

코로나19 전후 사람들의 헬스케어 관심도 변화 비교

봄사랑벗꽃말고AI는어때 

김나리, 김지원, 정승욱, 전준용



목차

분석 주제 및 목표

서론 - 아이디어의 시작

본론 1 - 뉴스기사 텍스트 분석

본론 2 - 관련 종목 주가 변화 시각화

분석 주제 및 목표



코로나19를 전후로한 사람들의 헬스케어 관심도 변화를 분석.

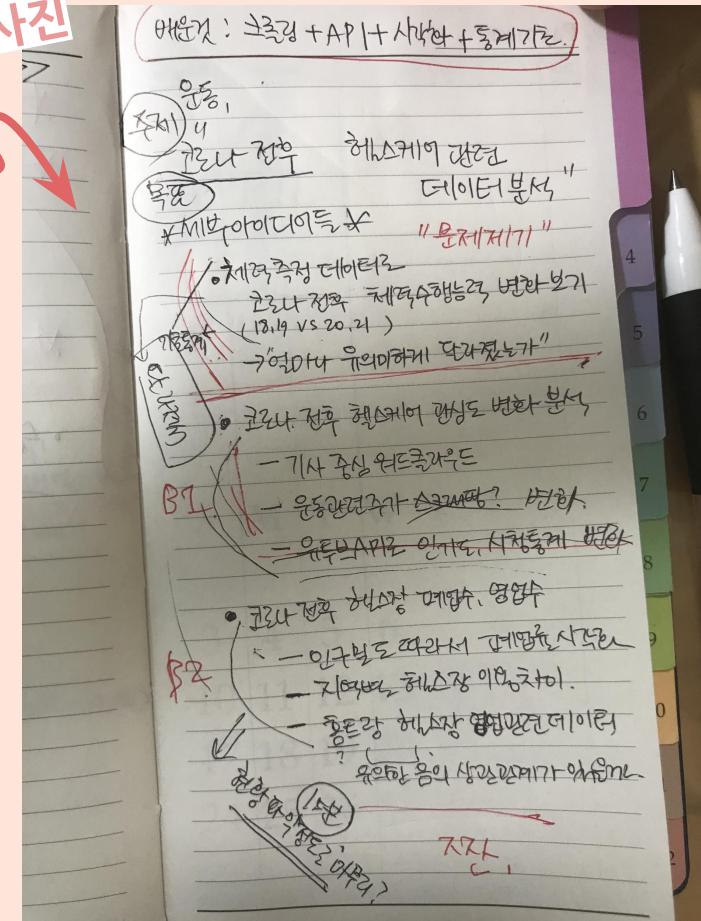


한국에서 코로나가 처음으로 발생한 2020년 1월을 기준으로

18~19년을 코로나 전, 20~21년을
코로나 후로 나누어서

헬스케어 관련 기사와 운동 관련 주가의 변화를 분석해본다.

첫 회의날 청사진



서론

아이디어의 시작

문제제기

‘국민체력 인증센터’의 ‘체력 능력 측정 데이터’로
코로나 전후 사람들의 체력 수행 능력의 변화가 유의미하게 달라졌는가를 확인해 봄.

- 데이터 수집 방법

방법1. 홈페이지 제공 CSV 파일 다운로드

방법2. OpenAPI로 xml데이터를 불러와서 데이터 프레임으로 저장

- 분석 프로세스

2018-2021년도 체력 측정 데이터를 달마다 최대 관측치 모두 끌어옴

api 데이터를 불러와서 코로나 전후 두개의 데이터프레임 생성

체력 수행 능력을 보여주는 비교 가능한 변수들만 독립표본 t-test를 시행

- 변수 예: f009 윗몸말아올리기(회), f023 의자에 앉았다 일어서기(회)
- 독립표본인 이유

[health_final.ipynb](#)

문제제기

- t-test 결과 (95% 신뢰수준)
 - 대체로 유의한 결과 BUT,
 - f013 : 일리노이 (초) : 민첩성 검사

	statistics	pvalue
itemF009	35.842683	3.052066e-279
itemF010	16.749551	7.131428e-63
itemF012	-24.252720	7.454614e-130
itemF013	-0.6667853	5.042288e-01
itemF014	-3.477364	5.065836e-04
itemF015	-3.393643	6.900188e-04
itemF016	-23.317033	5.417506e-120
itemF017	-3.487176	4.884027e-04
itemF019	-103.293125	0.000000e+00
itemF020	-21.041477	3.755607e-98
itemF021	45.172433	0.000000e+00
itemF022	-86.811160	0.000000e+00
itemF023	-51.302594	0.000000e+00
itemF024	-0.420716	6.740323e-01
itemF025	-29.440321	1.928301e-188
itemF026	7.636202	2.258954e-14
itemF027	24.752936	6.783534e-135
itemF030	84.257761	0.000000e+00
itemF031	-2.352666	1.864853e-02
itemF032	-41.718660	0.000000e+00
itemF033	-0.390744	6.959904e-01
itemF034	-2.194685	2.821360e-02
itemF035	-11.475477	2.123780e-30
itemF036	-96.239144	0.000000e+00
itemF037	-14.016121	1.403944e-44
itemF040	8.482460	1.866738e-15
itemF041	1.934856	5.375530e-02

프로세스 오류 1-1 : IOPub data rate exceeded.

- API를 불러오는 과정에서 오류 처럼 보이는 것 확인

```
In [6]: 1 url = callAPI(service_key,'1','10000','201903')
2 print(url)
3 response = requests.get(url).content
4 soup = BeautifulSoup(response, 'lxml-xml')
5 print(soup)

http://www.kspo.or.kr/openapi/service/nfaTestInfoService/getNfaTestRsItList?serviceKey=hDaSIy4ntwfjGKQRK49W8xYuR5Wl8HUCCr2pnL2wrGjq67
5JJRP0xb6e%2F9Xtg7N94DRG37oQr30uZY1JS6b3g%3D%3D&pageNo=1&numOfRows=10000&testYm=201903

IOPub data rate exceeded.
The notebook server will temporarily stop sending output
to the client in order to avoid crashing it.
To change this limit, set the config variable
'--NotebookApp.iopub_data_rate_limit'.

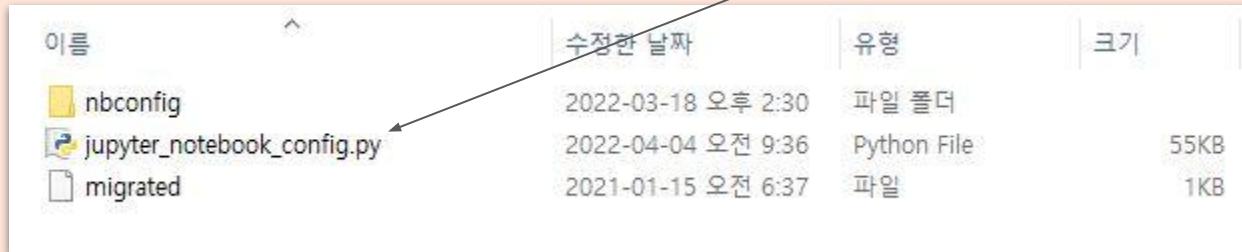
Current values:
NotebookApp.iopub_data_rate_limit=1000000.0 (bytes/sec)
NotebookApp.rate_limit_window=3.0 (secs)
```

- 원인 : Jupyter Notebook의 출력 데이터용량 제한 초과

해결 과정 : IOPub data rate exceeded.

