

6. 어떤 무선 청소기가 인기가 좋을까?

데이터 수집

```
# 검색 결과 웹 페이지의 URL: https://prod.danawa.com/list/?cate=102207&shortcutKeyword=무선청소
# selenium으로 크롬 브라우저를 생성하고 '무선 청소기'에 대한 다나와 검색 결과 페이지 URL로 접속
```

```
# 구글 드라이브 마운트
from google.colab import drive
drive.mount('/content/drive')
```

 Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

Selenium 설치 & 내 구글 드라이브에 chromedriver 설치

```
!pip install selenium
!apt-get update
```

```
Downloading selenium-4.23.1-py3-none-any.whl.metadata (7.1 kB)
Requirement already satisfied: urllib3<3,>=1.26 in /usr/local/lib/python3.10/dist-packages (from urllib3[socks]<3,>=1.26->selenium) (2.0.7)
Collecting trio~=0.17 (from selenium)
  Downloading trio-0.26.1-py3-none-any.whl.metadata (8.6 kB)
Collecting trio-websocket~=0.9 (from selenium)
  Downloading trio_websocket-0.11.1-py3-none-any.whl.metadata (4.7 kB)
Requirement already satisfied: certifi>=2021.10.8 in /usr/local/lib/python3.10/dist-packages (from selenium) (2024.7.4)
Requirement already satisfied: typing_extensions~=4.9 in /usr/local/lib/python3.10/dist-packages (from selenium) (4.12.2)
Requirement already satisfied: websocket-client~=1.8 in /usr/local/lib/python3.10/dist-packages (from selenium) (1.8.0)
Requirement already satisfied: attrs>=23.2.0 in /usr/local/lib/python3.10/dist-packages (from trio~=0.17->selenium) (23.2.0)
Requirement already satisfied: sortedcontainers in /usr/local/lib/python3.10/dist-packages (from trio~=0.17->selenium) (2.4.0)
Requirement already satisfied: idna in /usr/local/lib/python3.10/dist-packages (from trio~=0.17->selenium) (3.7)
Collecting outcome (from trio~=0.17->selenium)
  Downloading outcome-1.3.0.post0-py2.py3-none-any.whl.metadata (2.6 kB)
Requirement already satisfied: sniffio>=1.3.0 in /usr/local/lib/python3.10/dist-packages (from trio~=0.17->selenium) (1.3.1)
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from trio~=0.17->selenium) (1.2.2)
Collecting wsproto>=0.14 (from trio-websocket~=0.9->selenium)
  Downloading wsproto-1.2.0-py3-none-any.whl.metadata (5.6 kB)
Requirement already satisfied: pysocks!=1.5.7,<2.0,>=1.5.6 in /usr/local/lib/python3.10/dist-packages (from urllib3[socks]<3,>=1.26->selenium)
Collecting h11<1,>=0.9.0 (from wsproto>=0.14->trio-websocket~=0.9->selenium)
  Downloading h11-0.14.0-py3-none-any.whl.metadata (8.2 kB)
Downloading selenium-4.23.1-py3-none-any.whl (9.4 MB)
----- 9.4/9.4 MB 58.6 MB/s eta 0:00:00
Downloading trio-0.26.1-py3-none-any.whl (475 kB)
----- 475.7/475.7 kB 27.5 MB/s eta 0:00:00
Downloading trio_websocket-0.11.1-py3-none-any.whl (17 kB)
Downloading wsproto-1.2.0-py3-none-any.whl (24 kB)
Downloading outcome-1.3.0.post0-py2.py3-none-any.whl (10 kB)
Downloading h11-0.14.0-py3-none-any.whl (58 kB)
----- 58.3/58.3 kB 4.3 MB/s eta 0:00:00
Installing collected packages: outcome, h11, wsproto, trio, trio-websocket, selenium
Successfully installed h11-0.14.0 outcome-1.3.0.post0 selenium-4.23.1 trio-0.26.1 trio-websocket-0.11.1 wsproto-1.2.0
Get:1 https://cloud.r-project.org/bin/linux/ubuntu/jammy-cran40/ InRelease [3,626 B]
Get:2 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86\_64 InRelease [1,581 B]
Get:3 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86\_64 Packages [908 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Hit:5 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Ign:7 https://r2u.stat.illinois.edu/ubuntu jammy InRelease
Get:8 https://r2u.stat.illinois.edu/ubuntu jammy Release [5,713 B]
Get:9 https://ppa.launchpadcontent.net/deadsnakes/ppa/ubuntu jammy InRelease [18.1 kB]
Get:10 https://r2u.stat.illinois.edu/ubuntu jammy Release.gpg [793 B]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [2,118 kB]
Get:12 https://ppa.launchpadcontent.net/graphics-drivers/ppa/ubuntu jammy InRelease [24.3 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Hit:14 https://ppa.launchpadcontent.net/ubuntugis/ppa/ubuntu jammy InRelease
Get:15 https://r2u.stat.illinois.edu/ubuntu jammy/main amd64 Packages [2,552 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1,421 kB]
Get:17 https://ppa.launchpadcontent.net/deadsnakes/ppa/ubuntu jammy/main amd64 Packages [27.8 kB]
Get:18 https://ppa.launchpadcontent.net/graphics-drivers/ppa/ubuntu jammy/main amd64 Packages [49.2 kB]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [1,131 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2,787 kB]
Get:21 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2,884 kB]
Get:22 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2,396 kB]
Get:23 https://r2u.stat.illinois.edu/ubuntu jammy/main all Packages [8,218 kB]
Fetched 24.9 MB in 11s (2,350 kB/s)
Reading package lists... Done
W: Skipping acquire of configured file 'main/source/Sources' as repository 'https://r2u.stat.illinois.edu/ubuntu jammy InRelease' does not seem to have a Files section.
```

```
!apt install chromium-chromedriver
!cp /usr/lib/chromium-browser/chromedriver '/content/drive/MyDrive/Colab Notebooks' #
!pip install chromedriver-autoinstaller
```

```

Setting up squashfs-tools (1:4.5-3build1) ...
Setting up udev (249.11-0ubuntu3.12) ...
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Setting up libfuse3-3:amd64 (3.10.5-1build1) ...
Setting up snapd (2.63+22.04ubuntu0.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/snapd.apparmor.service → /lib/systemd/system/snapd.apparmor.service.
Created symlink /etc/systemd/system/multi-user.target.wants/snapd.autoimport.service → /lib/systemd/system/snapd.autoimport.service.
Created symlink /etc/systemd/system/multi-user.target.wants/snapd.core-fixup.service → /lib/systemd/system/snapd.core-fixup.service.
Created symlink /etc/systemd/system/multi-user.target.wants/snapd.recovery-chooser-trigger.service → /lib/systemd/system/snapd.recovery-chooser-trigger.service.
Created symlink /etc/systemd/system/multi-user.target.wants/snapd.seeded.service → /lib/systemd/system/snapd.seeded.service.
Created symlink /etc/systemd/system/cloud-final.service.wants/snapd.seeded.service → /lib/systemd/system/snapd.seeded.service.
Unit /lib/systemd/system/snapd.seeded.service is added as a dependency to a non-existent unit cloud-final.service.
Created symlink /etc/systemd/system/multi-user.target.wants/snapd.service → /lib/systemd/system/snapd.service.
Created symlink /etc/systemd/system/timers.target.wants/snapd.snap-repair.timer → /lib/systemd/system/snapd.snap-repair.timer.
Created symlink /etc/systemd/system/sockets.target.wants/snapd.socket → /lib/systemd/system/snapd.socket.
Created symlink /etc/systemd/system/final.target.wants/snapd.system-shutdown.service → /lib/systemd/system/snapd.system-shutdown.service.
Selecting previously unselected package chromium-browser.
(Reading database ... 124027 files and directories currently installed.)
Preparing to unpack .../chromium-browser_1%3a85.0.4183.83-0ubuntu2.22.04.1_amd64.deb ...
=> Installing the chromium snap
==> Checking connectivity with the snap store
==> System doesn't have a working snapd, skipping
Unpacking chromium-browser (1:85.0.4183.83-0ubuntu2.22.04.1) ...
Selecting previously unselected package chromium-chromedriver.
Preparing to unpack .../chromium-chromedriver_1%3a85.0.4183.83-0ubuntu2.22.04.1_amd64.deb ...
Unpacking chromium-chromedriver (1:85.0.4183.83-0ubuntu2.22.04.1) ...
Selecting previously unselected package systemd-hwe-hwdb.
Preparing to unpack .../systemd-hwe-hwdb_249.11.5_all.deb ...
Unpacking systemd-hwe-hwdb (249.11.5) ...
Setting up systemd-hwe-hwdb (249.11.5) ...
Setting up chromium-browser (1:85.0.4183.83-0ubuntu2.22.04.1) ...
update-alternatives: using /usr/bin/chromium-browser to provide /usr/bin/x-www-browser (x-www-browser) in auto mode
update-alternatives: using /usr/bin/chromium-browser to provide /usr/bin/gnome-www-browser (gnome-www-browser) in auto mode
Setting up chromium-chromedriver (1:85.0.4183.83-0ubuntu2.22.04.1) ...
Processing triggers for udev (249.11-0ubuntu3.12) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for libc-bin (2.35-0ubuntu3.4) ...
/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc_proxy.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_5.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_0.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbb.so.12 is not a symbolic link

Processing triggers for man-db (2.10.2-1) ...
Processing triggers for dbus (1.12.20-2ubuntu4.1) ...
Collecting chromedriver-autoinstaller
  Downloading chromedriver-autoinstaller-0.6.4-py3-none-any.whl.metadata (2.1 kB)
Requirement already satisfied: packaging>=23.1 in /usr/local/lib/python3.10/dist-packages (from chromedriver-autoinstaller) (24.1)
Downloading chromedriver-autoinstaller-0.6.4-py3-none-any.whl (7.6 kB)
Installing collected packages: chromedriver-autoinstaller
Successfully installed chromedriver-autoinstaller-0.6.4

```

```
!python --version
```

```
import selenium
print(selenium.__version__)
```

```
Python 3.10.12
4.31.1
```

라이브러리 임포트

```

from selenium import webdriver
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
import sys
from selenium.webdriver.common.keys import Keys
import urllib.request
import os
from urllib.request import urlretrieve

```

```

import time
import pandas as pd
import chromedriver_autoinstaller # setup chrome options

```

chrome_options 설정

```
chrome_path = "/content/drive/MyDrive/Colab Notebooks/chromedriver"
```

```
sys.path.insert(0,chrome_path)
chrome_options = webdriver.ChromeOptions()
chrome_options.add_argument('--headless') # ensure GUI is off
chrome_options.add_argument('--no-sandbox')
chrome_options.add_argument('--disable-dev-shm-usage') # set path to chromedriver as per your configuration
chrome_options.add_argument('lang=ko_KR') # 한국어
```

```
chromedriver_autoinstaller.install() # set the target URL
```

```
driver = webdriver.Chrome(options=chrome_options)
```

```
url='https://prod.danawa.com/list/?cate=102207&shortcutKeyword=무선청소기'
driver.get(url)
```

다나와 검색 웹 페이지에서 상품 정보 가져오기

데이터를 수집하기에 앞서 다나와 상품 검색 페이지를 살펴보겠습니다.
상품 검색 결과가 여러 페이지에 걸쳐 있으며, 각 페이지에 동일한 형식으로 상품이 표시됩니다.
검색 결과의 1페이지에 나온 30개의 상품 정보를 수집.

웹 페이지의 HTML 정보 가져오기

```
from bs4 import BeautifulSoup
html=driver.page_source
soup=BeautifulSoup(html,'html.parser')
```

검색 결과의 1페이지를 구성하는 HTML 정보를 가져와 BeautifulSoup으로 읽습니다.

```
prod_items=soup.select('li.prod_item')
len(prod_items)
```

```
# <li class="prod_item prod_layer" id="productItem41827157">
```

```
31
```

1페이지에 대한 무선청소기 정보 가져오기

현재 페이지에서 우리가 찾고자 하는 상품의 개수는 30개인데, 앞서 검색한 태그는
31개가 있는 것으로 나옵니다. 우리가 찾는 30개의 상품만 조회하도록 상위 태그 정보를 추가해서 검색.
prod_items=soup.select('ul.product_list > li.prod_item')
len(prod_items)

```
# <div class="main_prodlst main_prodlst_list">
# <ul class="product_list">
# <li class="prod_item prod_layer" id="productItem41827157">
```

```
31
```

31

```
print(prod_items[0])
```



```

<div class="prod_pricelist" data-cate="/10010110">
<!-- <ul> -->
<ul>
<li id="productInfoDetail_41827157">
<div class="box__vs-tooltip" style="display: block;"><div class="box__contents"><em>VS검색</em> 담고 <em>스펙 비교하기</em></div><button class=
427물
<label for="prodCompareCheck_41827157"><input id="prodCompareCheck_41827157" onmousedown="_trkEventLog
<input name="compareValue" type="hidden" value="41827157"/>
<input name="compareRepValue" type="hidden" value="41827157"/>
</p>
<p class="price_sect">
<a href="https://prod.danawa.com/info/?pcode=41827157&cate=102207" onmousedown="_trkEventLog('15상품리스트_가격')" target="_blank">
<strong>1,156,120</strong>원
</a>

<span class="more_btn_wrap">
<button class="ico_i_more" data-producttype="standard">가격정보 더보기</button>
<span class="layer_price_more" id="layer_price_more_41827157"></span>
<span class="sep_line"></span>
</span>
</p>
<input id="min_price_41827157" type="hidden" value="1,156,120"/>
<div class="over_preview">
<p class="memory_sect">
<span class="text"><카빙베이지, AX958BWE</span> <a href="https://prod.danawa.com/info/?pcode=41827157&cate=102207" onmousedown="_trkEventLo
</a>
</p>
</div>
</li>
</ul>
</div>
</div>
</li>

```



상품명

상품명 가져오기: <p class="prod_name"> dkfodml <a> 태그에 상품명 정보가 들어있기 때문에
select('p.prod_name > a')[0].text 명령으로 상품명 가져올 수 있습니다.

```

# <p class="prod_name">
# <strong class="pop_rank"> ... </strong>

```

```

# <a href="https://prod.danawa.com/info/?pcode=41827157&cate=102207" target="_blank" onmousedown="_trkEventLog('15상품리스트_상품명')">
# name="productName"> LG전자 오브제컬렉션 코드제로 A9S AX958
</a>

```

```

title=prod_items[0].select('p.prod_name>a')[0].text.strip()
print(title)

```

```

title1=prod_items[1].select('p.prod_name>a')[0].text.strip()
print(title1)

```

LG전자 오브제컬렉션 코드제로 A9S AX958
삼성전자 비스포크 제트 VS20B956AX

스펙 목록

스펙 목록 가져오기
<div class='spec_list'> 태그에 스펙 목록 정보가 들어있습니다.
spec_list=prod_items[0].select('div.spec_list')[0].text.strip()
print(spec_list)

```

spec_list1=prod_items[1].select('div.spec_list')[0].text.strip()
print(spec_list1)

```

핸드스틱청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0/먼지비용/충전/UVC LED/액세서리수납/스탠드거치/먼지비용시간: 30초/브러
핸드스틱청소기/무선/흡입형/흡입력: 220W/스테이션: 청정스테이션/먼지비용/충전/UVC LED/스탠드거치/먼지비용시간: 14초/브러쉬: 바둑/물걸레: 별매/솔



가격 정보 가져오기

다나와에서는 동일한 모델이라도 옵션별로 다양한 가격이 조회됩니다.
이 가운데 1위 가격을 가져오기 위해 1위 옵션 태그의 위치를 살펴보겠습니다.

```
# 스펙 목록 가져오기
# <div class='spec_list'> 태그에 스펙 목록 정보가 들어있습니다.
spec_list=prod_items[0].select('div.prod_pricelist')[0].text
print(spec_list)
```



VS검색 담고 스펙 비교하기달기

427몰
VS검색

1,156,120원

가격정보 더보기

카테고리, AX958BWE

```
# 스펙 목록 가져오기
# <div class='spec_list'> 태그에 스펙 목록 정보가 들어있습니다.
spec_list=prod_items[0].select('p.price_sect>a')[0].text.strip()
print(spec_list)
```

```
spec_list1=prod_items[1].select('p.price_sect>a')[0].text.strip()
print(spec_list1)
```



1,156,120원
478,930원

The screenshot shows a web browser displaying LG product listings. The first product is 'LG전자 오브제컬렉션 코드제로 A9S AX958'. The second product is '삼성전자 비스포크 제트 VS20B956AX'. The third product is 'LG전자 오브제컬렉션 코드제로 A9 AU9202WD'. The HTML source code on the right shows the product details for the second product, '삼성전자 비스포크 제트 VS20B956AX'. The HTML code includes the product name, price, and other details. The product name is '삼성전자 비스포크 제트 VS20B956AX' and the price is '478,950원'. The HTML code also includes the product name '삼성전자 비스포크 제트 VS20B956AX' and the price '478,950원'.

