Pandas 라이브러리 기본 알아보기

- 데이터 처리와 분석을 위한 파이썬 라이브러리
- R의 data.frame과 유사하게 설계한 DataFrame이라는 데이터 기반으로 만들어짐.
- 데이터 분석시에 속도도 빠르고 많은 기능을 가지고 있어, 머신러닝 데이터 분석을 수행시에 많이 사용됨.
- 데이터를 읽고, 쓰기가 비교적 용이함.

- 참조 url : https://pandas.pydata.org/

학습 내용

- 대표적인 데이터 셋 Iris의 기본 데이터 탐색
- 행열 확인, 데이터 앞뒤 확인, 데이터 요약값 확인 등
- 행열 선택

판다스의 2가지 자료형

- Series(시리즈) 자료형
- DataFrame(데이터 프레임) 자료형

01 판다스 불러오기

In [1]:
```
import pandas as pd
```

02 데이터 준비

In [2]:
```
import seaborn as sns
```

In [3]:
```
iris = sns.load_dataset("iris")
iris
```

Out[3]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

150 rows × 5 columns

In [4]:
```
### seaborn에서 불러온 데이터 셋은 판다스 데이터 프레임이 된다.
type(  iris  )
```

Out[4]:
```
pandas.core.frame.DataFrame
```

In [5]:
```
### 데이터 프레임의 데이터의 각 열은 시리즈이다.
type(  iris['species']  )
```

Out[5]:
```
pandas.core.series.Series
```

03. Pandas 기본

- 데이터 살펴보자.

In [6]:
```
### 행, 열 확인
iris.shape
```

Out[6]:
```
(150, 5)
```

In [7]:
```
### 컬럼명 확인
iris.columns
```

Out[7]:
```
Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',
       'species'],
      dtype='object')
```

In [8]:
```
### 앞의 데이터 살펴보기 - 기본값은 5, 숫자 지정시 해당 행이 보임
iris.head()
```

Out[8]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

In [9]:
```
### 뒤의 데이터 살펴보기 - 기본값은 5, 숫자 지정시 해당 행이 보임
iris.tail()
```

Out[9]:

	sepal_length	sepal_width	petal_length	petal_width	species
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

데이터 행수 개수를 지정해서 보기

In [10]:
```
print(  iris.head(7)  )
print()
print(  iris.tail(7)  )
```

```
   sepal_length  sepal_width  petal_length  petal_width species
0           5.1          3.5           1.4          0.2  setosa
1           4.9          3.0           1.4          0.2  setosa
2           4.7          3.2           1.3          0.2  setosa
3           4.6          3.1           1.5          0.2  setosa
4           5.0          3.6           1.4          0.2  setosa
5           5.4          3.9           1.7          0.4  setosa
6           4.6          3.4           1.4          0.3  setosa

     sepal_length  sepal_width  petal_length  petal_width    species
143           6.8          3.2           5.9          2.3  virginica
144           6.7          3.3           5.7          2.5  virginica
145           6.7          3.0           5.2          2.3  virginica
146           6.3          2.5           5.0          1.9  virginica
147           6.5          3.0           5.2          2.0  virginica
148           6.2          3.4           5.4          2.3  virginica
149           5.9          3.0           5.1          1.8  virginica
```

In [11]:
```
# 데이터 정보를 전체적으로 확인해 보기
iris.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   sepal_length  150 non-null    float64
 1   sepal_width   150 non-null    float64
 2   petal_length  150 non-null    float64
 3   petal_width   150 non-null    float64
 4   species       150 non-null    object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

In [12]:
```
# 데이터 수치형에 대해 알아보기
iris.describe()
```

Out[12]:

	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

In [13]:
```
# 데이터 범주형에 대한 정보 확인
iris.describe(include=['O'])
```

Out[13]:

	species
count	150
unique	3
top	setosa
freq	50

In [14]:
```
# 데이터에 비어 있는지 알아보기 (결측치 확인)
iris.isnull().sum()
```

Out[14]:
```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

행 선택

In [15]:
```
print( iris )

iris.iloc[ 0:10 , : ]  # 맨 앞에서 10행 선택
```

```
    sepal_length  sepal_width  petal_length  petal_width    species
0            5.1          3.5           1.4          0.2     setosa
1            4.9          3.0           1.4          0.2     setosa
2            4.7          3.2           1.3          0.2     setosa
3            4.6          3.1           1.5          0.2     setosa
4            5.0          3.6           1.4          0.2     setosa
..           ...          ...           ...          ...        ...
145          6.7          3.0           5.2          2.3  virginica
146          6.3          2.5           5.0          1.9  virginica
147          6.5          3.0           5.2          2.0  virginica
148          6.2          3.4           5.4          2.3  virginica
149          5.9          3.0           5.1          1.8  virginica

[150 rows x 5 columns]
```

Out[15]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa

열 선택

In [16]:
```
print( iris.columns )

iris.iloc[ : , 0:1 ] # 맨 첫번째 열 선택
```

```
Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',
       'species'],
      dtype='object')
```

Out[16]:

	sepal_length
0	5.1
1	4.9
2	4.7
3	4.6
4	5.0
...	...
145	6.7
146	6.3
147	6.5
148	6.2
149	5.9

150 rows × 1 columns

In [17]:
```
iris.loc[ 0:5 ,  : ] # 행 인덱스 0~5 선택
```

Out[17]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa

In [18]:
```
iris.loc[ : ,  'sepal_length':'petal_width' ] # 첫번째 컬럼부터 네번째 컬럼 선택
```

Out[18]:

	sepal_length	sepal_width	petal_length	petal_width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
...	...	...	...	...
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

150 rows × 4 columns

- last update : 24/06, @by DJ, Lim

In [ ]: