

LESSON

조회수 데이터 가져오기 (Selenium 모듈)

```
animLength = toTime - fromTime

# Ask user for directory
filePath = c4d.storage.SaveDialog()
filePath, objName = os.path.split(filePath)
objName = objName + ". "
filePath = filePath + "\\ "

# Ask for confirmation
questionDialogText = "Obj Sequence will be saved as:\n\n" \
    "" + filePath + objName + "####.obj\n\n" \
    "From frame " + str(fromTime) + " to " + str(toTime) + " "
proceedBool = c4d.gui.QuestionDialog(questionDialogText)

if proceedBool == True:

    # Loop through animation and export frames
    for x in range(0, animLength):

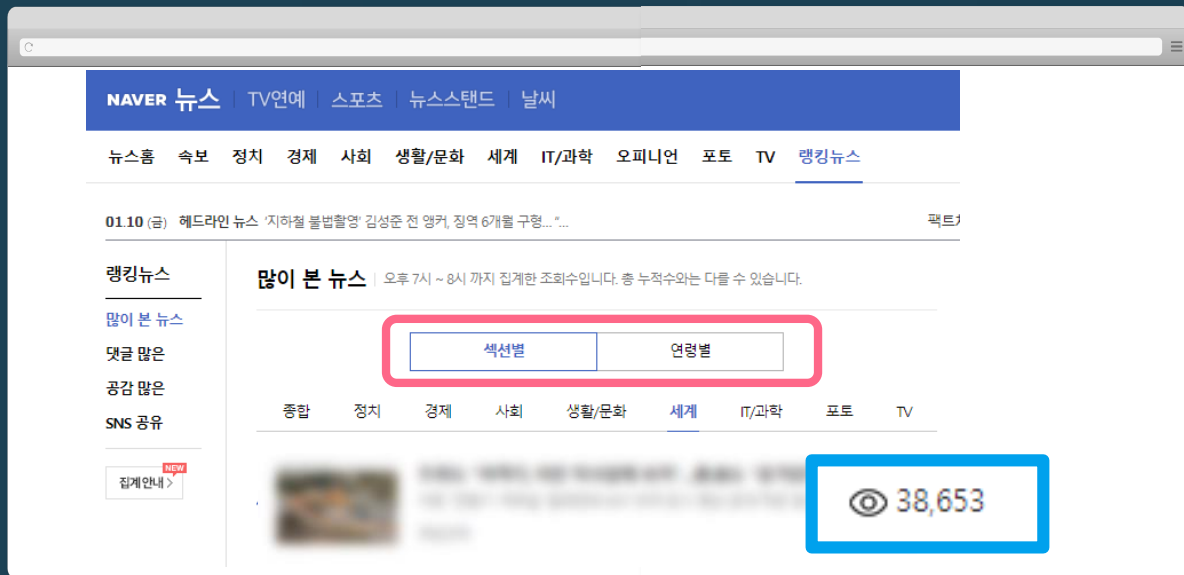
        # change frame, redraw view
        moveTime = c4d.BaseTime(fromTime, docFps) + c4d.BaseTime(x, docFps)
        doc.SetTime(moveTime)
        c4d.EventAdd(c4d.EVENT_FORCEREDRAW)
        c4d.DrawViews(c4d.DRAWFLAGS_FORCEFULLREDRAW)

        # update status bar
        c4d.StatusSetText("Exporting " + str(x) + " of " + str(animLength))
        c4d.StatusSetBar(100.0*x/animLength)

        # add buffer 0001
        bufferedNumber = str(doc.GetTime().GetFrame(docFps))
        if len(bufferedNumber) < 4:
```

‘랭킹 뉴스’ 페이지 접근해 데이터 가져오기

- Naver 뉴스의 많이 본 뉴스(랭킹 뉴스) 페이지에서
섹션별, 연령별 조회수 비교를 위한 데이터 가져오기



1. 랭킹 뉴스 페이지 접근 Step1

Naver 뉴스 페이지에 접근한다.

