

웹 크롤링 - selenium 적용 부분

🕒 작성일시	@2023년 3월 3일 오후 3:24
📄 강의 번호	
📄 유형	
📎 자료	
☑ 복습	<input type="checkbox"/>
≡ 학습 소스 출처 1	https://www.youtube.com/watch?v=yQ20jZwDjTE&list=PLMsa_0kAjjrd8hYYCwbAuDsXZmHpgHvIV&index=4
≡ 학습 소스 출처 2	
📅 날짜	


브라우저 닫기

```
browser = webdriver.Chrome()

# 브라우저 현재 탭 닫기
browser.close()

# 브라우저 전부 닫기
browser.quit()
```

자식요소 접근하기



Kentucky Brunch Brand Stout (2016) (Silver Wax)


Toppling Goliath Brewing Co.


Stout - Imperial / Double

This beer is no longer being produced by the brewery

This beer is the real McCoy. Barrel aged and crammed with coffee, none other will stand in it's way. Sought out for being delicious, it is notoriously...

[Read More](#)

13% ABV	N/A IBU	 (4.77)	2,009 Ratings	Added 01/13/17
---------	---------	--	---------------	----------------



Beer : Barrel : Time (2020)

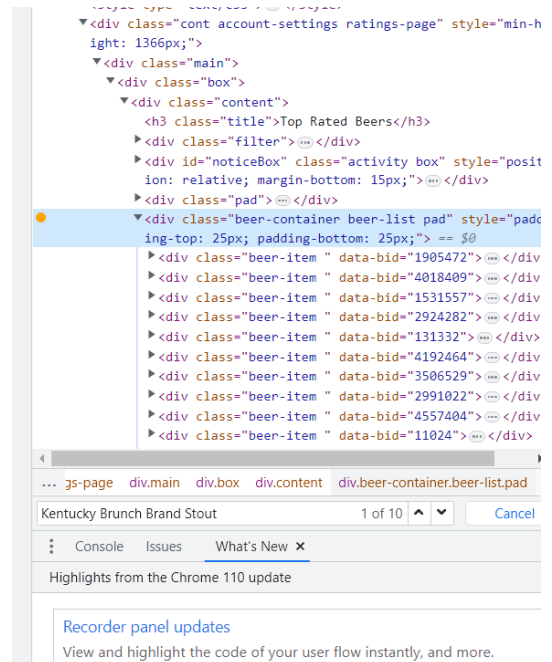
Side Project Brewing

Stout - Imperial / Double

This beer is no longer being produced by the brewery

For Beer : Barrel : Time 2020, we chose a blend of our stout recipes that consists of: O.W.K. recipe aged in Willett Bourbon barrels for 15 months O.W.K....

[Read More](#)



한페이지 안에서 `beer-container beer-list pad` 클래스의 자식으로 여러개의 `beer-item` 이 있는 상황이다.

여기서 각 `beer-item` 의 구조는 아래와 같다.

```

<div class="beer-item " data-bid="1905472">
  <a class="label" href="/b/toppling-goliath-brewing-co-kentucky-brunch-brand-stout-2016-silver-wax/1905472">
    
  </a>
  <div class="beer-details">
  <div class="details">
    <a class="add wishlist list-add-enhanced track-click tip" data-bid="1905472" data-track="topbeer" data-href=":addwish" data-wishlist="add" href="#" original-title="Add To List">Add</a>
  </div>

```

여기서 `beer-item` 의 리스트를 구하고 각 자식요소에게 접근하는 방법이 중요한데

```

# beer-item 클래스를 가진값들로 리스트를 만들었다.
beer_item_list = browser.find_elements(By.CSS_SELECTOR, '.beer-container.beer-list.pad>.beer-item')

```

위와 같이 `beer-item` 을 구한상태에서 자식요소를 구하는 과정에서 막혀있었다.

왜냐하면 selenium의 `webElement`에서 직접적으로 자식요소에게 접근하는 메소드가 없었기 때문에 현재 요소에서 부모나 자식요소에 접근하려면 매번 `find_element`를 사용하여 접근해야 했다.