- cmd : `jupyter notebook --generate-config`

```
C:\>jupyter notebook --generate-config  
Writing default config to: C:\Users\user\.jupyter\jupyter_notebook_config.py
```



해결 과정 : IOPub data rate exceeded.

```
## (bytes/sec)
#       Maximum rate at which stream output can be sent on iopub before they are
#       limited.
# Default: 1000000
# c.NotebookApp.iopub_data_rate_limit = 1000000
c.NotebookApp.iopub_data_rate_limit = 1.0e10
```

오류 해결 : IOPub data rate exceeded.

```
In [5]: 1 url = callAPI(service_key,'1','10000','201903')
2 print(url)
3 response = requests.get(url).content
4 soup = BeautifulSoup(response, 'lxml-xml')
5 print(soup)
```

```
http://www.kspo.or.kr/openapi/service/nfaTestInfoService/getNfaTestRsItList?se
675JJRP0xb6e%2F9Xtg7N94DRG37oQr30uZY1JS6b3g%3D%3D&pageNo=1&numOfRows=10000&tes
<?xml version="1.0" encoding="utf-8"?>
```

```
<response><header><resultCode>00</resultCode><resultMsg>NORMAL SERVICE.</resultMsg></header><body><items><item><ageClass>10</ageClas
s><ageDegree>15</ageDegree><ageGbn>청소년</ageGbn><certGbn>2등급</certGbn><itemF001>161.5</itemF001><itemF002>60.9</itemF002><itemF0
03>35.4</itemF003><itemF004>65</itemF004><itemF005>77</itemF005><itemF006>116</itemF006><itemF007>29.4</itemF007><itemF008>31.7</ite
mF008><itemF009></itemF009><itemF010>37</itemF010><itemF012>17.2</itemF012><itemF013>21.19</itemF013><itemF014>0.402</itemF014><ite
mF015>44.297</itemF015><itemF016>16</itemF016><itemF017>45.033</itemF017><itemF018>23.3</itemF018><itemF019></itemF019><itemF020>37
</itemF020><itemF021></itemF021><itemF022>165</itemF022><itemF023></itemF023><itemF024></itemF024><itemF025></itemF025><itemF026
></itemF026><itemF027></itemF027><itemF028>52.0</itemF028><itemF029>38.1</itemF029><itemF030></itemF030><itemF031></itemF031><itemF032></itemF032><it
emF033></itemF033><itemF034></itemF034><itemF035></itemF035><itemF036></itemF036><itemF037></itemF037><itemF038></itemF038><it
emF039></itemF039><itemF040></itemF040><itemF041></itemF041><presNote>본운동:줄넘기 운동, 엉덩관절 회전하기, 스텝퍼 뛰어서 오르내리
기, 스텝퍼 옆으로 뛰어넘기</presNote><testYm>201903</testYm></item><item><ageClass>10</ageClass><ageDegree>15</ageDegree><ageGbn>청소년
</ageGbn><certGbn>2등급</certGbn><itemF001>155.3</itemF001><itemF002>56.8</itemF002><itemF003>33.1</itemF003><itemF004>70</itemF00
4><itemF005>62</itemF005><itemF006>105</itemF006><itemF007>22.1</itemF007><itemF008>26.9</itemF008><itemF009></itemF009><itemF010>4
0</itemF010><itemF012>16.2</itemF012><itemF013>22</itemF013><itemF014>0.365</itemF014><itemF015>60.95</itemF015><itemF016>5</itemF01
6><itemF017>61.18</itemF017><itemF018>23.6</itemF018><itemF019></itemF019><itemF020>51</itemF020><itemF021></itemF021><itemF022>17
0</itemF022><itemF023></itemF023><itemF024></itemF024><itemF025></itemF025><itemF026></itemF026><itemF027></itemF027><itemF028
>47.3</itemF028><itemF029>42.9</itemF029><itemF030></itemF030><itemF031></itemF031><itemF032></itemF032><itemF033></itemF033><itemF034></itemF034><ite
```

```
In [6]: 1 url = callAPI(service_key,'1','10000','201903')
2 print(url)
3 response = requests.get(url).content
4 soup = BeautifulSoup(response, 'lxml-xml')
5 print(soup)

http://www.kspo.or.kr/openapi/service/nfaTestInfoService/getNfaTestRsItList?serviceKey=h0s5ly4ntvjj009k49y88yu85918hU0C7enL2+rgJd7
SJ1P9Px0x6es2F9ta7N94DRG37oQr30uZY1JS6b3g%3D%3D&pageNo=1&numOfRows=10000&testYm=201903

IOPub data rate exceeded.
The notebook server will temporarily stop sending output
to the client in order to avoid crashing it.
To change this limit, set the config variable
`NotebookApp.iopub_data_rate_limit`.

Current values:
NotebookApp.iopub_data_rate_limit=100000.0 (bytes/sec)
NotebookApp.rate_limit=10000.0 (secs)
```

프로세스 오류 1-2: ttest_ind() NaN 오류

- ttest_ind() 오류
 - 문제: 비교군 데이터를 넣어서 ttest를 진행했으나 결과 NaN으로 나옴.

```

1 for i in range(len(nums)):
2     result = stats.ttest_ind(healthdf_before.iloc[:,i], healthdf_after.iloc[:,i], equal_var = False)
3     print(result)
4 #     print(healthdf_before.columns[i], 'statistic : ', result[0])
5 #     print(healthdf_before.columns[i], 'p-value : ', result[1])
6 #     print("")

executed in 187ms, finished 10:19:57 2022-04-01

Ttest_indResult(statistic=nan, pvalue=nan)

```

- 원인: 결측치를 제거하지 않고 진행해서 생긴 문제
- 해결: df에 결측치를 제거하고 진행하였더니 제대로 결과가 출력됨.

분석 1

뉴스기사 키워드 분석 : 유사도 및 워드클라우드

분석 1

분석 내용:

- 코로나 전/후 '홈트레이닝' 관련 뉴스 기사 내용 수집
- 두 시기의 기사 내용들을 가지고 코사인 유사도 분석 & 워드클라우드 제작

분석 프로세스

- 스크래핑 대상 홈페이지, 키워드 및 기간 설정
- 데이터 수집 : 월별 3페이지씩 스크래핑, 엑셀파일로 정리
- 데이터 정제 : 정규화, 어근화, 불용어 확인 및 제거 등의 텍스트 정제
- 시각화 : 코사인 유사도 분석 + 해시맵, 워드클라우드
- 결과 해석

오류 처리

분석 1 - 데이터 수집

✓ <https://search.naver.com/search.naver?where=news&query=%EC%BD%94%EC%8A%A4%ED%8A%B8>

✓ 데이터 수집 : 크롤링(BeautifulSoup)

- 기사 URL
- 제목
- 발행 날짜
- 본문
- 발행 언론사

N | 홀트레이닝

통합 쇼핑 어학사전 이미지 VIEW 지식iN 인플루언서 동영상 뉴스 지도 ...

• 관련도순 • 최신순 • 오래된순 옵션

PICK 언론사가 선정한 주요기사 혹은 심층기획 기사입니다.

매일경제 1일 전 네이버뉴스
메타버스 홀트레이닝 '야핏 사이클' 1분기 매출 150억 원 기록
운동·교육 중심 메타버스 기업 아나두는 메타버스 홀트레이닝 서비스 '야핏 사이클'이 올해 1분기 150억 원의... 야핏 사이클은 가장 세계에서 게임 하듯 운동할 수...

메타버스 홀트레이닝 '야핏 사이클', 1분기 매출 150억 원 IT조선 1일 전
홀트레이닝 서비스 '야핏 사이클', 올 1분기 매... 한국경제 1일 전 네이버뉴스
아나두 홀트레이닝 '야핏 사이클' 1분기 매출 150억... 벤처스퀘어 1일 전
야핏 사이클 1분기 매출 150억 거둬 "메타버스 홀트레이닝..." 시선뉴스 1일 전

강원일보 22면 1단 1일 전 네이버뉴스
[10분 홀트레이닝]고강도 운동·휴식 수차례 반복 체력 기르고 다...
이번 회부터 5회에 걸쳐 소개할 동작은 짧은 동작으로 체력을 유지할 수 있는 하루 5분 타박타다. 타박타는 1996년 일본의 운동생리학자인 '이즈미 타박타'가 운...

스타뉴스 7일 전 네이버뉴스
'호적메이트' 허재X허훈, 조준현X조준호의 태릉식 홀트레이닝 공...
'호적메이트'에서 허재, 허훈 부자와 조준현, 조준호 형제가 본격 홀트레이닝에 들입했다. 29일 방송된 MBC... 이 밖에 조준호가 춤선 조준현을 위해 스파르타 홀트...

IT조선 2022.03.25.
롯데온 홀트레이닝 용품 매출 50%↑
행사기간 실내자전거, 워킹패드, 던별 등 홀트레이닝 용품을 최대 40% 할인 판매하고, 인기 웰스 유튜버 '육체미 빅터'와 손잡고 홀트레이닝 용품 리뷰를 담은 영...

롯데온, 30일까지 홀트레이닝 용품... 파이낸셜뉴스 2022.03.25. 네이버뉴스
롯데온, 홀트레이닝 용품 할인행사 스트레이트뉴스 2022.03.25.
롯데온, 30일까지 홀트레이닝 용품 최대 40% 할인 인더뉴스 2022.03.25.

분석 1 - 데이터 수집

✓ <https://search.naver.com/search.naver?where=news&query=%EC%9E%91%ED%85%84%ED%8A%A7%EB%8D%BC%ED%8A%A7%EB%8D%BC&pd=3&ds=2018.01.01&de=2018.01.31>

The screenshot shows a search filter dialog box from Naver News. It has five sections: 정렬 (Sort), 기간 (Date), 유형 (Type), 언론사 (Publisher), and 기자명 (Journalist). The '기간' section is highlighted with a red underline under the date range '2018.01.01. ~ 2018.01.31.'.

정렬 • 관련도순 • 최신순 • 오래된순

기간 • 전체 • 1시간 ▾ • 1일 • 1주 • 1개월 • 3개월 • 6개월 • 1년
• 2018.01.01. ~ 2018.01.31. ▾

유형 • 전체 • 포토 • 동영상 • 지면기사 • 보도자료 • 자동생성기사

언론사 • 전체 • 언론사 분류순 ▾ • 가나다순 ▾

기자명 기자명 입력 적용

분석 1 - 데이터 수집

툴트레이닝 중 저주파 운동기고 한친구들과 콘텐츠 강화를 a.info
한친구들과 콘텐츠 강화를 Color 76.75 × 17
건강한친구들, 엠티원티와
엠티원티, 건강한친구들과
건강한친구들과 엠티원티
엠티원티, 훌트레이닝 '마
관련뉴스 8건 전체보기 > Contrast
Name
Role
Keyboard-focusable
세계일보 | 2022.03.23. 네이버뉴스
U+스마트홈 '구글패키지' 가입자 10만명 달성 임박
특히 20대 이용자는 콘텐츠를 시청하며 훌트레이닝이나 요리를 하는 등 다양하게
활용하는 것으로 나타났다. LG유플러스 유플러스 엠상필 홈 IoT 사업담당(상무)...
LG유플러스, U+스마트홈 '구글패키지' 가입자 10만명 ... CNB뉴스 | 2022.03.23.

```
><a href="https://news.naver.com/main/read.naver?mode=LSD&mid=sec&sid1=105&oid=009&aid=0004944799" class="info" target="_blank"
  onclick="return goOtherCR(this, 'a=nws*a.nav&r=1&i=880000BC_0000000000000004944799&u='+urlencode(this.href));">...</a> == $0
```



기사 주소



find_all('a',{'class','info'})

분석 1 - 데이터 수집

The screenshot shows the element inspector for the article title. The selected element is `h3#articleTitle.tts_head`. The properties panel shows the following details:

- Color:** #000000
- Font:** 30px "apple-system, BlinkMacSystemFont, Arial, sans-serif"
- Margin:** 0px 3px 1px 0px
- Contrast:** Aa 21 (green checkmark)
- Name:** 인니 홈클리닝 '오케이홈' 3억 투자 유치
- Role:** heading
- Keyboard-focusable:** (green checkmark)

The main content area displays the title: **인니 홈클리닝 '오케이홈' 3억 투자 유치**.

✓ 기사 제목

✓ `find('h3', {'id' : 'articleTitle'})`

The screenshot shows the element inspector for the date '04.06(수)'. The selected element is `span.t11`. The properties panel shows the following details:

- Color:** #888888
- Font:** 12px "Helvetica Neue", Arial, Tahoma, sans-serif
- Margin:** 0px 6px 0px 0px
- Contrast:** Aa 3.54 (yellow circle)
- Name:** 04.06(수)
- Role:** generic
- Keyboard-focusable:** (yellow circle)

The main content area displays the date: **04.06(수)**.

✓ 기사 날짜

✓ `find('span', {'class' : 't11'})`

분석 1 - 데이터 수집

The screenshot shows the Google Chrome DevTools Elements tab. A specific element, `<div id="articleBody" class="article_body _font_setting_target size3 font1">`, is selected and highlighted in blue. The right-hand panel displays detailed information about this element, including its dimensions (727 x 1852), color (#000000), font (Font 17px -apple-system, BlinkMacSystemFont...), padding (0px 40px 2px), and accessibility details. Below the element's details, the DOM tree shows the overall structure of the page.

```

<!-- 기사 헤더 -->
<div class="article_header">...</div>
<!-- // 기사 헤더 -->
<div class="article_body _font_setting_target size3 font1" id="articleBody">...</div>
<script type="text/javascript"> new FontSettingMenu(); </script>
<!-- 언론광재법 -->
    
```

본문

`find('h3', {'id' : 'articleTitle'})`

The screenshot shows the Google Chrome DevTools Elements tab. An element, `MoneyToday`, is selected and highlighted in blue. The right-hand panel displays detailed information about this element, including its dimensions (64.53 x 13), color (#444444), font (Font 11px 나눔고딕, NanumGothic, 돈ゴ, Dotu...), and accessibility details. Below the element's details, the DOM tree shows the overall structure of the page.

```

<ul>...</ul>
<p class="copyright">본원주의 저작권은 저작자 또는 네이버에 있으며 이를 무단 이용하는 경우 저작권법에 따른 법적 책임을 질 수 있습니다.</p>
<address class="address_cp nclicks(fot_presscr)">
    "Copyright © "
    <a href="http://www.mt.co.kr/" target="_blank">MoneyToday</a> => 50
    " All Rights Reserved."
</address>
<address class="address_nhn">...</address>
    
```

