

Lý Thuyết Cơ Bản Về XPath - Hướng Dẫn Chi Tiết

XPath (**XML Path Language**) là ngôn ngữ được sử dụng để xác định vị trí của phần tử trong tài liệu XML hoặc HTML. Nó thường được dùng trong kiểm thử tự động (Selenium, Appium), Web Scraping và xử lý dữ liệu.

1. Cấu Trúc Cơ Bản Của XPath

- ♦ Chọn phần tử theo tên thẻ

```
//tagname
```

 Ví dụ:

```
//div → Tìm tất cả các thẻ `<div>` trong trang  
//button → Tìm tất cả thẻ `<button>` trong trang
```

- ♦ Chọn phần tử bằng ID hoặc Class

 Chọn phần tử có **ID** cụ thể:

xpath

```
//*[@id='myID']
```

 Chọn phần tử có **Class** cụ thể:

```
//*[@class='myClass']
```

Ví dụ thực tế:

```
//div[@id='main-container'] → Chọn `<div>` có `id="main-container"`
```

2. Xác Định Phần Tử Bằng Attribute (Thuộc Tính)

- ♦ Sử dụng dấu @ để lọc phần tử theo thuộc tính

```
//tagname[@attribute='value']
```

 Ví dụ:

```
//input[@type='text'] → Chọn thẻ `<input>` có `type="text"`  
//a[@href='https://example.com'] → Chọn thẻ `<a>` có  
`href="https://example.com"`
```

- ♦ Chọn phần tử với nhiều thuộc tính

```
//input[@type='text' and @name='username']
```

 Lọc `<input>` có `type="text"` và `name="username"`.

3. Chọn Phần Tử Theo Văn Bản (Text)

- ♦ Chọn phần tử có nội dung chính xác

```
//tagname[text()='Exact Text']
```

📌 Ví dụ:

```
//button[text()='Login'] → Chọn nút `<button>` có nội dung chính xác là "Login"
```

- ♦ Chọn phần tử có chứa một đoạn văn bản (Không cần chính xác hoàn toàn)

```
//tagname[contains(text(), 'Partial Text')]
```

📌 Ví dụ:

```
//p[contains(text(), 'Welcome')] → Chọn đoạn văn `<p>` chứa chữ "Welcome"
```

- ♦ Kết hợp `normalize-space()` để loại bỏ khoảng trắng thừa

```
//button[normalize-space()='Submit']
```

📌 Dùng khi trang web có khoảng trắng thừa trong HTML.

🎯 4. Sử Dụng Toán Tử **AND**, **OR**, **NOT**

- ♦ Sử dụng **AND** để lọc phần tử có nhiều điều kiện

```
//input[@type='text' and @placeholder='Username']
```

📌 Chọn `<input>` có `type="text"` và `placeholder="Username"`.

- ♦ Sử dụng **OR** để chọn một trong nhiều điều kiện

```
//button[@id='login' or @class='btn-primary']
```

📌 Chọn `<button>` có `id="login"` hoặc `class="btn-primary"`.

- ♦ Sử dụng **NOT** để loại bỏ phần tử

```
//input[not(@disabled)]
```

📌 Chọn `<input>` không có thuộc tính `disabled`.

🎯 5. Xác Định Cấu Trúc Cha - Con - Tổ Tiên

- ♦ Chọn phần tử con (Direct Child)

```
//div[@id='parent']/child::tagname
```

📌 Ví dụ:

```
//div[@id='menu']/ul → Chọn <ul> là con trực tiếp của <div id="menu">
```

- ♦ Chọn phần tử con (Dù ở mức nào trong cây DOM)

```
//div[@id='menu']/li
```

📌 Chọn tất cả `<li>` bên trong `<div id="menu">`, dù ở cấp nào.

♦ Chọn phần tử cha (Parent)

```
//span[@class='icon']/parent::*
```

📌 Chọn phần tử **cha** của `<span>` có class `"icon"`.

♦ Chọn tổ tiên (Ancestor)

```
//span[@class='icon']/ancestor::div
```

📌 Tìm **tất cả tổ tiên** là `<div>` của `<span>` có class `"icon"`.

♦ Chọn anh chị em (Sibling)

```
//h2[@class='title']/following-sibling::p
```

📌 Chọn `<p>` nằm **ngay sau** `<h2 class="title">`.

```
//h2[@class='title']/preceding-sibling::p
```

📌 Chọn `<p>` nằm **ngay trước** `<h2 class="title">`.

🎯 6. Chọn Phần Tử Theo Vị Trí

- ♦ **Chọn phần tử đầu tiên**

```
(//button[@class='btn'])[1]
```

📌 Chọn nút đầu tiên có class "btn".

- ♦ **Chọn phần tử cuối cùng**

```
(//button[@class='btn'])[last()]
```

📌 Chọn nút cuối cùng có class "btn".

- ♦ **Chọn phần tử thứ n**

```
(//button[@class='btn'])[2]
```

📌 Chọn nút thứ hai có class "btn".

🎯 7. Xpath Động (Dành cho các trang có nội dung thay đổi)

Nếu trang có phần tử thay đổi ID hoặc class động, có thể dùng:

xpath

```
//*[contains(@class, 'dynamic-class')]
```

📌 Chọn tất cả phần tử có class chứa "dynamic-class".

Hoặc nếu ID thay đổi một phần:

```
//div[starts-with(@id, 'user_')]
```

📌 Chọn tất cả `<div>` có ID **bắt đầu** với `"user_"`.

🎯 8. Tổng Kết Các Câu Lệnh Quan Trọng

Câu lệnh	Ý nghĩa
<code>//tagname</code>	Chọn tất cả thẻ theo tên
<code>//*[@attribute='value']</code>	Chọn phần tử theo thuộc tính
<code>//tagname[contains(text(), 'text')]</code>	Chọn phần tử chứa văn bản
<code>//tagname[contains(@attribute, 'value')]</code>	Chọn phần tử có giá trị chứa một phần của attribute
<code>//tagname[normalize-space()='text']</code>	Chọn phần tử có văn bản, loại bỏ khoảng trắng thừa
<code>//tagname[@attr1='val1' and @attr2='val2']</code>	Chọn phần tử có cả hai thuộc tính
<code>//tagname[@attr1='val1' or @attr2='val2']</code>	Chọn phần tử có một trong hai thuộc tính
<code>//tagname/following-sibling::tagname</code>	Chọn phần tử kế tiếp
<code>//tagname/preceding-sibling::tagname</code>	Chọn phần tử trước đó
<code>(//tagname)[1]</code>	Chọn phần tử đầu tiên
<code>(//tagname)[last()]</code>	Chọn phần tử cuối cùng

Kết Luận

- XPath giúp xác định chính xác phần tử trên trang web.
- Có thể chọn phần tử dựa vào **thẻ, thuộc tính, văn bản, quan hệ cha-con, anh chị em, tổ tiên**.
- Dùng **`contains()`**, **`normalize-space()`**, **`starts-with()`** để chọn phần tử linh hoạt hơn.
- **Kết hợp AND, OR, NOT** để lọc phần tử mạnh mẽ hơn.

 **Bây giờ bạn đã có thể viết XPath chuyên nghiệp!**