반복문으로 검색 결과의 1페이지에 대한 상품 정보 추출

```
prod_data=[]
for prod_item in prod_items[1]:
    try: # 상품명 가져오기
        title=prod_item.select('p.prod_name>a')[0].text.strip()
    except:
        title=''

    try: # 스펙 목록 가져오기
        spec_list=prod_item.select('div.spec_list')[0].text.strip()
    except:
        spec_list=''

    prod_data.append([title, spec_list])
```

```
print(len(prod_data))
```

```
print(prod_data)
```

```
5
[['', '1,156,120원'], ['', '1,156,120원'], ['', '1,156,120원'], ['삼성전자 비스포크 제트 VS20B956AX', '핸디스틱청소기/무선/흡입형/흡입력: 220W/스
```

```
prod_data=[]
for prod_item in prod_items[0:30]:
    try: # 상품명 가져오기
        title=prod_item.select('p.prod_name>a')[0].text.strip()
        # print('title값:',title)
    except:
        title=''

    try: # 스펙 목록 가져오기
        spec_list=prod_item.select('div.spec_list')[0].text.strip()
    except:
        spec_list=''

    try: # 가격 정보 가져오기
        price=prod_items[0].select('p.price_sect>a')[0].text.strip()
    except:
        price=0
    prod_data.append([title, spec_list, price])
```

```
print(len(prod_data))
```

```
print(prod_data)
```

검색 결과의 1페이지에 존재하는 총 30개의 결과가 잘 정리 된 것을 확인할 수 있습니다.

상품별로 수집할 정보(상품명, 스펙 목록, 가격)의 값이 없는 경우가 존재할 수도 있으므로 try/except 구문 활용.

```
30
[['LG전자 오브제컬렉션 코드제로 A9S AX958', '핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0/먼지비움/충전/UVC LED/액,
```

상품 정보 태그에서 원하는 정보를 추출하는 함수

```
def get_prod_items(prod_items):
    prod_data=[]

    for prod_item in prod_items[0:30]:
        try: # 상품명 가져오기
            title=prod_item.select('p.prod_name>a')[0].text.strip()
            # print('title값:',title)
        except:
            title=''

        try: # 스펙 목록 가져오기
            spec_list=prod_item.select('div.spec_list')[0].text.strip()
        except:
            spec_list=''

        try: # 가격 정보 가져오기
            price=prod_items[0].select('p.price_sect>a')[0].text.strip()
        except:
            price=0
        prod_data.append([title, spec_list, price])

    print(len(prod_data))
    print(prod_data)

    return prod_data
# 각 상품의 수집 항목(상품명, 스펙 목록, 가격 정보)을 추출한 결과 리스트인 prod_data가 반환됩니다.

prod_items=soup.select('div.main_prodlist>ul.product_list>li.prod_item')
prod_items
```




```

<div class="hnx view-more">
  </div>

```

```

prod_data=get_prod_items(prod_items)
print(len(prod_data))

```

```

30
[[ 'LG전자 오브제컬렉션 코드제로 A9S AX958', '핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0/먼지비움/충전/UVC LED/액.
30

```

데이터 수집: 여러 페이지에 걸친 다나와 검색 페이지 크롤링

이번 절에서는 이 함수와 반복문을 이용해 전체 페이지의 상품 정보 데이터를 수집.
다나와는 page가 url에 따라 달라지지 않게 바뀌어서 네이버 쇼핑으로 변경

page1) pagingIndex=1: 페이지수

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

page2)

https://search.shopping.naver.com/search/all?adQuery=무선청소기&or igQuery=무선청소기&pagingIndex=2&pagingSize=40&productSet=total&query=무선청소기&

page3)

https://search.shopping.naver.com/search/all?adQuery=무선청소기&or igQuery=무선청소기&pagingIndex=3&pagingSize=40&productSet=total&query=무선청소기&sort

★ 4.89 (238) · 찜 52 · 등록일 2022.06. · 정보 수정요청



일렉트로룩스 WQ71-201B
최저 213,850원 무료 판매처 58
 디지털가전 > 청소기 > 무선청소기 > 핸디스틱청소기
 형태: 하중심 | 출시년도: 2020년 | 집진방식: 싸이클론 | 작동방식: 회전식 | 청소방식: 흡입전용
 | 흡입력조절: 2단계 | 사용시간: 50분 | 충전시간: 4시간30분 | 충전방식: 스탠드충전
 ★ 4.78 (664) · 찜 139 · 등록일 2020.10. · 정보 수정요청

브랜드 카탈로그

| | |
|------------|-----------|
| 쿠팡 | ↓ 213,850 |
| G마켓 | 213,850 |
| 옥션 | 215,000 |
| 롯데백화점 | 229,000 |
| elelctro Z | 229,000 |



삼성전자 비스포크제트 250W VS25C972DRH
최저 739,120원 무료 판매처 79
 디지털가전 > 청소기 > 무선청소기 > 핸디스틱청소기
 형태: 상중심 | 모터: 디지털인버터모터 | 출시년도: 2023년 | 집진방식: 제트싸이클론
 | 작동방식: 회전식 | 청소방식: 흡입+걸레겸용 | 흡입력: 250W | 흡입력조절: 5단계
 ★ 4.87 (102) · 찜 72 · 등록일 2023.04. · 정보 수정요청

브랜드 카탈로그

| | |
|----------|-----------|
| G마켓 | ↓ 739,120 |
| 삼성공식파... | 799,000 |
| 롯데ON | 819,870 |
| 11번가 | 842,230 |
| SSG닷컴 | 842,380 |

① 네이버쇼핑에서는 각쇼핑몰에서 받은 상품 정보만을 제공하며, 쇼핑몰의 정보와 일치하지 않을 수 있으므로 반드시 해당 쇼핑몰에서 정확한 정보를 확인하시기 바랍니다. [법적고지 보기](#) >
 포인트최종적립금액은 네이버페이 포인트 사용, 쿠폰 사용 여부 및 옵션 가격에 따라 달라질 수 있습니다.
 현금결제시 주의사항안내: 무통장입금 등의 현금결제시 거래안전 확보를 위해 반드시 에스스코 결제를 이용하여 주시기 바랍니다. [자세히 보기](#) >

« 이전 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 다음 »

가장 중요한 값: query, page

이 두 파라미터 값을 달리해서 변경하면 원하는 검색 페이지를 요청할 수 있습니다.

검색어와 페이지 정보만 변경해서 다음과 같은 URL을 만들어서 웹 브라우저로 페이지를 열어서 호출할 수 있다면 여러 페이지를 가져오도록 처리할 수 있을 것!

ex) <http://search.danawa.com/dsearch.php?query={검색어}&volumeType=allvs&page={페이지}&limit=30&sort=saveDESC&list=list&boost=true&addDelivery=N&tab=무선청소기>
 # '무선청소기'로 검색한 결과 페이지를 for문을 통해 차례차례 변경하면서, 처리할 수 있다면 간단하게 여러 페이지를 크롤링할 수 있을 것입니다.

```
def get_search_page_url(page):
```

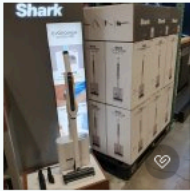
```
    return 'https://search.shopping.naver.com/search/all?adQuery=무선청소기&or igQuery=무선청소기&pagingIndex={}&pagingSize=40&productSet=total&query=무
```

```
page=1
```

```
url=get_search_page_url(page)
```

```
# print(url)
```

★4.53(213) · 찜 54 · 등록일 2009.03. · 정보 수정요청



[당일발송] 샤크 예보파워 시스템 + 무선 스틱 클리너 CS150KRAE 먼지비움 스탠드 코스트코

379,900원 50,000원 오늘출발

디지털/가전 > 청소기 > 무선청소기 > 핸디스틱청소기

★4.66(58) · 구매 261 · 찜 603 · 등록일 2024.04. · 신고하기

톡톡

마켓n송지

정보

상품만 보기 >

빅파워

Npay+ 포인트 최대 3,799원

적립 | 쿠폰 | 할인 | 구매정보

네이버쇼핑에서는 각 쇼핑물에서 받은 상품정보만을 제공하며, 쇼핑물의 정보와 일치하지 않을 수 있으므로 반드시 해당 쇼핑물에서 정확한 정보를 확인하시기 바랍니다. [법적고지 보기](#) >

포인트 최종적립금액은 네이버페이 포인트 사용, 쿠폰 사용 여부 및 옵션 가격에 따라 달라질 수 있습니다.

현금결제시 주의사항안내: 무통장입금 등의 현금결제시 거래안전 확보를 위해 반드시 에스프로 결제를 이용하여 주시기 바랍니다. [자세히 보기](#) >

1 2 3 4 5 6 7 8 9 10 다음 >

def get_search_page_url(page):

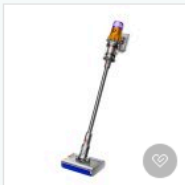
return 'https://search.shopping.naver.com/search/all?adQuery=무선청소기&origQuery=무선청소%BB%B0&pagingIndex={}&pagingSize=40&productSet=total&quer

page=2

url=get_search_page_url(page)

print(url)

여러 페이지를 크롤링할 때는 페이지 URL을 변경해 가며, 페이지별 정보를 수집하고 병합하는 과정을 반복하면 됩니다.



다이슨 V12S 디택트 슬림 서브마린

최저 790,000원 무료 판매처 44

디지털/가전 > 청소기 > 무선청소기 > 핸디스틱청소기

형태: 핸디스틱형 | 출시년도: 2023년 | 작동방식: 회전식 | 청소방식: 흡입전용 | 센서: 먼지센서, 먼지감지 | 흡입력: 150W | 사용시간: 1시간 | 충전시간: 4시간 | 먼지통: 안티탱글(영킴방지)

★4.78(232) · 찜 120 · 등록일 2023.06. · 정보 수정요청

브랜드 카탈로그

다이슨코리아 ↓790,000

쿠팡 890,000

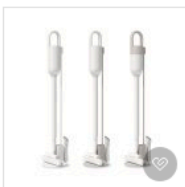
다이슨 코... Npay+ 890,000

GSSHOP Npay 890,000

CJ온스타일 Npay 890,000

· 엘로우,니켈 790,000원 | 30개

· 골드,골드 990,000원 | 14개



무궁화전자 모온 오비콤

최저 299,000원 무료 판매처 23

디지털/가전 > 청소기 > 무선청소기 > 핸디스틱청소기

형태: 핸디스틱형 | 모터: BLDC모터 | 출시년도: 2022년 | 청소방식: 흡입전용 | 흡입력: 90W | 사용시간: 1시간 | 충전시간: 3시간 | 충전방식: 스탠드충전 | 충전알림: 충전표시등(LED)

★4.85(2,265) · 찜 217 · 등록일 2024.03. · 정보 수정요청

브랜드 카탈로그

MO-ON Npay+ ↓299,000

쿠팡 299,000

현대Hmall Npay 299,000

CJ온스타일 Npay 299,000

SSO닷컴 299,000

네이버쇼핑에서는 각 쇼핑물에서 받은 상품정보만을 제공하며, 쇼핑물의 정보와 일치하지 않을 수 있으므로 반드시 해당 쇼핑물에서 정확한 정보를 확인하시기 바랍니다. [법적고지 보기](#) >

포인트 최종적립금액은 네이버페이 포인트 사용, 쿠폰 사용 여부 및 옵션 가격에 따라 달라질 수 있습니다.

현금결제시 주의사항안내: 무통장입금 등의 현금결제시 거래안전 확보를 위해 반드시 에스프로 결제를 이용하여 주시기 바랍니다. [자세히 보기](#) >

< 이전 1 2 3 4 5 6 7 8 9 10 다음 >

진행표시줄 처리

페이지별 URL을 이용해 반복문으로 전체 페이지의 상품 정보를 크롤링할 수 있습니다.

반복문을 실행하기에 앞서 tqdm 라이브러리를 이용해 현재 진행 상황을 표시하는 진행 표시줄을 만들어보겠습니다.

tqdm 모듈을 이용하면 전체 과정 중 현재 진행 단계, 지금까지의 소요 시간, 1회 작업 시 소요시간, 예상 완료시간 등을 확인할 수 있습니다.

