# 셀레니움

사용자가 사용하는 웹 드라이버가 필요함 크롬을 사용하기 때문에 크롬 드라이버 설치 ->

In [15]:
```python
# python selenium 설치하기
!pip install selenium
```

Requirement already satisfied: selenium in /home/user/anaconda3/lib/python3.7/site-packages (4.11.2)
Requirement already satisfied: trio~=0.17 in /home/user/anaconda3/lib/python3.7/site-packages (from selenium) (0.22.2)
Requirement already satisfied: urllib3[socks]<3,>=1.26 in /home/user/anaconda3/lib/python3.7/site-packages (from selenium) (2.0.7)
Requirement already satisfied: certifi>=2021.10.8 in /home/user/anaconda3/lib/python3.7/site-packages (from selenium) (2023.11.17)
Requirement already satisfied: trio-websocket~=0.9 in /home/user/anaconda3/lib/python3.7/site-packages (from selenium) (0.11.1)
Requirement already satisfied: exceptiongroup>=1.0.0rc9; python_version < "3.11" in /home/user/anaconda3/lib/python3.7/site-packages (from trio~=0.17->selenium) (1.1.3)
Requirement already satisfied: idna in /home/user/anaconda3/lib/python3.7/site-packages (from trio~=0.17->selenium) (2.8)
Requirement already satisfied: sniffio in /home/user/anaconda3/lib/python3.7/site-packages (from trio~=0.17->selenium) (1.3.0)
Requirement already satisfied: attrs>=20.1.0 in /home/user/anaconda3/lib/python3.7/site-packages (from trio~=0.17->selenium) (23.1.0)
Requirement already satisfied: sortedcontainers in /home/user/anaconda3/lib/python3.7/site-packages (from trio~=0.17->selenium) (2.1.0)
Requirement already satisfied: outcome in /home/user/anaconda3/lib/python3.7/site-packages (from trio~=0.17->selenium) (1.3.0.post0)
Requirement already satisfied: pysocks!=1.5.7,<2.0,>=1.5.6; extra == "socks" in /home/user/anaconda3/lib/python3.7/site-packages (from urllib3[socks]<3,>=1.26->selenium) (1.7.1)
Requirement already satisfied: wsproto>=0.14 in /home/user/anaconda3/lib/python3.7/site-packages (from trio-websocket~=0.9->selenium) (1.2.0)
Requirement already satisfied: importlib-metadata; python_version < "3.8" in /home/user/anaconda3/lib/python3.7/site-packages (from attrs>=20.1.0->trio~=0.17->selenium) (0.23)
Requirement already satisfied: h11<1,>=0.9.0 in /home/user/anaconda3/lib/python3.7/site-packages (from wsproto>=0.14->trio-websocket~=0.9->selenium) (0.14.0)
Requirement already satisfied: zipp>=0.5 in /home/user/anaconda3/lib/python3.7/site-packages (from importlib-metadata; python_version < "3.8"->attrs>=20.1.0->trio~=0.17->selenium) (0.6.0)
Requirement already satisfied: typing-extensions; python_version < "3.8" in /home/user/anaconda3/lib/python3.7/site-packages (from h11<1,>=0.9.0->wsproto>=0.14->trio-websocket~=0.9->selenium) (4.7.1)
Requirement already satisfied: more-itertools in /home/user/anaconda3/lib/python3.7/site-packages (from zipp>=0.5->importlib-metadata; python_version < "3.8"->attrs>=20.1.0->trio~=0.17->selenium) (7.2.0)

In [24]:
```
!pip install pandas numpy tqdm selenium webdriver_manager beautiful
soup4
```

In [24]:
```
!pip install pandas numpy tqdm selenium webdriver_manager beautiful
soup4
```

