파이썬 실습

실습

Pandas 실습

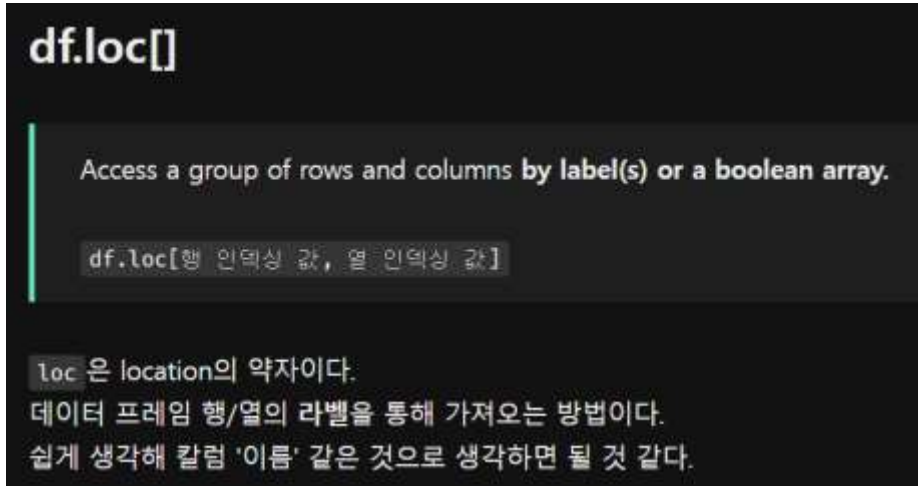
```
import pandas as pd

df = pd.read_csv("train.csv")
df
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

Pandas 실습




```
df.loc[0]
```

```
PassengerId                      1
Survived                         0
Pclass                           3
Name           Braund, Mr. Owen Harris
Sex                           male
Age                           22.0
SibSp                            1
Parch                            0
Ticket                   A/5 21171
Fare                          7.25
Cabin                          NaN
Embarked                         S
Name: 0, dtype: object
```

Pandas 실습

- (참고로 `df.loc[[0]]`으로 하면 이렇게 볼 수도 있다)

```
df.loc[[0]]
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.25	NaN	S

Pandas 실습

두 번째 row의 `Name` 이 알고 싶다면?

```
df.loc[1, 'Name']
```

```
'Cumings, Mrs. John Bradley (Florence Briggs Thayer)'
```

슬라이싱을 통해 여러 값 가져오기

```
df.loc[:, 'Name']
```

```
0                                    Braund, Mr. Owen Harris
1          Cumings, Mrs. John Bradley (Florence Briggs Th...
2                                    Heikkinen, Miss. Laina
3            Futrelle, Mrs. Jacques Heath (Lily May Peel)
4                                  Allen, Mr. William Henry
                                  ...
886                                  Montvila, Rev. Juozas
887                           Graham, Miss. Margaret Edith
888               Johnston, Miss. Catherine Helen "Carrie"
889                                  Behr, Mr. Karl Howell
890                                    Dooley, Mr. Patrick
Name: Name, Length: 891, dtype: object
```

Pandas 실습

```
df.loc[:, :'Name']
```

	PassengerId	Survived	Pclass	Name
0	1	0	3	Braund, Mr. Owen Harris
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...
2	3	1	3	Heikkinen, Miss. Laina
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)
4	5	0	3	Allen, Mr. William Henry
...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas
887	888	1	1	Graham, Miss. Margaret Edith
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"
889	890	1	1	Behr, Mr. Karl Howell
890	891	0	3	Dooley, Mr. Patrick

891 rows × 4 columns

Pandas 실습

```
df.loc[:4, :'Name']
```

	PassengerId	Survived	Pclass	Name
0	1	0	3	Braund, Mr. Owen Harris
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...
2	3	1	3	Heikkinen, Miss. Laina
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)
4	5	0	3	Allen, Mr. William Henry

Pandas 실습

```
condition = (df['Pclass'] == 3) & (df['Survived'] == 1)

df.loc[condition]
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333	NaN	S
10	11	1	3	Sandstrom, Miss. Marguerite Rut	female	4.0	1	1	PP 9549	16.7000	G6	S
19	20	1	3	Masselmani, Mrs. Fatima	female	NaN	0	0	2649	7.2250	NaN	C
22	23	1	3	McGowan, Miss. Anna "Annie"	female	15.0	0	0	330923	8.0292	NaN	Q
...	...	...	...	...	...	...	...	...	...	...	...	...
838	839	1	3	Chip, Mr. Chang	male	32.0	0	0	1601	56.4958	NaN	S
855	856	1	3	Aks, Mrs. Sam (Leah Rosen)	female	18.0	0	1	392091	9.3500	NaN	S
858	859	1	3	Baclini, Mrs. Solomon (Latifa Qurban)	female	24.0	0	3	2666	19.2583	NaN	C
869	870	1	3	Johnson, Master. Harold Theodor	male	4.0	1	1	347742	11.1333	NaN	S
875	876	1	3	Najib, Miss. Adele Kiamie "Jane"	female	15.0	0	0	2667	7.2250	NaN	C

119 rows × 12 columns

df.iloc[]

> Purely integer-location based indexing for selection by position.

`iloc`은 Integer location의 약자이다.
데이터 프레임 행/열의 순서를 나타내는 정수를 통해 가져오는 방법이다. `df.loc[]`이 라벨을 사용한다면 `df.iloc[]`은 각 행렬의 순번을 사용하는 차이가 있다.

첫번째 ROW 추출하기

