

```
import requests
```

```
from bs4 import BeautifulSoup
```

```
response = requests.get('https://movie.naver.com/movie/sdb/rank/rmovie.naver')
response
```

<Response [200]>

```
html = response.text
html
```

[illegible]

In [5]:

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

Out[5]:

```
<!DOCTYPE html>

<html lang="ko">
<head>
<meta content="text/html; charset=utf-8" http-equiv="Content
-Type"/>
<meta content="IE=edge" http-equiv="X-UA-Compatible"/>
<meta content="http://imgmovie.naver.com/today/naverme/n
averme_profile.jpg" property="me2:image">
<meta content="네이버영화 " property="me2:post_tag">
<meta content="네이버영화" property="me2:category1"/>
<meta content="" property="me2:category2"/>
<meta content="랭킹 : 네이버 영화" property="og:title"/>
<meta content="영화, 영화인, 예매, 박스오피스 랭킹 정보 제공" prop
erty="og:description"/>
<meta content="article" property="og:type"/>
<meta content="https://movie.naver.com/movie/sdb/rank/rmo
vie.naver" property="og:url"/>
<meta content="http://static.naver.net/m/movie/icons/OG_270
```

In [6]:

```
wordAddr = soup.select('.tit3 a')
```

wordAddr

Out[6]:

```
[<a href="/movie/bi/mi/basic.naver?code=208077" title="스파이
더맨: 노 웨이 홈">스파이더맨: 노 웨이 홈</a>,
<a href="/movie/bi/mi/basic.naver?code=190400" title="경관의
피">경관의 피</a>,
<a href="/movie/bi/mi/basic.naver?code=159893" title="킹스
맨: 퍼스트 에이전트">킹스맨: 퍼스트 에이전트</a>,
<a href="/movie/bi/mi/basic.naver?code=204649" title="해피 뉴
이어">해피 뉴 이어</a>,
<a href="/movie/bi/mi/basic.naver?code=191600" title="매트릭
스: 리저렉션">매트릭스: 리저렉션</a>,
<a href="/movie/bi/mi/basic.naver?code=187320" title="쌍2게
더">쌍2게더</a>,
<a href="/movie/bi/mi/basic.naver?code=192614" title="돈 룩
업">돈 룩 업</a>,
<a href="/movie/bi/mi/basic.naver?code=205623" title="드라이
브 마이 카">드라이브 마이 카</a>,
<a href="/movie/bi/mi/basic.naver?code=205625" title="램">램
</a>,
<a href="/movie/bi/mi/basic.naver?code=210895" title="타이거
마스크">타이거마스크</a>]
```

In [7]:

```

num=1
for i in wordAddr:
    title =i.attrs['title']
    href =i.attrs['href']

    print("{} . {}=> {}".format(num,title,href))
    num +=1

```

```

1. 스파이더맨: 노 웨이 홈=> /movie/bi/mi/basic.naver?code=208077
2. 경관의 피=> /movie/bi/mi/basic.naver?code=190400
3. 킹스맨: 퍼스트 에이전트=> /movie/bi/mi/basic.naver?code=159893
4. 해피 뉴 이어=> /movie/bi/mi/basic.naver?code=204649
5. 매트릭스: 리저렉션=> /movie/bi/mi/basic.naver?code=191600
6. 씽2게더=> /movie/bi/mi/basic.naver?code=187320
7. 돈 룩 업=> /movie/bi/mi/basic.naver?code=192614
8. 드라이브 마이 카=> /movie/bi/mi/basic.naver?code=205623
9. 램=> /movie/bi/mi/basic.naver?code=205625
10. 타이거마스크=> /movie/bi/mi/basic.naver?code=210895
11. 특송=> /movie/bi/mi/basic.naver?code=186233
12. 엔칸토: 마법의 세계=> /movie/bi/mi/basic.naver?code=210282
13. 레지던트 이블: 라쿤시티=> /movie/bi/mi/basic.naver?code=212886
14. 라임크라임=> /movie/bi/mi/basic.naver?code=198621
15. 고스트버스터즈 라이즈=> /movie/bi/mi/basic.naver?code=182578
16. 하우스 오브 구찌=> /movie/bi/mi/basic.naver?code=200892
17. 유체이탈자=> /movie/bi/mi/basic.naver?code=182362
18. 맨다린 프로젝트=> /movie/bi/mi/basic.naver?code=201004

```

```
#old_content > table > tbody > tr:nth-child(2) > td.title > div > a
```

```
#old_content > table > tbody > tr:nth-child(3) > td.title > div > a
```

```
td.title>div>a div.tit3>a
```

