

Locators

LOCATING WEBELEMENTS

- **id** - Locates elements whose **ID** attribute matches the search value
- **Name** - Locates elements whose **NAME** attribute matches the search value
- **Link Text** - Locates anchor elements whose visible text matches the search value
- **Partial Link Text** - Locates anchor elements whose visible text contains the search value. If multiple elements are matching, only the first one will be selected.
- **Class Name** - Locates elements whose class name contains the search value
- **Tag Name** - Locates elements whose tag name matches the search value
- **CSS Selector** - Locates elements matching a CSS selector
- **XPath** - Locates elements matching an XPath expression

XPath Full Tutorial

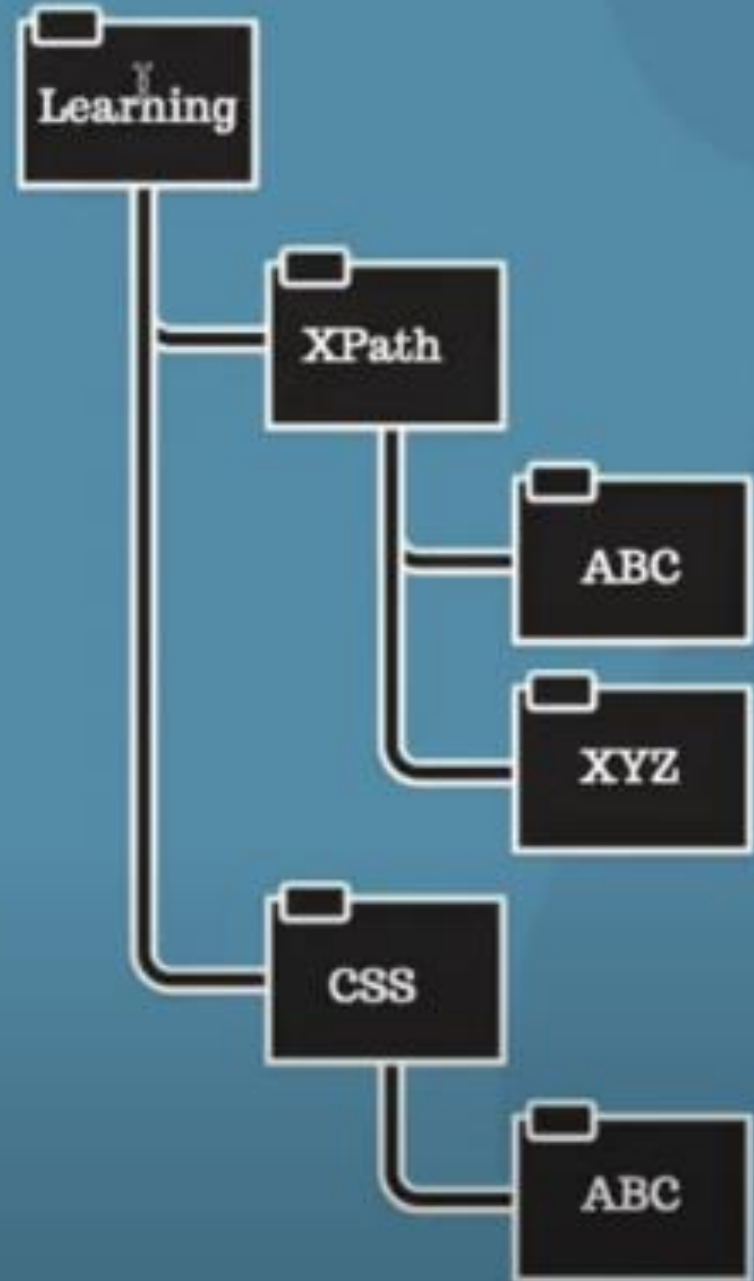
1. Introduction to XPath | Understand basic format?
2. Relative vs Absolute XPath
3. XPath function - "starts-with"
4. XPath function - "contains"
5. XPath function - "text()" method
6. How to use AND & OR in Selenium XPath
7. XPath Axes Methods (Parent, Child, Self)
8. XPath Axes Methods (descendant, descendant-or-self)
9. XPath Axes Methods (ancestor, ancestor-or-self)
10. XPath Axes Methods (following, following-sibling)
11. XPath Axes Methods (preceding, preceding-sibling)

WHAT IS XPATH?

- XPath XML İletişim Yolu Dili anlamına gelir
- XPath HTML DOM yapısını kullanan bir web sayfasında herhangi bir öğenin konumunu bulmak için kullanılır
- World Wide Web Consortium (W3C) tarafından tanımlanmıştır)

«XPath, herhangi bir XML belgesindeki düğümlerde gezinmek için kullanılır.

« XPath uses “path like” syntax



```

<?xml version="1.0" encoding="iso-8859-8" standalone="yes" ?>
<CURRENCIES>
  <LAST_UPDATE>2004-07-29</LAST_UPDATE>
  <CURRENCY>
    <NAME>dollar</NAME>
    <UNIT>1</UNIT>
    <CURRENCYCODE>USD</CURRENCYCODE>
    <COUNTRY>USA</COUNTRY>
    <RATE>4.527</RATE>
    <CHANGE>0.044</CHANGE>
  </CURRENCY>
  <CURRENCY>
    <NAME>euro</NAME>
    <UNIT>1</UNIT>
    <CURRENCYCODE>EUR</CURRENCYCODE>
    <COUNTRY>European Monetary Union</COUNTRY>
    <RATE>5.4417</RATE>
    <CHANGE>-0.013</CHANGE>
  </CURRENCY>
</CURRENCIES>

```

XML

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Example</title>
5      <link rel="stylesheet" href="st:
6    </head>
7    <body>
8      <h1>
9        <a href="/">Header</a>
10     </h1>
11     <nav>
12       <a href="one/">One</a>
13       <a href="two/">Two</a>
14       <a href="three/">Three</a>
15     </nav>

```

HTML

Basic format of XPath

Xpath=//tagname[@Attribute='Value']

Types of XPath

Absolute XPath:

- Sayfanın kök öğesinden istenen öğeye giden yolun tamamını içerir
- Absolute XPath kök düğüm ile başlar- Single forward slash (/)
- Absolute Xpath kullanmanın dezavantajı - DOM'daki herhangi bir küçük değişiklik XPath'i geçersiz kılar.

Relative XPath:

- With relative XPath, XPath HTML DOM yapısının ortasından başlar.
- Çift ileri eğik çizgi ile başlar (//)
- Daha az kırılgandır.

XPath function - "starts-with"

- **"starts-with"** fonksiyonu dinamik web elemanlarını bulmada çok faydalıdır.
- Statik kalan web öğesinin başlangıç değerini eşleştirmek için kullanabilirsiniz.

(Example - ID = session62354624, session76576744)

- **"starts-with"** fonksiyonu statik attribute değerlerini de bulabilir.

Format:

XPath=//tagname[starts-with(@Attribute, 'Value')]

XPath function - "contains "

- "contains" fonksiyonu dinamik web elemanlarını bulmak için kullanılır.
- Web ögesini bulmak için herhangi bir parçalı ögenin özelliklerini girebilirsiniz.

Format:

Xpath=//tagname[contains(@Attribute,'Value')]

XPath function - "text()" method

- "text()" methodu tam metin eşleşmesi olan öğeyi bulmak için kullanılır

Format:

Xpath=//tagname[text()='ActualText']

Selenium Xpath de AND & OR ile Kullanımı

- AND & OR ifadeleri Selenium XPATH ifadesinde de kullanılabilir
- Web sayfasında öge bulmak için ikiden fazla özellik kullanmak istiyorsanız çok yararlıdır.

Syntax is:

Xpath=//tagname[@Attribute='Value' or @Attribute='value']

Xpath=//tagname[@Attribute='Value' and @Attribute='value']

XPath Axes Methods (Parent, Child, Self)

Her bir node yapısı bir özelliği temsil eder.

Yapıdaki diğer node'lar bu özelliğe göre konumlandırılırlar.

- **parent**: Context (Current) node geçerli node'un üst ögesini seçer.

XPath Format : //tagname[@Attribute='Value']/parent::tagname

- **child** : Geçerli node'un tüm child'larını seçer.

XPath Format : //tagname[@Attribute='Value']/child::tagname

- **self**: Geçerli node içinde seçim yapmamızı sağlar.

XPath Format : //tagname[@Attribute='Value']/self::tagname

XPath Axes Methods

(descendant, descendant-or-self)

descendant : context (geçerli) node tüm torunlarını (çocuklar, torunlar, vb.) seçer.

XPath Format : //tagname[@Attribute="Value"]//descendant::tagname

descendant-or-self : Eğer descendants ve self tagları aynı ise bu context (geçerli) node ve tüm torunlarını (çocuklar, torunlar, vb.) seçer.

XPath Format : //tagname[@Attribute='Value']//descendant-or-self::tagname

XPath Axes Methods (ancestor, ancestor-or-self)

ancestor : Bu context (Geçerli) node 'un tüm atalarını (ebeveyn, büyükbaba, vb.) seçer.

XPath Format : //tagname[@Attribute="Value"]//ancestorstagname

ancestor-or-self :eğer ancestors ve self tagname'leri aynı ise bu context (geçerli) node'u ve (ebeveyn, büyükbaba, vb.) gibi tüm atalarını seçer.

XPath Format : //tagname[@Attribute='Value']//ancestor-or-self:tagname

XPath Axes Methods (following, following-sibling)

following : Bu context (geçerli) node'dan sonra görünen tüm düğümleri seçer.

XPath Format :

`//tagname[@Attribute='Value']//following::tagname`

following-sibling : Bu context (geçerli) node'u ve bu node'dan sonra aynı ebeveyne sahip tüm düğümleri seçer

XPath Format :

`//tagname[@Attribute='Value']//following-sibling::tagname`

XPath Axes Methods **(preceding, preceding-sibling)**

preceding: Bu context (geçerli) node'dan önce görünen tüm düğümleri seçer.

XPath Format :

`//tagname[@Attribute="Value"]//preceding::tagname`

preceding-sibling: Bu context (geçerli) node ile aynı üst öğeye sahip olan tüm node'ları seçer.

XPath Format :

`//tagname[@Attribute='Value']//preceding-sibling::tagname`

CSS Full Tutorial

- Introduction to CSS and CSS Selector
- Write CSS Selector using Class Attribute
- Advanced CSS Selectors (Using mix of Tag, ID and Classname)
- CSS Selector Substring Matching
- Finding Direct Child or Sub-child Elements
- CSS Selector - Next Sibling | Adjacent Sibling Combinator
- CSS Selector Pseudo-classes - First-Child, Last-Child, Nth-Child, Nth-Last-Child
- CSS Selector Pseudo-classes - First-of-Type, Last-of-Type, Nth-of-type()

WHAT IS CSS AND CSS SELECTOR?

(BASIC CSS SELECTOR SYNTAX)

- CSS kademelendirilmiş/basamaklandırılmış Stil Sayfaları anlamına gelir.
- CSS HTML belgesinin sunumunu tanımlayan bir stil sayfası dilidir.
- CSS Selector'leri web sayfasındaki HTML öğelerini hedeflemek için kullanılır

Syntax: `tagName[AttributeName='AttributeValue']`

Example: `input[id=first_name]`

CSS SELECTOR: SELECT BY ID

- Web öğesinin bir ID attribute varsa, CSS selector'ü ID attribute'nün ayrıntıları için kullanabilirsiniz.
- CSS Selector içinde ID attribute hash (#) işareti ile kullanılır.

Syntax : tagName#elementID

Example : input#first_name

CSS SELECTOR USING

CLASS ATTRIBUTE

- eğer webelement CLASS attribute'ne sahipse CSS selector içinde Class attribute'nün özelliklerini kullanabiliriz.
- CSS Selector stntax'ı içinde CLASS attribute nokta sembolü ile kullanılır (.)

Syntax : tagName.elementID

Example : input.signup

CSS SELECTOR USING OTHER **WEBELEMENT ATTRIBUTES**

Webelementlerindeki "type", "placeholder", "value" gibi attribute'lerini de CSS selector'lerde kullanabiliriz.

Syntax: `tagName[AttributeName='AttributeValue']`

Example: `input[value='Sign me up']`

ADVANCED CSS SELECTORS

(USING MIX OF TAG, ID AND CLASSNAME)

Webelement'lerindeki diğer attribute'leri ID VE CLASSNAME ile karıştırıp CSS selector içinde kullanabiliriz.

Syntax: tagName.classValue[AttributeName='AttributeValue']
 tagName#idValue[AttributeName='AttributeValue']

Examples: input.signup[type='submit'][value='Sign me up']
 input#submit_btn[type='submit'][value='Sign me up ']

CSS SELECTOR - SUB-STRING

- Dinamik webelement tanımlamalarında Sub-string ve partial string eşleşmelerini kullanmak çok kullanışlıdır.
- CSS selectorlerinde sub- string eşleşmelerinde kullanılan 3 önemli karakter vardır :

"^" İşareti – Textin Ön ekini kullanmamızı sağlar.

`input[name^='country_c']`

"\$" İşareti - Textin Son ekini kullanmamızı sağlar

`input[name$='y_client']`

"*" İşareti - Textin içinde bir sub-string ögesini kullanmamızı sağlar.

`input[name*='try_cl']`

FINDING CHILD OR SUBCHILD ELEMENTS

- Direct Child

- Direct olarak child'ler da seçim yapmak için Child combinator (>) kullanılır.

Syntax: tagName[AttributeNames'AttributeValue'] > tagName[AttributeNames'AttributeValue']

Example: select#country > option[value="AU"]

- Child or Subchild

- Child veya Subchild seçmek için Descendant combinator () kullanılır.

Syntax: tagName[AttributeNames='AttributeValue'] tagName[AttributeName='AttributeValue']

Example: form#deorg form div

CSS SELECTOR - NEXT SIBLING

- Sibling(Kardeş) elementleri locate etmek için + operatörü kullanılır.
- Sibling combinatorü(+) aynı Webelementin altında olan kardeş webelementleri seçmemizi sağlar.

Format:

tagName[AttributeName='AttributeValue']+tagName[AttributeNames='AttributeValue']

CSS SELECTOR PSEUDO-CLASSES

FIRST-CHILD, LAST-CHILD, NTH-CHILD, NTH-LAST-CHILD

- A CSS pseudo-class bir seçiciye eklenen ve seçilen web ögesinin özel bir durumunu belirten bir anahtar sözcüktür bu sözcükler;
 - **:first-child** – Kardeş elementlerden ilk olanı döndürür.
 - **:last-child** Kardeş olan elementlerden son olanı döndürür.
 - **:nth-child()** – Kardeşlerin oluşturduğu gruptan belirlediğimiz elementi döndürür.
 - **:nth-last-child()** - Kardeşlerin oluşturduğu gruptan belirlediğimiz elementi sondan itibaren sayarak döndürür
- Examples `select#job_role> :first-child`

CSS SELECTOR PSEUDO-CLASSES CONT.

FIRST-OF-TYPE, LAST-OF-TYPE, NTH-OF-TYPE()

- :first-of-type – Kardeş elementlerden belirttiğimiz türün ilk elemanını döndürür.
- :last-of-type - Kardeş elementlerden belirttiğimiz türün son elemanını döndürür.
- :nth-of-type() - Kardeş elementlerden belirttiğimiz türün belirttiğimiz konumdaki elemanını döndürür.
- Examples: `form#deorg.form>div:first-of-type`

Dinlediğiniz için

teşekkür ederim,

Umarım faydalı bir sunum olmuştur.