이럴때는 XPATH를 사용하여 접근하면 간편하다.

XPATH 기본문법

/author	현재 컨텍스트 내에 있는 모든 <author> 요소 이 식은 다음 행에 나온 식과 같습니다.
author	현재 컨텍스트 내에 있는 모든 <author> 요소
first.name	현재 컨텍스트 내에 있는 모든 <first-name> 요소
/bookstore	이 문서의 문서 요소(<bookstore>)
//author	문서에 있는 모든 <author> 요소
book[bookstore/@specialty=@style]	style 특성 값이 문서의 루트에 있는 <bookstore> 요소의 specialty 특성 값과 같은 모든 <book> 요소
author/first-name	<author> 요소의 자식인 모든 <first-name> 요소
bookstore/title	<bookstore> 요소(임의의 하위 항목)에서 수준이 하나 이상인 모든 세부 <title> 요소. 이 식은 다음 행에 나온 식과 다릅니다.
bookstore/*title	<bookstore> 요소의 최하위인 모든 <title> 요소
bookstore/book/excerpt/emph	<bookstore> 요소 내에 있는 <book> 요소의 <excerpt> 자식 내에 있는 모든 <emph> 요소
//title	현재 컨텍스트에서 수준이 하나 이상인 모든 세부 <title> 요소. 이 상황은 유일하게 기간 노데이선이 필요한 경우입니다.
author/*	<author> 요소의 모든 자식 요소
book/*last-name	<book> 요소의 최하위인 모든 <last-name> 요소
/*	현재 컨텍스트의 모든 최하위 요소
*[@specialty]	specialty 특성이 있는 모든 요소
@style	현재 컨텍스트의 style 특성

price/@exchange	현재 컨텍스트 내에 있는 <price> 요소의 exchange 특성
price/@exchange/total	특성에 요소 자식이 포함되지 않았으므로 빈 노드 집합을 반환합니다. 이 식은 XPath(XML Path Language) 문법상 허용되지만 엄격히 말하면 유효하지 않습니다.
book[@style]	현재 컨텍스트의 style 특성이 있는 모든 <book> 요소
book/@style	현재 컨텍스트의 모든 <book> 요소에 대한 style 특성
@*	현재 요소 컨텍스트의 모든 특성
/first-name	현재 컨텍스트 노드에 있는 모든 <first-name> 요소 이 식은 다음 행에 나온 식과 같습니다.
first-name	현재 컨텍스트 노드에 있는 모든 <first-name> 요소
author[1]	현재 컨텍스트 노드에서 첫 번째 <author> 요소
author[first-name][3]	<first-name> 자식 요소가 있는 세 번째 <author> 요소
my:book	my 네임스페이스의 <book> 요소
my:*	my 네임스페이스의 모든 요소
@my:*	my 네임스페이스의 모든 특성 여기에는 my 네임스페이스의 요소에서 비정규화된 특성이 포함되지 않습니다.

출처 : [https://learn.microsoft.com/ko-kr/previous-versions/ms256086\(v=vs.120\)?redirectedfrom=MSDN](https://learn.microsoft.com/ko-kr/previous-versions/ms256086(v=vs.120)?redirectedfrom=MSDN)

여기서 이미 `beer_item_list` 로 `beer_item` 요소에 접근이 가능하므로 , 그의 자식에 접근할때는

```
# '' 안에 원하는 요소를 입력하면 첫번째 beer-item 의 요소에 바로 접근이가능하다.
cur_beer_item = beer_item_list[0].find_element(By.XPATH, '')
```

csv 파일로 추출하기

csv 파일로 만들기 편하기 위해 pandas 프레임워크와 lxml 라이브러리를 터미널에서 다운받는다



```
pip install pandas
pip install lxml
```

또한 csv 파일을 사용하기위한 코드를 작성한다.

```
import csv

# beer csv 파일 작성을 위한 준비
```

```
f = open("beer_table.csv" , "w" , encoding="utf8",newline="")  
writer = csv.writer(f)
```