```
df.iloc[0]
```

```
PassengerId                     1
Survived                        0
Pclass                          3
Name            Braund, Mr. Owen Harris
Sex                          male
Age                          22.0
SibSp                           1
Parch                           0
Ticket                   A/5 21171
Fare                          7.25
Cabin                          NaN
Embarked                         S
Name: 0, dtype: object
```

두번째 Row의 Name

```
df.iloc[1, 3]
```

```
'Cumings, Mrs. John Bradley (Florence Briggs Thayer)'
```

```
df.iloc[:, 3]
```

```
0                                      Braund, Mr. Owen Harris
1      Cumings, Mrs. John Bradley (Florence Briggs Th...
2                                      Heikkinen, Miss. Laina
3           Futrelle, Mrs. Jacques Heath (Lily May Peel)
4                                      Allen, Mr. William Henry
                                ...
886                                    Montvila, Rev. Juozas
887                             Graham, Miss. Margaret Edith
888                 Johnston, Miss. Catherine Helen "Carrie"
889                                    Behr, Mr. Karl Howell
890                                      Dooley, Mr. Patrick
Name: Name, Length: 891, dtype: object
```

Pandas 실습

```
df.iloc[:, :3]
```

	PassengerId	Survived	Pclass
0	1	0	3
1	2	1	1
2	3	1	3
3	4	1	1
4	5	0	3
...	...	...	...

```
df.iloc[:3, :3]
```

	PassengerId	Survived	Pclass
0	1	0	3
1	2	1	1
2	3	1	3

Pandas 실습

바로 인덱싱하기 df['column명']

```
df['Name']
```

```
0                                    Braund, Mr. Owen Harris
1          Cumings, Mrs. John Bradley (Florence Briggs Th...
2                                    Heikkinen, Miss. Laina
3            Futrelle, Mrs. Jacques Heath (Lily May Peel)
4                                    Allen, Mr. William Henry
                                    ...
886                                  Montvila, Rev. Juozas
887                           Graham, Miss. Margaret Edith
888               Johnston, Miss. Catherine Helen "Carrie"
889                                  Behr, Mr. Karl Howell
890                                    Dooley, Mr. Patrick
Name: Name, Length: 891, dtype: object
```

Pandas 실습

df[[‘a column’, ‘b comumn’]]

```
df[['Pclass', 'Name']]
```

	Pclass	Name
0	3	Braund, Mr. Owen Harris
1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...
2	3	Heikkinen, Miss. Laina
3	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)
4	3	Allen, Mr. William Henry
...	...	...
886	2	Montvila, Rev. Juozas
887	1	Graham, Miss. Margaret Edith
888	3	Johnston, Miss. Catherine Helen "Carrie"
889	1	Behr, Mr. Karl Howell
890	3	Dooley, Mr. Patrick

891 rows × 2 columns

- 참고로 리스트 슬라이싱을 할 때 칼럼은 안됨
- row에 대한 슬라이싱은 df['a':'d’]
- 대신 loc, iloc을 주로 사용

Matplotlib 실습

jupyter Untitled Last Checkpoint: 2023.08.06. (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted Python 3 (ipykernel)

Run Code

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
from numpy import pi

a = np.arange(10, 30, 5) # 10부터 30까지 5간격(시작점, 끝점, 간격)
b = np.arange(0, 2, 0.3) # 0부터 2까지 0.3간격

print(a)
print(b)

c = np.linspace(0, 2, 15) # 0~ 2까지 15개숫자

print(c)

x = np.linspace(0, 2*pi, 100) # 0부터 2*pi 100개 만들기
f = np.sin(x)
print(f)

points = np.arange(-5, 5, 0.01)
xs, ys = np.meshgrid(points, points)

z = np.sqrt(xs**2 + ys**2)

plt.imshow(z, cmap=plt.cm.gray)
plt.colorbar()
plt.title("image")
```

```
[10 15 20 25]
[0.  0.3 0.6 0.9 1.2 1.5 1.8]
[0.         0.14285714 0.28571429 0.42857143 0.57142857 0.71428571
 0.85714286 1.         1.14285714 1.28571429 1.42857143 1.57142857
 1.71428571 1.85714286 2.        ]
[ 0.00000000e+00  6.34239197e-02  1.26592454e-01  1.89251244e-01
  2.51147987e-01  3.12033446e-01  3.71662456e-01  4.29794912e-01
```

Linspace_sin.ipynb