! pip install tqdm

반복문 하나를 실행하는 것이 iteration이 하나 진행됐다고 하며, 현재 진행 상태를 확인할 수 있습니다.

Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (4.66.4)

```
import time
from tqdm import tqdm_notebook
```

```
total_page=10
for page in tqdm_notebook(range(1,total_page+1)):
    time.sleep(5) # 페이지가 로딩 완료되기 위한 시간을 5초로 준다.
```

```
<ipython-input-31-d73230a74bae>:5: TqdmDeprecationWarning: This function will be removed in tqdm==5.0.0
Please use `tqdm.notebook.tqdm` instead of `tqdm.tqdm_notebook`
    for page in tqdm_notebook(range(1,total_page+1)):
```

100%

10/10 [00:50<00:00, 5.01s/it]

여러 페이지에 걸친 상품 정보 수집

```
url="https://prod.danawa.com/list/?cate=102207&shortcutKeyword=무선청소기"
driver.get(url)
```

```
html=driver.page_source
soup=BeautifulSoup(html, 'html.parser')
```

```
prod_items=soup.select('li.prod_item')
len(prod_items)
```

31

```
prod_items=soup.select('div.main_prodlst>ul.product_list>li.prod_item')
len(prod_items)
```

31

```
title=prod_items[0].select('p.prod_name > a')[0].text.strip()
print(title)
```

LG전자 오브제컬렉션 코드제로 A9S AX958

```
spec_list=prod_items[0].select('div.spec_list')[0].text.strip()
print(spec_list)
```

```
핸드스틱청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0/먼지비움/충전/UVC LED/액세서리수납/스탠드거치/먼지비움시간: 30초/브러:
```

```
price=prod_items[0].select('p.price_sect>a')[0].text.strip()
print(price)
```

1,156,120원

```
# 반복문으로 검색 결과의 1페이지에 대한 상품 정보 추출
prod_data=[]
```

```
for prod_item in prod_items[0:30]:
    try:
        title=prod_item.select('p.prod-name>a')[0].text.strip()
    except:
        title=''

    try:
        spec_list=prod_item.select('div.spec_list')[0].text.strip()
    except:
        spec_list=''

    try:
        price=prod_item.select('p.price_sect>a')[0].text.strip()
    except:
        price=0

    prod_data.append([title, spec_list, price])
```

```
print(len(prod_data))
print(prod_data)
```

```
[[', '핸드스틱청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0/먼지비움/충전/UVC LED/액세서리수납/스탠드거치/먼지비움시간: 30:
```

상품 정보 태그에서 원하는 정보를 추출하는 함수

```
def get_prod_items(prod_items):
# 반복문으로 검색 결과의 1페이지에 대한 상품 정보 추출
prod_data=[]

for prod_item in prod_items[0:30]:
    try:
        title=prod_item.select('p.prod_name>a')[0].text.strip()
    except:
        title=''

    try:
        spec_list=prod_item.select('div.spec_list')[0].text.strip()
    except:
        spec_list=''

    try:
        price=prod_item.select('p.price_sect>a')[0].text.strip()
    except:
        price=0

    prod_data.append([title, spec_list, price])

print(len(prod_data))
print(prod_data)
return prod_data

prod_items=soup.select('div.main_prodlst >ul.product_list>li.prod_item')
prod_data=get_prod_items(prod_items)
print(len(prod_data))
```

30
[['', '핸디스탁청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0/먼지비움/충전/UVC LED/액세서리수납/스탠드거치/먼지비움시간: 30:30
30

prod_items



```

<div class="meta_item mt_comment">
<a href="/info/?pcode=7718509&cate=102207&deliveryYN=N&bookmark=cm_opinion&companyReviewYN=Y#bookmark_cm_opinion"
onmousedown="_trkEventLog('상품리스트_상품정보_쇼핑올리뷰_블로그이동_리스트형')">target="_blank">
<span class="dt_behind">상품리뷰</span>
<div class="star-single"><span class="blind">별점</span>
<div class="box_star"></div><span class="text_score">4.7</span>
<div class="text_review"><span class="blind">리뷰수</span><span class="text_number">4,254</span></div>
</div>
</a>

```

URL 변화

<https://search.danawa.com/dsearch.php?query=무선청소기&tab=main>

1페이지 진입 시 (최초 URL과 동일)

<http://search.danawa.com/dsearch.php?query=무선청소기&originalQuery=무선청소기&previousKeyword=EB%AC%B4%EC%84%A0%EC%B2%AD%EC%86%8C%EA%B8%B0%&volumeT>

2페이지 진입 시

<http://search.danawa.com/dsearch.php?query=무선청소기&originalQuery=무선청소기&previousKeyword=무선청소기&volumeType=allvs&page=2&limit=30&sort=save>

<http://search.danawa.com/dsearch.php?query=무선청소기&originalQuery=무선청소기&previousKeyword=무선청소기&volumeType=allvs&page=3&limit=30&sort=saveDE>



File "<ipython-input-42-a60eb4e5a436>". line 4

<http://search.danawa.com/dsearch.php?query=무선청소기&originalQuery=무선청소기&previousKeyword=EB%AC%B4%EC%84%A0%EC%B2%AD%EC%86%8C%EA%B8%B0%&volumeType=allvs&page=1&limit=30&sort=saveDESC&list&-boost=true&addDelivery=N&tab=goods&tab=goods>

SyntaxError: invalid decimal literal

다음 단계: [오류 수정](#)

```

from selenium import webdriver
import time
from bs4 import BeautifulSoup
from tqdm import tqdm_notebook
driver.implicitly_wait(3)
keyword='무선청소기'
total_page=10
prod_data_total=[]

```

```

def get_search_page_url(keyword, total_page):
    return 'http://search.danawa.com/dsearch.php?query={}&originalQuery={}&previousKeyword={}&volumeType=allvs&page={}&limit=30&sort=saveDESC&list&-bo

```

```

for page in tqdm_notebook(range(1,total_page+1)):
    url=get_search_page_url(keyword, page)
    driver.get(url)
    # 페이지가 로딩 완료되기 위한 시간으로 5초를 할당
    # 페이지 URL을 생성하는 함수를 이용해 특정 페이지로 이동합니다.
    time.sleep(5)

```

```

# 현재 페이지의 HTML 정보 가져오기
html=driver.page_source
soup=BeautifulSoup(html, 'html.parser')

```

```

# 상품 정보 추출
prod_items=soup.select('div#productListArea > div.main_prodlst > ul.product_list > li.prod_item')
prod_item_list=get_prod_items(prod_items)

```

```

# 추출 데이터 저장
prod_data_total=prod_data_total+prod_item_list

```

[illegible]

prod_data_total

'97,200원'],
['LG전자 오브제컬렉션 코드제로 A9S AU9882'],
'핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 220W/스테이션: 올인원타워/먼지비용/충전/액세서리수납/스탠드거치/먼지비용시간: 36초/브러쉬: 와이드바닥/물걸레: 스팀/솔형/틈새/먼지봉투: 2.5L/[배터리] 사용시간: 1시간(최대)/분리형(2개)/리튬이온/[청소] LED라이트/싸이클론흡입/자동물공급/스마트인버터모터/워셔블헤파필터/크기(가로x세로x깊이): 300x1120x260mm'],
'839,820원'],
['다이슨 V8'],
'핸디스틱청소기/무선/흡입형/흡입력: 115AW/벽걸이거치/브러쉬: 바닥/솔형/틈새/[배터리] 사용시간: 1시간(최대)/리튬이온/싸이클론흡입/먼지필터/크기(가로x세로x깊이): 221x1256x250mm'],
'352,320원'],
['LG전자 오브제컬렉션 코드제로 A9 AU9472WD'],
'핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 220W/스테이션: 올인원타워/먼지비용/충전/액세서리수납/스탠드거치/먼지비용시간: 60초/브러쉬: 바닥/물걸레: 일반/솔형/틈새/먼지봉투: 2.5L/[배터리] 사용시간: 1시간(최대)/분리형(1개)/리튬이온/[청소] 싸이클론흡입/자동물공급/스마트인버터모터/워셔블헤파필터/크기(가로x세로x깊이): 250x1120x260mm'],
'744,390원'],
['LG전자 오브제컬렉션 코드제로 A9S AX9604'],
'핸디스틱청소기/무선/흡입형/흡입력: 250W/스테이션: 올인원타워/먼지비용/충전/UVC LED/액세서리수납/스탠드거치/먼지비용시간: 30초/브러쉬: 와이드바닥/물걸레: 별매/솔형/틈새/먼지봉투: 2.5L/[배터리] 사용시간: 30분(최대)/분리형(1개)/리튬이온/[청소] LED라이트/싸이클론흡입/스마트인버터모터/워셔블헤파필터/크기(가로x세로x깊이): 300x1120x245mm'],
'907,900원'],
['LG전자 오브제컬렉션 코드제로 A9S AX958'],
'핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0/먼지비용/충전/UVC LED/액세서리수납/스탠드거치/먼지비용시간: 30초/브러쉬: 와이드바닥/물걸레: 스팀, 고온, 일반/청구/솔형/틈새/먼지봉투: 2.5L/[배터리] 사용시간: 30분(최대)/분리형(2개)/리튬이온/[청소] LED라이트/싸이클론흡입/오토스탑엔고/자동물공급/스마트인버터모터/워셔블헤파필터/크기(가로x세로x깊이): 300x1100x245mm'],
'1,156,120원'],
['삼성전자 비스포크 제트 VS20B956AX'],
'핸디스틱청소기/무선/흡입형/흡입력: 220W/스테이션: 청정스테이션/먼지비용/충전/UVC LED/스탠드거치/먼지비용시간: 14초/브러쉬: 바닥/물걸레: 별매/솔형/틈새/연장관/먼지봉투: 1.2L/[배터리] 사용시간: 1시간(최대)/분리형(1개)/리튬이온/[청소] LED라이트/싸이클론흡입/디지털인버터모터/워셔블헤파필터/크기(가로x세로x깊이): 250x930x202mm'],
'478,930원'],
['LG전자 오브제컬렉션 코드제로 A9 AU9202WD'],
'핸디스틱청소기/무선/흡입형/흡입력: 220W/스테이션: 올인원타워/먼지비용/충전/액세서리수납/스탠드거치/먼지비용시간: 60초/브러쉬: 바닥/물걸레: 별매/솔형/틈새/먼지봉투: 2.5L/[배터리] 사용시간: 1시간(최대)/분리형(1개)/리튬이온/[청소] 싸이클론흡입/스마트인버터모터/헤파필터/크기(가로x세로x깊이): 250x1120x260mm'],
'629,950원'],
['LG전자 오브제컬렉션 코드제로 A9 AU9272WD'],
'핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 220W/스테이션: 올인원타워/먼지비용/충전/액세서리수납/스탠드거치/먼지비용시간: 60초/브러쉬: 바닥/물걸레: 일반/솔형/틈새/먼지봉투: 2.5L/[배터리] 사용시간: 1시간(최대)/분리형(1개)/리튬이온/[청소] 싸이클론흡입/자동물공급/스마트인버터모터/워셔블헤파필터/크기(가로x세로x깊이): 250x1120x260mm'],
'695,610원'],
['다이슨 싸이클론 V10'],
'핸디스틱청소기/무선/흡입형/흡입력: 151AW/충전/벽걸이거치/브러쉬: 바닥/팻/솔형/틈새/[배터리] 사용시간: 1시간(최대)/리튬이온/[청소] 싸이클론흡입/디지털모터/헤파필터/크기(가로x세로x깊이): 242x1238x250mm'],
'399,000원'],
['샤오미 미홀 M22'],
'핸디스틱청소기/무선/흡입형/벽걸이거치/브러쉬: 바닥/물걸레: 별매/청구/솔형/틈새/[배터리] 사용시간: 50분(최대)/분리형(1개)/[청소] LED라이트/BLDC모터/워셔블헤파필터/크기(가로x세로x깊이): 240x1040x204mm'],
'159,000원'],
['LG전자 오브제컬렉션 코드제로 A9 AS9202WD'],
'핸디스틱청소기/무선/흡입형/흡입력: 210W/스탠드거치/브러쉬: 바닥/물걸레: 별매/솔형/틈새/[배터리] 사용시간: 1시간(최대)/분리형(1개)/리튬이온/[청소] 싸이클론흡입/스마트인버터모터/헤파필터/크기(가로x세로x깊이): 250x1120x260mm'],
'512,230원'],
['LG전자 오브제컬렉션 코드제로 A9S AX9984'],
'핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 250W/스테이션: 올인원타워/먼지비용/충전/UVC LED/액세서리수납/스탠드거치/먼지비용시간: 30초/브러쉬: 와이드바닥/물걸레: 스팀, 고온, 일반/청구/솔형/틈새/먼지봉투: 2.5L/[배터리] 사용시간: 30분(최대)/분리형(2개)/리튬이온/[청소] LED라이트/싸이클론흡입/자동물공급/스마트인버터모터/워셔블헤파필터/크기(가로x세로x깊이): 300x1120x245mm'],
'1,028,300원'],
['LG전자 오브제컬렉션 코드제로 A9S AX9474'],
'핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 320W/스테이션: 올인원타워/먼지비용/충전/UVC LED/액세서리수납/스탠드거치/먼지비용시간: 30초/

수집 데이터저장

```
import pandas as pd
data=pd.DataFrame(prod_data_total)
data.columns=['상품명','스펙 목록','가격']
data.to_excel('danawa_crawling_result.xlsx',index=False)
```

다나와 크롤링 데이터 전처리

```
# 수집한 데이터를 분석에 용이하도록, 상품명 데이터에서 회사명과 제품명 분리
# 일반 문자열로 저장된 스펙 목록을 필요한 스펙으로 분류 및 단위 통합
```

다나와 크롤링 데이터 불러오기

```
import pandas as pd
data=pd.read_excel('/content/danawa_crawling_result.xlsx')
data.info()
data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300 entries, 0 to 299
Data columns (total 3 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   상품명      300 non-null     object
1   스펙 목록    300 non-null     object
2   가격        300 non-null     object
dtypes: object(3)
memory usage: 7.2+ KB
```

| | 상품명 | 스펙 목록 | 가격 |
|---|------------------------------|---|------------|
| 0 | LG전자 오브제컬렉션 코드제로 A9S AX958 | 핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 280W/스테이션: 올인원타워4.0... | 1,156,120원 |
| 1 | 삼성전자 비스포크 제트 VS20B956AX | 핸디스틱청소기/무선/흡입형/흡입력: 220W/스테이션: 청정스테이션/먼지비움/충전/... | 478,930원 |
| 2 | LG전자 오브제컬렉션 코드제로 A9 AU9202WD | 핸디스틱청소기/무선/흡입형/흡입력: 220W/스테이션: 올인원타워/먼지비움/충전/액... | 629,950원 |
| 3 | LG전자 오브제컬렉션 코드제로 A9 AU9272WD | 핸디스틱청소기/무선/흡입+물걸레(동시)/흡입력: 220W/스테이션: 올인원타워/먼지... | 695,610원 |
| 4 | 다이슨 싸이클론 V10 | 핸디스틱청소기/무선/흡입형/흡입력: 151AW/충전/벽걸이거치/브러쉬: 바닥/펫/솔... | 399,000원 |

다음 단계:

[data변수로 코드 생성](#)

[추천 차트 보기](#)

[New interactive sheet](#)

```
# 총 300개의 무선청소기 데이터를 가져왔음을 확인할 수 있고, 상품명, 스펙 목록은 문자열 타입(object)인데, 이는 보통 문자열 데이터일때 object 데이터 타
# 가격은 숫자형 데이터 타입(int)으로 구성돼 있습니다.
# 가격은 모두 300개로서 모든 항목에 데이터가 잘 채워진 것을 확인할 수 있습니다.
```

회사명,모델명 정리

```
data['상품명'][:10]
```

```
0   LG전자 오브제컬렉션 코드제로 A9S AX958
1   삼성전자 비스포크 제트 VS20B956AX
2   LG전자 오브제컬렉션 코드제로 A9 AU9202WD
3   LG전자 오브제컬렉션 코드제로 A9 AU9272WD
4   다이슨 싸이클론 V10
5   샤오미 미홀 M22
6   LG전자 오브제컬렉션 코드제로 A9 AS9202WD
7   LG전자 오브제컬렉션 코드제로 A9S AX9984
8   LG전자 오브제컬렉션 코드제로 A9S AX947A
9   LG전자 오브제컬렉션 코드제로 A9S AX958A
Name: 상품명, dtype: object
```

```
title="LG전자 오브제컬렉션 코드제로 A9 AX9984"
info=title.split(' ',1)
print(info)
# split() 함수: 특정 문자를 기준으로 문자열을 나누는 함수.
# split(' ', 1): 1= 첫번째 공백에 대해서만 구분하게 합니다.
```

```
['LG전자', '오브제컬렉션 코드제로 A9 AX9984']
```


회사명+모델명 분리

```
company_list=[]
product_list=[]

for title in data['상품명']:
    title_info=title.split(' ',1)
    company_name=title_info[0]
    product_name=title_info[1]

    company_list.append(company_name)
    product_list.append(product_name)

print('len(data):', len(data))
print('len(company_list):', len(company_list))
```



```
len(data): 300
len(company_list): 300
```