The screenshot shows the Naver homepage with the DevTools console open. The console displays the HTML structure of the navigation bar, specifically the `` link. A red arrow points from this link in the console to the '뉴스' link in the navigation bar. The navigation bar also includes links for '메일', '카페', '블로그', '지식IN', '쇼핑', 'Pay', 'TV', '사전', '증권', '부동산', '지도', '영화', '음악', '책', '웹툰', and '더보기'.

```
<div class="u_skip"></div>
<!-- //스킵 내비게이션 -->
<!-- 크롬/웨일/엣지 -->
<div id="whale_promotion_banner" class="banner_area type_chrome" style="display: block; overflow: visible; height: 48px; position: relative;">
  <a id="whale_promotion_download_file" style="display: none" href="http://update.whale.naver.net/downloads/installers/WhaleSetup.exe" download></a>
  <div id="PM_ID_ct" class="wrap">
    <!-- 헤더 -->
    <div class="header" role="banner">
      <div class="special_bg"></div>
      <div class="section_navbar">
        <div class="area_navigation" role="navigation">
          <ul class="an_l">
            <ul id="PM_ID_serviceNavi" class="an_l">
              <li class="an_item">
                <a href="https://news.naver.com/" class="an_a mn_news" data-clk="svc.news">
                  <span class="an_icon"></span>
                  <span class="an_txt">뉴스</span>
                </a>
              </li>
            </ul>
          </ul>
        </div>
      </div>
    </div>
  </div>
</div>
```

1. 랭킹 뉴스 페이지 접근 Step1

Naver 뉴스 페이지에 접근한다.

크롬 웹 드라이버 설치 경로

```
import selenium
import time
from selenium import webdriver
```

```
path = "/Users/zoostar/Downloads/chromedriver"
driver = webdriver.Chrome(path)
driver.get('https://www.naver.com')
```

```
element = driver.find_element_by_class_name("mn_news")
element.click()
```



1. 랭킹 뉴스 페이지 접근 Step2

랭킹 뉴스 페이지에 접근한다.



```
<div id="lnb" class="lnb_menu">
```

```
<li class="lnb_menu_home on"><a href="/main/home.nhn" class="nclicks(LNB.home)"><span class="tx">뉴스홈</span> <span class="blind">선택됨</span></a></li>
<li><a href="/main/list.nhn?mode=LS&mid=sec&sid1=001" class="nclicks(LNB.flash)"><span class="tx">속보</span></a></li>
<li><a href="/main/main.nhn?mode=LS&mid=shms&sid1=100" class="nclicks(LNB.pol)"><span class="tx">정치</span></a></li>
<li><a href="/main/main.nhn?mode=LS&mid=shms&sid1=101" class="nclicks(LNB.eco)"><span class="tx">경제</span></a></li>
<li><a href="/main/main.nhn?mode=LS&mid=shms&sid1=102" class="nclicks(LNB.soc)"><span class="tx">사회</span></a></li>
<li><a href="/main/main.nhn?mode=LS&mid=shms&sid1=103" class="nclicks(LNB.lif)"><span class="tx">생활/문화</span></a></li>
<li><a href="/main/main.nhn?mode=LS&mid=shms&sid1=104" class="nclicks(LNB.wor)"><span class="tx">세계</span></a></li>
<li><a href="/main/main.nhn?mode=LS&mid=shms&sid1=105" class="nclicks(LNB.sci)"><span class="tx">IT/과학</span></a></li>
<li><a href="/main/opinion/home.nhn" class="nclicks(LNB.opi)"><span class="tx">오피니언</span></a></li>
<li><a href="/main/photo/index.nhn?mid=pho" class="nclicks(LNB.pho)"><span class="tx">포토</span></a></li>
<li><a href="/main/tv/index.nhn?mid=tyh" class="nclicks(LNB.tv)"><span class="tx">TV</span></a></li>
<li><a href="/main/ranking/popularDay.nhn?mid=etc&sid1=111" class="nclicks(LNB.ranking)"><span class="tx">랭킹뉴스</span></a></li>
</ul>
<form name="lnb_searchForm" id="lnb_searchForm" method="get" action="https://search.naver.com/search.naver" target="_blank" accept-charset="UTF-8">
  <fieldset>
    <legend>뉴스 검색</legend>
    <input type="text" title="뉴스 검색" name="query" accesskey="s" class="text_index" style="ime-mode:active;" />
    <input type="hidden" name="where" value="news">
    <input type="hidden" name="ie" value="utf8">
    <input type="hidden" name="sm" value="nws_hyt">
    <button type="submit" class="btn_search_lnb nclicks(LNB.search)"><span class="tx"><span class="blind">검색</span></span></button>
  </fieldset>
</div>
</div>
```

1. 랭킹 뉴스 페이지 접근 Step2

랭킹 뉴스 페이지에 접근한다.

```

ss="nclicks(LNB.home)"><span class="tx">뉴스홈</span> <span class="tx">속보</span></a></li>
ass="nclicks(LNB.flash)"><span class="tx">속보</span></a></li>
ass="nclicks(LNB.pol)"><span class="tx">정치</span></a></li>
ass="nclicks(LNB.eco)"><span class="tx">경제</span></a></li>
ass="nclicks(LNB.soc)"><span class="tx">사회</span></a></li>
ass="nclicks(LNB.lif)"><span class="tx">생활/문화</span></a></li>
ass="nclicks(LNB.wor)"><span class="tx">세계</span></a></li>
ass="nclicks(LNB.sci)"><span class="tx">IT/과학</span></a></li>
pi)"><span class="tx">오피니언</span></a></li>
s(LNB.pho)"><span class="tx">포토</span></a></li>
NB.tv)"><span class="tx">TV</span></a></li>
l" class="nclicks(LNB.ranking)"><span class="tx">랭킹뉴스</span></a></li>

action="https://search.naver.com/search.naver" target="_blank"

key="s" class="text_index" style="ime-mode:active;"/>

NB.search)"><span class="tx"><span class="blind">검색</span></span></a></li>
    
```



랭킹 뉴스 카테고리의 class는 'tx'로 되어 있으며 동일한 클래스의 요소 중 뒤에서 두 번째에 위치해 있는 것을 확인

```

import selenium
import time
from selenium import webdriver
    
```

```

path = "/Users/zoostar/Downloads/chromedriver"
driver = webdriver.Chrome(path)
driver.get('https://www.naver.com')
    
```

```

element = driver.find_element_by_class_name("mn_news")
    
```

```

element = driver.find_element_by_id("lbn").find_elements_by_class_name("tx")
element[len(element)-2].click()
    
```

2. 섹션별 페이지 접근 Step1

섹션별 데이터 규칙을 찾는다.

01.05 (일) 헤드라인 뉴스 한겨레 '우리의 정치인들' 연봉 2억... 불친척 놓고 '명장'

랭킹뉴스

많이 본 뉴스

댓글 많은

공감 많은

SNS 공유

많이 본 뉴스 | 오후 9시 ~ 10시 까지 집계한 조화수입니다. 총 누적수와는 다를 수 있습니다.

li.ranking_category_item.is_selected 58.5 x 37 명별

종합 정치 경제 사회 생활/문화 세계 IT/과학

Elements Console Sources Network Performance Memory Application Security Audits

```

<td class="snb"></td>
<td class="content">
  <div class="content">
    <div class="list_header ranking_header"></div>
    <div class="ranking_type"></div>
    <div class="ranking_category">
      ::before
      <div class="ranking_category_inner">
        <ul class="ranking_category_list">
          <li class="ranking_category_item"></li>
          <li class="ranking_category_item is_selected">
            ::before
            <a href="/main/ranking/popularDay.nhn?rankingType=popular_day&sectionId=100&date=20200105" class="nclinks(rank.tabsec2)"></a> == $0
          </li>
          <li class="ranking_category_item"></li>
          <li class="ranking_category_item"></li>
          <li class="ranking_category_item"></li>
          <li class="ranking_category_item"></li>
          <li class="ranking_category_item"></li>
          <li class="ranking_category_item"></li>
          <li class="ranking_category_item"></li>
        </ul>
      </div>
    </div>
  </td>
</tr>

```


2. 섹션별 페이지 접근 Step1 섹션별 데이터 규칙을 찾는다.

```
import selenium
import time
from selenium import webdriver

path = "/Users/zoostar/Downloads/chromedriver"
driver = webdriver.Chrome(path)
driver.get('https://www.naver.com')

element = driver.find_element_by_class_name("mn_news")
element.click()
element = driver.find_element_by_id("lnb").find_elements_by_class_name("tx")
element[len(element)-2].click()

#정치, 경제, 사회, 생활/문화, 세계, IT/과학, 포토, TV
element = driver.find_elements_by_class_name("ranking_category_item")
```



```
><div class="list_header ranking_header">...</div>
><div class="ranking_type">...</div>
▼<div class="ranking_category">
  ::before
  ▼<div class="ranking_category_inner">
    ▼<ul class="ranking_category_list">
      ▶<li class="ranking_category_item">...</li>
      ▼<li class="ranking_category_item is_selected">
        ::before
```


2. 섹션별 페이지 접근 Step2

반복문을 활용하여 섹션별 페이지 소스를 BeautifulSoup 객체로 저장한다.

```
import selenium
import time
from selenium import webdriver
from bs4 import BeautifulSoup

path = "/Users/zoostar/Downloads/chromedriver"
driver = webdriver.Chrome(path)
driver.get('https://www.naver.com')

element = driver.find_element_by_class_name("mn_news")
element.click()
element = driver.find_element_by_id("lnb").find_elements_by_class_name("tx")
element[len(element)-2].click()

#정치, 경제, 사회, 생활/문화, 세계, IT/과학, 포토, TV
news_list = {}
element = driver.find_elements_by_class_name("ranking_category_item")

for i in range(1, len(element)):
    element = driver.find_elements_by_class_name("ranking_category_item")
    element[i].click()
    soup = BeautifulSoup(driver.page_source, 'html.parser')
    news_list[soup.title.string.split(',')[0]] = soup
    time.sleep(1)
```

종합

정치

경제

사회

생활/문화

세계

IT/과학

포토

TV

```
print(soup.title.string)
```

TV,

많이 본 뉴스 : 네이버 뉴스

3. 연령별 페이지 접근 Step1

연령별 데이터 섹션에 접근한다.



연령별 페이지는 섹션별 탭
다음에 위치한 탭을 클릭해야
나타나기 때문에 해당 요소를
클릭 할 수 있도록 요소 검색

01.05 (일) 헤드라인 뉴스 文대통령도 보낸 축하화환...사람 화환 아무것도 안보낸 ...

랭킹뉴스

많이 본 뉴스

댓글 많은

공감 많은

SNS 공유

많이 본 뉴스 | 오후 9시 ~ 10시 까지 집계한 조화수입니다. 총 누적수와는 다를 수 있습니다.

ul.ranking_type_list 327 x 34

섹션별

연령별

10대

20대

30대

40대

50대

60대 이상

```

Elements Console Sources Network Performance Memory Application Security Audits
<tbody>
  <tr>
    <td class="snb">...</td>
    <td class="content">
      <div class="content">
        <div class="list_header_ranking_header">...</div>
        <div class="ranking_type">
          <ul class="ranking_type_list">
            <li class="ranking_type_item nclicks(rnk.tabsec)">...</li>
            <li class="ranking_type_item is_selected nclicks(rnk.tabage)">
              <a href="/main/ranking/popularDay.nhn?rankingType=age">...</a> == $0
            </li>
          </ul>
        </div>
      </div>
    </td>
  </tr>
</tbody>
    
```

3. 연령별 페이지 접근 Step1

연령별 데이터 섹션에 접근한다.

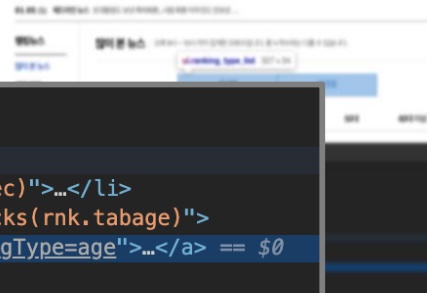
```
import selenium
import time
from selenium import webdriver
from bs4 import BeautifulSoup
```

```
path = "/Users/zoostar/Downloads/chromedriver"
driver = webdriver.Chrome(path)
driver.get('https://www.naver.com')
```

```
element = driver.find_element_by_class_name("mn_news")
element.click()
element = driver.find_element_by_id("lnb").find_elements_by_class_name("tx")
element[len(element)-2].click()
```

```
element = driver.find_elements_by_class_name("ranking_type_item")
element[1].click()
```

```
><div class="list_header ranking_header">...</div>
▼<div class="ranking_type">
  ▼<ul class="ranking_type_list">
    ▶<li class="ranking_type_item nclicks(rnk.tabsec)">...</li>
    ▼<li class="ranking_type_item is_selected nclicks(rnk.tabage)">
      ▶<a href="/main/ranking/popularDay.nhn?rankingType=age">...</a> == $0
    </li>
  </ul>
  </div>
```



3. 연령별 페이지 접근 Step2

반복문을 활용하여 섹션별 페이지 소스를 BeautifulSoup 객체로 저장한다.

```
import selenium
import time
from selenium import webdriver
from bs4 import BeautifulSoup

path = "/Users/zoostar/Downloads/chromedriver"
driver = webdriver.Chrome(path)
driver.get('https://www.naver.com')

element = driver.find_element_by_class_name("mn_news")
element.click()
element = driver.find_element_by_id("lnb").find_elements_by_class_name("tx")
element[len(element)-2].click()

element = driver.find_elements_by_class_name("ranking_type_item")
element[1].click()

age_list = {}
element = driver.find_elements_by_class_name("ranking_category_item")
for i in range(len(element)):
    element = driver.find_elements_by_class_name("ranking_category_item")
    element[i].click()
    soup = BeautifulSoup(driver.page_source, 'html.parser')
    age_list[soup.find('ul', class_='ranking_category_list').find('li', class_='is_selected')
                .find('a').text.split('선택됨')[0].strip()] = soup
    time.sleep(1)
```

3. 연령별 페이지 접근 Step2

반복문을 활용하여 섹션별 페이지 소스를 BeautifulSoup 객체로 저장한다.

```
soup = BeautifulSoup(driver.page_source, 'html.parser')
age_list[soup.find('ul', class_='ranking_category_list').find('li', class_='is_selected')
            .find('a').text.split('선택됨')[0].strip()] = soup
time.sleep(1)
```

```
soup.find('ul', class_='ranking_category_list')
<ul class="ranking_category_list">
<li class="ranking_category_item">
<a class="nclicks(rank.tabage2)" href="/main/ranking/popularDay.nhn?rankingType=age&subType=10">10대 <span class="blind">선택됨</span></a>
</li>
<li class="ranking_category_item">
<a class="nclicks(rank.tabage2)" href="/main/ranking/popularDay.nhn?rankingType=age&subType=20">20대 <span class="blind">선택됨</span></a>
</li>
<li class="ranking_category_item">
<a class="nclicks(rank.tabage2)" href="/main/ranking/popularDay.nhn?rankingType=age&subType=30">30대 <span class="blind">선택됨</span></a>
</li>
<li class="ranking_category_item">
<a class="nclicks(rank.tabage2)" href="/main/ranking/popularDay.nhn?rankingType=age&subType=40">40대 <span class="blind">선택됨</span></a>
</li>
<li class="ranking_category_item">
<a class="nclicks(rank.tabage2)" href="/main/ranking/popularDay.nhn?rankingType=age&subType=50">50대 <span class="blind">선택됨</span></a>
</li>
<li class="ranking_category_item is_selected">
<a class="nclicks(rank.tabage2)" href="/main/ranking/popularDay.nhn?rankingType=age&subType=60">60대 이상 <span class="blind">선택됨</span></a>
</li>
</ul>
```

LESSON

데이터 파싱 및 가공 (BeautifulSoup 모듈)

```
animLength = toTime - fromTime

# Ask user for directory
filePath = c4d.storage.SaveDialog()
filePath, objName = os.path.split(filePath)
objName = objName + ". "
filePath = filePath + "\\ "

# Ask for confirmation
questionDialogText = "Obj Sequence will be saved as:\n\n" \
    "" + filePath + objName + "####.obj\n\n" \
    "From frame " + str(fromTime) + " to " + str(toTime) + " "
proceedBool = c4d.gui.QuestionDialog(questionDialogText)

if proceedBool == True:

    # Loop through animation and export frames
    for x in range(0, animLength):

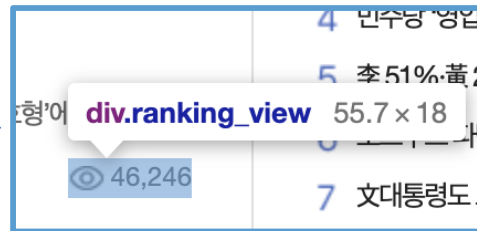
        # change frame, redraw view
        moveTime = c4d.BaseTime(fromTime, docFps) + c4d.BaseTime(x, docFps)
        doc.SetTime(moveTime)
        c4d.EventAdd(c4d.EVENT_FORCEREDRAW)
        c4d.DrawViews(c4d.DRAWFLAGS_FORCEFULLREDRAW)

        # update status bar
        c4d.StatusSetText("Exporting " + str(x) + " of " + str(animLength))
        c4d.StatusSetBar(100.0*x/animLength)

        # add buffer 0001
        bufferedNumber = str(doc.GetTime().GetFrame(docFps))
        if len(bufferedNumber) < 4:
```

1. 섹션별 조회수 파싱 Step1 조회수 패턴을 찾는다.

많이 본 뉴스 | 오후 9시 ~ 10시 까지 집계한 조회수입니다. 총 누적수는 다를 수 있습니다.



```
news_list['정치'].find('div', class_="ranking_view")
```

```
<div class="ranking_view">46,246</div>
```

1. 섹션별 조회수 파싱 Step2 반복문을 활용한 하나의 섹션 전체 조회수를 파싱한다.

```
print(news_list['정치'].find('div',class_="ranking_view").string)
```

46,246

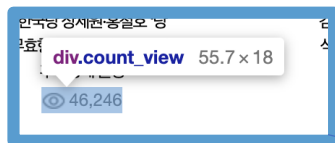
```
sum_view = 0
for i in news_list['정치'].find_all('div',class_="ranking_view"):
    sum_view += int(i.string.replace(",",""))
print(sum_view)
```

329730

1. 섹션별 조회수 파싱 Step3 반복문을 활용한 전체 섹션 전체 조회수를 파싱한다.

```
news_view_list = {}
for i in news_list:
    sum_view = 0
    if(i!='포토' and i!='TV'):
        for j in news_list[i].find_all('div',class_='ranking_view'):
            sum_view += int(j.string.replace(",",""))
    else:
        for j in news_list[i].find_all('div',class_='count_view'):
            sum_view += int(j.string.replace(",",""))
    news_view_list[i] = sum_view
print(news_view_list)
```

```
{'정치': 329730, '경제': 118733, '사회': 370581, '생활/문화': 174381, '세계': 265289, 'IT/과학': 56683, '포토': 517171, 'TV': 368719}
```



2. 연령별 조회수 파싱 Step1

반복문을 활용한 하나의 섹션 전체 조회수를 파싱한다.

많이 본 뉴스 | 오후 9시 ~ 10시까지 집계한 조회수입니다. 총 누적수와는 다를 수 있습니다.

색선별

연령별

10대

20대

30대

40대

50대

60대 이상

1

2

3

4

1,783

1,277

167

158

div ranking_view

1,783

```
print(age_list['10대'].find('div',class_="ranking_view"))
```

<div class="ranking_view">1,783</div>

2. 연령별 조회수 파싱 Step2

반복문을 활용한 하나의 연령별 조회수를 파싱한다.

```
age_view_list = {}
for i in age_list:
    sum_view = 0
    for j in age_list[i].find_all('div', class_='ranking_view'):
        sum_view += int(j.string.replace(",", ''))

    age_view_list[i] = sum_view
print(age_view_list)
```

```
{'10대': 3951, '20대': 21984, '30대': 50417, '40대': 34213, '50대': 20125, '60대 이상': 16521}
```

LESSON

분석 결과 시각화 (matplotlib 모듈)

```
animLength = toTime - fromTime

# Ask user for directory
filePath = c4d.storage.SaveDialog()
filePath, objName = os.path.split(filePath)
objName = objName + ". "
filePath = filePath + "\\ "

# Ask for confirmation
questionDialogText = "Obj Sequence will be saved as:\n\n" \
    "" + filePath + objName + "####.obj\n\n" \
    "From frame " + str(fromTime) + " to " + str(toTime) + " "
proceedBool = c4d.gui.QuestionDialog(questionDialogText)

if proceedBool == True:

    # Loop through animation and export frames
    for x in range(0,animLength):

        # change frame, redraw view
        moveTime = c4d.BaseTime(fromTime,docFps) + c4d.BaseTime(x)
        doc.SetTime(moveTime)
        c4d.EventAdd(c4d.EVENT_FORCEREDRAW)
        c4d.DrawViews(c4d.DRAWFLAGS_FORCEFULLREDRAW)

        # update status bar
        c4d.StatusSetText("Exporting " + str(x) + " of " + str(animLength))
        c4d.StatusSetBar(100.0*x/animLength)

        # add buffer 0001
        bufferedNumber = str(doc.GetTime().GetFrame(docFps))
        if len(bufferedNumber) < 4:
```

