

## [ 5-2. Excel 파일 읽고 쓰기 ]

### 1. DataFrame을 Excel 파일로 쓰기

Pandas의 DataFrame의 'to\_excel' 메서드를 사용하면 DataFrame을 바로 Excel 파일로 변환할 수 있습니다.

```
In [ ]: # 예시 데이터프레임 작성
import pandas as pd
from pandas import DataFrame

data = {
    'Name': ['John', 'Anna', 'Peter', 'Linda'],
    'Age': [28, 34, 29, 32],
    'City': ['New York', 'Paris', 'Berlin', 'London']
}

df = DataFrame(data)
print(df)
```

	Name	Age	City
0	John	28	New York
1	Anna	34	Paris
2	Peter	29	Berlin
3	Linda	32	London

```
In [ ]: # csv 파일로 저장하기
df.to_excel('excel_data2.xlsx', index=False, engine='openpyxl')
```

- 위 코드는 'excel\_data2.xlsx' 라는 이름으로 DataFrame의 내용을 엑셀 파일로 저장합니다.
- index=False 옵션은 DataFrame의 인덱스를 엑셀 파일에 포함하지 않도록 합니다.
- engine=openpyxl 옵션은 엑셀 파일을 생성하기 위해 openpyxl 엔진을 사용하겠다는 것을 명시합니다.

### 2. Excel 파일을 DataFrame으로 읽어들이기

pd.read\_excel 메서드를 사용하여 저장된 excel 파일을 다시 불러와 DataFrame으로 변환할 수 있습니다.

```
In [ ]: # Excel 파일을 DataFrame으로 불러오기
loaded_df = pd.read_excel('excel_data2.xlsx')

print(loaded_df)
```

	Name	Age	City
0	John	28	New York
1	Anna	34	Paris
2	Peter	29	Berlin
3	Linda	32	London

### 3. openpyxl 라이브러리를 이용하여 excel 파일을 읽고, DataFrame으로 설정하기

- 엑셀 파일을 읽을 때는 openpyxl의 load\_workbook을 이용하여 엑셀 워크북을 읽어와 wb 객체를 생성합니다.
- ws(work sheet) 객체는 wb.active 또는 wb['Sheet1'] 과 같이 특정 시트를 지정하는 방식으로 생성할 수 있습니다.
- ws객체의 iter\_rows 메서드를 이용하여 한 행씩 읽어들이면서 첫번째 행은 columns 변수에, 나머지는 딕셔너리 형태로 변환하여 data 변수에 저장합니다.
- 마지막으로 data 객체를 DataFrame으로 변환합니다.

```
In [ ]: from openpyxl import load_workbook
import pandas as pd

# 엑셀 파일 로드
wb = load_workbook('excel_data2.xlsx')
ws = wb.active

# 첫번째 행(컬럼 이름) 가져오기
columns = [
    cell.value
    for cell
    in next(ws.iter_rows(min_row=1, max_row=1))
]

# 데이터를 딕셔너리 리스트로 변환
data = []
for row in ws.iter_rows(min_row=2):
    record = {columns[i]: cell.value for i, cell in enumerate(row)}
    data.append(record)

# 딕셔너리 리스트를 DataFrame으로 변환
df = pd.DataFrame(data)

print(df)
```

	Name	Age	City
0	John	28	New York
1	Anna	34	Paris
2	Peter	29	Berlin
3	Linda	32	London

## 4. openpyxl 라이브러리를 이용하여 DataFrame을 Excel 파일로 쓰기

```
In [ ]: from openpyxl import Workbook

# 새로운 Workbook 생성
wb = Workbook()
ws = wb.active

# DataFrame의 컬럼명을 엑셀 파일의 첫번째 행으로 추가
ws.append(list(df.columns))

# DataFrame의 각 행을 엑셀 파일에 추가
for index, row in df.iterrows():
    ws.append(row.tolist())

# 엑셀 파일 저장
wb.save('excel_data3.xlsx')
```

```
In [ ]: # 파일 출력이 제대로 되었는지 확인
pd.read_excel('excel_data3.xlsx')
```

```
Out [ ]:
```

	Name	Age	City
0	John	28	New York
1	Anna	34	Paris
2	Peter	29	Berlin
3	Linda	32	London

```
In [ ]:
```