

目录

前言	1.1
基本语法	1.2
使用场景和举例	1.3
常见函数	1.3.1
逻辑操作	1.3.2
判断比较	1.3.3
常见问题	1.4
附录	1.5
官网资料	1.5.1
参考资料	1.5.2

XPath知识总结

- 最新版本: `v0.6`
- 更新时间: `20210730`

简介

整理在上层软件领域，尤其是Web领域，常用到的基础通用知识XPath。先进行基本语法的介绍；再总结常用场景和具体例子，包括常见函数、逻辑操作、判断比较等；以及一些常见问题；最后附上官网资料供参考。

源码+浏览+下载

本书的各种源码、在线浏览地址、多种格式文件下载如下：

Gitbook源码

- [crifan/xpath_summary](#): XPath知识总结

如何使用此Gitbook源码去生成发布为电子书

详见: [crifan/gitbook_template: demo how to use crifan gitbook template and demo](#)

在线浏览

- XPath知识总结 [book.crifan.com](#)
- XPath知识总结 [crifan.github.io](#)

离线下载阅读

- XPath知识总结 PDF
- XPath知识总结 ePub
- XPath知识总结 Mobi

版权说明

此电子书教程的全部内容，如无特别说明，均为本人原创和整理。其中部分内容参考自网络，均已备注了出处。如有发现侵犯您的版权，请通过邮箱联系我 `admin` 艾特 `crifan.com`，我会尽快删除。谢谢合作。

鸣谢

感谢我的老婆陈雪的包容理解和悉心照料，才使得我 crifan 有更多精力去专注技术专研和整理归纳出这些电子书和技术教程，特此鸣谢。

更多其他电子书

本人 crifan 还写了其他 100+ 本电子书教程，感兴趣可移步至：

[crifan/crifan_ebook_readme: Crifan的电子书的使用说明](#)

crifan.com，使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新： 2021-07-30 19:09:52

XPath基本语法介绍

常见符号

XPath 中有很多常用符号，下面来解释其具体含义和基本的逻辑：

- 查找和定位元素的位置和层次时用到的：
 - `/`：表示绝对路径，绝对路径是指从根目录开始
 - `//`：表示相对路径
 - 典型的用法是表示从任意路径去搜索
 - `.`：表示当前层
 - `..`：表示上一层
- 想要具体查找和定位哪个元素时用到的：
 - `*`：表示通配符
- 找到了该元素还要判断其属性时用到的：
 - `[]`：属性的判断条件表达式
 - `@`：表示属性，`@属性`
 - 比如常用的 `@class`，`@name`，`@id` 等等

常见函数

- `contains()`：是否包含对应的值
- `starts-with()`：是否以某种字符串开始
- `ends-with()`：是否以某种字符串结束
- `last()`：列表/集合类元素的最后一个值
- `text()`：一个节点的文本值

逻辑判断

- `and`：逻辑 且
- `not`：否定
 - 一般情况下会与返回值为 `true` 或者 `false` 的函数组合起来使用。
 - 比如上面提到的 `contains()` 与 `starts-with()`
 - 当然 `not()` 还有一种特殊用法就是直接 `not`

crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新： 2021-07-29 13:29:21

场景和举例

下面就来介绍一下XPath的常见使用场景和具体的示例。

XPath常见使用场景

比如之前自己就遇到过的：

- Docbook 的 xml 的处理中的 XSLT 中使用 xpath
- BeautifulSoup 中的html的提取
- Selenium 中元素的选择
- Scrapy 的 Selector 中提取内容

XPath使用举例

TODO： 1.把selenium相关的帖子中的示例代码，尤其是关于xpath的，整理过来供参考

常见符号和属性判断

@属性 举例

对于判断元素中是否包含某个属性（注意不是去判断属性的值），则语法是：`[@someProperty]`

比如：

```
<span aria-label="Price" class="x-hidden-focus">$799.00</span>
```

和

```
<div class="price-text srv_price">  
  <s class="srv_saleprice" aria-label="Full price was $1,299.00">  
  <span>&nbsp;</span>  
  <div class="price-disclaimer ">  
    <span aria-label="Now $1,299.00" class="x-hidden-focus">  
  </div>  
</div>
```

只想要判断span中存在aria-label属性即可，则可以用：`//span[@aria-label]`

更多例子如下：

- `@class="dropdown-menu"`
 - `cartNumOptionElemList = driver.find_elements_by_xpath('//*[@class="dropdown-menu"]/li[@role="option"]')`
- `@name="loginfmt"`
 - `inputEmailElement = WebDriverWait(driver, gCfg["waitTimeout"]).until(EC.presence_of_element_located((By.XPATH, '//input[@type="email" and @name="loginfmt"]')))`
- `@id="idSIButton9"`
 - `nextElement = driver.find_element_by_xpath('//input[@type="submit" and @id="idSIButton9" and @value="Next"]')`

selenium中特殊列表元素的选择

之前遇到列表选项的html不是 `select` 的 `option` , 而是:

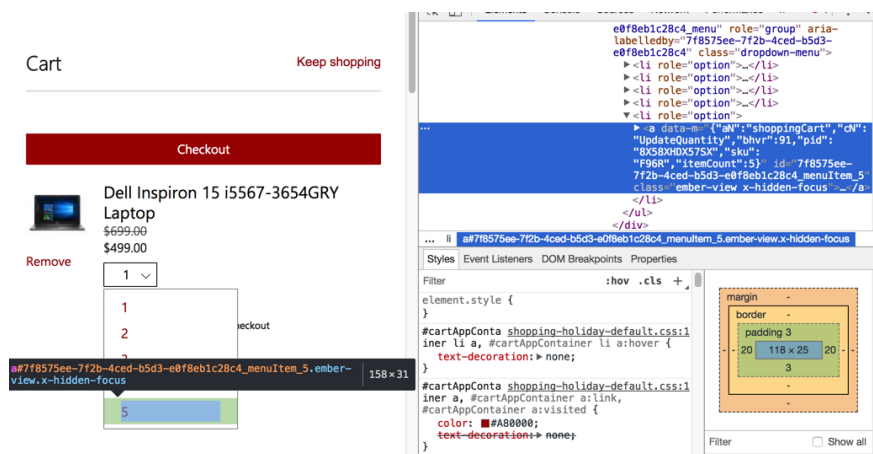
```
<ul id="22bc22dd-ef9d-4d3c-8de9-1e7bc704f9f9_menu" role="group">
  <li role="option">
    <a data-m="{&quot;a&quot;:&quot;shoppingCart&quot;,&quot;c
  </a>
  </li>
  <li role="option">
  ...
```

即, `ul` 的 `li` 的列表

最后用代码:

```
cartNumOptionElemList = driver.find_elements_by_xpath('//*[@class="dropdown-menu"]/li[@role="option"]')
cartNumOptionCount = len(cartNumOptionElemList)
logging.info("cartNumOptionElemList=%s, cartNumOptionCount=%s", cartNumOptionElemList, cartNumOptionCount)
if cartNumOptionCount < gCfg["msStore"]["onceBuyNum"]:
    logging.error("Current Cart select max number %s < expected %s", cartNumOptionCount, gCfg["msStore"]["onceBuyNum"])
    driver.quit()
toSelectIdx = gCfg["msStore"]["onceBuyNum"] - 1
# carNumSelect = Select(cartNumOptionElemList)
# carNumSelect.select_by_index(gCfg["msStore"]["onceBuyNum"])
carNumSelectElem = cartNumOptionElemList[toSelectIdx]
logging.info("carNumSelectElem=%s", carNumSelectElem)
carNumSelectElem.click()
# aLinkElem = carNumSelectElem.find_element_by_link_text("shoppingCart")
# logging.info("aLinkElem=%s", aLinkElem)
# aLinkElem.click()
```

去实现，找到列表的元素，点击后下拉显示所有的选项，然后点击选中某个选项：



crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新: 2021-07-30 19:09:05

常见函数

- `start-with()` : 以某值开头的
- `text()` : 文本值
- `last()` : 最后一个
- `concat()` : 拼接多个内容

下面详细介绍:

start-with

举例:

html:

```
<li class="b_ad">
  xxx
</li>

<li class="b_ad b_adBottom">
  yyy
</li>

<li class="b_algo">
  aaa
</li>

<li class="b_algo">
  bbb
</li>
```

希望同时找到class是

- 普通元素的: `b_algo`
- 广告Ad元素的: 以 `b_ad` 开始的

代码:

```
allLiXpath = "//li[@class='b_algo' or starts-with(@class,
resultLiList = resultElem.find_elements_by_xpath(allLiXpath)
```

text()

- 对于HTML代码:


```
<a target="_self" href="/s?rsv_idx=1&wd=111&usr
```

定位英文结果即可使用: `//a[text()='英文结果']`

另外一个例子:

- `clickHereBtnElement = driver.find_element_by_xpath('//a[text()='click here'])` 中的 `text()='click'`

last()

- `//input[@name="identity"][last()]`

concat()

举例:

- docbook的xml的处理 -> XSLT中使用xpath

```
<xsl:key name="book" match="books/book" use="concat(e
```

