Pandas를 이용한 데이터 준비

Soongsil University

숭실대학교
베어드교양대학
강의선 교수
iami86@ssu.ac.kr

Pandas란?

- 데이터분석을 위한 Python 라이브러리
  - Panel Data Analysis
- 대용량의 데이터 처리를 지원함.
  - 자동화된 분석을 지원
  - 대용량의 데이터 처리 지원
  - 머신러닝, 시각화 등의 데이터 사이언스 관련 라이브러리에서 사용

[pandas.pydata.org]

판다스 vs 엑셀

- 자동화
  - 파이썬 : 코딩을 통한 자동화
  - 엑셀 : 기본적으로 사람의 손으로 작업 (VB로 자동화 가능하기는 함)
- 대용량 데이터 처리
  - 엑셀은 큰 데이터 처리에 부적합함 (로딩도 안됨)
  - 데이터 처리 속도가 느림
- 분석 방법
  - 엑셀은 일반적으로 지원되는 기능에 한정하여 작업
  - 사용자가 코딩을 통해 다양한 창의적인 데이터 분석이 가능함

설치방법 및 라이브러리 선언

- 콘솔에서 아래의 명령으로 설치
  - Path 설정에 문제가 있는 경우 아래 명령이 실행 안될 수 있음
  - 그럴 경우 pip 명령이 있는 위치로 경로를 이동한 후 실행

```
pip install pandas
```

- 아나콘다 설치
  - 자동으로 설치됨
  - 주피터 노트북에서 사용할 수 있음
- 라이브러리 선언

```
import pandas as pd
```

데이터 입력하기 : DataFrame 생성(데이터 생성)

- DataFrame : 행과 열로 구성된 일종의 스프레드시트

```
import pandas as pd

no    = [20211021, 20205412, 20210578]
name  = ['박형식','공유','아이유']
major = ['영어영문학과', '화학과', '수학과']

df = pd.DataFrame({'학번':no, '이름':name, '학과':major})
df
```

	학번	이름	학과
0	20211021	박형식	영어영문학과
1	20205412	공유	화학과
2	20210578	아이유	수학과

	A	B	C
1	학번	이름	학과
2	20211021	박형식	영어영문학과
3	20205412	공유	화학과
4	20210578	아이유	수학과

Soongsil University

데이터 입력하기 : DataFrame 생성(데이터 생성)

```
import pandas as pd

Al_class = [[20211021,'박형식','영어영문학과'],
            [20205412,'공유','화학과'],
            [20210578,'아이유','수학과']
           ]

df = pd.DataFrame(Al_class, columns=['학번', '이름','학과'])
df
```

```
import pandas as pd

df = pd.DataFrame([[20211021,'박형식','영어영문학과'],
                   [20205412,'공유','화학과'],
                   [20210578,'아이유','수학과']
                  ],columns=['학번', '이름','학과'])
df
```

	A	B	C
1	학번	이름	학과
2	20211021	박형식	영어영문학과
3	20205412	공유	화학과
4	20210578	아이유	수학과

	학번	이름	학과
0	20211021	박형식	영어영문학과
1	20205412	공유	화학과
2	20210578	아이유	수학과

Soongsil University

txt, csv 파일 불러오기

Soongsil University

콤마(,)로 구분된 파일

- CSV(Comma Separated Values)
- 아래 내용을 파일로 저장하기 (c:/data/exam1.txt)

```
name, score, absent
kim, 95, 3
choi, 100, 0
lee, 90, 2
park, 85, 1
cho, 77, 5
```

```
import pandas as pd

df = pd.read_csv('c:/data/exam1.txt')
print( df )
```

```
   name   score   absent
0   kim      95        3
1  choi     100        0
2   lee      90        2
3  park      85        1
4   cho      77        5
```


Soongsil University

탭(tab)으로 구분된 파일

- 아래 내용을 파일로 저장하기 (c:/data/exam2.txt)

name	score	absent
kim	95	3
choi	100	0
lee	90	2
park	85	1
cho	77	5

```
1  import pandas as pd
2
3  df = pd.read_csv('c:/data/exam2.txt', delimiter='₩t')
4  print( df )
5
```

```
   name  score  absent
0   kim     95       3
1  choi    100       0
2   lee     90       2
3  park     85       1
4   cho     77       5
```


Soongsil University

쉼표(,)로 구분되고 헤더가 없는 파일

- 첫 줄에 헤더(컬럼 명)이 없는 경우

kim, 95, 3
choi, 100, 0
lee, 90, 2
park, 85, 1
cho, 77, 5

```
1  import pandas as pd
2
3  df = pd.read_csv('c:/data/exam.txt', delimiter=',', header=None)
4  print( df )
```

```
      0    1  2
0   kim   95  3
1  choi  100  0
2   lee   90  2
3  park   85  1
4   cho   77  5
```


Soongsil University

Excel 파일 불러오기

Soongsil University

엑셀파일 준비하기

- 연도별 출생인구 엑셀 파일 준비

	A	B	C
1	연도	출생아수	천명당 출생률
2	1951	728,175	37.7
3	1952	775,630	39.6
4	1953	830,330	41.6
5	1954	892,236	43.4
6	1955	961,055	45.4
7	1956	999,005	45.2
8	1957	1,016,573	44.8
9	1958	1,046,011	44.5
10	1959	1,074,876	44.2
11	1960	1,099,294	44
12	1961	1,099,164	42.7
13	1962	1,089,951	41.1

- C:₩Data 폴더 만들기
- C:₩Data 폴더에 연도별출생인구.xlsx 저장하기
- 그림과 같이 3개의 컬럼의 데이터로 구성된 엑셀 파일을 준비합니다.