In [34]:
```python
import csv
import pandas as pd
import numpy as np
import time
import requests

from tqdm import tqdm
from selenium import webdriver
from webdriver_manager.chrome import ChromeDriverManager
from selenium.webdriver.common.by import By
from bs4 import BeautifulSoup
from selenium.webdriver.common.keys import Keys
from itertools import repeat  # 반복 가능한 데이터를 다루는데 도움을 주는 도구 제공
# repeat 함수 : 지정된 값을 무한히 반복하는 이터레이터 생성
from urllib.request import urlopen  # URL을 열고 읽는데 사용되는 함수 제공
# urlopen 함수 : 주어진 URL을 열고 해당 URL에서 읽은 데이터 반환
import re  # 정규 표현식을 사용하여 문자열을 처리하는 함수 제공

# 드라이버 생성 설치 경로 정확하게 기재 ( 드라이버 로드)
chromedriver = '/home/user/다운로드'

# 드라이버 사용하겠다.
driver = webdriver.Chrome()

# #크롤링할 사이트 무신사 호출
# url = "https://www.musinsa.com/app/"
# driver.get(url)
# time.sleep(1)

# 무신사 스토어 > 상의( 무신사 추천순) 해당 url오픈
html = urlopen("https://www.musinsa.com/categories/item/001")
bsObject = BeautifulSoup(html, "html.parser")

print(bsObject)

# 상품들의 정보가 담긴 li_box 모두 가져오기
item_list = bsObject.findAll('li', {'class': 'li_box'})
print(item_list)

# 빈 리스트
brand = []
name = []
price = []
number = []
number_box = 0 # 순위 카운트

for item in item_list:

    # 브랜드
    brand_box = item.findAll('p', {'class': 'item_title'})
    if len(brand_box) == 1:
        brand.append(brand_box[0].get_text())
    elif len(brand_box) == 2:
        brand.append(brand_box[1].get_text())

    # 상품명 담기
    name_box = item.find('a', {'class': 'img-block'}).get('title')
    print(name_box)
    name.append(name_box.strip())
```

```python
    # 가격 담기
    price_box = item.find('p', {'class': 'price'}).get_text().split
()
    if len(price_box) == 1:
        price.append(price_box[0])
    elif len(brand_box) == 2:
        price.append(price_box[1])

    # 순위 담기
    number_box += 1
    number.append(str(number_box))

# 확인
print(brand)
print(name)
print(price)

if my_variable is not None:
    # 변수가 None이 아닌 경우에만 strip 등의 메서드 호출
    stripped_value = my_variable.strip()
    # 나머지 코드 실행
else:
    # 변수가 None인 경우에 대한 처리
    print("변수가 None입니다.")
```

```
<!DOCTYPE html>

<html lang="ko">
<head>
<script type="text/javascript">
        var dataLayer = dataLayer || [];
    </script>
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
        new Date().getTime(),event:'gtm.js'});var f=d.getElemen
tsByTagName(s)[0],
        j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=t
rue;j.src=
        'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parent
Node.insertBefore(j,f);
    })(window,document,'script','dataLayer','GTM-TXDSFSF');</script
>
<!-- End Google Tag Manager -->
<script>
        dataLayer.push({
            mainInfo: {
                'email': ""
            }
        });
    </script>
<title>상의 | 무신사 추천 브랜드</title>
<meta charset="utf-8"/>
<meta content="IE=edge" http-equiv="X-UA-Compatible">
<meta content="width=device-width, initial-scale=1" name="viewpor
t"/>
<meta content="무신사에서 상의 관련 상품 리스트와 쇼핑 기획에 대한 자세한 정보를 확
인하세요." name="description"/>
<!-- 페이스북 오픈 그래프 메타태그  -->
<meta content="648837775133366" id="fbAddId" property="fb:app_id">
<meta content="website" id="fbOgType" property="og:type">
<meta content="https://www.musinsa.comnull?_gf=A" id="fbOgUrl" prop
erty="og:url"/>
<meta content="상의 | 무신사 추천 브랜드" id="fbOgTitle" property="og:tit
le"/>
<meta content="https://image.msscdn.net/mfile_s01/fb/share_musinsa.
png" id="fbOgImage" property="og:image"/>
<meta content="무신사에서 상의 관련 상품 리스트와 쇼핑 기획에 대한 자세한 정보를 확
인하세요." id="fbOgDescription" property="og:description"/>
<!--// 페이스북 오픈 그래프 메타태그 -->
<link href="//image.msscdn.net/favicon.ico?20200213" rel="shortcut
icon" type="image/x-icon"/>
<link href="//static.msscdn.net/platform/css/common.css" rel="style
sheet"/>
<link href="//image.msscdn.net/ui/musinsa/resources/common/css/ico
n.min.css?1697673600000" rel="stylesheet"/>
<link href="//static.msscdn.net/ui/build/pc/css/common.css" rel="st
ylesheet"/>
<link href="//static.msscdn.net/ui/build/pc/css/review.css" rel="st
ylesheet"/>
<link href="//static.msscdn.net/skin/musinsa/css/magazine_common.cs
s?1697673600000" rel="stylesheet" type="text/css">
<link href="//static.msscdn.net/skin/musinsa/css/store_common.css?1
697673600000" rel="stylesheet" type="text/css">
<link href="//static.msscdn.net/skin/musinsa/css/layout.min.css?169
7673600000" rel="stylesheet" type="text/css">
<link href="//static.msscdn.net/skin/musinsa/css/guide.min.css?1697
```

```
</p>
<p class="txt_cnt_like" data-goodsno="2415381"></p>
</div>
</div>
</li>, <li class="li_box" data-no="2860619">
<!-- 상단 좌측 라벨 -->
<div class="icon_new">SALE 55%</div>
<!-- 상단 우측 라벨 -->
<div class="box-icon-right">
<span class="icon-reverse label-outlet">아울렛</span>
</div>
<div class="icon_group">
<ul>
<li class="icon_man sight_out" title="남성">남성</li>
</ul>
</div>
<div class="li_inner">
<div class="list_img">
<a class="img-block" href="//www.musinsa.com/app/goods/2860619" nam
e="goods_link" title="MLU2A5TS1996 ~XXL  빅사이즈 프린트로고 후드티">
<img alt="머렐(MERRELL) MLU2A5TS1996 ~XXL  빅사이즈 프린트로고 후드티" clas
s="lazyload lazy" data-original="https://image.msscdn.net/images/go
ods_img/20221013/2860619/2860619_2_125.jpg" src="//image.msscdn.ne
t/images/no_image_125.png"/>
</a>
</div>
<div class="article_info">
<p class="item_title"><a href="//www.musinsa.com/brands/merrell">머렐
</a></p>
<p class="list_info">
<a href="//www.musinsa.com/app/goods/2860619" name="goods_link" tit
le="MLU2A5TS1996 ~XXL  빅사이즈 프린트로고 후드티">
<strong class="txt_reserve"></strong>
                        MLU2A5TS1996 ~XXL  빅사이즈 프린트로고 후드티
                    </a>
</p>
<p class="price">
<del>69,000원</del>
                    31,050원
                </p>
<p class="membership" onclick="viewMemberPrice('sPrice_2860619', '2
860619', '0');">MEMBERSHIP PRICE<span class="fr">▼</span></p>
<div class="member_price" id="sPrice_2860619" style="display:none;
z-index: 9999;">
<p>무신사 회원가<span class="fr"><a href="#" onclick="viewMemberPrice
('sPrice_2860619', '2860619', '0');return false;">X</a></span></p>
<ul>
<li class="pertinent" style="border-bottom:1px solid #ddd;">
<span>&gt; </span>
                        비회원
                        <span class="txt_price_member">31,050원</span></
li>
</ul>
</div>
<p class="txt_cnt_like" data-goodsno="2860619"></p>
</div>
</div>
</li>]
None
```

```
---------------------------------------------------------------------
--------
AttributeError                                  Traceback (most recent ca
ll last)
<ipython-input-34-917a1c9ec237> in <module>
     57        name_box = item.find('a', {'class': 'img-block'}).get
('title')
     58        print(name_box)
---> 59        name.append(name_box.strip())
     60
     61        # 가격 담기

AttributeError: 'NoneType' object has no attribute 'strip'
```

