

빅 데이터 분석을 위한 알고리즘 활용

빅 데이터분석을 위한 알고리즘 소개 및 Python을 활용한 실습



CONTENTS

DATA Analysis

01

파이썬 프로그래밍 기초

파이썬에서 활용 가능한 자료의 형태와 기능들에 대해 학습함

02

데이터 수집

데이터 수집과정을 소개하고 파이썬을 이용하여 데이터를 수집하는 방법을 학습함

03

데이터 전처리 및 시각화

분석 데이터의 구조 및 형태와 파이썬을 활용하여 분석 데이터를 생성하기 위한 방법들을 학습함

04

AI 알고리즘 맛보기

회귀모형과, Tree기반의 앙상블 모형에 대해 간단히 학습하고 파이썬으로 생성함

05

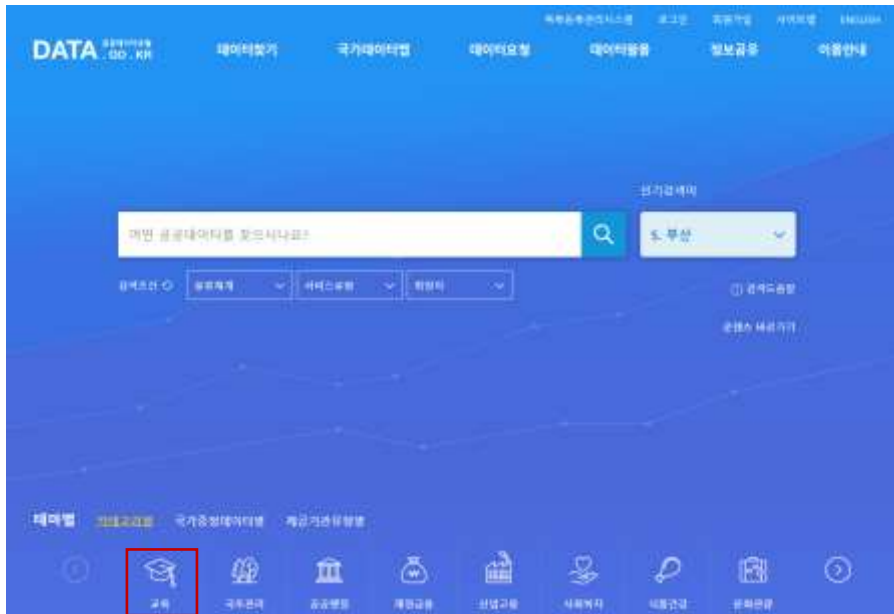
모형 평가 방법

다양한 AI 알고리즘들을 생성하고 평가하는 방법에 대해 학습함

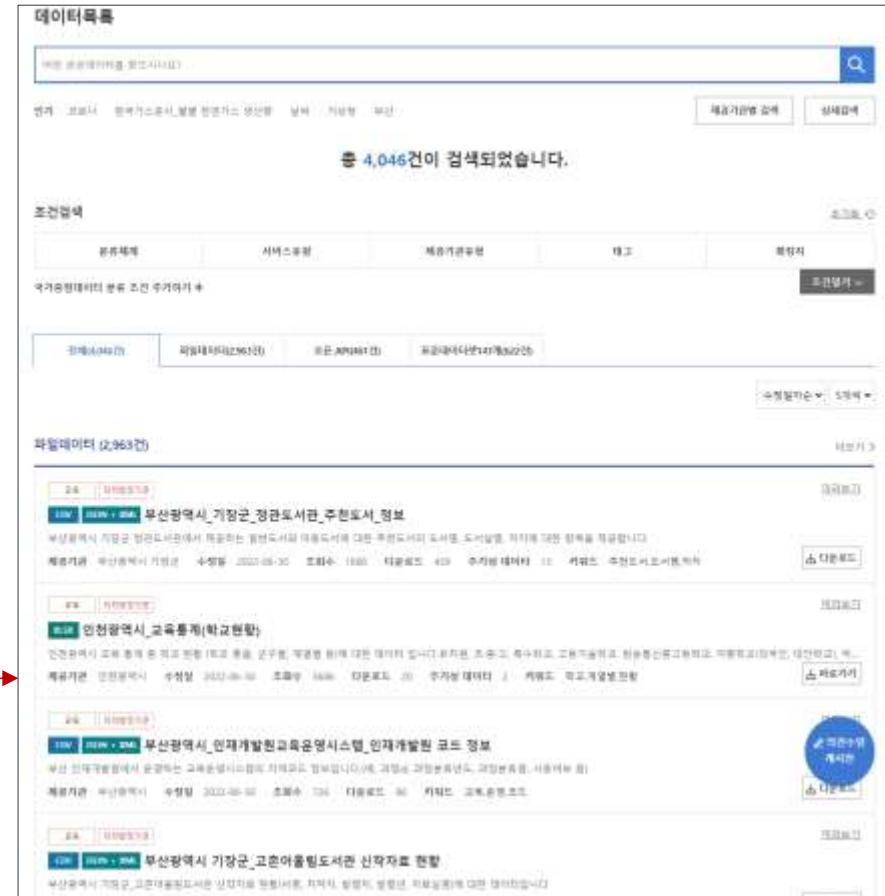
2.1 데이터 수집

02. 데이터 수집

- 활용해 볼 수 있는 데이터로 쉽게 수집할 수 있는 방법은 “공공 데이터 포털”을 활용하는 것 입니다. 이를 활용하여 데이터 분석 연습 뿐만 아니라 실제 분석 결과를 통한 활용도 가능합니다.
- 공공 데이터 포털 뿐만 아니라 행동 데이터를 의미하는 로그 데이터 혹은 웹사이트에 있는 영화 리뷰 데이터 등과 같은 정보도 크롤링을 통해 수집이 가능합니다.



• <https://www.data.go.kr/>



2.1 데이터 수집

02. 데이터 수집

- 활용해 볼 수 있는 데이터로 쉽게 수집할 수 있는 방법은 “공공 데이터 포털”을 활용하는 것 입니다. 이를 활용하여 데이터 분석 연습 뿐만 아니라 실제 분석 결과를 통한 활용도 가능합니다.
- 공공 데이터 포털 뿐만 아니라 행동 데이터를 의미하는 로그 데이터 혹은 웹사이트에 있는 영화 리뷰 데이터 등과 같은 정보도 크롤링을 통해 수집이 가능합니다.

NAVER 영화

영화홈
상영작 예정작
영화평점
예매
사사회 이벤트
평점 리뷰
내리온 리뷰
영화리뷰
다들보고
영화리뷰

내리온 평점 - 140자평

전체 평점: 140자평 보기
개별 평점: 140자평 보기

전체 리스트

개별 평점: 140자평 보기

번호 평점 140자평 글쓴이 날짜

13369433 ★★★★★ 10 포 하나의 약속(100654)

13369432 ★★★★★ 1 거룩한 계보(58085)

13369424 ★★★★★ 9 창산별(137696)

13369423 ★★★★★ 8 감기(72522)

13369419 ★★★★★ 1 늑대소년(88253)

13369416 ★★★★★ 10 프리덤(137991)

13369413 ★★★★★ 8 빠르게 살자(65540)

13369413 ★★★★★ 9 A-특공대(50598)

13369413 ★★★★★ 9 악마를 보았다(72408)

13369413 ★★★★★ 9 파파로티(85640)

13369412 ★★★★★ 2 악녀(155256)

13369412 ★★★★★ 4 소셜포비아(122457)

13369412 ★★★★★ 10 불남은 칸다(31801)

13369412 ★★★★★ 10 인터스텔라(45290)

13369412 ★★★★★ 10 아이 엠 샘(34227)

13369412 ★★★★★ 10 명랑(93756)

13369409 ★★★★★ 8 제이슨 본(144968)

13369409 ★★★★★ 7 아가씨(123519)

13369409 ★★★★★ 10 베테랑(115977)

13369409 ★★★★★ 10 사도(121922)

13369409 ★★★★★ 10 미션 임파서블: 로그네이션(95541)

13369400 ★★★★★ 10 무서운 집(126389)

13369400 ★★★★★ 9 뷰티 인사이드(129050)

13369400 ★★★★★ 9 워터월드(17116)

13369400 ★★★★★ 10 복수혈전(11354)

13369400 ★★★★★ 10 세 일간제(73372)

13369400 ★★★★★ 8 번호인(101901)

13369400 ★★★★★ 10 바람(70773)

13369400 ★★★★★ 8 한반도(41705)

13369400 ★★★★★ 9 악의교전(98420)

13369396 ★★★★★ 7 남과 여(122133)

A	B	C	D	F	G
id	nickname	date	movieID	point	movie
13369433	hms****	2014-02-08	100654	10	포 하나의 약속(100654)
13369432	cjfw****	2017-05-12	58085	1	거룩한 계보(58085)
13369424	roma****	2017-08-28	137696	9	창산별(137696)
13369423	yuns****	2014-09-09	72522	8	감기(72522)
13369419	saty****	2017-06-28	88253	1	늑대소년(88253)
13369416	kso7****	2015-11-24	137991	10	프리덤(137991)
13369413	jsp6****	2017-08-21	65540	8	빠르게 살자(65540)
13369413	jsp6****	2017-08-21	50598	9	A-특공대(50598)
13369413	jsp6****	2017-07-21	72408	9	악마를 보았다(72408)
13369413	jsp6****	2017-07-21	85640	9	파파로티(85640)
13369412	dlsd****	2017-07-18	155256	2	악녀(155256)
13369412	dlsd****	2015-03-21	122457	4	소셜포비아(122457)
13369412	dlsd****	2015-02-17	31801	10	불남은 칸다(31801)
13369412	dlsd****	2015-02-05	45290	10	인터스텔라(45290)
13369412	dlsd****	2015-01-22	34227	10	아이 엠 샘(34227)
13369412	dlsd****	2014-12-18	93756	10	명랑(93756)
13369409	love****	2016-08-17	144968	8	제이슨 본(144968)
13369409	love****	2016-07-25	123519	7	아가씨(123519)
13369409	love****	2015-10-07	115977	10	베테랑(115977)
13369409	love****	2015-10-05	121922	10	사도(121922)
13369409	love****	2015-08-29	95541	10	미션 임파서블: 로그네이션(95541)
13369400	brai****	2016-01-26	126389	10	무서운 집(126389)
13369400	brai****	2015-11-27	129050	9	뷰티 인사이드(129050)
13369400	brai****	2014-06-22	17116	9	워터월드(17116)
13369400	brai****	2014-05-16	11354	10	복수혈전(11354)
13369400	brai****	2014-02-02	73372	10	세 일간제(73372)
13369400	brai****	2013-12-23	101901	8	번호인(101901)
13369400	brai****	2013-10-30	70773	10	바람(70773)
13369400	brai****	2013-10-20	41705	8	한반도(41705)
13369400	brai****	2013-10-20	98420	9	악의교전(98420)
13369396	rlog****	2016-06-21	122133	7	남과 여(122133)

DATA GO.KR | 데이터찾기 | 국가데이터맵 | 데이터요청 | 데이터활용 | 정보공유 | 이용안내

1. 검색어: 일별 미세먼지

2. "일별 미세먼지"에 대해 총 470건이 검색되었습니다.

조건검색

분류체계	서비스유형	제공기관유형	태그	확장자
국가중점데이터 분류 조건 추가하기 +				

전체(470건) | 파일데이터(379건) | 오픈 API(91건) | 표준데이터셋(0건)

정확도순 | 5개씩 | 정렬

파일데이터 (379건)

환경가상	지리정보	제목	제공기관	수정일	조회수	다운로드	키워드
CSV		서울특별시_일별 평균 대기오염도 정보	서울특별시	2021-06-16	1200	33	미세먼지, 오존, 이산화질소
PDF		대전광역시_미세먼지 현황	대전광역시	2022-07-26	4598	106	미세먼지 주위, 미세먼지, 미세먼지 농도

2.2 공공 데이터 포털

02. 데이터 수집

3
메타데이터 다운로드

파일명	서울특별시_일별 평균 대기오염도 정보_20210616		
분류체계	환경 - 대기	제공기관	서울특별시
관리부서명	빅데이터담당관	관리부서 전화번호	02-
보유근거		수집방법	
업데이트 주기	수시 (3월 1회 업데이트)	차기 등록 예정일	
제공유형	텍스트	현재 행	1
확정자	CSV	다운로드(바로그가)	33
데이터 형식		키워드	미지
등록	2021-06-16	수정	2021-
제공형태	기관자원에서 다운로드(제공데이터(URL)기재)		
URL	http://data.seoul.go.kr/data-as/CSV/2719/5710/09aseview.do		
설명	대기 환경지수, 미세먼지, 오존, 이산화질소, 일산화탄소, 이황산가스 등의 평균 대기오염도 일별 정보		
기타 유의사항			
비공표유형	무로	비공표유형 및 단위	건
이용허락범위	공공저작물_출격유치		

환경

서울시 일별 평균 대기오염도 정보

대기 환경지수, 미세먼지, 오존, 이산화질소, 일산화탄소, 이황산가스 등의 평균 대기오염도 일별 정보를 제공합니다.
*오존넷 서비스는 최근 1년 이내의 데이터만 출력합니다.

4

파일내려받기

NO	형식	파일명	용량(MB)	수집일	내려받기
1	데이터	일별평균대기오염도_2020.csv	0.79	2021.05.03	다운로드
2	데이터	일별평균대기오염도_2019.csv	1.01	2020.10.30	다운로드
3	데이터	일별평균대기오염도_2018.xlsx	0.96	2019.01.23	다운로드
4	데이터	일별평균대기오염도_2017.xlsx	0.65	2019.01.23	다운로드
5	데이터	일별평균대기오염도_2016.xlsx	0.65	2019.01.23	다운로드

3
메타데이터 다운로드

파일명	서울특별시_시간별 (초)미세먼지		
분류체계	환경 - 대기	제공기관	서울특별시
관리부서명	빅데이터담당관	관리부서 전화번호	02-2133-4280

서울특별시_시간별 (초)미세먼지

서울특별시 대기질 자료(초미세먼지, 미세먼지)입니다.
2006년 1월부터 2021년 5월까지의 자료로 차차구별 시간 평균 자료(서울시 평균 자료 포함)입니다.
본 측정자료는 측정 전 실시간 자료입니다. 참고용으로만 활용 가능하며, 행정 목적으로는 사용할 수 없습니다.

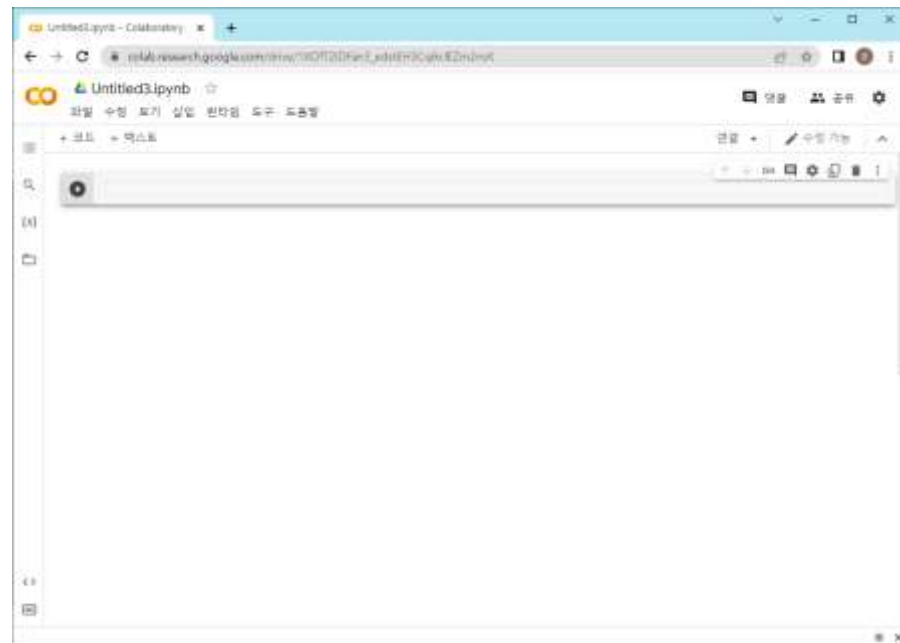
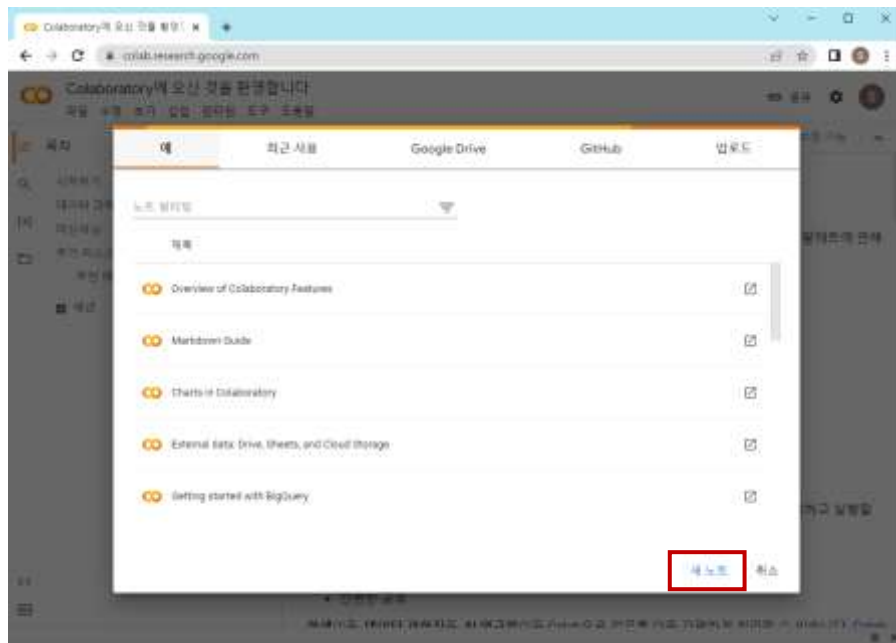
자료 문의 : 대기정책과 02-2133-3655

[다운로드](#)

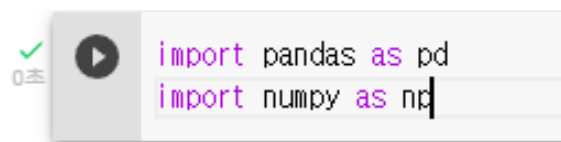
오류신고 및
담당자 문의

6

- 구글에서 코랩을 검색하거나 <https://colab.research.google.com>에 직접 접속합니다.
- 새 노트를 클릭합니다.



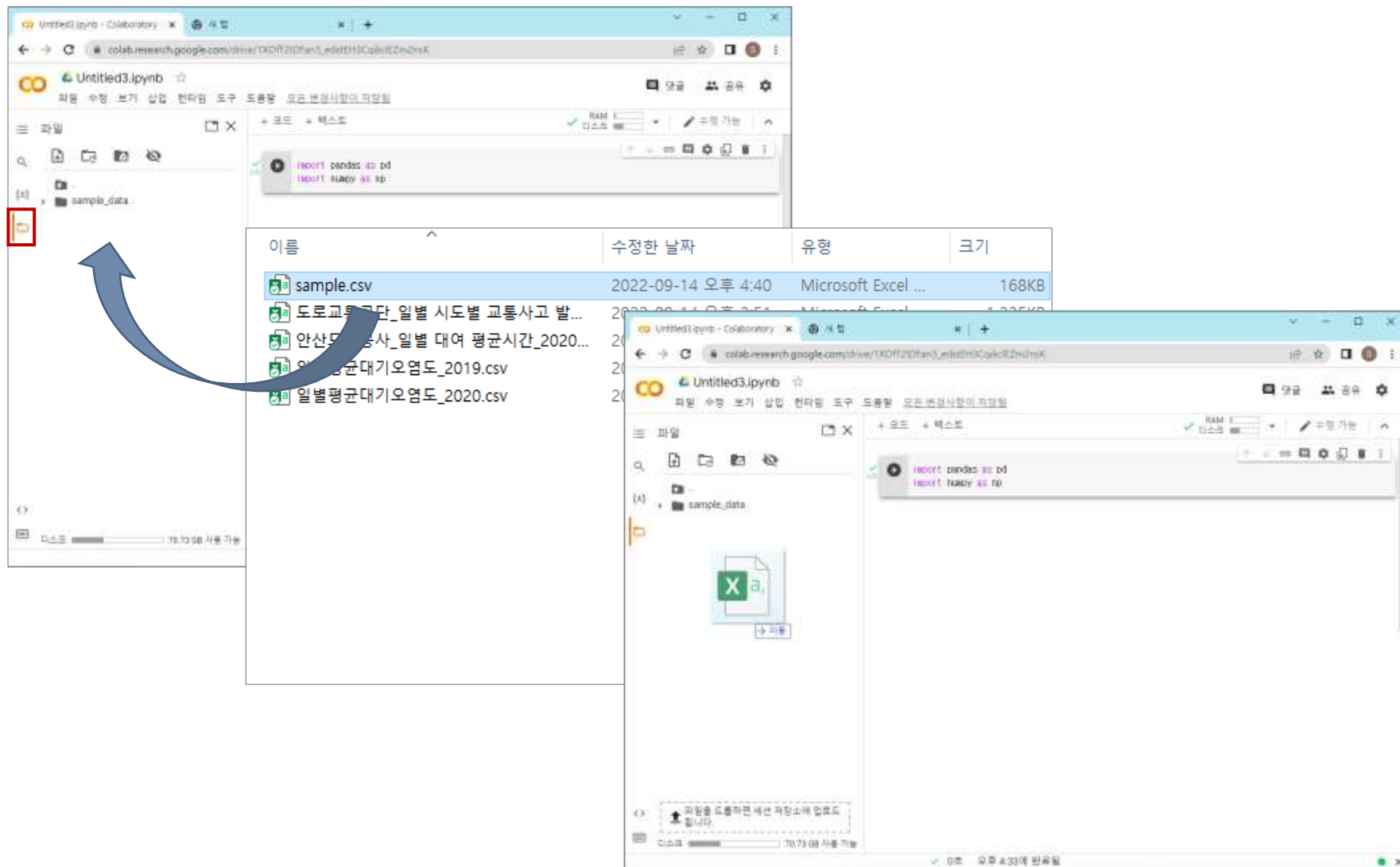
- 각 셀들을 실행시키고 싶은 경우 셀을 클릭 후, Ctrl+Enter 혹은 Shift+Enter를 누릅니다.
- 각 셀들이 실행이 완료되는 경우 녹색 체크표시가 나옵니다.



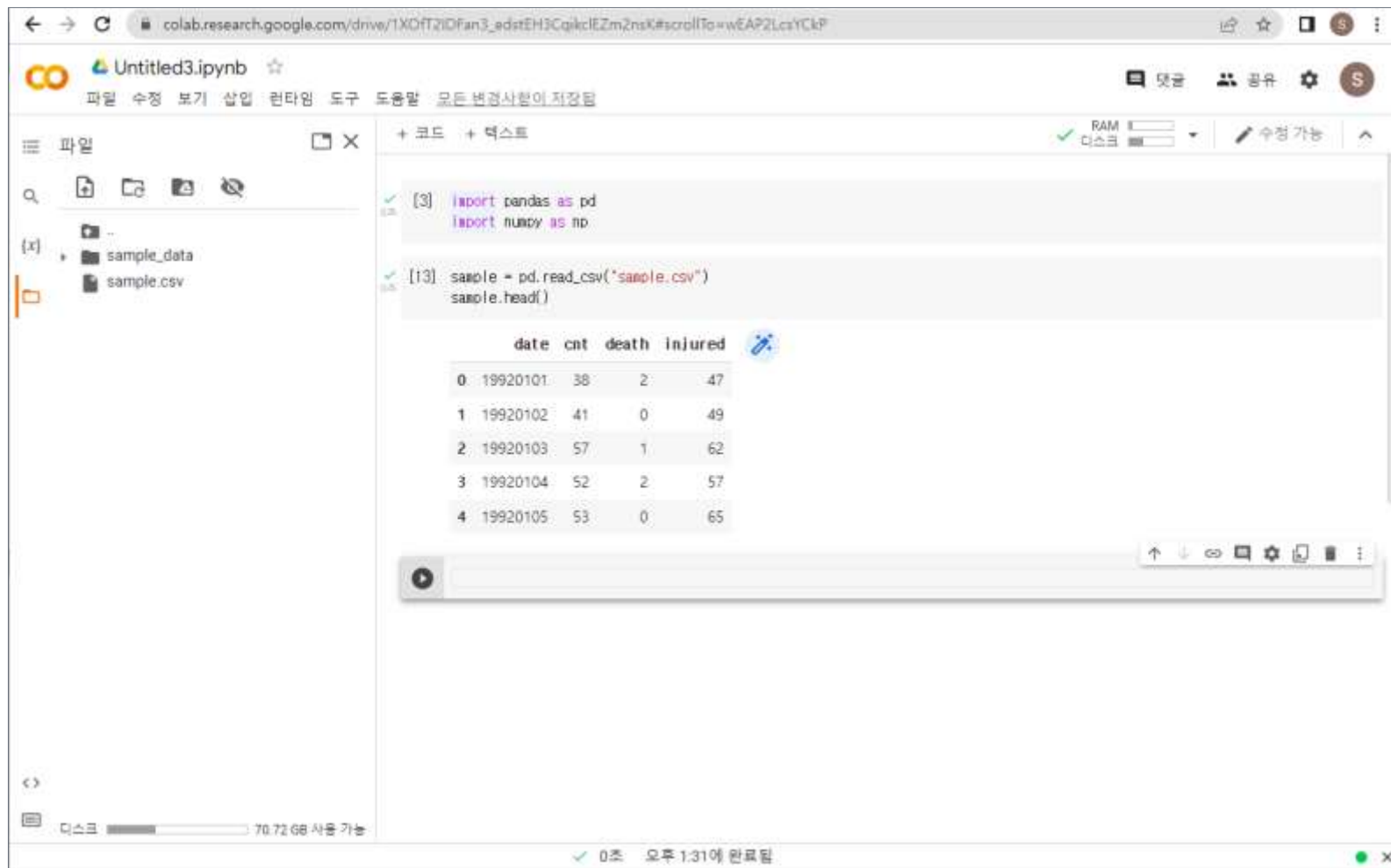
2.2 공공 데이터 포털

02. 데이터 수집

- 왼쪽 사이드바에 폴더를 클릭합니다.
- 원하는 데이터를 드래그해서 옮겨줍니다.



- 기본적으로 데이터를 핸들링하는데 주로 사용하게될 패키지 두개를 가져옵니다.
- * import pandas as pd , import numpy as np



The screenshot shows a Google Colab notebook titled 'Untitled3.ipynb'. The left sidebar displays a file explorer with a folder named 'sample_data' containing a file 'sample.csv'. The main code area shows two code cells. The first cell imports pandas as 'pd' and numpy as 'np'. The second cell reads 'sample.csv' into a DataFrame named 'sample' and displays its head. The output of the second cell is a table with 5 rows and 4 columns: 'date', 'cnt', 'death', and 'injured'.

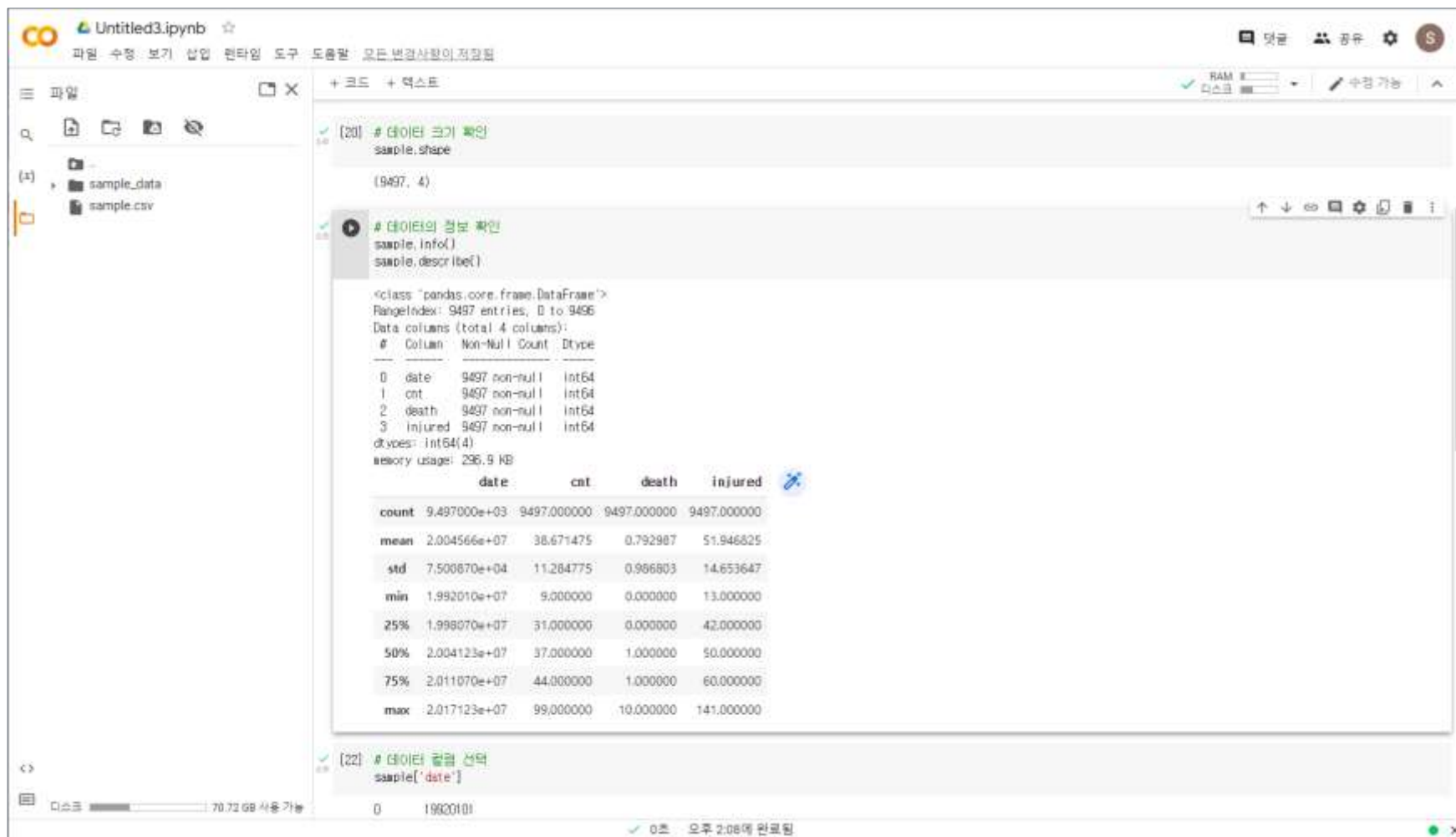
```
[3]: import pandas as pd
import numpy as np

[13]: sample = pd.read_csv('sample.csv')
sample.head()
```

	date	cnt	death	injured
0	19920101	38	2	47
1	19920102	41	0	49
2	19920103	57	1	62
3	19920104	52	2	57
4	19920105	53	0	65

At the bottom of the notebook, a status bar indicates '0초 오후 1:31에 완료됨' (Completed at 1:31 PM in 0 seconds).

- 데이터의 기본 정보를 알기 위해 데이터 크기를 확인하고, null값이 있는지 확인합니다.
- 데이터의 요약통계량에 해당하는 값들도 함께 확인합니다.



The screenshot shows a Jupyter Notebook interface with the following content:

File Explorer (Left): Untitled3.ipynb, sample_data, sample.csv

Code Cell 1:

```
[20] # 데이터 크기 확인
sample.shape
```

Output:

```
(9497, 4)
```

Code Cell 2:

```
# 데이터의 정보 확인
sample.info()
sample.describe()
```

Output:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9497 entries, 0 to 9496
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype  
---  -
0    date      9497 non-null   int64  
1    cnt       9497 non-null   int64  
2    death     9497 non-null   int64  
3    injured   9497 non-null   int64  
dtypes: int64(4)
memory usage: 296.9 KB
```

	date	cnt	death	injured
count	9.497000e+03	9497.000000	9497.000000	9497.000000
mean	2.004566e+07	38.671475	0.792987	51.946825
std	7.500670e+04	11.284775	0.986803	14.653647
min	1.992010e+07	9.000000	0.000000	13.000000
25%	1.998070e+07	31.000000	0.000000	42.000000
50%	2.004123e+07	37.000000	1.000000	50.000000
75%	2.011070e+07	44.000000	1.000000	60.000000
max	2.017123e+07	99.000000	10.000000	141.000000

Code Cell 3:

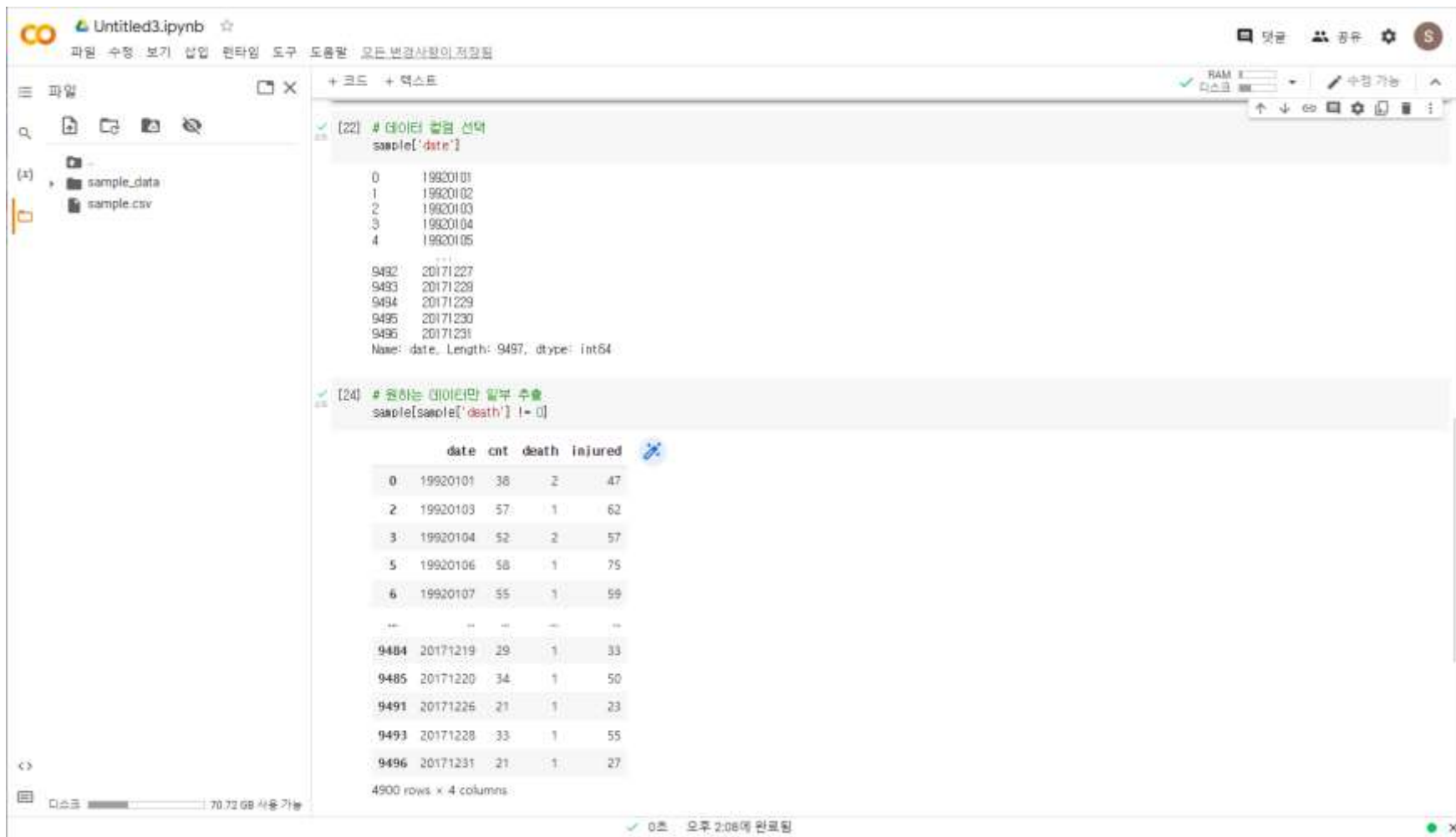
```
[22] # 데이터 컬럼 선택
sample['date']
```

Output:

```
0    19020101
```

Status Bar: 0초 오후 2:08에 완료됨

- 원하는 컬럼만을 선택해서 가져오기 위해 ['컬럼명']을 활용합니다.
- 원하는 데이터만 일부 추출하기 위해 데이터의 조건을 설정해 줍니다.



The screenshot shows a Jupyter Notebook interface with two code cells. The first cell, labeled [22], contains the code `sample['date']` and displays a preview of the 'date' column with 10 rows of dates from 19920101 to 20171231. The second cell, labeled [24], contains the code `sample[sample['death'] != 0]` and displays a preview of the filtered data with 4 columns: 'date', 'cnt', 'death', and 'injured'. The preview shows rows where 'death' is not zero.