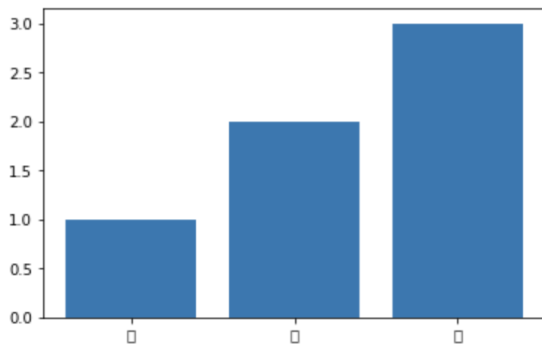
matplotlib 모듈을 이용한 데이터 시각화

- matplotlib 모듈을 이용해 그래프 등을 구현할 경우
한글로 된 글자가 제대로 보이지 않는 문제가 있음
→ 한글이 지원되는 폰트로 변경을 먼저 해주어야 함

```
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

figure = plt.figure()
axes = figure.add_subplot(111)

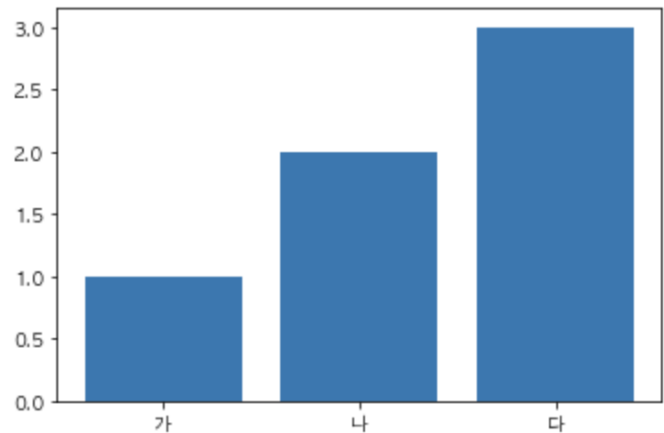
x = ['가', '나', '다']
y = [1, 2, 3]
axes.bar(x, y)
```



1. 한글 폰트 설정하기 matplotlib.rc 함수로 font를 한글이 지원되는 폰트로 변경한다.

```
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
matplotlib.rc('font',family='AppleGothic')
figure = plt.figure()
axes = figure.add_subplot(111)

x = ['가', '나', '다']
y = [1, 2, 3]
axes.bar(x, y)
```

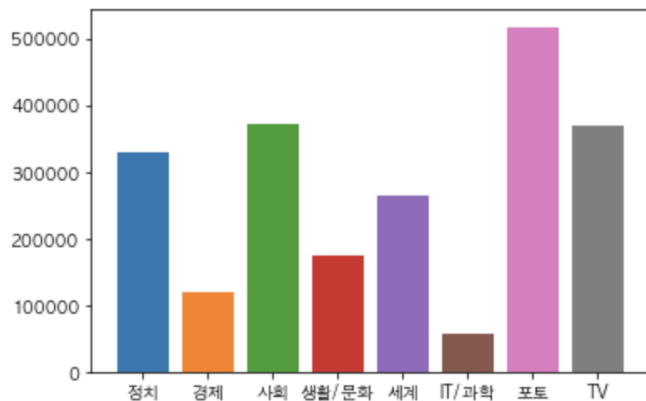


2. 데이터 시각화 섹션별, 연령별 조회수 데이터를 활용해 각각 차트를 만든다.

```
figure = plt.figure()
axes = figure.add_subplot(111)

for i in news_view_list:
    axes.bar(i,news_view_list[i])

plt.show()
```



```
figure = plt.figure()
axes = figure.add_subplot(111)

for i in age_view_list:
    axes.bar(i,age_view_list[i])

plt.show()
```