언론사

`find('address', {'class' : 'address_cp'}).find('a')`

프로세스 오류 2-1: .get_text()의 Nonetype 오류

- 원인: 스크래핑하면서 특정 뉴스 페이지들만 쓰는 태그가 달라질 때

```
In [49]: 1 title = source_news.find('h3', {'id': 'articleTitle'}).get_text()
2 date = source_news.find('span', {'class': 't11'}).get_text()

-----
AttributeError                                 Traceback (most recent call last)
<ipython-input-49-affbb62dab6f> in <module>
----> 1 title = source_news.find('h3', {'id': 'articleTitle'}).get_text()
      2 date = source_news.find('span', {'class': 't11'}).get_text()

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

- 해결: try/ except로 nonetype 오류가 나면

해당 페이지는 스크래핑 하지 않고 다음으로 넘어가게 함.

분석 1 - 데이터 수집 (괄호 내용 제거)

```
# [ ] < > ( ) 괄호로 감싸여 있는 부분 삭제
pattern = re.compile(r'[^<()][\s\S+,+-,:=%]*[^>]')')
article_content = pattern.sub(' ', article_content)
print("대괄호 제거후 : ", article_content)
```

이메일 이후 제거: [사진=월다 폭스 인스타그램] 출산 이후 홈트레이닝만으로 26kg 감량에 성공한 엄마가 있다. 1일(현지시간) 데일리메일 호주판은 꾸준한 노력으로 1년만에 깜짝 놀랄 변신에 성공한 월다 폭스(28)를 소개했다. 호주 멜버른에 살고 있는 월다는 지난 2016년 첫 아들 할리를 낳은 뒤 몸무게가 80kg까지 불어났다. 출산 전 68kg였던 월다는 날씬한 몸매를 얻기 위해 유행한다는 다이어트는 다 도전했다. 그러나 번번히 실패한 월다는 더 짹버린 살에 자신감을 잃었다. [사진=월다 폭스 인스타그램] 아이를 낳은 뒤에는 더욱 헬스장을 찾기가 쉽지 않았다. 첫 아이로 육아에 서툴었던데다 아이를 돌봐줄 사람도 마땅치 않았기 때문이다. 결국 월다는 다니던 헬스장을 끊고 집에서 홈 트레이닝을 하기로 결심했다. 월다는 할리를 재운 뒤 20~25분 짧지만 강도 높은 운동을 시작했다. 주로 했던 동작은 스쿼트, 버피 테스트, 플랭크 등이었다. 시간이 나면 1시간 정도 집 주변을 조깅했다. [사진=월다 폭스 인스타그램] 몸이 점점 달라지는 것을 느끼면서 월다는 식단에도 신경쓰기 시작했다. 그는 현미와 갑자로 탄수화물을 보충하고 채식 위주의 식사를 했다. 간식으로는 과일을 먹었다.

괄호 제거 전

괄호 제거후 : 출산 이후 홈트레이닝만으로 26kg 감량에 성공한 엄마가 있다. 1일 데일리메일 호주판은 꾸준한 노력으로 1년만에 깜짝 놀랄 변신에 성공한 월다 폭스를 소개했다. 호주 멜버른에 살고 있는 월다는 지난 2016년 첫 아들 할리를 낳은 뒤 몸무게가 80kg까지 불어났다. 출산 전 68kg였던 월다는 날씬한 몸매를 얻기 위해 유행한다는 다이어트는 다 도전했다. 그러나 번번히 실패한 월다는 더 짹버린 살에 자신감을 잃었다. 아이를 낳은 뒤에는 더욱 헬스장을 찾기가 쉽지 않았다. 첫 아이로 육아에 서툴었던데다 아이를 돌봐줄 사람도 마땅치 않았기 때문이다. 결국 월다는 다니던 헬스장을 끊고 집에서 홈 트레이닝을 하기로 결심했다. 월다는 할리를 재운 뒤 20~25분 짧지만 강도 높은 운동을 시작했다. 주로 했던 동작은 스쿼트, 버피 테스트, 플랭크 등이었다. 시간이 나면 1시간 정도 집 주변을 조깅했다.

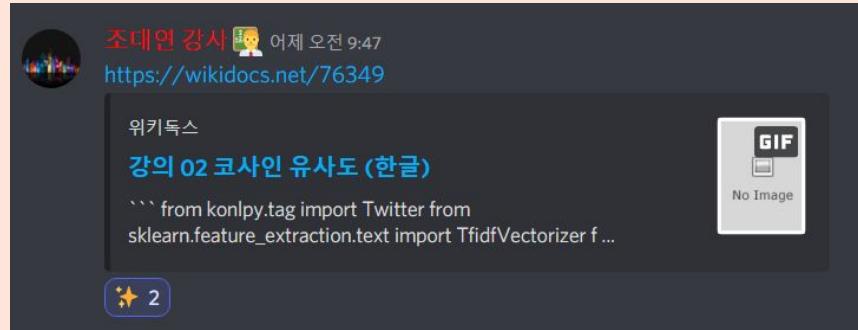
괄호 제거 후

분석 1 - 데이터 수집 엑셀 파일

A	B	C	D	E	F	G
1	Title	Date	Article	URL	PressCompany	
2	[지금은 흠	2018.01.14 -비용 부담	https://new	헤럴드경제		
3	몸매관리도	2018.01.26 G마켓, 몸	https://new	연합뉴스		
4	원마이 챔	2018.01.29 홈케어, 흠	https://new	MBN		
5	[친절한 경	2018.01.26 동영상 뉴	https://new	SBS & SBSi		
6	옷도 렌탈	2018.01.21 코오롱Fn	https://new	The Internet Hankyoreh		
7	매쉬업엔젤	2018.01.23 사진=매쉬	https://new	edaily		
8	롯데렌탈,	2018.01.18 롯데월드E	https://new	연합뉴스		
9	아디다스	2018.02.08 아이다스하	https://new	스포츠조선		
10	카카오VX,	2018.02.13 [02월 13일]	https://new	한경닷컴		
11	가성비甲	· 2018.02.08 최근 국내(https://new	디지털타임스		
12	[2018년 키	2018.02.07 7일 카카오	https://new	디지털데일리		
13	'하반기 IP(2018.02.07 - 상반기 If	https://new	edaily		
14	카카오게	2018.02.07 사진= 좌쪽	https://new	포모스		
15	'빅4 아닌	· 2018.02.08 남궁훈 카	https://new	디지털데일리		
16	유산소+근	2018.03.26 웰스기기 :	https://new	financial news		
17	을고 웃은	2018.03.01 출산 이	https://new	financial news		
18	[TAPAS]흡	2018.03.30 미세먼지의	https://new	헤럴드경제		
19	부산국제별	2018.03.29 【부산=뉴	https://new	뉴시스		
20	벡스코서	: 2018.03.29 【부산=뉴	https://new	뉴시스		
21	【부산국제영화제】	2018.03.29 2018부산국제영화제' 기	https://new	하경다크		

프로세스 오류 2-2: 한글데이터 코사인 유사도 분석

- 질문 : 한글 데이터도 우리가 배운것처럼 영어로 코사인 유사도 분석하는 것처럼 분석이 가능한 것인가? → 질문을 위해 강사님 소환 🎀
- 해결 : 가능하다. 관련 위키독스를 참고 <https://wikidocs.net/76349>



분석 결과 - 코사인 유사도 분석

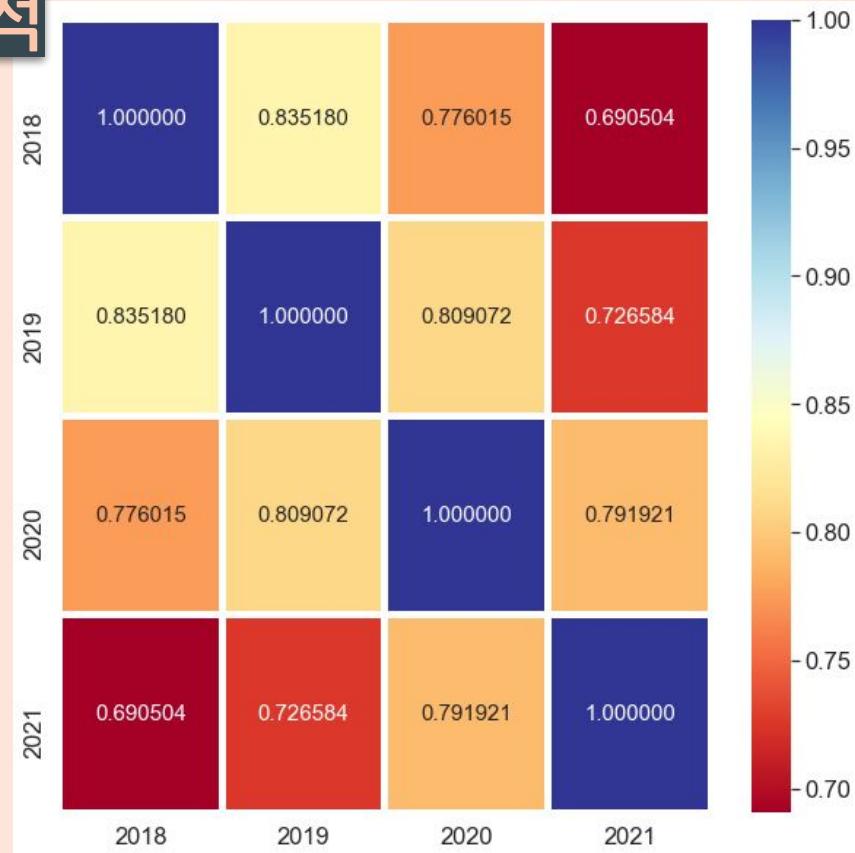
```
1 # seaborn 시각화를 위해 DataFrame 객체로 만들어줍니다.  
2 cos_df = pd.DataFrame(cosine_similarity(corpus), index=['2018', '2019', '2020', '2021'], columns=['2018', '2019', '2020', '2021'])  
3 cos_df
```

	2018	2019	2020	2021
2018	1.000000	0.835180	0.776015	0.690504
2019	0.835180	1.000000	0.809072	0.726584
2020	0.776015	0.809072	1.000000	0.791921
2021	0.690504	0.726584	0.791921	1.000000

분석 결과 - 코사인 유사도 분석

✓ 2018-2019 사이, 2020-2021 사이의 기사글이
서로 유사도가 높게 나타남.

✓ 단, 2019년과 2020년의 홈트레이닝 관련 뉴스 기사의
경우, 다른 연도와 비교해서 유사도가 높은 것으로
나타남.
2019년과 2020년 홈트레이닝 관련 기사의 내용에 코로나외의
요인에 대한 영향으로 볼 수 있을 것 같음.
(코로나 이전에도 헬스케어 디바이스의 발전으로 운동,
홈트에 대한 관심이 증가한 부분, 텍스트 데이터의 부족 등)



분석 결과 - 워드클라우드

```
1 # apple로고 워드클라우드를 만드는 사용자 함수
2 def appleWordCloud(word_dic, filename):
3     apple_logo = np.array(Image.open('apple1.jpg'))
4     image_colors = ImageColorGenerator(apple_logo)
5
6     word_cloud = WordCloud(font_path="C:/Windows/Fonts/malgun.ttf",
7                             width = 2000, height = 1000,
8                             mask = apple_logo,
9                             background_color = 'white').generate_from_frequencies(word_dic)
10
11    word_cloud = word_cloud.recolor(color_func=image_colors)
12    word_cloud.to_file(filename='{}.jpg'.format(filename))
13    plt.figure(figsize=(15,15))
14    plt.imshow(word_cloud, interpolation='bilinear')
15    plt.axis("off")
16    plt.tight_layout(pad=0)
17    plt.show()
```



원본 이미지

분석 결과 - 워드클라우드



2018-2019



2020-2021

분석 2

주가 분석 : 네이버 금융 스크래이핑, 주가 변동 시각화

분석 2

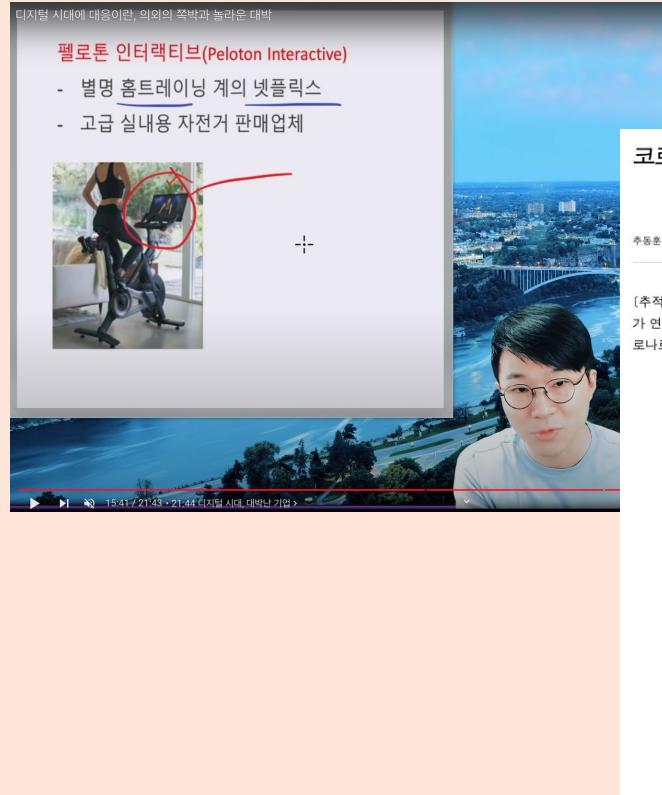
데이터 수집 방법

- 국내 데이터 - 스크래핑
- 해외 데이터 - 라이브러리

분석 프로세스

- 대상 종목 선정
- 데이터 수집
- 데이터 정제
- 시각화
- 결과 해석

오류 처리



분석 2 - 종목 선정 및 데이터 수집

헬스케어 관련 주

1. InBody → 국내 KOSDAQ: 041830
2. 펠로톤 인터렉티브(PTON) → 미국 NASDAQ: PTON
3. 플래닛 피트니스(PLNT) → 미국 NYSE: PLNT
4. 노틸러스(NLS) → 미국 NYSE : NLS

헬스복 관련

1. 룰루레몬 → 미국 NASDAQ: LULU
 - 레깅스 업체
2. 나이키 → 미국 NYSE: NKE
3. 언더아머 → 미국 NYSE: UAA
4. 아디다스 → 독일 ETR: ADS
5. 브랜드엑스코퍼레이션 → 국내 KOSDAQ: 337930
 - 레깅스 업체

닭가슴살 관련

1. 푸드나무 → 국내 KOSDAQ: 290720
 - a. 랭킹닭컴

✓ 대상 종목 선정 : 분야별 선정

- 헬스케어, 헬스복, 헬스푸드

✓ 종목별 주가 데이터 정의

- 2018 ~ 2019: 코로나 이전 주가 (일별 종가)
- 2020 ~ 2022: 코로나 이후 주가 (일별 종가)

✓ 데이터 수집

스크래핑 (Selenium)

& 라이브러리 (FinanceDataReader)

- 국내 : 네이버 금융 일별 시세
- 해외 : 라이브러리 이용

분석 2 - 데이터 수집 (Selenium)