## pyautogui 라이브러리로 간단한 팝업창 띄우기

- 마우스, 키보드 매크로 라이브러리
- 간단한 팝업창 띄우기 가능
- `pyautogui.prompt("검색어 입력")`

In [8]:

```
!pip install pyautogui
```

```
Requirement already satisfied: pyautogui in c:\users\yj\anaconda3\lib\site-packages (0.9.53)
Requirement already satisfied: pymsgbox in c:\users\yj\anaconda3\lib\site-packages (from pyautogui) (1.0.9)
Requirement already satisfied: pygetwindow>=0.0.5 in c:\users\yj\anaconda3\lib\site-packages (from pyautogui) (0.0.9)
Requirement already satisfied: mouseinfo in c:\users\yj\anaconda3\lib\site-packages (from pyautogui) (0.1.3)
Requirement already satisfied: PyTweening>=1.0.1 in c:\users\yj\anaconda3\lib\site-packages (from pyautogui) (1.0.4)
Requirement already satisfied: pyscreeze>=0.1.21 in c:\users\yj\anaconda3\lib\site-packages (from pyautogui) (0.1.28)
Requirement already satisfied: pyrect in c:\users\yj\anaconda3\lib\site-packages (from pygetwindow>=0.0.5->pyautogui) (0.1.4)
Requirement already satisfied: pyperclip in c:\users\yj\anaconda3\lib\site-packages (from mouseinfo->pyautogui) (1.8.2)
```

## 실전프로젝트1) 뉴스 데이터 수집하기 03. 여러 페이지 결과 모두 가져오기

- url 분석하기, 여러 페이지의 결과 가져오기 위해 이동하면서 url이 어떻게 변하는지 살펴보기
- 뉴스 다음 페이지로 이동시 start 파라미터의 값이 10씩 증가
- 1페이지: start=1, 2페이지: start=11, 3페이지: start=21...10페이지:start=91

In [15]:

```

import pyautogui

keyword = pyautogui.prompt("검색어를 입력하세요.")
userPage = int(pyautogui.prompt("페이지를 입력하세요.)) #사용자가 입력하는 마지막 페이지
userPage1 = userPage*10

pageNum=1
articleNum=1
for i in range(1, userPage1, 10):

    #f-string 사용, f"문자열(변수)"
    response3 = requests.get(f"https://search.naver.com/search.naver?where=news&s

    html3 = response3.text #웹 사이트의 html소스 코드 가져옴
    soup3 = BeautifulSoup(html3, 'html.parser')
    links3 = soup3.select('.news_tit') #결과가 리스트 형태로 나옴

    #리스트의 값을 하나씩 for 문으로 출력

    for link in links3:
        title = link.text #태그 내 텍스트 요소 가져옴
        url = link.attrs['href'] #속성 href의 값을 가져옴
        print(articleNum, ",", title, url)
        articleNum +=1
    print(f"-----{pageNum}페이지-----")
    pageNum +=1

```

```

1. '반도체 호황'에 삼성전자 작년 매출 279조원...역대 최고(종합2보) http://yna.kr/AKR20220107035751003?did=1195m (http://yna.kr/AKR20220107035751003?did=1195m)
2. 삼성전자, 4분기 매출 76조 영업이익 13.8조 http://www.joseilbo.com/news/news\_read.php?uid=443151&class=17&grp=\(http://www.joseilbo.com/news/news\_read.php?uid=443151&class=17&grp=\)
3. 삼성전자 작년 연간 매출 279조원...사상 최대치 기록 https://imnews.imbc.com/news/2022/econo/article/6330770\_35687.html (https://imnews.imbc.com/news/2022/econo/article/6330770\_35687.html)
4. '반도체의 힘' 삼성전자, 창사 이래 최대 매출...“올해 더 좋다” https://www.seoul.co.kr/news/newsView.php?id=20220107500059&wlog\_tag3=naver (https://www.seoul.co.kr/news/newsView.php?id=20220107500059&wlog\_tag3=naver)
5. 삼성전자 작년 매출 신기록...IM 부문도 빛났다 http://www.fnnews.com/news/202201071005428358 (http://www.fnnews.com/news/202201071005428358)
6. OLED TV 초고화질 전쟁 점화... 삼성 QD vs LG EX https://biz.chosun.com/it-science/ict/2022/01/07/TQTK2SLTYFGGDLWR3KTHBAUVE/?utm\_source=naver&utm\_medium=original&utm\_campaign=

```

In [ ]: