In [1]:

```
!pip install pandas
```

```
Requirement already satisfied: pandas in /Users/mohit/opt/anaconda3/li
b/python3.8/site-packages (1.2.4)
Requirement already satisfied: python-dateutil>=2.7.3 in /Users/mohit/
opt/anaconda3/lib/python3.8/site-packages (from pandas) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in /Users/mohit/opt/anacon
da3/lib/python3.8/site-packages (from pandas) (2021.1)
Requirement already satisfied: numpy>=1.16.5 in /Users/mohit/opt/anaco
nda3/lib/python3.8/site-packages (from pandas) (1.20.1)
Requirement already satisfied: six>=1.5 in /Users/mohit/opt/anaconda3/
lib/python3.8/site-packages (from python-dateutil>=2.7.3->pandas) (1.1
5.0)
```

In [96]:

```
import pandas as pd
import numpy as np
```

In [3]:

```
df = pd.read_csv('gapminder.csv')
df.head()
```

Out[3]:

	country	continent	year	lifeExp	population	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 [4]:

```
type(df)
```

Out[4]:

```
pandas.core.frame.DataFrame
```

In [7]:

```
df.shape
```

Out[7]:

```
(1704, 6)
```

In [10]:

```
df.columns
```

Out[10]:

```
Index(['country', 'continent', 'year', 'lifeExp', 'population', 'gdpPe
rCap'], dtype='object')
```

In [11]:

```
df
```

Out[11]:

	country	continent	year	lifeExp	population	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

In [12]:

```
df.head()
```

Out[12]:

	country	continent	year	lifeExp	population	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 [18]:

```
df.head(n=10)
```

Out[18]:

	country	continent	year	lifeExp	population	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

In [13]:

```
df.tail()
```

Out[13]:

	country	continent	year	lifeExp	population	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 [19]:

```
df.tail(n=8)
```

Out[19]:

	country	continent	year	lifeExp	population	gdpPerCap
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

In [21]:

```
df.columns
```

Out[21]:

```
Index(['country', 'continent', 'year', 'lifeExp', 'population', 'gdpPe
rCap'], dtype='object')
```

In [23]:

```
df.keys()
```

Out[23]:

```
Index(['country', 'continent', 'year', 'lifeExp', 'population', 'gdpPe
rCap'], dtype='object')
```

In [26]:

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

Out[26]:

```
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 [27]:

```
type(df['country'])
```

Out[27]:

```
pandas.core.series.Series
```

In [36]:

```
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   population  1704 non-null   int64  
 5   gdpPerCap   1704 non-null   float64
dtypes: float64(2), int64(2), object(2)
memory usage: 80.0+ KB
```

In [31]:

```
df.dtypes
```

Out[31]:

```
country        object
continent      object
year            int64
lifeExp       float64
population      int64
gdpPerCap     float64
dtype: object
```

In [34]:

```
# df['country'].unique()
```

In [40]:

```
df.describe().round(2)
```

Out[40]:

	year	lifeExp	population	gdpPerCap
count	1704.00	1704.00	1.704000e+03	1704.00
mean	1979.50	59.47	2.960121e+07	7215.33
std	17.27	12.92	1.061579e+08	9857.45
min	1952.00	23.60	6.001100e+04	241.17
25%	1965.75	48.20	2.793664e+06	1202.06
50%	1979.50	60.71	7.023596e+06	3531.85
75%	1993.25	70.85	1.958522e+07	9325.46
max	2007.00	82.60	1.318683e+09	113523.13

In [44]:

```
df.describe(include='all')
```

Out[44]:

	country	continent	year	lifeExp	population	gdpPerCap
count	1704	1704	1704.00000	1704.000000	1.704000e+03	1704.000000
unique	142	5	NaN	NaN	NaN	NaN
top	Ethiopia	Africa	NaN	NaN	NaN	NaN
freq	12	624	NaN	NaN	NaN	NaN
mean	NaN	NaN	1979.50000	59.474439	2.960121e+07	7215.327081
std	NaN	NaN	17.26533	12.917107	1.061579e+08	9857.454543
min	NaN	NaN	1952.00000	23.599000	6.001100e+04	241.165876
25%	NaN	NaN	1965.75000	48.198000	2.793664e+06	1202.060309
50%	NaN	NaN	1979.50000	60.712500	7.023596e+06	3531.846988
75%	NaN	NaN	1993.25000	70.845500	1.958522e+07	9325.462346
max	NaN	NaN	2007.00000	82.603000	1.318683e+09	113523.132900

Basic Ops

In [49]:

```
df['country'].head()
```

Out[49]:

```
0    Afghanistan
1    Afghanistan
2    Afghanistan
3    Afghanistan
4    Afghanistan
Name: country, dtype: object
```

In [50]:

```
df
```

Out[50]:

	country	continent	year	lifeExp	population	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

In [55]:

```
lst_of_keys = ['country', 'lifeExp']
df[lst_of_keys].head()
```

Out[55]:

	country	lifeExp
0	Afghanistan	28.801
1	Afghanistan	30.332
2	Afghanistan	31.997
3	Afghanistan	34.020
4	Afghanistan	36.088

In [56]:

```
df.columns
```

Out[56]:

```
Index(['country', 'continent', 'year', 'lifeExp', 'population', 'gdpPe
rCap'], dtype='object')
```

In [59]:

```
df.rename(columns={'country': 'COUNTRY'}, inplace=True)
```

In [60]:

```
df.head()
```

Out[60]:

	COUNTRY	continent	year	lifeExp	population	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 [63]:

```
df.rename({'COUNTRY': 'country'}, axis=1, inplace=True)
```

In [65]:

```
df.head()
```

Out[65]:

	country	continent	year	lifeExp	population	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 [66]:

```
# changing all the existing columns to the new ones

# df.columns = ['A', 'B', 'C', 'D', 'E', 'F']
```

Out[66]:

```
Index(['country', 'continent', 'year', 'lifeExp', 'population', 'gdpPe
rCap'], dtype='object')
```

In [71]:

```
df.country
```

Out[71]:

```
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 [72]:

```
df.country is df['country']
```

Out[72]:

```
True
```

In [73]:

```
df.head()
```

Out[73]:

	country	continent	year	lifeExp	population	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 [78]:

```
df.drop('continent', axis=1)
```

Out[78]:

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

1704 rows × 5 columns

In [80]:

```
df.drop(columns=['continent'], inplace= True)
```

In [81]:

```
df.head()
```

Out[81]:

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

In [83]:

```
# df['new']
```

In [91]:

```
df['year'] + df['lifeExp']
```

Out[91]:

```
0       1980.801
1       1987.332
2       1993.997
3       2001.020
4       2008.088
          ...
1699    2049.351
1700    2052.377
1701    2043.809
1702    2041.989
1703    2050.487
Length: 1704, dtype: float64
```

In [92]:

```
df['new'] = df['year']+df['lifeExp']
```

In [93]:

```
df.head()
```

Out[93]:

	country	year	lifeExp	population	gdpPerCap	new
0	Afghanistan	1952	28.801	8425333	779.445314	1980.801
1	Afghanistan	1957	30.332	9240934	820.853030	1987.332
2	Afghanistan	1962	31.997	10267083	853.100710	1993.997
3	Afghanistan	1967	34.020	11537966	836.197138	2001.020
4	Afghanistan	1972	36.088	13079460	739.981106	2008.088

In [94]:

```
df['sub'] = df['year'] - df['lifeExp']
```

In [95]:

```
df.head()
```

Out[95]:

	country	year	lifeExp	population	gdpPerCap	new	sub
0	Afghanistan	1952	28.801	8425333	779.445314	1980.801	1923.199
1	Afghanistan	1957	30.332	9240934	820.853030	1987.332	1926.668
2	Afghanistan	1962	31.997	10267083	853.100710	1993.997	1930.003
3	Afghanistan	1967	34.020	11537966	836.197138	2001.020	1932.980
4	Afghanistan	1972	36.088	13079460	739.981106	2008.088	1935.912

In [ ]:

```

```

In [100]:

```
vec = np.random.randint(1, 50, 1704)
vec
```

Out[100]:

```
array([11, 21, 37, ..., 27, 26, 21])
```

In [101]:

```
df['vec'] = vec
```

In [102]:

```
df.head()
```

Out[102]:

	country	year	lifeExp	population	gdpPerCap	new	sub	vec
0	Afghanistan	1952	28.801	8425333	779.445314	1980.801	1923.199	11
1	Afghanistan	1957	30.332	9240934	820.853030	1987.332	1926.668	21
2	Afghanistan	1962	31.997	10267083	853.100710	1993.997	1930.003	37
3	Afghanistan	1967	34.020	11537966	836.197138	2001.020	1932.980	40
4	Afghanistan	1972	36.088	13079460	739.981106	2008.088	1935.912	14

