

데이터 입출력_p57

- pd.read_csv()
- pd.read_json()
- pd.read_excel()
- pd.read_html()
- pd.to_csv()

CSV 파일 읽기

데이터 값을 쉼표(,)로 구분하고 있다는 의미로 CSV(comma-separated values)라고 부르는 텍스트 파일

```
# ./sample.csv
# 현재 코드를 작성 중인 폴더 안에 sample.csv 파일이 있다는 말이다.
# 아래의 4줄이 포함되어 있다.
c0,c1,c2,c3
0,1,4,7
1,2,5,8
2,3,6,9
```

CSV 파일 읽기 실습_p58

```
import pandas as pd
file_path= './sample.csv'
df1 = pd.read_csv(file_path)
df2 = pd.read_csv(file_path, header=None)
df3 = pd.read_csv(file_path, index_col='c0')
# df6 = pd.read_csv(file_path, delimiter=',')
# df6 = pd.read_csv(file_path, delimiter='\t')

# df1
c0 c1 c2 c3
0 0 1 4 7
1 1 2 5 8
2 2 3 6 9

# df2
0 1 2 3
0 c0 c1 c2 c3
1 0 1 4 7
2 1 2 5 8
3 2 3 6 9

# df1
c1 c2 c3
c0
0 1 4 7
1 2 5 8
2 3 6 9
```

CSV 파일 만들기

to_csv() 메소드를 활용하여 저장

```
data = {'name' : [ 'Jerry', 'Riah', 'Paul'],
        'algol' : [ "A", "A+", "B"],
        'basic' : [ "C", "B", "B+"],
        'c++' : [ "B+", "C", "C+"],
       }
df = pd.DataFrame(data)
df2 = pd.DataFrame(data)
df2.set_index('name', inplace=True)
print(df)
print(df2)
```

```
# output
    name algol basic c++
0   Jerry      A      C  B+
1   Riah      A+     B    C
2   Paul      B      B+  C+
```

```
      algol basic c++
name
Jerry      A      C  B+
Riah      A+     B    C
Paul      B      B+  C+
```

CSV 파일 만들기

to_csv() 메소드를 활용하여 저장

```
data = {'name' : [ 'Jerry', 'Riah', 'Paul'],
        'algol' : [ "A", "A+", "B"],
        'basic' : [ "C", "B", "B+"],
        'c++' : [ "B+", "C", "C+"],
       }
df = pd.DataFrame(data)
df2 = pd.DataFrame(data)
df2.set_index('name', inplace=True)
print(df)
print(df2)
```

```
# output
    name algol basic c++
0   Jerry      A      C  B+
1   Riah      A+     B    C
2   Paul      B      B+  C+
```

```
    algol basic c++
name
Jerry      A      C  B+
Riah      A+     B    C
Paul      B      B+  C+
```

CSV 파일 변환

이전의 데이터프레임 자체를 CSV 파일로 변환했다.

```
df.to_csv("./df_sample.csv")  
'''
```

```
# 결과값  
,name,algol,basic,c++  
0,Jerry,A,C,B+  
1,Riah,A+,B,C  
2,Paul,B,B+,C+  
'''
```

```
df2.to_csv("./df_sample2.csv")  
'''
```

```
# 결과값  
name,algol,basic,c++  
Jerry,A,C,B+  
Riah,A+,B,C  
Paul,B,B+,C+  
'''
```

CSV 파일 다시 불러오기

| 저장한 csv파일을 다시 출력

```
result1 = pd.read_csv('./df_sample.csv')
result2 = pd.read_csv('./df_sample2.csv')
print(result1)
print(result2)

# output
    Unnamed: 0      name  algol  basic  c++
0            0     Jerry      A      C    B+
1            1     Riah     A+      B      C
2            2     Paul      B     B+    C+


      name  algol  basic  c++
0  Jerry      A      C    B+
1  Riah     A+      B      C
2  Paul      B     B+    C+
```

result1을 보면 unnamed값이 포함되어 있는 것을 확인 할 수 있다. 따라서 재활용성을 생각해서 필요에 맞게 CSV 파일을 만들어 놓는 것이 중요하다고 할 수 있다.