In [29]:
```
# ㅋㅋ..한번에 주석달기 ctrl /
```

In [30]:
```
# # 카테고리 설정 및 페이지 접근
# # 하의 크롤링

# def get_from_page_musinsa_no(b_catagory_no, start_page, page):
#        musinsa_link_li = []  # 무신사 제품 no 리스트

#        # GET page 개수만큼 무신사 페이지에서 무신사 제품 no
#        for p in tqdm(range(start_page, start_page+page)):
#            url = f'https://www.musinsa.com/categories/item/
{b_catagory_no}?d_cat_cd={b_catagory_no}&brand=&list_kind=small&sor
t=pop_category&sub_sort=&page={p}&display_cnt=90\
#                        &group_sale=&exclusive_yn=&sale_goods=&ti
mesale_yn=&ex_soldout=&plusDeliveryYn=&kids=&color=&price1=&price2=
&shoeSizeOption=&tags=&campaign_id=&includeKeywords=&measure='
#            driver.get(url)
#            time.sleep(0.1)
#            link1 = 'https://www.musinsa.com/app/goods/'

#            # GET p페이지에서 무신사 제품 no 리스트
#            temp_no_li = []
#            temp = driver.find_elements(By.XPATH, '//*[@id="s
earchList"]/*[@class="li_box"]')
#            for code in temp:
#                link2 = code.get_attribute('data-no')
#                temp_no_li.append(link1 + link2)

#            musinsa_link_li += temp_no_li

#        return musinsa_link_li
```

In [ ]: