

XPath - Cheat Sheet



LX_schlee

Content

1. Why are XPath Locators important?
2. XPath-Syntax
3. Relative and Absolute XPath
4. Difference between "/" and "/" in XPath-Expressions
5. Parents and Siblings Logic

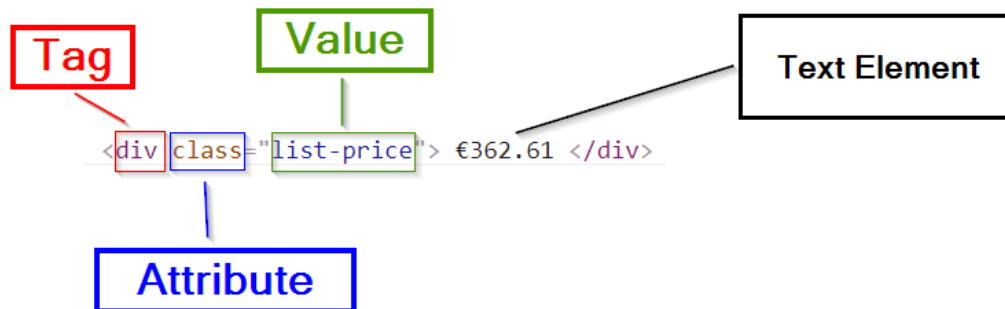
1. Why are XPath Locators important?

- Different locators to target specific elements : *ID*, *Class*, *LinkText* etc.
- Alternative: XPath (don't rely on other locators)
- XML Path you use to navigate through the HTML structure of a website
- Our overall goal is to create an expression which points to a unique element/list of elements (-> XPath)

Don't get confused ! – we will cover everything with examples 😊

2. XPath-Syntax

`//tag[@attribute="value"]`



Example 1: List of Elements – not unique!

```
... <li class="searchResultItem"> == $0
  <div class="view_items">
    <div class="img pruImage">...</div>
    <div class="name">
      ...
    </div>
  </div>
</li>
...

html body div#wrapper.warp2015 div#content.clearfix div#content-main ul.gridView li.searchResultItem

//li[@class="searchResultItem"] 1 of 40
```

Example 2: List of an Element (!) - (unique!)

```
... <li class="searchResultItem"> == $0
  <div class="view_items">
    ...
  </div>
</li>
...

html body div#wrapper.warp2015 div#content.clearfix div#content-main ul.gridView li.searchResultItem

//li[@class="searchResultItem"][1] 1 of 1
```

Example 3: List of Elements – not unique!

```
... <div class="price">€398.64</div> == $0
  <div class="list-price"> €362.61 </div>
  <div class="review">...</div>
  <div class="favit_free">...</div>
</div>
...

html body div#wrapper.warp2015 div#content.clearfix div#content-main ul.gridView li.searchResultItem div.view_items div.price

//div[@class="price"] 1 of 40
```

Example 4: List of an Element (!) - (unique!)

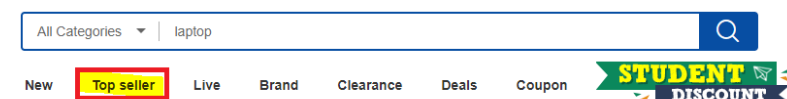
..	<div class="price">€308,64</div>	== \$0
	<div class="list-price">€362.61</div>	
	<div class="review">...</div>	
html	body	div#wrapper.warp2015
	div#content.clearfix	div#content-main
	ul.gridView	li.searchResultItem
	div.view_items	div.price
[//div[@class="price"]][1]		1 of 1

3. Absolute and Relative XPath

Absolute XPath: Complete Path from Root Element

Relative XPath: You choose the Element you want to start from (important: Use “//” if you choose the Relative XPath)

Use Case Example (we want to target the „Top Seller” Text Element)



```
<div class="head_menus">
  <div class="shop_cate" id="shopCate">...</div>
  <div class="menu_main">
    <div class="menu_li">...</div>
    <div class="menu_li"> == $0
      <a href="/bestselling/">Top seller</a>
    </div>
    <div class="menu_li">...</div>
    <div class="menu_li">...</div>
```

Relative XPath:

Starting Point → `<div class="menu_main">`

Ending Point → `Top seller`

XPath Expression → `//div[@class="menu_main"]/div[2]/a`

1 of 1

Absolute Xpath:

```
<!DOCTYPE html>
<html class style="height: 100%;">
  <head>...</head>
  <body style="position: relative; min-height: 100%; top: 0px;">
    <div class="head_warp"> == $0
      <div class="head_r_naiv_bg">...</div>
      <div class="head_M_navi">...</div>
      <div class="head_menus">
        <div class="shop_cate" id="shopCate">...</div>
        <div class="menu_main">
          <div class="menu_li">...</div>
          <div class="menu_li">
            <a href="/bestselling/">Top seller</a>
          </div>
        </div>
      </div>
```

Starting Element (Root Element)

Ending Element

html body div.head_warp

html/body/div/div[3]/div[2]/div[2]/a

XPath Expression

1 of 1

4. Difference between “/” and “//” in XPath-Expressions

- “/” (single Slash) is used to target the immediate children of a node.

Example: `//div[@class="menu-main"]/div[2]/div[3]/div[1]/ul/li/a`

- “//” (double Slash) is used to target child elements but this child elements do not have to be the immediate children

Example: `//div[@class="menu-main"]//ul/li/a`

5. Parents and Siblings Logic

Parent XPath

Syntax: XPath-Expression(Child Element)/parent::<tag>

The screenshot shows a DOM tree with the following structure:

- <div class="head_menus">
 - <div class="shop_cate" id="shopCate">...</div>
 - <div class="menu_main"> (highlighted with a green box and labeled "Ending Element (Parent Element)")
 - <div class="menu_li">...</div>
 - <div class="menu_li">...</div>
 - <div class="menu_li">...</div>
 - <div class="menu_li">...</div>
 - <div class="menu_li">...</div> (highlighted with a green box and labeled "Starting Element (5th Child Element)")
 - <div class="menu_li">...</div>

The breadcrumb trail at the bottom is: html > body > div#wrapper.warp2015 > div#content.clearfix > div#content-main > ul.gridView

The XPath expression entered in the console is: `//div[@class="menu_li"][5]//parent::div` (highlighted with a red box and labeled "XPath Expression")

Sibling (Preceding)

Syntax: XPath-Expression/preceding-sibling::<tag>

The screenshot shows a DOM tree with the following structure:

- <div class="menu_main">
 - <div class="menu_li">...</div>
 - ... <div class="menu_li">...</div> == \$0
 - <div class="menu_li">...</div>
 - <div class="menu_li">...</div>
 - <div class="menu_li">...</div>
 - <div class="menu_li">...</div> (highlighted with a green box and labeled "Ending Point")
 - <div class="menu_li">...</div> (highlighted with a green box and labeled "Starting Point")

The breadcrumb trail at the bottom is: html > body > div.head_warp > div.head_menus > div.menu_main > div.menu_li

The XPath expression entered in the console is: `//div[@class="menu_li"][7]/preceding-sibling::div[1]` (highlighted with a red box and labeled "Xpath Expression")

Sibling (Following)

Syntax: XPath-Expression/following-sibling::<tag>

The diagram illustrates the XPath 'following-sibling' syntax using an XML snippet and a breadcrumb trail.

XML Snippet:

```
<div class="menu_main">
  <div class="menu_li">...</div>
  <div class="menu_li">...</div> == $0
  <div class="menu_li">...</div>
  <div class="menu_li">...</div>
  <div class="menu_li">...</div>
  <div class="menu_li">...</div>
  <div class="menu_li">...</div>
  <a class="navi_banner xiaomi" href="https://de.geekbuying.com/help/student_discount">...</a>
</div>
```

Annotations:

- Starting Point:** Points to the second `<div class="menu_li">...</div>` element, which is also labeled `== $0`.
- Ending Point:** Points to the third `<div class="menu_li">...</div>` element.

Breadcrumb Trail:

html body div.head_warp div.head_menus div.menu_main div.menu_li

XPath Expression:

```
//div[@class="menu_li"][2]/following-sibling::div[2]
```