```

<table cellspacing="0" class="type2">
  <tbody>
    <tr>
      <th>날짜</th>
      <th>종가</th>
      <th>전일비</th>
      <th>시가</th>
      <th>고가</th>
      <th>저가</th>
      <th>거래량</th>
    </tr>
    <tr>...</tr>
    <tr onmouseover="mouseOver(this)" onmouseout="mouseOut(this)">
      <td align="center">
        <span class="tah p10 gray03">2022.04.05</span>
      </td>
      <td class="num">
        <span class="tah p11">28,800</span>
      </td>
      <td class="num">
         150 </span>
      </td>
      <td class="num"> == $0
        <span class="tah p11">28,500</span>
      </td>
      <td class="num">
    
```

finance.naver.com/item/sise_day.nhn?code=041830&page=

일별 시세				
날짜	종가	전일비	시가	고가
2022.04.05	28,800	▲ 150	28,500	
2022.04.04	28,650	▼ 250	29,200	
2022.04.01	28,900	▲ 900	27,700	
2022.03.31	28,000	▲ 150	27,850	
2022.03.30	27,850	▲ 400	27,400	
2022.03.29	27,450	▲ 250	27,300	
2022.03.28	27,200	▼ 700	27,900	
2022.03.25	27,900	▲ 950	27,300	
2022.03.24	26,950	▲ 300	26,500	
2022.03.23	26,650	▲ 350	26,500	

네이버 금융 종목 일별 시세

1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 다음 | 맨뒤 | >

✓ 칼럼 이름 : <table> 첫번째 <tr>태그

✓ 본문 : title 제외하고 1행 부터

✓ 추출 데이터: 날짜 (연,월,일) & 종가 (Close)

스크래핑 중 오류 2가지 발생 → 오류 별도 설명

분석 2 - 데이터 수집 (Library)

```
[6]: import pandas as pd
2 import numpy as np
3 import FinanceDataReader as fdr
4 #한국 주식 가격, 미국주식 가격, 지수, 환율, 암호화폐 가격, 종목 리스팅 등 금융 데이터 수집 라이브러리
5 fdr.__version__
'0.9.33'

