

Cheat Sheet



WWW.FREERANGETESTERS.COM



XPath Locators

XPath es un lenguaje que permite navegar a través de elementos y atributos en un documento HTML.

Selección Básica

- **Elemento con atributo específico:** `//tagname[@attribute='value']`
- **Todos los elementos con un atributo específico:** `//*[@attribute='value']`

Uso de Contiene

- **Contiene texto:** `//*[contains(text(),'partial text')]`
- **Contiene atributo:** `//tagname[contains(@attribute, 'partial value')]`

Relaciones de Elementos

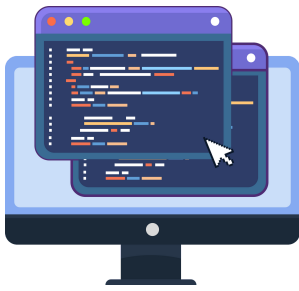
- **Elemento hijo:** `//parentTag/childTag`
- **Descendiente:** `//ancestorTag//descendantTag`
- **Elemento hermano siguiente:** `//precedingElement/following-sibling::tagname`

Usando Índices

- **Elemento específico en una lista:** `(//tagname)[index]`

Combinación de Criterios

- **Múltiples atributos:** `//tagname[@attr1='value1'][@attr2='value2']`
- **Combinación de texto y atributos:** `//tagname[contains(text(),'text')][@attribute='value']`



CSS Selectors

Los selectores CSS proporcionan una forma de seleccionar elementos basándose en su estilo y estructura en el documento.

- **Selección Básica**
 - Elemento específico: *tagname*
 - Elemento con ID específico: *#id*
 - Elemento con clase específica: *.classname*
- **Relaciones de Elementos**
 - Elemento hijo: *parentTag > childTag*
 - Elemento descendiente: *ancestorTag descendantTag*
 - Elemento hermano adyacente: *precedingElement + tagname*
 - Todos los hermanos siguientes: *precedingElement ~ tagname*
- **Atributos y Valores**
 - Atributo específico: *tagname[attribute='value']*
 - Contiene atributo: *tagname[attribute*='partial value']*
 - Empieza con un atributo: *tagname[attribute^='start value']*
 - Termina con un atributo: *tagname[attribute\$='end value']*
- **Pseudo-Clases y Pseudo-Elementos**
 - Primer hijo: *tagname:first-child*
 - Último hijo: *tagname:last-child*
 - Único hijo: *tagname:only-child*



Ejemplo 1: HTML con Lista de Elementos

```
Fragmento HTML

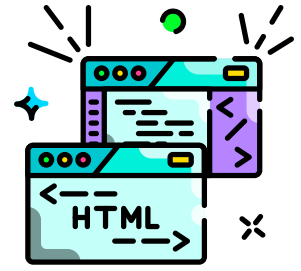
<div id="menu">
<ul>
<li class="item">Inicio</li>
<li class="item">Servicios</li>
<li class="item">Contacto</li>
</ul>
</div>
```

Ubicación de Elementos:

- Para seleccionar el segundo elemento de la lista (Servicios):
 - XPath: `//div[@id='menu']/ul/li[2]`
 - CSS Selector: `div#menu ul li:nth-child(2)`

Explicación

- En XPath, estamos navegando desde el div con id='menu', pasando por su ul hijo, hasta llegar al segundo li.
- En CSS, usamos nth-child(2) para seleccionar el segundo hijo li bajo ul que está dentro del div con id='menu'.



Ejemplo 2: HTML con Campos de Formulario

```
Fragmento HTML 2

<form id="loginForm">
  <input type="text" id="username" name="user_name" />
  <input type="password" name="password" />
  <button type="submit">Ingresar</button>
</form>
```

Ubicación de Elementos:

- **Para seleccionar el campo de contraseña:**
 - **XPath:** `//form[@id='loginForm']//input[@type='password']`
 - **CSS Selector:** `form#loginForm input[type='password']`

Explicación

- En XPath, estamos buscando cualquier input con type='password' dentro del form con id='loginForm'.
- En CSS, estamos haciendo algo similar, seleccionando el input con type='password' dentro del form con id='loginForm'.

Consejos generales

- Evitar dependencia de estructuras complejas: No depender de la ruta completa desde la raíz del documento.
- Preferencia por ID y Clase: Utilizar identificadores únicos (id) y nombres de clase siempre que sea posible.
- Robustez Ante Cambios: Evitar detalles que puedan cambiar frecuentemente, como índices o texto específico, a menos que sea inevitable.
- Legibilidad y Mantenibilidad: Escribir locators que sean fáciles de entender y mantener.