```
[22] # 데이터 컬럼 선택
sample['date']

0    19920101
1    19920102
2    19920103
3    19920104
4    19920105
...
9492   20171227
9493   20171228
9494   20171229
9495   20171230
9496   20171231
Name: date, Length: 9497, dtype: int64
```

```
[24] # 원하는 데이터만 일부 추출
sample[sample['death'] != 0]
```

	date	cnt	death	injured
0	19920101	38	2	47
2	19920103	57	1	62
3	19920104	52	2	57
5	19920106	58	1	75
6	19920107	55	1	59
...
9484	20171219	29	1	33
9485	20171220	34	1	50
9491	20171226	21	1	23
9493	20171228	33	1	55
9496	20171231	21	1	27

4900 rows x 4 columns

- 분석 데이터에는 분석 알고리즘에는 직접적으로는 사용되진 않지만 여러 데이터를 연결하기 위한 “key” 컬럼이 존재합니다.
- 또한 직접적으로 알고리즘에 사용하기 위한 target 변수와 설명 변수가 존재합니다.

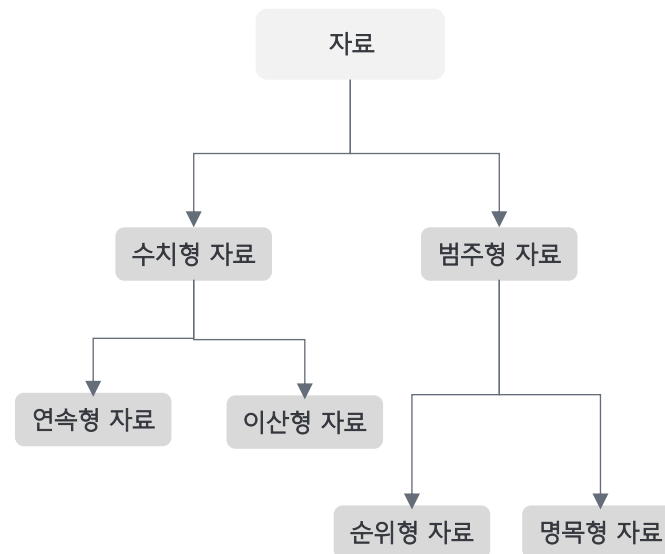
분석 데이터 구조

■ 분석 데이터 구조

- 데이터의 구조는 한 단위(ex. 사람)의 정보들을 포함하는 row값들과 동일한 정보를 가지고 있는 column값들로 구분됨
- 데이터 내에는 기본적으로 “학번”과 같이 정보들을 이어주는 “key” 컬럼과, 실제 분석가가 예측을 하고자 하는 타겟 컬럼, 그리고 타겟을 예측하기 위한 컬럼들로 구성 되어있음
- 보통 컬럼의 단위를 분석에서는 “변수”라고 명칭 하며, 타겟을 예측하기 위한 컬럼들을 설명 변수, 혹은 feature라고 명칭함

Key	Target	등급	성별	과제	결석	과거성적	...
학번	성적						
20312	80	A	여	0	0	85	...
20421	95	A+	남	0	1	90	...
21125	72	B+	남	1	0	80	...
21217	78	A	여	1	1	75	...

자료의 형태



- 데이터를 직접 다운로드 할 수 있겠지만, 데이터가 따로 업로드 되어있지 않은 경우 웹 사이트에 있는 원하는 데이터를 추출하는 방법을 스크래핑 혹은 크롤링 이라고 합니다.
- “BeautifulSoup”은 파이썬에서 스크래핑을 하기위해 사용하는 대표적인 라이브러리 중 하나입니다.

스크래핑

■ 웹 스크래핑이란?

- 웹 스크래핑(Web Scraping)은 인터넷에 있는 웹사이트에서 원하는 정보를 자동으로 추출하는 기술을 의미합니다.
- 웹사이트는 보통 HTML 구조로 이루어져 있기때문에 웹 스크래핑을 하기 위해서는 HTML구조에 대해 이해해야 합니다.

■ 웹 스크래핑이란?

```
<html>
<head>
  <title>페이지 제목</title>
</head>
<body>
  <h1>안녕하세요</h1>
  <p>내용1</p>
  <a href="https://www.example.com">예시링크</a>
</body>
</html>
```

BeautifulSoup 사용 예시

```
import requests
from bs4 import BeautifulSoup

# 1. 웹페이지 요청
url = "https://news.naver.com/" # 네이버 뉴스 메인 페이지
response = requests.get(url)

# 2. HTML 파싱
soup = BeautifulSoup(response.text, features="html.parser")

# 3. 원하는 데이터 추출 (기사 제목)
# strong 태그 중 class="cnf_news_title"만 추출
titles = soup.find_all("strong", class_="cnf_news_title")

print("== 네이버 뉴스 기사 제목 ==")
for t in titles[:10]:
    print(t.get_text(strip=True))

== 네이버 뉴스 기사 제목 ==
Former Interior Minister Lee Sang-min indicted over alleged role in Yoon's
열차 사고 CCTV 살펴보니... 안전장치 작동했나? 집중 조사
경부 "중대재해 발생 기업 공공 공사 참여 제한"
목 김영철, 이 대통령 실명 비난..."외교 상대도 아냐"
조국 "사과한다고 2030 마음 열겠다" 발언에...박용진 "부적절"
[속보] 실종됐던 '이태원 참사' 출동 소방관 숨진 채 발견
백종원, 점주들에 300억 풀더니 결국...개미를 '비명' [종목+]
국민의힘 " '뇌물' 김용 풀려나, 다음은 정진상·이화영인가?...면죄부 공화국"
"美정부, 삼성전자 지분 확보 검토" ...오히려 기업엔 기회 될수도?
윤석열 정부 합참, '북한 도발 시 전면전' 계획 세웠다
```