[7]: health_care_dict = {'인바디' : '041830', '펠로톤' : 'PTON', '플래닛피트니스' : 'PLNT', '노틸러스' : 'NLS',
1       '클루레몬' : 'LULU', '나이키' : 'NKE', '언더아머' : 'UAA', '브랜드엑스코퍼레이션' : '337930',
2       '푸드나무' : '290720'}
3
4
5 item_list = []
6 for item_code in health_care_dict.values():
7     close = fdr.DataReader(item_code, '2018')['Close']
8     item_list.append(close)
9
10
11 df_all_stocks = pd.concat(item_list, axis = 1)
12 df_all_stocks.columns = health_care_dict.keys()
13 df_all_stocks
```

인바디 펠로톤 플래닛피트니스 노틸러스 클루레몬 나이키 언더아머 브랜드엑스코퍼레이션 푸드나무

Date	인바디	펠로톤	플래닛피트니스	노틸러스	클루레몬	나이키	언더아머	브랜드엑스코퍼레이션	푸드나무
2018-01-02	40100.0	NaN	32.96	13.35	79.69	63.49	15.03	NaN	NaN
2018-01-03	38750.0	NaN	32.90	13.30	78.59	63.48	15.72	NaN	NaN
2018-01-04	37450.0	NaN	32.51	12.95	79.85	63.44	15.92	NaN	NaN
2018-01-05	38200.0	NaN	33.70	13.15	79.43	63.98	15.87	NaN	NaN
2018-01-08	39950.0	NaN	33.49	12.90	79.04	64.55	15.98	NaN	NaN
...
2022-03-30	27850.0	28.44	85.00	4.16	376.92	138.54	17.66	8660.0	26450.0
2022-03-31	28000.0	26.42	84.48	4.12	365.23	134.56	17.02	9350.0	28450.0
2022-04-01	28900.0	26.31	84.69	4.15	367.44	133.52	16.77	8980.0	27800.0
2022-04-04	28650.0	27.81	84.39	4.08	384.18	134.34	16.96	9470.0	27800.0
2022-04-05	28800.0	26.32	82.94	4.05	376.73	133.11	16.49	9210.0	30150.0

1105 rows × 9 columns

✓ 라이브러리 FinanceDataReader 소개

한국, 미국 거래소 내 종목 가격 데이터 제공:

KOSPI, KOSDAQ, NASDAQ, NYSE 등

활용예시: df = fdr.DataReader('종목코드',

'시작연도', '끝연도')



✓ 날짜, 일별 종가, 종목 명 데이터 추출,

데이터프레임

실시간 시계열 데이터 이므로 정의된 데이터

CSV로 추출 후 작업

Data Preprocessing

Raw Stock Price Line Chart



2018년 ~ 현재 헬스케어 관련 주가



✓ 스케일 조정

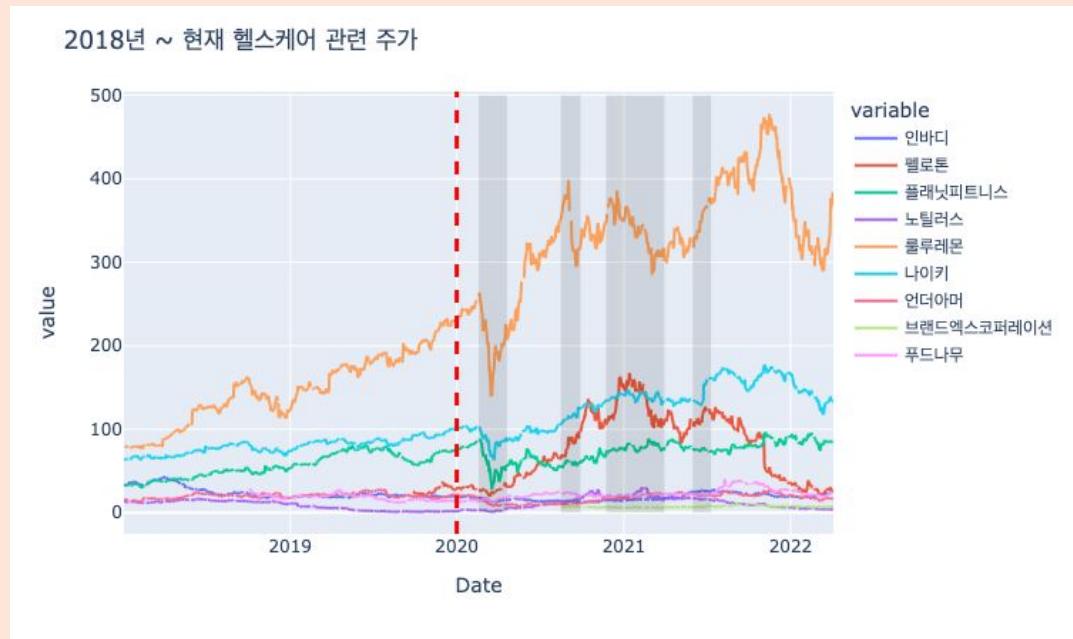
한국, 미국 주식 표시 가격 차이

→ 환율로 나누어 달러화로 스케일 조정

(4개년 평균 환율)

Visualization 시각화

- 코로나 대 유행 시기 음영처리
- 전체 주가 시각화
- 코로나 전후 시기 구분 : 빨간색 점선
- 4회 중 앞의 3회 유행 시기에서
주가가 크게 떨어짐



Visualization 시각화 - 헬스 케어

- 코로나 대 유행 시기 음영처리
- 전체 주가 시각화
- 코로나 전후 시기 구분 : 빨간색 점선
- 4회 중 앞의 3회 유행 시기에서
주가가 크게 떨어짐



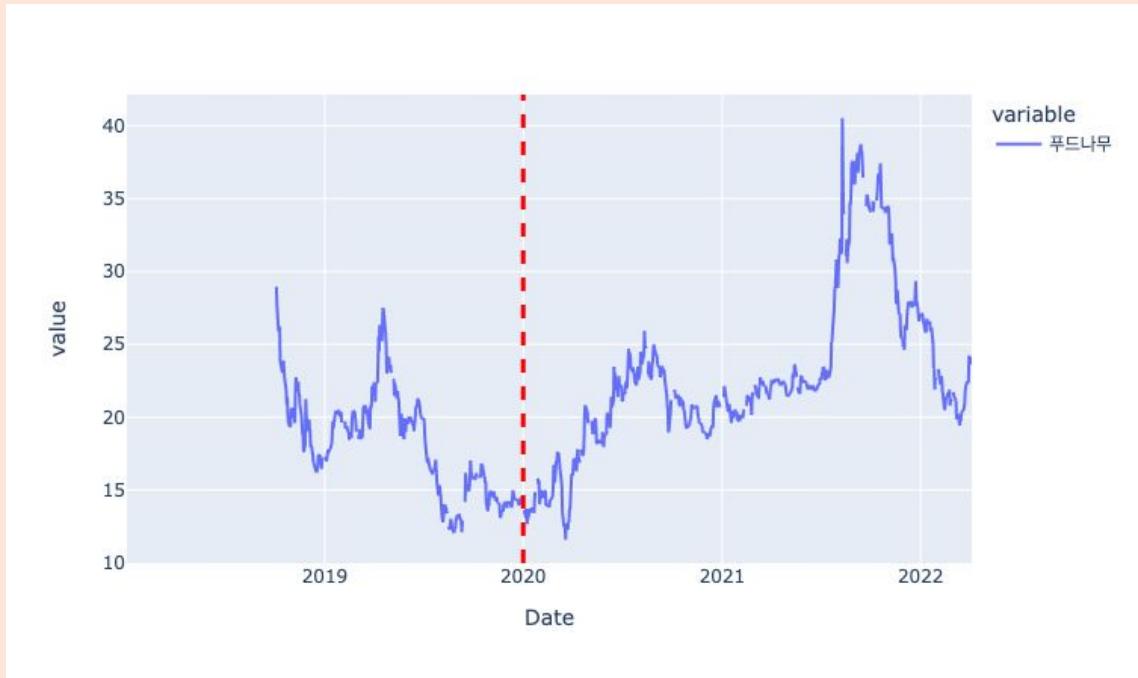
Visualization 시각화 - 헬스복

- ✓ 코로나 대 유행 시기 음영처리
- ✓ 전체 주가 시각화
- ✓ 코로나 전후 시기 구분 : 빨간색 점선
- ✓ 4회 중 앞의 3회 유행 시기에서
주가가 크게 떨어짐

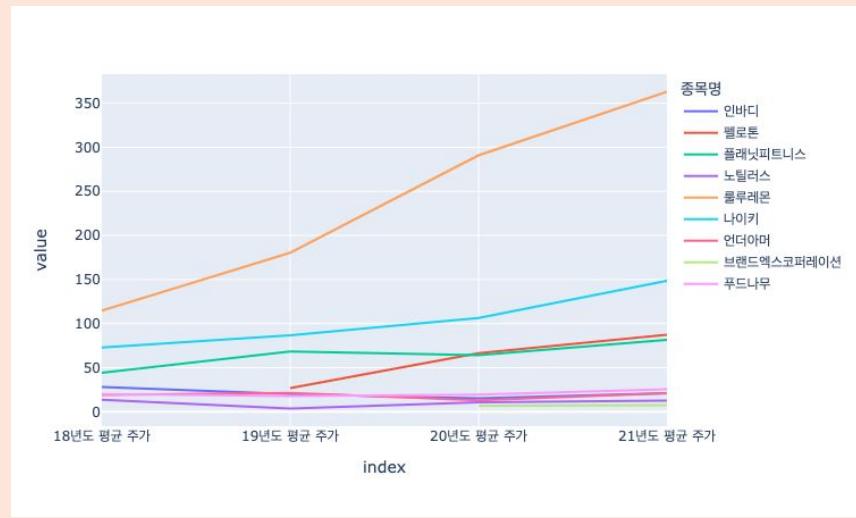
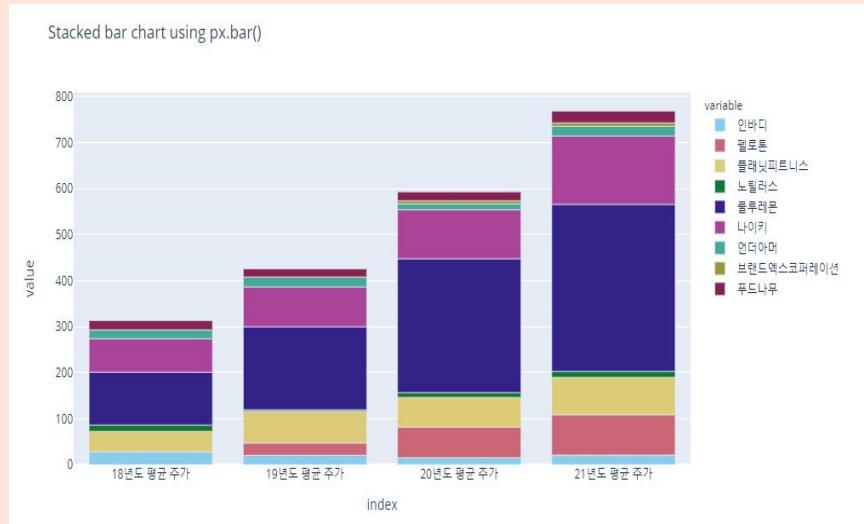


Visualization 시각화 - 헬스 푸드

- 코로나 대 유행 시기 음영처리
- 전체 주가 시각화
- 코로나 전후 시기 구분 : 빨간색 점선
- 4회 중 앞의 3회 유행 시기에서
주가가 크게 떨어짐



Visualization 시각화 - 종목별 연 평균 주가



✓ 연도별 종목별 주가 평균 데이터 생성

Series to DF & Transpose
휴장일은 결측치 이므로 제외



유의사항

상장 시기, 주식 거래일 차이

- 휴장일은 결측치로 정의
- 평균 계산시 결측치 제거

프로세스 오류 3-1: 스크래핑

Error 2)

```

1 #korea_url = "http://finance.naver.com/item/sise.naver" + "?code=" + "041830"
2 korea_url = "http://finance.naver.com/item/sise_day.nhn" + "?code=" + "005930" + "&page="
3 print(korea_url)
4
5 res = requests.get(korea_url).content
6 soup = BeautifulSoup(res.decode('euc-kr', 'replace'), 'html.parser')
7 # 한글깨짐 해결 -> content.decode('euc-kr', 'replace')

[68] ✓ 0.5s
...
http://finance.naver.com/item/sise_day.nhn?code=005930&page=

> ^
1 #korea_url = "http://finance.naver.com/item/sise.naver" + "?code=" + "041830"
2 korea_url = "http://finance.naver.com/item/sise_day.nhn" + "?code=" + "005930" + "&page="
3 print(korea_url)
4
5 res = requests.get(korea_url).content
6 soup = BeautifulSoup(res.decode('euc-kr', 'replace'), 'html.parser')
7 # 한글깨짐 해결 -> content.decode('euc-kr', 'replace')

[69] ✓ 0.2s
...
http://finance.naver.com/item/sise_day.nhn?code=005930&page=

> ^
1 table0 = soup.find('table', {'class': 'type2'})
2 print(table0)

[70] ✓ 0.2s
...
None

```

* **문제:** 네이버 금융 내 모든 시세 테이블 클래스 “type2”로 동일

→ Request 방법

원하는 일별 시세 테이블 뽑을 수 없었음

→ full XPath 방법

nonetype으로 정보 읽지 못함

* **해결방안:** 일별 시세만 나와있는 URL에서 작업

→ Selenium & BeautifulSoup 으로 일별 시세 추출 성공

프로세스 오류 3-1:

Error 2)

```

1 #korea_url = "http://finance.naver.com/item/sise.naver"
2 korea_url = "http://finance.naver.com/item/sise_day.nhn"
3 print(korea_url)
4
5 res = requests.get(korea_url).content
6 soup = BeautifulSoup(res.decode('euc-kr', 'replace'),
7 # 한글깨짐 해결 -> content.decode('euc-kr', 'replace')

```

[68] ✓ 0.5s

... http://finance.naver.com/item/sise_day.nhn?code=005930&page=

```

1 #korea_url = "http://finance.naver.com/item/sise.naver"
2 korea_url = "http://finance.naver.com/item/sise_day.nhn"
3 print(korea_url)
4
5 res = requests.get(korea_url).content
6 soup = BeautifulSoup(res.decode('euc-kr', 'replace'),
7 # 한글깨짐 해결 -> content.decode('euc-kr', 'replace')

```

[69] ✓ 0.2s

... http://finance.naver.com/item/sise_day.nhn?code=005930&page=

```

1 table0 = soup.find('table', {'class': 'type2'})
2 print(table0)

```

[70] ✓ 0.2s

... None

날짜	종가	전일비	시가	고가	저가	거래량
2022.03.21	25,950	▼ 750	26,650	26,700	25,600	158,163
2022.03.18	26,700	▲ 1,300	25,400	26,800	25,200	168,033
2022.03.17	25,400	▼ 50	25,850	25,850	25,400	36,167
2022.03.16	25,450	▲ 350	25,400	25,550	25,100	19,041
2022.03.15	25,100	0	25,100	25,600	24,950	27,663
2022.03.14	25,100	▼ 550	25,600	25,600	25,100	57,803
2022.03.11	25,650	▼ 100	25,700	26,350	25,500	44,447
2022.03.10	25,750	▲ 400	25,700	26,100	25,100	50,434
2022.03.08	25,350	▼ 50	25,150	25,900	25,050	47,810
2022.03.07	25,400	▼ 600	25,900	26,100	25,300	88,929

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

시세정보 바로가기 더보기 ▾

코스피 | 코스닥 | 선물 | 코넥스
코스피200 | ETF | 업종별 | 테마별

Elements Console Sources Network Performance Memory Application Security >

> <head> ... </head>
 > <body marginheight="0" data-new-gr-c-s-check-loaded="14.1055.0" data-gr-ext-installed>
 > <script language="JavaScript"> ... </script>
 > <h4 class="tline2"> ... </h4>
 > <table cellspacing="0" class="type2">
 > <tbody> == \$0
 > <tr> ... </tr>
 > <tr> ... </tr>
 > <tr onmouseover="mouseOver(this)" onmouseout="mouseOut(this)" style="background-color: rgb(255, 255, 255);">
 > <td align="center"> ... </td>
 > <td class="num">
 > 25,950
 ... iv#wrap div#middle.new_totalinfo div.content_wrap div#content div.section.inner_sub iframe html body table.type2 tbody ...
 table
 : Console What's New x
 view and edit @supports at-rules
 The CSS @supports at-rules are now displayed and editable in the Styles pane.
 Recorder panel improvements

프로세스 오류 3-1: 스크래핑

Error 2)

```

1 #korea_url = "http://finance.naver.com/item/sise.naver" + "?code=" + "041830"
2 korea_url = "http://finance.naver.com/item/sise_day.nhn" + "?code=" + "005930" + "&page="
3 print(korea_url)
4
5 res = requests.get(korea_url).content
6 soup = BeautifulSoup(res.decode('euc-kr', 'replace'), 'html.parser')
7 # 한글깨짐 해결 -> content.decode('euc-kr', 'replace')

[68] ✓ 0.5s
...
http://finance.naver.com/item/sise_day.nhn?code=005930&page=

> ^
1 #korea_url = "http://finance.naver.com/item/sise.naver" + "?code=" + "041830"
2 korea_url = "http://finance.naver.com/item/sise_day.nhn" + "?code=" + "005930" + "&page="
3 print(korea_url)
4
5 res = requests.get(korea_url).content
6 soup = BeautifulSoup(res.decode('euc-kr', 'replace'), 'html.parser')
7 # 한글깨짐 해결 -> content.decode('euc-kr', 'replace')

[69] ✓ 0.2s
...
http://finance.naver.com/item/sise_day.nhn?code=005930&page=

> ^
1 table0 = soup.find('table', {'class': 'type2'})
2 print(table0)

[70] ✓ 0.2s
...
None

```

* **문제:** 네이버 금융 내 모든 시세 테이블 클래스 “type2”로 동일

→ Request 방법

원하는 일별 시세 테이블 뽑을 수 없었음

→ full xPath 방법

nonetype으로 정보 읽지 못함

* **해결방안:** 일별 시세만 나와있는 URL에서 작업

→ Selenium & BeautifulSoup 으로 일별 시세 추출 성공



나리

승욱

준용

지원



감사합니다