In [103]:

```
df['zero'] = 0
```

In [104]:

```
df.head()
```

Out[104]:

	country	year	lifeExp	population	gdpPerCap	new	sub	vec	zero
0	Afghanistan	1952	28.801	8425333	779.445314	1980.801	1923.199	11	0
1	Afghanistan	1957	30.332	9240934	820.853030	1987.332	1926.668	21	0
2	Afghanistan	1962	31.997	10267083	853.100710	1993.997	1930.003	37	0
3	Afghanistan	1967	34.020	11537966	836.197138	2001.020	1932.980	40	0
4	Afghanistan	1972	36.088	13079460	739.981106	2008.088	1935.912	14	0

In [105]:

```
df.drop(columns=['new', 'sub', 'vec', 'zero'], inplace=True, axis=1)
```

In [106]:

```
df.head()
```

Out[106]:

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

In [107]:

```
df.head()
```

Out[107]:

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

In [108]:

```
df.drop(3, axis=0)
```

Out[108]:

	country	year	lifeExp	population	gdpPerCap
0	Afghanistan	1952	28.801	8425333	779.445314
1	Afghanistan	1957	30.332	9240934	820.853030
2	Afghanistan	1962	31.997	10267083	853.100710
4	Afghanistan	1972	36.088	13079460	739.981106
5	Afghanistan	1977	38.438	14880372	786.113360
...	...	...	...	...	...
1699	Zimbabwe	1987	62.351	9216418	706.157306
1700	Zimbabwe	1992	60.377	10704340	693.420786
1701	Zimbabwe	1997	46.809	11404948	792.449960
1702	Zimbabwe	2002	39.989	11926563	672.038623
1703	Zimbabwe	2007	43.487	12311143	469.709298

1703 rows × 5 columns

In [109]:

```
df.drop(index=3)
```

Out[109]:

	country	year	lifeExp	population	gdpPerCap
0	Afghanistan	1952	28.801	8425333	779.445314
1	Afghanistan	1957	30.332	9240934	820.853030
2	Afghanistan	1962	31.997	10267083	853.100710
4	Afghanistan	1972	36.088	13079460	739.981106
5	Afghanistan	1977	38.438	14880372	786.113360
...	...	...	...	...	...
1699	Zimbabwe	1987	62.351	9216418	706.157306
1700	Zimbabwe	1992	60.377	10704340	693.420786
1701	Zimbabwe	1997	46.809	11404948	792.449960
1702	Zimbabwe	2002	39.989	11926563	672.038623
1703	Zimbabwe	2007	43.487	12311143	469.709298

1703 rows × 5 columns

In [110]:

```
df
```

Out[110]:

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

1704 rows × 5 columns

In [ ]:

```

```

Working with Rows

In [112]:

```
ser = df['country']
ser
```

Out[112]:

```
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 [116]:

```
ser[20]
```

Out[116]:

```
'Albania'
```

In [120]:

```
ser[6:15]
```

Out[120]:

```
6     Afghanistan
7     Afghanistan
8     Afghanistan
9     Afghanistan
10    Afghanistan
11    Afghanistan
12        Albania
13        Albania
14        Albania
Name: country, dtype: object
```

In [121]:

```
list(ser[6:15])
```

Out[121]:

```
['Afghanistan',
 'Afghanistan',
 'Afghanistan',
 'Afghanistan',
 'Afghanistan',
 'Afghanistan',
 'Albania',
 'Albania',
 'Albania']
```

In [ ]:

```

```

In [123]:

```
df.columns
```

Out[123]:

```
Index(['country', 'year', 'lifeExp', 'population', 'gdpPerCap'], dtype
='object')
```

In [127]:

```
df.index[2]
```

Out[127]:

```
2
```

In [125]:

```
# df.index = np.arange(0, 1704) + 10
```

In [130]:

```
data = pd.Series(['a', 'b', 'c'], index=[1,5,3])
data
```

Out[130]:

```
1    a
5    b
3    c
dtype: object
```

In [133]:

