

제 14 강 Web Scraping (2)

과업 #2

네이버 쇼핑 신라면 가격 최저가 리스트 만들기

2023년

https://search.shopping.naver.com/search/all?frm=NVSHATC&origQuery=신라면&pagingIndex=1&pagingSize=40&productSet=total&query=신라면&sort=price_asc×tamp=&viewType=list

2024년

https://search.shopping.naver.com/search/all?adQuery=신라면&origQuery=신라면&pagingIndex=1&pagingSize=40&productSet=total&query=신라면&sort=price_asc×tamp=&viewType=list

The screenshot shows the Naver Shopping search results for '신라면' (Shin Ramyun). The search filters on the left include '카테고리' (Category), '키워드추천' (Keyword Recommendation), '가격' (Price), and '배송/혜택/색상' (Delivery/Benefit/Color). The search results are sorted by '낮은 가격순' (Lowest Price First), which is highlighted with a red box. The first product listed is '한봉지 미고랭 핫앤스파이시 라면 400g 야식 신라면 - UnKnown' (One pack, no-boil, hot & spicy ramen 400g, late-night meal, Shin Ramyun - UnKnown) for 10,000 won. The product image shows a red and white package of Shin Ramyun. The search results also show a '추천 낮은 가격순' (Recommended lowest price) banner and a 'TMON' logo.

네이버 접속 및 페이지 획득

```
import requests
from bs4 import BeautifulSoup

word = '신라면'
url =
f'https://search.shopping.naver.com/search/all?origQuery={word}&pagingIndex=1&pagingSize=40&productSet=total&query={word}&sort=price_asc&timestamp=&viewType=list'

# 2023년 작동, BUT 2024년엔 작동하지 않음.
Headers = {
    'User-Agent' : 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36 '
}

# 2024년 작동함.
Headers = {
    'User-Agent' : 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) '
}

r = requests.get(url, headers=headers)
r.raise_for_status()
```

검색 요소 찾아봅시다!

신라면 : 네이버 쇼핑

https://search.shopping.naver.com/search/all?adQuery=신라면&origQuery=신라면&pagingIndex=1&pagingSize=40&productSet=total&query...

전체 41,385

가격비교

네이버페이

백화점홈쇼핑


쇼핑원도

해외직구

· 네이버 랭킹순 · **낮은 가격순** · 높은 가격순 · 리뷰 많은순 · 리뷰 좋은순 · 등록일순

쇼핑물선택 · 상품타입(전체) · 40개씩 보기

추천 낮은 가격순 으로 정렬되었습니다.(1만원 이상) 적용기준



Essential kitchen

농심 신라면 멀티 120g 5개입 1봉 얼큰하게 매운맛 [농심] 신라면 멀티 5입 5개

10,000원 4,000원

가격비교

식품 > 라면/면류 > 라면 > 봉지라면


등록일 2023.12. · ♥ 찜하기 0 · 신고하기

coupang

플래티넘

쿠팡캐시 100원

적립 구매정보



100% Korean

농심 신라면 건면 멀티 5개입 x 2팩 / 총10봉_MC

10,000원 3,000원

식품 > 라면/면류 > 라면 > 봉지라면


신한/롯데/현대/KB국민/삼성 100만원 이상 결제 시 최대 22개월 무이자

등록일 2024.05. · ♥ 찜하기 0 · 신고하기

11D

플래티넘

구매정보



무엇을 상상하든, 그 이상!!
[무료배송]

라면 랜덤박스

라면랜덤박스 진짱봉 너구리 신라면 진라면 짜파게티

10,000원 무료


식품 > 라면/면류 > 라면 > 봉지라면

리뷰 10 (네이버페이 5) · 등록일 2019.01. · ♥ 찜하기 1 · 신고하기

디라이스

Npay+ 1%적립

적립 구매정보



농심

농심 신라면블럭 134g 4개입 멀티팩 간편한

10,000원 3,000원

오늘출발

Gmarket

플래티넘

TOP

네이버 랭킹순 ☒ 낮은 가격순 · 높은 가격순 · 리뷰 많은순 · 리뷰 좋은순 · 등록일순

쇼핑물선택 ▾

i 추천 낮은 가격순 으로 정렬되었습니다.(1만원 이상) 적용기준 ☒

div.product_info_area__xxCTi 577 × 155



농심 신라면블랙 134g 4개입 멀티팩 간편한

10,000원 ~~13,000원~~

식품 > 라면/면류 > 라면 > 봉지라면

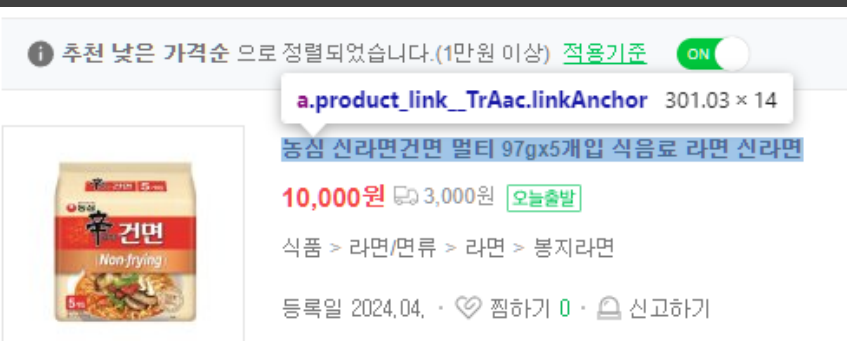
연간 4,900원으로 신세계 유니버스 클럽 가입! 이마트, 스타벅스, SSG.COM가 포함된 6개사의 혜택을 한 번에 누리세요!

등록일 2024.03. · 찜하기 0 · 신고하기

```
▼ <div class="product_inner__gr8QR">
  ▶ <div class="product_img_area__cllrkn">...</div>
  ▼ <div class="product_info_area__xxCTi"> = $0
    ▼ <div class="product_title__Mmw2K">
      <a target="_blank" rel="noopener" class="product_link__TrAac I
    </div>
    ▼ <div class="product_price_area__eTg7l">
      ▼ <strong class="product_price__52o09">
        ▼ <span class="price">
          ▼ <span class="price_num__S2p_v">
            <em>10,000</em>
            "원"
          </span>
        </span>
      </strong>
      ▶ <span class="price_delivery__yw_We">...</span>
```

제품 이름 element

- class 는 product_link_0000
- title 이라는 속성에 이름이 들어있음.
- 링크는 href 에 들어있음.



Copilot 질의



Copilot

class 이름이 `product_link` 으로 시작하고, `title` 속성이 있는 요소를 찾는 CSS selector는 다음과 같습니다:

CSS



```
[class^="product_link"][title]
```

AI-generated code. Review and use carefully. [More info on FAQ.](#)

이 CSS selector는 다음을 의미합니다:

- `[class^="product_link"]` : class 이름이 `product_link` 으로 시작하는 요소를 선택합니다.
- `[title]` : `title` 속성이 있는 요소를 선택합니다.

https://www.w3schools.com/cssref/css_selectors.php

제품 이름 출력

```
elems = soup.select('[class^="product_link"][title]')  
  
for e in elems:  
    print(e['title'])
```


퀴즈: 가격 출력 ?

제품명과 가격 동시에 출력

- 공통 상위 부모 요소를 먼저 선택하고, 하위 요소를 다시 검색

```
elems = soup.select('[class^="product_item"]')
for e in elems:
    title = e.select_one('[title]')['title']
    price = e.select_one('.price').text
    print(f'제품: {title} 가격: {price}')
```

문제점

- JavaScript 를 통한 반응형 웹의 경우, request module 로 제대로 읽어낼 수 없음.

```
▼ <ul class="list_basis">  
  ▼ <div> == $0  
    ▶ <div>...</div>  
    ▶ <div>...</div>  
    ▶ <div>...</div>  
    ▶ <div>...</div>  
    ▶ <div>...</div>  
  </div>  
</ul>
```

Selenium

- 웹 자동화 테스트 도구.
 - 각종 사용자의 입력(클릭, 텍스트입력, 리스트 선택, ...)을 프로그램으로 처리.
- 인터랙티브 웹 사이트 크롤링에 활용.
- 설치
 - selenium 라이브러리 설치 - `pip install selenium`
 - 자동화를 위한 브라우저 프로그램 설치 - working folder 아래에 복사.
 - Chrome driver - PC 에 설치되어 있는 chrome 의 버전에 맞는 드라이버 설치 필요.
 - <https://chromedriver.chromium.org/downloads>

Playwright

- Microsoft 사가 개발한 웹 테스트 및 자동화 프레임워크
- Selenium과의 비교 우위
 - 자동 대기 - 엘리먼트들이 준비될 때까지 대기
 - 테스트 레코딩 & 재생
 - 여러 개의 브라우저에서 동시 실행
- 설치 (2단계) - Command Prompt 활용
 - `pip install pytest-playwright`
 - `python -m playwright install`

네이버 홈

```
from playwright.sync_api import sync_playwright

p = sync_playwright().start()
browser = p.chromium.launch(headless=False)
page = browser.new_page()
page.goto("https://www.naver.com")
print(page.title())
browser.close()
p.stop()
```

쇼핑 페이지 콘텐츠 확보

```
from playwright.sync_api import sync_playwright
from bs4 import BeautifulSoup
import time

word = '신라면'
url =
f'https://search.shopping.naver.com/search/all?origQuery={word}&pagingIndex=1&pagingSize=40&productSet=total&que
ry={word}&sort=price_asc&timestamp=&viewType=list'
user_agent = 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0
Safari/537.36 Edg/124.0.0.0'

p = sync_playwright().start()
browser = p.chromium.launch(headless=False).new_context(
    user_agent=user_agent, viewport={'width':1920, 'height':1080}
)
page = browser.new_page()
page.goto(url)

src_size = 0
while src_size < len(page.content()):
    src_size = len(page.content())
    page.keyboard.press('End')
    time.sleep(1)

soup = BeautifulSoup(page.content(), features='html.parser')
```

Copilot



You

다음 웹 페이지에서 가격을 추출하는 python code를 작성하세요. https://search.shopping.naver.com/search/all?adQuery=%EC%8B%A0%EB%9D%BC%EB%A9%B4&origQuery=%EC%8B%A0%EB%9D%BC%EB%A9%B4&pagingIndex=1&pagingSize=40&productSet=total&query=%EC%8B%A0%EB%9D%BC%EB%A9%B4&sort=price_asc×tamp=&viewType=list#



Copilot

네이버 쇼핑 페이지에서 가격을 추출하는 Python 코드는 BeautifulSoup 라이브러리를 사용하여 작성할 수 있습니다¹². 아래는 간단한 예시입니다:

Python



```
import requests
from bs4 import BeautifulSoup

# 웹 페이지 URL
url = "https://search.shopping.naver.com/search/all?adQuery=%EC%8B%A0%EB%9D%BC%EB%A9%B4&origQuery=%EC%8B%A0%EB%9D%BC%EB%A9%B4&pagingIndex=1&pagingSize=40&productSet=total&query=%EC%8B%A0%EB%9D%BC%EB%A9%B4&sort=price_asc&timestamp=&viewType=list#"

# 웹 페이지 요청
response = requests.get(url)

# BeautifulSoup 객체 생성
soup = BeautifulSoup(response.text, 'html.parser')

# 가격 정보가 있는 요소 선택
price_elements = soup.select('.price_num__2WUXn')

# 각 요소에서 가격 추출
prices = [element.text for element in price_elements]

# 가격 출력
for price in prices:
    print(price)
```

AI-generated code. Review and use carefully. [More info on FAQ.](#)



You

네이버 쇼핑 REST API 를 이용해서 신라면의 최저가 20개를 출력하는 Pytho 코드를 작성



Copilot

네이버 쇼핑 REST API를 사용하여 신라면의 최저가 20개를 출력하는 Python 코드를 작성하려면, 아래와 같이 작성할 수 있습니다.

Python



```
import requests

def get_lowest_prices(product_name, count):
    url = "https://openapi.naver.com/v1/search/shop.json"
    headers = {
        "X-Naver-Client-Id": "<YOUR_CLIENT_ID>",
        "X-Naver-Client-Secret": "<YOUR_CLIENT_SECRET>"
    }
    params = {
        "query": product_name,
        "display": count,
        "sort": "asc"
    }
    response = requests.get(url, headers=headers, params=params)
    result = response.json()
    items = result.get("items", [])
    for item in items:
        print(f"Product: {item['title']}, Lowest Price: {item['lprice']}")

get_lowest_prices("신라면", 20)
```

AI-generated code. Review and use carefully. [More info on FAQ.](#)

설정 - 자바스크립트

chrome://settings/content/javascript

설정

설정 검색

내 Google 서비스 설정

자동 완성 및 비밀번호

개인 정보 보호 및 보안

성능

모양

검색엔진

기본 브라우저

시작 그룹

언어

다운로드

접근성

시스템

설정 초기화

확장 프로그램

Chrome 정보

자바스크립트

검색

사이트에서는 일반적으로 비디오 게임 또는 웹 양식과 같은 대화형 기능을 실행하는 데 JavaScript를 사용합니다.

기본 동작

사이트를 방문할 때 이 설정을 자동으로 따릅니다.

☒ <> 사이트에서 JavaScript를 사용할 수 있음

☐ <> 사이트에서 JavaScript를 사용하도록 허용하지 않음

맞춤설정된 동작

다음 사이트는 기본 설정이 아닌 맞춤 설정을 따릅니다.

JavaScript 사용이 허용되지 않음

[*].localhost

JavaScript 사용이 허용됨

추가된 사이트 없음

추가

추가

- 신라면 이미지 자동 다운로드



img 184 × 184

광고입니다. ⓘ



농심 신라면 (120g X 30개)



농심 신라면 건면 40개

```
<div class="api_popup_wrap">
  <div class="api_flicking_wrap_fe_image_tab_shop_ad_conveyer_container" style="position: relative;
  select: none; -webkit-user-drag: none; touch-action: auto;">
    <div class="flick_bx">
      <div class="ad_inner">
        <a href="https://adcr.naver.com/adcr?x=CPwHel tBSAOW16Us9vnyBf ///w==k7DBowIBxH...tLJqi9+huBNvj:
        W2mv6kiFomM1dCNfpxsbuN" class="thumb_area" target="_blank" onclick="window.tcr("a=sad.img&i=
        ...
        <img src=
        "https://search.pstatic.net/common/?src=https%3A%2F%2Fshopping-phinf.pstatic.net%2Fmain_87
        " width="184" height="184" alt onerror="naver.common.handleImgError(this)"> = $0
        ::after
      </a>
      <a href="https://adcr.naver.com/adcr?x=CPwHel tBSAOW16Us9vnyBf ///w==k7DBowIBxH...tLJqi9+huBNvj:
      W2mv6kiFomM1dCNfpxsbuN" class="info_area" target="_blank" onclick="window.tcr("a=sad.txt&i=
      </a>
    </div>
  </div>
```

```
word = '신라면'
url =
f'https://search.naver.com/search.naver?ssc=tab.image.all&where=image&sm=tab_jum&query={word}'
user_agent = 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/124.0.0.0 Safari/537.36 Edg/124.0.0.0'

p = sync_playwright().start()
browser = p.chromium.launch(headless=False).new_context(
    user_agent=user_agent, viewport={'width':1920, 'height':1080}
)
page = browser.new_page()
page.goto(url)

soup = BeautifulSoup(page.content(), features='html.parser')

elms = soup.select('._fe_image_tab_content_thumbnail_image')
for e in elms:
    print(e['src'])
```

request 를 통한 이미지 다운로드

```
import uuid
import mimetypes

def download_image(image_url):
    r = requests.get(image_url)
    r.raise_for_status()
    ext = mimetypes.guess_extension(r.headers['Content-Type'])
    fname = f'img_{uuid.uuid4().int}{ext}'
    print(f'Downloading {fname} .....')
    with open(fname, 'wb') as wf:
        print(f'Downloading {fname} .....')
        wf.write(r.content)

for e in elms:
    print(e)
    download_image(e['src'])
```

로딩 늦어지는 리소스에 대한 처리

```
for e in elms:  
    download_image(e.get('data-lazysrc', e['src']))
```