemento
[last()-1]	Penúltimo
[@attr]	Tem atributo
[@attr='val']	Atributo igual ao valor
[element]	Tem elemento filho
[element='text']	Elemento filho contém texto
[not(@attr)]	Não tem atributo

Predicados Encadeados

```
//div[@class='list']//a[1] # primeiro <a> em div.list
//input[@type='text'][@name='q'] # condição AND
//book[price>30][@lang='en'] # múltiplas condições
```

Funções

Funções de String

contains(s, sub)	Verdadeiro se s contém sub
starts-with(s, pre)	Verdadeiro se s começa com pre
string-length(s)	Comprimento da string
normalize-space(s)	Remover espaços extras e colapsar
concat(a, b, ...)	Concatenar strings
substring(s, pos, len)	Extrair substring (base 1)
translate(s, from, to)	Substituição caractere por caractere

Funções Numéricas

sum(node-set)	Soma de valores numéricos
count(node-set)	Número de nós
floor(n)	Arredondar para baixo
ceiling(n)	Arredondar para cima
round(n)	Arredondar para o inteiro mais próximo
number(val)	Converter para número

Exemplos de Funções

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

Operadores

Operadores de Comparação

=	Igual
!=	Diferente
<	Menor que
<=	Menor ou igual
>	Maior que
>=	Maior ou igual

Lógicos e Aritméticos

and	AND lógico
or	OR lógico
not()	NOT lógico (função)
+	Adição
-	Subtração
*	Multiplicação
div	Divisão
mod	Módulo
	União de conjuntos de nós

Exemplos de Operadores

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

Testes de Nó

Tipos de Nó

node()	Qualquer nó (elemento, texto, comentário, PI)
text()	Apenas nó de texto
comment()	Apenas nó de comentário
processing-instruction()	Nó de instrução de processamento
*	Qualquer nó elemento
@*	Qualquer nó atributo
element-name	Elemento com nome específico

Exemplos de Teste de Nó

```
//p/text() # conteúdo de texto de <p>
//div/comment() # comentários dentro de <div>
//body/node() # todos os nós filhos de <body>
//div/* # todos os filhos elemento de <div>
```

Funções Booleanas

true()	Booleano verdadeiro
false()	Booleano falso
boolean(expr)	Converter para booleano
not(expr)	Negar booleano
lang(code)	Verdadeiro se o idioma do nó corresponde

Abreviações

Forma Curta vs Longa

(nenhuma)	child:: (eixo padrão)
@	attribute::
//	/descendant-or-self::node()/
.	self::node()
..	parent::node()
[n]	[position()=n]

Exemplos Abreviados

```
# Estes pares são equivalentes:
child::div → div
attribute::href → @href
/descendant-or-self::node()/p → //p
self::node() → .
parent::node() → ..
```

Padrões Abreviados Comuns

```
//div[@id='main'] # div com id="main"
//table//td # todos os <td> em qualquer <table>
../sibling # irmão via pai
//span # descendentes span do contexto
```

Padrões Comuns

Web Scraping / Testes

```
//input[@name='username'] # input de formulário por nome
//button[text()='Submit'] # botão por texto
//div[contains(@class, 'error')] # elemento por classe parcial
//a[contains(@href, 'login')] # link por href parcial
```

Seleção Condicional

```
//div[@class='item'][./span[@class='price']]
//tr[td[1]='Active'] # linha onde 1ª célula = Active
//*[contains(text(), 'Warning')] # qualquer elemento com texto
```

Referência Rápida de XPath

XPath em Python (lxml)

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

XPath no Selenium

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