데이터 프레임 불러오기

csv 파일 불러오기 : read_csv()
excel 파일 불러오기 : read_excel()

tsv 파일은 따로 불러오는 함수가 없기 때문에 read_csv() 함수의 매개변수를 조정해주어야 함

In [1]:

```
import pandas as pd
```

tsv 파일 불러오기

- \t : Tab키를 의미함
- sep 매개변수를 통해 데이터를 구분하는 기준을 설정할 수 있음

In [2]:

```
df = pd.read_csv("gapminder.tsv", sep='₩t')
df
```

Out[2]:

	country	continent	year	lifeExp	pop	gdpPercap
0	Afghanistan	Asia	1952	28.801	8425333	779.445314
1	Afghanistan	Asia	1957	30.332	9240934	820.853030
2	Afghanistan	Asia	1962	31.997	10267083	853.100710
3	Afghanistan	Asia	1967	34.020	11537966	836.197138
4	Afghanistan	Asia	1972	36.088	13079460	739.981106
...	...	...	...	...	...	...
1699	Zimbabwe	Africa	1987	62.351	9216418	706.157306
1700	Zimbabwe	Africa	1992	60.377	10704340	693.420786
1701	Zimbabwe	Africa	1997	46.809	11404948	792.449960
1702	Zimbabwe	Africa	2002	39.989	11926563	672.038623
1703	Zimbabwe	Africa	2007	43.487	12311143	469.709298

1704 rows × 6 columns

열에 접근하기

In [3]:

```
df['country']
```

Out[3]:

```
0        Afghanistan
1        Afghanistan
2        Afghanistan
3        Afghanistan
4        Afghanistan
            ...
1699        Zimbabwe
1700        Zimbabwe
1701        Zimbabwe
1702        Zimbabwe
1703        Zimbabwe
Name: country, Length: 1704, dtype: object
```

In [4]:

```
df[['country', 'year']]
```

Out[4]:

	country	year
0	Afghanistan	1952
1	Afghanistan	1957
2	Afghanistan	1962
3	Afghanistan	1967
4	Afghanistan	1972
...	...	...
1699	Zimbabwe	1987
1700	Zimbabwe	1992
1701	Zimbabwe	1997
1702	Zimbabwe	2002
1703	Zimbabwe	2007

1704 rows × 2 columns

행에 접근하기

- loc : 이름으로 접근
- iloc : 순서로 접근

[행,열]로 접근 가능

In [5]:

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

Out[5]:

```
country        Afghanistan
continent             Asia
year                  1952
lifeExp             28.801
pop                8425333
gdpPercap       779.445314
Name: 0, dtype: object
```

In [6]:

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

Out[6]:

```
country        Afghanistan
continent             Asia
year                  1952
lifeExp             28.801
pop                8425333
gdpPercap       779.445314
Name: 0, dtype: object
```

In [7]:

```
df.loc[[0,10,100,1000]]
```

Out[7]:

	country	continent	year	lifeExp	pop	gdpPercap
0	Afghanistan	Asia	1952	28.801	8425333	779.445314
10	Afghanistan	Asia	2002	42.129	25268405	726.734055
100	Bangladesh	Asia	1972	45.252	70759295	630.233627
1000	Mongolia	Asia	1972	53.754	1320500	1421.741975

In [8]:

```
df.iloc[[0,10,100,1000]]
```

Out[8]:

	country	continent	year	lifeExp	pop	gdpPercap
0	Afghanistan	Asia	1952	28.801	8425333	779.445314
10	Afghanistan	Asia	2002	42.129	25268405	726.734055
100	Bangladesh	Asia	1972	45.252	70759295	630.233627
1000	Mongolia	Asia	1972	53.754	1320500	1421.741975

In [9]:

```
df.loc[[0,10,100,1000], ['country', 'year']]
```

Out[9]:

	country	year
0	Afghanistan	1952
10	Afghanistan	2002
100	Bangladesh	1972
1000	Mongolia	1972

In [10]:

```
df.iloc[[0,10,100,1000], [0, 2]]
```

Out[10]:

	country	year
0	Afghanistan	1952
10	Afghanistan	2002
100	Bangladesh	1972
1000	Mongolia	1972

원하는 행만 추출하는 방법

year의 값이 year의 평균값보다 큰 행만 추출하기

In [11]:

```
df[df['year'] > df['year'].mean()]
```

Out[11]:

	country	continent	year	lifeExp	pop	gdpPercap
6	Afghanistan	Asia	1982	39.854	12881816	978.011439
7	Afghanistan	Asia	1987	40.822	13867957	852.395945
8	Afghanistan	Asia	1992	41.674	16317921	649.341395
9	Afghanistan	Asia	1997	41.763	22227415	635.341351
10	Afghanistan	Asia	2002	42.129	25268405	726.734055
...	...	...	...	...	...	...
1699	Zimbabwe	Africa	1987	62.351	9216418	706.157306
1700	Zimbabwe	Africa	1992	60.377	10704340	693.420786
1701	Zimbabwe	Africa	1997	46.809	11404948	792.449960
1702	Zimbabwe	Africa	2002	39.989	11926563	672.038623
1703	Zimbabwe	Africa	2007	43.487	12311143	469.709298

852 rows × 6 columns

year의 값이 year의 평균값보다 크고 동시에 lifeExp값도 lifeExp값의 평균보다 큰 행만 추출하기

In [12]:

```
df[(df['year'] > df['year'].mean()) & (df['lifeExp'] > df['lifeExp'].mean())]
```

Out[12]:

	country	continent	year	lifeExp	pop	gdpPercap
18	Albania	Europe	1982	70.420	2780097	3630.880722
19	Albania	Europe	1987	72.000	3075321	3738.932735
20	Albania	Europe	1992	71.581	3326498	2497.437901
21	Albania	Europe	1997	72.950	3428038	3193.054604
22	Albania	Europe	2002	75.651	3508512	4604.211737
...	...	...	...	...	...	...
1678	Yemen, Rep.	Asia	2002	60.308	18701257	2234.820827
1679	Yemen, Rep.	Asia	2007	62.698	22211743	2280.769906
1698	Zimbabwe	Africa	1982	60.363	7636524	788.855041
1699	Zimbabwe	Africa	1987	62.351	9216418	706.157306
1700	Zimbabwe	Africa	1992	60.377	10704340	693.420786

566 rows × 6 columns

year의 값이 year의 평균값보다 크거나 또는 lifeExp값도 lifeExp값의 평균보다 큰 행만 추출하기

In [13]:

```
df[(df['year'] > df['year'].mean()) | (df['lifeExp'] > df['lifeExp'].mean())]
```

Out[13]:

	country	continent	year	lifeExp	pop	gdpPercap
6	Afghanistan	Asia	1982	39.854	12881816	978.011439
7	Afghanistan	Asia	1987	40.822	13867957	852.395945
8	Afghanistan	Asia	1992	41.674	16317921	649.341395
9	Afghanistan	Asia	1997	41.763	22227415	635.341351
10	Afghanistan	Asia	2002	42.129	25268405	726.734055
...	...	...	...	...	...	...
1699	Zimbabwe	Africa	1987	62.351	9216418	706.157306
1700	Zimbabwe	Africa	1992	60.377	10704340	693.420786
1701	Zimbabwe	Africa	1997	46.809	11404948	792.449960
1702	Zimbabwe	Africa	2002	39.989	11926563	672.038623
1703	Zimbabwe	Africa	2007	43.487	12311143	469.709298

1181 rows × 6 columns

데이터 프레임 모양 확인하기

In [14]:

```
df.shape
```

Out[14]:

```
(1704, 6)
```

데이터 프레임 열 타입 확인하기

In [17]:

```
df.dtypes
```

Out[17]:

```
country       object
continent     object
year           int64
lifeExp      float64
pop            int64
gdpPercap    float64
dtype: object
```

In [18]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1704 entries, 0 to 1703
Data columns (total 6 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   country    1704 non-null   object 
 1   continent  1704 non-null   object 
 2   year       1704 non-null   int64  
 3   lifeExp    1704 non-null   float64
 4   pop        1704 non-null   int64  
 5   gdpPercap  1704 non-null   float64
dtypes: float64(2), int64(2), object(2)
memory usage: 80.0+ KB
```

데이터 프레임 앞 부분 확인하기

In [19]:

```
df.head()
```

Out[19]:

	country	continent	year	lifeExp	pop	gdpPercap
0	Afghanistan	Asia	1952	28.801	8425333	779.445314
1	Afghanistan	Asia	1957	30.332	9240934	820.853030
2	Afghanistan	Asia	1962	31.997	10267083	853.100710
3	Afghanistan	Asia	1967	34.020	11537966	836.197138
4	Afghanistan	Asia	1972	36.088	13079460	739.981106

In [20]:

```
df.head(10)
```

Out[20]:

	country	continent	year	lifeExp	pop	gdpPercap
0	Afghanistan	Asia	1952	28.801	8425333	779.445314
1	Afghanistan	Asia	1957	30.332	9240934	820.853030
2	Afghanistan	Asia	1962	31.997	10267083	853.100710
3	Afghanistan	Asia	1967	34.020	11537966	836.197138
4	Afghanistan	Asia	1972	36.088	13079460	739.981106
5	Afghanistan	Asia	1977	38.438	14880372	786.113360
6	Afghanistan	Asia	1982	39.854	12881816	978.011439
7	Afghanistan	Asia	1987	40.822	13867957	852.395945
8	Afghanistan	Asia	1992	41.674	16317921	649.341395
9	Afghanistan	Asia	1997	41.763	22227415	635.341351

데이터 프레임 꼬리 부분 확인하기

In [21]:

```
df.tail()
```

Out[21]:

	country	continent	year	lifeExp	pop	gdpPercap
1699	Zimbabwe	Africa	1987	62.351	9216418	706.157306
1700	Zimbabwe	Africa	1992	60.377	10704340	693.420786
1701	Zimbabwe	Africa	1997	46.809	11404948	792.449960
1702	Zimbabwe	Africa	2002	39.989	11926563	672.038623
1703	Zimbabwe	Africa	2007	43.487	12311143	469.709298

In [22]:

```
df.tail(10)
```

Out[22]:

	country	continent	year	lifeExp	pop	gdpPercap
1694	Zimbabwe	Africa	1962	52.358	4277736	527.272182
1695	Zimbabwe	Africa	1967	53.995	4995432	569.795071
1696	Zimbabwe	Africa	1972	55.635	5861135	799.362176
1697	Zimbabwe	Africa	1977	57.674	6642107	685.587682
1698	Zimbabwe	Africa	1982	60.363	7636524	788.855041
1699	Zimbabwe	Africa	1987	62.351	9216418	706.157306
1700	Zimbabwe	Africa	1992	60.377	10704340	693.420786
1701	Zimbabwe	Africa	1997	46.809	11404948	792.449960
1702	Zimbabwe	Africa	2002	39.989	11926563	672.038623
1703	Zimbabwe	Africa	2007	43.487	12311143	469.709298

데이터 프레임 열 확인하기

In [23]:

```
df.columns
```

Out[23]:

```
Index(['country', 'continent', 'year', 'lifeExp', 'pop', 'gdpPercap'], dtype='objec
t')
```