pandas report

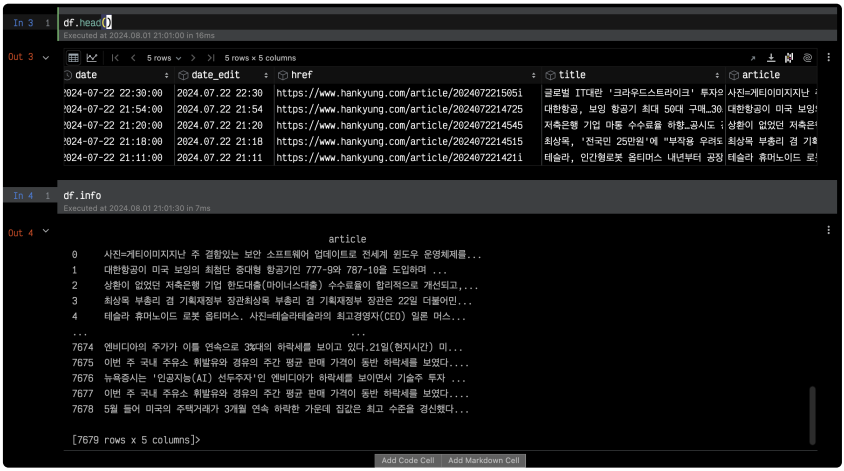
공식문서 내용: copy on write

```
In [1]: df = pd.DataFrame({"foo": [1, 2, 3], "bar": [4, 5, 6]})

In [2]: subset = df["foo"]

In [3]: subset.iloc[0] = 100

In [4]: df
Out[4]:
   foo  bar
0  100    4
1    2    5
2    3    6
```

과거에는 DataFrame에서 추출된 view를 수정하면 원본 DataFrame의 값도 변하게 동작이 되었습니다. 이는 딱 봐도 찾기 어려운 오류를 유발하는 형태였습니다. pandas 3.0부터 기본적으로 동시에 2개 이상의 pandas 객체를 수정하지 못도록 copy on write 원칙을 도입했습니다.

```
In [5]: pd.options.mode.copy_on_write = True

In [6]: df = pd.DataFrame({"foo": [1, 2, 3], "bar": [4, 5, 6]})

In [7]: subset = df["foo"]

In [8]: subset.iloc[0] = 100

In [9]: df
Out[9]:
   foo  bar
0    1    4
1    2    5
2    3    6
```

크롤링 json pandas로 불러오기



데이터 저장 포멧

- pickle
  - 파이썬 객체를 직렬화, 역직렬화 즉 객체와 파일형태로 변환에서 쓰입니다.
    - python에서 객체를 로드하거나 학습된 모델을 저장할때 많이 사용됩니다.
- CSV, TSV
  - 텍스트 파일 형식으로, 데이터를 csv는 쉼표 tsv는 탭으로 구분하여 저장합니다. 간단한 형식으로, 각 줄이 한 레코드를 나타내며, 각 필드는 쉼표로 구분됩니다.
  - 데이터분석에서 널리활용됩니다.
- JSON
  - key-value형태로 데이터를 저장합니다. 가독성이 xml같은 것에 비해 좋습니다
  - 웹api, db저장 포멧으로 많이 쓰입니다.
- HTML
  - 웹페이지의 마크업 언어입니다. 웹페이지의 구조와 내용을 정의합니다.
  - 웹개발에서 필수적입니다
- XML
  - 가독성이 많이 떨어집니다. 태그를 사용하여 데이터를 계층적으로 표현합니다.
  - 설정파일, 웹api 에서 포멧으로 많이 활용됩니다.
- Parquet

    - 하둡에서 사용되는 방식이다. 열기반으로 저장하는 포멧이다.열에는 유사한 데이터들이 있기 때문에 압축에 유리하다!
    - 빅데이터 처리(하둡)에서 많이 쓰이는 포멧이
- YAML
    - 인간의 가독성에 신경쓴 포멧입니다. 들여쓰기(2칸)으로 계층적으로 데이터를 표현합니다.
    - 설정파일, 문서화의 포멧으로 자주 활용됩니다.
- TOML
    - 구문이 간단해서 YAML과 비슷하게 가독성이 좋지만 유연성은 부족합니다.
    - 설정파일의 포맷으로 자주 활용됩니다.