- 其他例子
 - `concat('un', 'grateful')` -> `ungrateful`
 - `concat('Thy ', (), 'old ', "groans", "", ' ring', 'yet', ' in', ' my', ' ancient', ' ears.')` -> `Thy old groans ring yet in my ancient ears.`

crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新: 2021-07-30 19:04:27

逻辑操作

Xpath中的逻辑操作，主要有：

- `or` : 逻辑 或
- `and` : 逻辑 与
- `not` : 逻辑 非

or 逻辑或

举例：

可以同时搜到 `class`是**`b_algo`** 或 `class`值以**`b_ad`**开头的 `li`元素

```
allLiXPath = "//li[@class='b_algo' or starts-with(@class,  
resultLiList = resultElem.find_elements_by_xpath(allLiXPath)
```

not 逻辑非

举例：

- `//input[@name="identity" and not
(contains(@class, 'A'))]`
- `//input[not(@class)]`

crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新: 2021-07-30 19:06:47

判断比较

- `!=` : 不等于

!=

Selenium的Chrome的driver中用xpath去查找元素，是可以通过：

```
//a[@id="uhf-shopping-cart" and @aria-label!="0 items in sh
```

实现判断，只匹配到

```
id="uhf-shopping-cart" aria-label="2 items in shopping cart
```

这类元素，而不匹配：

```
id="uhf-shopping-cart" aria-label="0 items in shopping cart
```

即xpath支持：

```
@someProperty!="some not expected text"
```

这种写法的

crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新： 2021-07-30 19:06:41

常见问题

在当前节点下面搜索子节点=相对路径搜索

```
<html>
  <ol id="b_results">
    <li class="b_ad">
      <ul>
        <li class="b_adLastChild">
          <div class="sb_add sb_adTA">
            <h2>
              <a id="tile_link_cn" href="http://www.drv5.cn">
            </h2>
          </div>
        </li>
      </ul>
    </li>
    <li class="b_algo">
      <h2>
        <a target="_blank" href="http://www.drv5.cn">
      </h2>
    </li>
  </ol>
</html>
```

- `/` : 从根节点开始算起, 向下搜索
 - 说明:
 - 类似于文件夹的绝对路径
 - 举例
 - `/Users/limao/dev/`
 - 举例
 - `/html/ol/li[@class='b_ad']` 可定位到 `<li class="b_ad">`
- `//` : 从 (根节点向下) 任意层级开始搜索
 - 举例
 - `//ol/li[@class='b_ad']` 或 `//li[@class='b_ad']` 均可定位到 `<li class="b_ad">`
- `./` 或 `./.` : 点表示相对路径, 相对当前元素, 向下开始搜索子节点
 - 举例
 - 此处已获取到 `<li class="b_algo">` 元素, 想要去搜 ``

```
href="http://www.driv5.cn/azgame/77154.html">yyy</a>
```

，则

- 可以用
- `./h2/a` 或 `./a`
- 不能用
 - `//h2/a`
 - 否则会误搜出 文件根节点开始的第一个 h2 的 a ，此处即 `<li class="b_ad">` 下面的 `xxx`

crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新: 2021-07-30 19:06:23

附录

下面列出相关官网的语法教程和参考资料。

crifan.com, 使用[署名4.0国际\(CC BY 4.0\)协议](#)发布 all right reserved,
powered by Gitbook最后更新: 2021-07-29 13:29:21

官网资料

- [w3schools](#)
 - [XPath Tutorial](#)
- [w3.org](#)
 - [XML Path Language \(XPath\)](#)
 - [Selectors](#)
- [zvon.org](#)
 - [XPath 1.0 Tutorial @ZVON.org](#)
 - [starts-with \[XPath 2.0 Reference @ Zvon.org\]](#)
 - [contains \[XPath 2.0 Reference @ Zvon.org\]](#)
 - [matches \[XPath 2.0 Reference @ Zvon.org\]](#)
 - [ends-with \[XPath 2.0 Reference @ Zvon.org\]](#)
 - [concat \[XPath 2.0 Reference @ Zvon.org\]](#)

crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新: 2021-07-30 19:01:43

参考资料

- [【已解决】 Selenium如何实现2种class属性值的or或去查找元素](#)
- [【未解决】 Selenium的Chrome模拟bing必应搜索结果页面提取搜索结果](#)
- [【已解决】 Selenium中用xpath查询元素节点找到结果不对](#)
- [XPath 运算符 | 菜鸟教程](#)
- [【已解决】 Xpath中如何判断属性值不等于某个值](#)
- [【已解决】 xpath中如何实现或者or的判断](#)
- [【已解决】 Selenium如何点击下拉框并选择某个值](#)
- [查找元素 — Selenium-Python中文文档 2 documentation](#)
- [xml - Multiple conditions \(and operator\) in XPath - Stack Overflow](#)
- [Selenium 自动化测试从零实战 - CSDN博客](#)

crifan.com, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新: 2021-07-30 19:04:47