

DataFrame 내보내기

DataFrame을 Excel이나 CSV 형식으로 저장하기

VSCode에서의 결과 확인을 위해 Excel Viewer Extension을 설치하는 것이 좋다

#01. 패키지 참조

```
from pandas import read_excel
```

#02. 데이터 가져오기

```
df = read_excel("https://data.hossam.kr/grade.xlsx", index_col='이름')
df
```

#03. 데이터 내보내기

1) 엑셀 파일로 내보내기

원본 파일 그대로 저장하기

```
df.to_excel("1.xlsx")
```

파라미터 설정

인덱스를 포함하는 경우

```
df.to_excel("2.xlsx",          # 파일이름
            index=True,        # 인덱스 포함 여부 (생략시 True)
            index_label="이름"  # 인덱스 필드의 제목 (생략시 빈칸으로 설정됨)
            )
```

인덱스를 포함하지 않는 경우

```
df.to_excel("3.xlsx",          # 파일이름
            index=False,        # 인덱스 포함 여부(기본값=True)
            columns=['학년', '성별', '국어'], # 저장할 컬럼 지정
            header=['level', 'sex', 'kor']    # 컬럼별 제목 설정
            )
```

2) CSV로 내보내기

encoding을 설정하는 것 외에는 엑셀파일을 내보내는 것과 동일하다.

원본 그대로 저장하기

```
df.to_csv("1.csv", encoding='utf-8')
```

파라미터 설정

인덱스를 포함하는 경우

```
df.to_csv("2.csv",          # 파일이름
          encoding="utf-8",  # 파일 인코딩
          index=True,        # 인덱스 포함 여부 (생략시 True)
          index_label="이름" # 인덱스 필드의 제목 (생략시 빈칸으로 설정됨)
        )
```

인덱스를 포함하지 않는 경우

```
df.to_csv("3.csv",          # 파일이름
          encoding="utf-8",  # 인코딩
          index=False,       # 인덱스 포함 여부(기본값=True)
          columns=['학년', '성별', '국어'], # 저장할 컬럼 지정
          header=['level', 'sex', 'kor']    # 컬럼별 제목 설정
        )
```

실습 결과물 모두 삭제

```
import os
import glob as gl

list1 = gl.glob("*.xlsx")
list2 = gl.glob("*.csv")
list3 = list1 + list2

for items in list3:
    os.remove(items)
    print("%s가 삭제됨" % items)
```