```
# error

# data[0]
```

In [131]:

```
data[1] # uses explicit indexing
```

Out[131]:

```
'a'
```

In [134]:

```
data[1:3] # uses implicit indexing
```

Out[134]:

```
5    b
3    c
dtype: object
```

loc, iloc

In [135]:

```
data.loc[1]
```

Out[135]:

```
'a'
```

In [136]:

```
data.loc[1:3]
```

Out[136]:

```
1    a
5    b
3    c
dtype: object
```

In [137]:

```
# implicit
data.iloc[1]
```

Out[137]:

```
'b'
```

In [138]:

```
data.iloc[0:2]
```

Out[138]:

```
1    a
5    b
dtype: object
```

In [ ]:

```

```

In [139]:

```
df
```

Out[139]:

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

1704 rows × 5 columns

In [142]:

```
df.loc[4]
```

Out[142]:

```
country       Afghanistan
year                 1972
lifeExp            36.088
population       13079460
gdpPerCap      739.981106
Name: 4, dtype: object
```

In [143]:

```
df.iloc[4]
```

Out[143]:

```
country       Afghanistan
year                 1972
lifeExp            36.088
population       13079460
gdpPerCap      739.981106
Name: 4, dtype: object
```

In [146]:

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

Out[146]:

	country	year	lifeExp	population	gdpPerCap
10	Afghanistan	2002	42.129	25268405	726.734055
100	Bangladesh	1972	45.252	70759295	630.233627
1000	Mongolia	1972	53.754	1320500	1421.741975

In [147]:

```
df.iloc[-1]
```

Out[147]:

```
country          Zimbabwe
year                 2007
lifeExp            43.487
population       12311143
gdpPerCap      469.709298
Name: 1703, dtype: object
```

In [148]:

```
df.loc[-1]
```

```
---------------------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
 last)
~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexes/range.
py in get_loc(self, key, method, tolerance)
    350                 try:
--> 351                     return self._range.index(new_key)
    352                 except ValueError as err:

ValueError: -1 is not in range

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call
 last)
<ipython-input-148-c8e3ddd1947a> in <module>
----> 1 df.loc[-1]

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
__getitem__(self, key)
    893 
    894             maybe_callable = com.apply_if_callable(key, self.o
bj)
--> 895             return self._getitem_axis(maybe_callable, axis=axi
s)
    896 
    897     def _is_scalar_access(self, key: Tuple):

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
_getitem_axis(self, key, axis)
   1122         # fall thru to straight lookup
   1123         self._validate_key(key, axis)
-> 1124         return self._get_label(key, axis=axis)
   1125 
   1126     def _get_slice_axis(self, slice_obj: slice, axis: int):

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
_get_label(self, label, axis)
   1071     def _get_label(self, label, axis: int):
   1072         # GH#5667 this will fail if the label is not present i
n the axis.
-> 1073         return self.obj.xs(label, axis=axis)
   1074 
   1075     def _handle_lowerdim_multi_index_axis0(self, tup: Tuple):

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/generic.py in
xs(self, key, axis, level, drop_level)
   3737                 raise TypeError(f"Expected label or tuple of l
abels, got {key}") from e
   3738         else:
-> 3739             loc = index.get_loc(key)
   3740 
   3741             if isinstance(loc, np.ndarray):

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexes/range.
```

```
py in get_loc(self, key, method, tolerance)
    351                     return self._range.index(new_key)
    352                 except ValueError as err:
--> 353                     raise KeyError(key) from err
    354             raise KeyError(key)
    355         return super().get_loc(key, method=method, tolerance=t
olerance)

KeyError: -1
```

In [ ]:

```

```

In [149]:

```
df.iloc[5:15]
```

Out[149]:

	country	year	lifeExp	population	gdpPerCap
5	Afghanistan	1977	38.438	14880372	786.113360
6	Afghanistan	1982	39.854	12881816	978.011439
7	Afghanistan	1987	40.822	13867957	852.395945
8	Afghanistan	1992	41.674	16317921	649.341395
9	Afghanistan	1997	41.763	22227415	635.341351
10	Afghanistan	2002	42.129	25268405	726.734055
11	Afghanistan	2007	43.828	31889923	974.580338
12	Albania	1952	55.230	1282697	1601.056136
13	Albania	1957	59.280	1476505	1942.284244
14	Albania	1962	64.820	1728137	2312.888958

In [ ]:

```

```

In [150]:

```
temp = df.set_index('country')
temp
```

Out[150]:

	year	lifeExp	population	gdpPerCap
country				
Afghanistan	1952	28.801	8425333	779.445314
Afghanistan	1957	30.332	9240934	820.853030
Afghanistan	1962	31.997	10267083	853.100710
Afghanistan	1967	34.020	11537966	836.197138
Afghanistan	1972	36.088	13079460	739.981106
...	...	...	...	...
Zimbabwe	1987	62.351	9216418	706.157306
Zimbabwe	1992	60.377	10704340	693.420786
Zimbabwe	1997	46.809	11404948	792.449960
Zimbabwe	2002	39.989	11926563	672.038623
Zimbabwe	2007	43.487	12311143	469.709298

1704 rows × 4 columns

In [151]:

```
temp.loc['Afghanistan']
```

Out[151]:

	year	lifeExp	population	gdpPerCap
country				
Afghanistan	1952	28.801	8425333	779.445314
Afghanistan	1957	30.332	9240934	820.853030
Afghanistan	1962	31.997	10267083	853.100710
Afghanistan	1967	34.020	11537966	836.197138
Afghanistan	1972	36.088	13079460	739.981106
Afghanistan	1977	38.438	14880372	786.113360
Afghanistan	1982	39.854	12881816	978.011439
Afghanistan	1987	40.822	13867957	852.395945
Afghanistan	1992	41.674	16317921	649.341395
Afghanistan	1997	41.763	22227415	635.341351
Afghanistan	2002	42.129	25268405	726.734055
Afghanistan	2007	43.828	31889923	974.580338

In [152]:

```
temp.iloc[0]
```

Out[152]:

```
year          1.952000e+03
lifeExp       2.880100e+01
population    8.425333e+06
gdpPerCap     7.794453e+02
Name: Afghanistan, dtype: float64
```

In [ ]:

```
# temp.drop(index= 'Afg')
```

In [154]:

```
temp.reset_index()
```

Out[154]:

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

1704 rows × 5 columns

In [ ]:

```

```

Work rows and columns

In [155]:

```
df
```

Out[155]:

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

1704 rows × 5 columns
```

In [156]:

```
df.iloc[2:5, 1:3]
```

Out[156]:

```
   year  lifeExp
2  1962   31.997
3  1967   34.020
4  1972   36.088
```

In [157]:

```
df.iloc[1:100:10, 1:4]
```

Out[157]:

	year	lifeExp	population
1	1957	30.332	9240934
11	2007	43.828	31889923
21	1997	72.950	3428038
31	1987	65.799	23254956
41	1977	39.483	6162675
51	1967	65.634	22934225
61	1957	70.330	9712569
71	2007	81.235	20434176
81	1997	77.510	8069876
91	1987	70.750	454612

In [160]:

```
df.iloc[1:100:10, 1:4].reset_index(drop = True)
```

Out[160]:

	year	lifeExp	population
0	1957	30.332	9240934
1	2007	43.828	31889923
2	1997	72.950	3428038
3	1987	65.799	23254956
4	1977	39.483	6162675
5	1967	65.634	22934225
6	1957	70.330	9712569
7	2007	81.235	20434176
8	1997	77.510	8069876
9	1987	70.750	454612

In [161]:

```
df.iloc[-500::-1 , 2:5]
```

Out[161]:

	lifeExp	population	gdpPerCap
1204	55.448	13954700	5937.827283
1203	51.445	12132200	5788.093330
1202	49.096	10516500	4957.037982
1201	46.263	9146100	4245.256698
1200	43.902	8025700	3758.523437
...	...	...	...
4	36.088	13079460	739.981106
3	34.020	11537966	836.197138
2	31.997	10267083	853.100710
1	30.332	9240934	820.853030
0	28.801	8425333	779.445314

1205 rows × 3 columns

In [163]:

```
df
```

Out[163]:

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

1704 rows × 5 columns

In [164]:

```
df.loc[ 2:5  , 1:3 ]
```

```
---------------------------------------------------------------------
-----
TypeError                                 Traceback (most recent call
 last)
<ipython-input-164-771d76e11c52> in <module>
----> 1 df.loc[ 2:5  , 1:3 ]

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
__getitem__(self, key)
    887                     # AttributeError for IntervalTree get_valu
e
    888                     return self.obj._get_value(*key, takeable=
self._takeable)
--> 889             return self._getitem_tuple(key)
    890         else:
    891             # we by definition only have the 0th axis

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
_getitem_tuple(self, tup)
   1067             return self._multi_take(tup)
   1068
-> 1069         return self._getitem_tuple_same_dim(tup)
   1070
   1071     def _get_label(self, label, axis: int):

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
_getitem_tuple_same_dim(self, tup)
    773                 continue
    774
--> 775             retval = getattr(retval, self.name)._getitem_axis(
key, axis=i)
    776             # We should never have retval.ndim < self.ndim, as
that should
    777             #  be handled by the _getitem_lowerdim call above.

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
_getitem_axis(self, key, axis)
   1100         if isinstance(key, slice):
   1101             self._validate_key(key, axis)
-> 1102             return self._get_slice_axis(key, axis=axis)
   1103         elif com.is_bool_indexer(key):
   1104             return self._getbool_axis(key, axis=axis)

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexing.py in
_get_slice_axis(self, slice_obj, axis)
   1134
   1135         labels = obj._get_axis(axis)
-> 1136         indexer = labels.slice_indexer(
   1137             slice_obj.start, slice_obj.stop, slice_obj.step, k
ind="loc"
   1138         )

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexes/base.p
y in slice_indexer(self, start, end, step, kind)
   5275         slice(1, 3, None)
   5276         """
```

```
-> 5277         start_slice, end_slice = self.slice_locs(start, end, s
tep=step, kind=kind)
   5278
   5279         # return a slice

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexes/base.p
y in slice_locs(self, start, end, step, kind)
   5474         start_slice = None
   5475         if start is not None:
-> 5476             start_slice = self.get_slice_bound(start, "left",
 kind)
   5477         if start_slice is None:
   5478             start_slice = 0

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexes/base.p
y in get_slice_bound(self, label, side, kind)
   5384         # For datetime indices label may be a string that has
 to be converted
   5385         # to datetime boundary according to its resolution.
-> 5386         label = self._maybe_cast_slice_bound(label, side, kind
)
   5387
   5388         # we need to look up the label

~/opt/anaconda3/lib/python3.8/site-packages/pandas/core/indexes/base.p
y in _maybe_cast_slice_bound(self, label, side, kind)
   5336         # reject them, if index does not contain label
   5337         if (is_float(label) or is_integer(label)) and label no
t in self.values:
-> 5338             raise self._invalid_indexer("slice", label)
   5339
   5340         return label

TypeError: cannot do slice indexing on Index with these indexers [1] o
f type int
```

In [165]:

```
df.columns
```

Out[165]:

```
Index(['country', 'year', 'lifeExp', 'population', 'gdpPerCap'], dtype
='object')
```

In [166]:

```
df.loc[ 2:5  , 'year'  : 'population' ]
```

Out[166]:

	year	lifeExp	population
2	1962	31.997	10267083
3	1967	34.020	11537966
4	1972	36.088	13079460
5	1977	38.438	14880372

In [172]:

```
# df.loc[0:5005]
```

In [176]:

```
df.values
```

Out[176]:

```
array([['Afghanistan', 1952, 28.801, 8425333, 779.4453145],
       ['Afghanistan', 1957, 30.332, 9240934, 820.8530296],
       ['Afghanistan', 1962, 31.997, 10267083, 853.10071],
       ...,
       ['Zimbabwe', 1997, 46.809, 11404948, 792.4499603],
       ['Zimbabwe', 2002, 39.989, 11926563, 672.0386227],
       ['Zimbabwe', 2007, 43.487, 12311143, 469.7092981]], dtype=objec
t)
```

In [ ]:

```

```

In [178]:

```
df['lifeExp'].sum()
```

Out[178]:

```
101344.44467999999
```

In [179]:

```
df['lifeExp'].mean()
```

Out[179]:

```
59.47443936619714
```

In [180]:

```
df['lifeExp'].median()
```

Out[180]:

```
60.7125
```

In [181]:

```
df['lifeExp'].min()
```

Out[181]:

```
23.599
```

In [182]:

```
df['lifeExp'].max()
```

Out[182]:

```
82.603
```

In [186]:

```
df['lifeExp'].shape[0]
```

Out[186]:

```
1704
```

In [190]:

```
df.mean()
```

Out[190]:

```
year          1.979500e+03
lifeExp       5.947444e+01
population    2.960121e+07
gdpPerCap     7.215327e+03
dtype: float64
```

In [192]:

```
df
```

Out[192]:

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

1704 rows × 5 columns

In [195]:

```
df.sort_values(by='year', ascending=False)
```

Out[195]:

	country	year	lifeExp	population	gdpPerCap
1703	Zimbabwe	2007	43.487	12311143	469.709298
491	Equatorial Guinea	2007	51.579	551201	12154.089750
515	Ethiopia	2007	52.947	76511887	690.805576
527	Finland	2007	79.313	5238460	33207.084400
539	France	2007	80.657	61083916	30470.016700
...	...	...	...	...	...
1116	Niger	1952	37.444	3379468	761.879376
1128	Nigeria	1952	36.324	33119096	1077.281856
1140	Norway	1952	72.670	3327728	10095.421720
1152	Oman	1952	37.578	507833	1828.230307
852	Kuwait	1952	55.565	160000	108382.352900

1704 rows × 5 columns

In [197]:

```python
df.sort_values(by=['year', 'lifeExp'], ascending=[False, True])
```

Out[197]:

	country	year	lifeExp	population	gdpPerCap
1463	Swaziland	2007	39.613	1133066	4513.480643
1043	Mozambique	2007	42.082	19951656	823.685621
1691	Zambia	2007	42.384	11746035	1271.211593
1355	Sierra Leone	2007	42.568	6144562	862.540756
887	Lesotho	2007	42.592	2012649	1569.331442
...	...	...	...	...	...
408	Denmark	1952	70.780	4334000	9692.385245
1464	Sweden	1952	71.860	7124673	8527.844662
1080	Netherlands	1952	72.130	10381988	8941.571858
684	Iceland	1952	72.490	147962	7267.688428
1140	Norway	1952	72.670	3327728	10095.421720

1704 rows × 5 columns

In [ ]:

```python
```

Creating DF fromm scratch

In [200]:

```
rand_mat = np.random.randint(1, 50, (10, 3))

pd.DataFrame(rand_mat, columns= ['a','b', 'c'])
```

Out[200]:

	a	b	c
0	7	26	43
1	11	4	6
2	16	48	12
3	34	33	7
4	48	5	49
5	29	27	4
6	49	6	37
7	30	10	24
8	13	22	44
9	41	31	46

In [201]:

```
d = {
    'a' : [1,2,3,4,5],
    'b' : [10,20,30,40,50],
    'c' : [10,9,8,7,6]

}
```

In [202]:

```
pd.DataFrame(d)
```

Out[202]:

	a	b	c
0	1	10	10
1	2	20	9
2	3	30	8
3	4	40	7
4	5	50	6

In [ ]:

```

```

Concat

In [204]:

```
a = pd.DataFrame({'A':[10,30], 'B':[20,40]})
b = pd.DataFrame({'A':[10,30], 'C':[20,40]})
```

In [205]:

```
a
```

Out[205]:

	A	B
0	10	20
1	30	40

In [206]:

```
b
```

Out[206]:

	A	C
0	10	20
1	30	40

In [ ]:

```
np.concatenate
```

In [209]:

```
pd.concat([a, b], axis=0)
```

Out[209]:

	A	B	C
0	10	20.0	NaN
1	30	40.0	NaN
0	10	NaN	20.0
1	30	NaN	40.0

In [212]:

```
temp  =  pd.concat([a, b], axis=1, )
temp
```

Out[212]:

	A	B	A	C
0	10	20	10	20
1	30	40	30	40

In [213]:

```
temp['A']
```

Out[213]:

	A	A
0	10	10
1	30	30

In [214]:

```
arr=np.arange(16).reshape(4,4)
arr
```

Out[214]:

```
array([[ 0,  1,  2,  3],
       [ 4,  5,  6,  7],
       [ 8,  9, 10, 11],
       [12, 13, 14, 15]])
```

In [219]:

```
# arr[]
```

In [218]:

```
# arr[[0,2],[1,3]]=99
```

In [220]:

```
# arr
```

In [ ]:

```

```

Merge

In [211]:

```
# pd.merge
```

In [ ]:

```

```