Soongsil University

Excel 데이터 가져와서 출력하기

```
1 import pandas as pd
2
3 birthData = pd.read_excel("c:/data/연도별출생인구.xlsx")
4 birthData
```

	연도	출생아수	천명당출생률
0	1951	728175	37.7
1	1952	775630	39.6
2	1953	830330	41.6
3	1954	892236	43.4
4	1955	961055	45.4
...	...	...	...
63	2014	435435	8.6
64	2015	438420	8.6
65	2016	406243	7.9
66	2017	357771	7.0
67	2018	326900	6.4

68 rows × 3 columns

read_excel 함수로 간단하게 로딩

print(birthData) 혹은 birthData 로 출력
(많은 경우 일부분만 표시)

경로구분은 / 혹은 ₩₩ 를 사용함.
(₩는 오류발생할 수 있음)

Soongsil University

info 함수로 데이터 파악하기

```
birthData.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 68 entries, 0 to 67
Data columns (total 3 columns):
연도          68 non-null int64
출생아수        68 non-null int64
천명당 출생률    68 non-null float64
dtypes: float64(1), int64(2)
memory usage: 1.7 KB
```

인덱스는 0부터 67까지
즉, 레코드 개수는 68개임.

3개의 칼럼(열)로 구성되어 있음.

연도는 정수 데이터
출생아수는 정수 데이터
천명당 출생률은 실수 데이터

Soongsil University

describe 함수로 데이터 파악하기

```
birthData.describe()
```

	연도	출생아수	천명당 출생률
count	68.00000	6.800000e+01	68.000000
mean	1984.50000	7.451443e+05	22.483824
std	19.77372	2.271235e+05	12.619395
min	1951.00000	3.269000e+05	6.400000
25%	1967.75000	5.394685e+05	11.250000
50%	1984.50000	7.246800e+05	16.700000
75%	2001.25000	9.621715e+05	33.225000
max	2018.00000	1.099294e+06	45.400000

Soongsil University

행과 열 일부 선택하기

Soongsil University

특정 레코드 선택하기

- 인덱스(index)를 활용한 슬라이싱(Slicing) 방식

```
birthData[0:3]
```

	연도	출생아수	천명당 출생률
0	1951	728175	37.7
1	1952	775630	39.6
2	1953	830330	41.6

```
birthData[50:55]
```

	연도	출생아수	천명당 출생률
50	2001	554895	11.6
51	2002	492111	10.2
52	2003	490543	10.2
53	2004	472761	9.8
54	2005	435031	8.9

```
birthData[:4]
```

	연도	출생아수	천명당 출생률
0	1951	728175	37.7
1	1952	775630	39.6
2	1953	830330	41.6
3	1954	892236	43.4

```
birthData[65:]
```

	연도	출생아수	천명당 출생률
65	2016	406243	7.9
66	2017	357771	7.0
67	2018	326900	6.4

리스트에서 특정 요소를 슬라이싱 하는 방식과 동일함

특정 칼럼(열) 선택하기

[방식1]

```
birthData.연도
```

```
0     1951
1     1952
2     1953
3     1954
4     1955
...
63    2014
64    2015
65    2016
66    2017
67    2018
```

```
birthData.출생아수
```

```
0     728175
1     775630
2     830330
3     892236
4     961055
...
63    435435
64    438420
65    406243
66    357771
67    326900
```

방식2 : 따옴표, 칼럼 이름에 띄어쓰기가 있는 경우

```
birthData['연도']
```

```
0     1951
1     1952
2     1953
3     1954
4     1955
...
63    2014
64    2015
65    2016
66    2017
67    2018
```

```
birthData['천명당 출생률']
```

```
0     37.7
1     39.6
2     41.6
3     43.4
4     45.4
...
63    8.6
64    8.6
65    7.9
66    7.0
67    6.4
```


행과 열 선택하기

```
birthData.연도[0:5]
```

```
0    1951
1    1952
2    1953
3    1954
4    1955
```

방식1

```
birthData['연도'][0:5]
```

```
0    1951
1    1952
2    1953
3    1954
4    1955
```

방식2

```
birthData.출생아수[10:15]
```

```
10    1099164
11    1089951
12    1075203
13    1057241
14    1040544
```

방식1

```
birthData['출생아수'][10:15]
```

```
10    1099164
11    1089951
12    1075203
13    1057241
14    1040544
```

방식2

여러 개의 칼럼(열) 선택하기

```
birthData[ ['연도', '출생아수'] ]
```

대괄호가 2개 사용

	연도	출생아수
0	1951	728175
1	1952	775630
2	1953	830330
3	1954	892236
4	1955	961055

응용 : 행과 열의 선택 / 새로운 DF 로 저장

```
df2 = birthData.출생아수[0:5]
```

```
df2
```

```
0    728175
1    775630
2    830330
3    892236
4    961055
Name: 출생아수, dtype: int64
```

```
df3 = birthData[['연도', '출생아수'] ][10:15]
```

```
df3
```

	연도	출생아수
10	1961	1099164
11	1962	1089951
12	1963	1075203
13	1964	1057241
14	1965	1040544

Soongsil University

조건에 맞는 데이터 선택하기

Soongsil University

query 질의 함수 활용하기

1990년부터 2000년까지의 데이터

```
birthData.query( '1990<=연도<=2000')
```

	연도	출생아수	천명당 출생률
39	1990	649738	15.2
40	1991	709275	16.4
41	1992	730678	16.7
42	1993	715826	16.0
43	1994	721185	16.0

2000년 이후 50만명 이상 출생한 연도 데이터

```
birthData.query( '연도>=2000 and 출생아수>=500000')
```

	연도	출생아수	천명당 출생률
49	2000	634501	13.3
50	2001	554895	11.6

Soongsil University

수고하셨습니다.

Soongsil University