```
# The Series Data Structure

import pandas as pd
pd.Series?

animals=['Tiger','Bear','Moose']
animals
```

```
['Tiger', 'Bear', 'Moose']
```

```
pd.Series(animals)
```

```
0    Tiger
1     Bear
2    Moose
dtype: object
```

```
numbers=[1,2,3]
pd.Series(numbers)
```

```
0    1
1    2
2    3
dtype: int64
```

저장이 완료되었습니다.

```
0    Tigers
1      Bear
2      None
dtype: object
```

```
numbers=[1,2,None]
pd.Series(numbers)
```

```
0    1.0
1    2.0
2    NaN
dtype: float64
```

```
import numpy as np
np.nan==None
```

```
False
```

```
np.nan==np.nan
```

```
False
```

```
np.isnan(np.nan)
```

```
True
```

```
sports={'Archery':'Bhutan',
        'Golf':'Scotland',
        'Sumo':'Japan',
        'Taekwondo':'South Korea'}
s=pd.Series(sports)
s
```

```
Archery           Bhutan
Golf            Scotland
Sumo               Japan
Taekwondo    South Korea
dtype: object
```

```
s.index
```

```
Index(['Archery', 'Golf', 'Sumo', 'Taekwondo'], dtype='object')
```

```
s=pd.Series(['Tiger','Bear','Moose'],index=['India','America','Canada'])
s
```

```
India      Tiger
America     Bear
Canada     Moose
```

저장이 완료되었습니다.

```
sports={'Archery':'Bhutan',
        'Golf':'Scotland',
        'Sumo':'Japan',
        'Taekwondo':'South Korea'}
s=pd.Series(sports,index=['Golf','Sumo','Hockey'])
s
```

```
Golf      Scotland
Sumo         Japan
Hockey         NaN
dtype: object
```

```
# Querying a Series
```

```
sports={'Archery':'Bhutan',
        'Golf':'Scotland',
        'Sumo':'Japan',
        'Taekwondo':'South Korea'}
s=pd.Series(sports)
s
```

```
Archery           Bhutan
Golf            Scotland
Sumo               Japan
Taekwondo    South Korea
dtype: object
```

```
s.iloc[2]
```

```
'Japan'
```

```
s.loc['Sumo']
```

```
'Japan'
```

```
s[2]
```

```
'Japan'
```

```
s['Sumo']
```

```
'Japan'
```

```
s[1]
```

```
'Scotland'
```

저장이 완료되었습니다.

```
 6:'Scotland',
 7:'Japan',
 8:'South Korea'}
s=pd.Series(sports)
s
```

```
5         Bhutan
6       Scotland
7          Japan
8    South Korea
dtype: object
```

```
s[0]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-25-c9c96910e542> in <module>()
----> 1 s[0]

                                 1 frames
/usr/local/lib/python3.6/dist-packages/pandas/core/indexes/base.py in get_value(self, series,
   4402         k = self._convert_scalar_indexer(k, kind="getitem")
   4403         try:
-> 4404             return self._engine.get_value(s, k, tz=getattr(series.dtype, "tz", None))
   4405         except KeyError as e1:
   4406             if len(self) > 0 and (self.holds_integer() or self.is_boolean()):

pandas/_libs/index.pyx in pandas._libs.index.IndexEngine.get_value()

pandas/_libs/index.pyx in pandas._libs.index.IndexEngine.get_value()

pandas/_libs/index.pyx in pandas._libs.index.IndexEngine.get_loc()

pandas/_libs/hashtable_class_helper.pxi in pandas._libs.hashtable.Int64HashTable.get_item()

pandas/_libs/hashtable_class_helper.pxi in pandas._libs.hashtable.Int64HashTable.get_item()

KeyError: 0
```

SEARCH STACK OVERFLOW

저장이 완료되었습니다.

```
'Bhutan'
```

```
s.iloc[1]
```

```
'Scotland'
```

```
s=pd.Series([100,120,101,3])
s
```

```
0    100
1    120
2    101
3      3
dtype: int64
```

```
total = 0
for item in s:
    total+=item
print(total)
```

```
324
```

```
total=np.sum(s)
print(total)
```

```
324
```

```
s=pd.Series(np.random.randint(0,1000,10000))
s
```

```
0       615
1       290
2       610
3       996
4       916
       ...
9995    273
9996      2
9997    640
9998    698
9999    688
Length: 10000, dtype: int64
```

```
s.head()
```

```
0    615
1    290
2    610
3    996
```



```
%%timeit -n 100
summary=0
for item in s:
    summary+=item
```

```
100 loops, best of 3: 1.22 ms per loop
```

```
%%timeit -n 100
summary=np.sum(s)
```

```
100 loops, best of 3: 83.6 µs per loop
```

```
s=pd.Series([1,2,3])
s
```

```
0    1
1    2
2    3
dtype: int64
```

```
s.loc['Animal']='Bears'
s
```

```
0            1
1            2
2            3
Animal    Bears
dtype: object
```

```
sports={'Archery':'Bhutan',
        'Golf':'Scotland',
        'Sumo':'Japan',
        'Taekwondo':'South Korea'}
s=pd.Series(sports)
s
```

```
Archery           Bhutan
Golf            Scotland
Sumo               Japan
Taekwondo    South Korea
dtype: object
```

```
criket_countries=pd.Series(['Australia','Pakistan','England'],
                           index=['Cricket','Cricket','Cricket'])
```

```
criket_countries
```

```
Cricket    Australia
Cricket     Pakistan
```

저장이 완료되었습니다.

```
all_countries=s.append(criket_countries)
```

```
all_countries
```

```
Archery           Bhutan
Golf            Scotland
Sumo               Japan
Taekwondo    South Korea
Cricket        Australia
Cricket         Pakistan
Cricket          England
dtype: object
```

```
s
```

```
Archery           Bhutan
Golf            Scotland
Sumo               Japan
Taekwondo    South Korea
dtype: object
```

```
all_countries.loc['Cricket']
```

```
Cricket    Australia
Cricket     Pakistan
Cricket      England
dtype: object
```

```
#The DataFrame Data Structure

purchase_1=pd.Series({'Name':'Chris',
                      'Item Purchased':'Dog Food',
                      'Cost':22.50})
purchase_2=pd.Series({'Name':'Kevin',
                      'Item Purchased':'Kitty Litter',
                      'Cost':2.50})
purchase_3=pd.Series({'Name':'Vinod',
                      'Item Purchased':'Bird Seed',
                      'Cost':5.00})
df=pd.DataFrame([purchase_1,purchase_2,purchase_3],
                index=['Store1','Store1','Store2'])
df.head()
```

| | Name | Item Purchased | Cost |
|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 |
| **Store1** | Kevin | Kitty Litter | 2.5 |
| **Store2** | Vinod | Bird Seed | 5.0 |

저장이 완료되었습니다.

```
Name                 Vinod
Item Purchased    Bird Seed
Cost                      5
Name: Store2, dtype: object
```

```
type(df.loc['Store2'])
```

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

```
type(df)
```

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

```
df.loc['Store1']
```

| | Name | Item Purchased | Cost |
|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 |
| **Store1** | Kevin | Kitty Litter | 2.5 |

```
df.loc['Store1']['Cost']# 비추천
```

```
df.loc['Store1','Cost']#비효율
```

```
Store1    22.5
Store1     2.5
Name: Cost, dtype: float64
```

```
df.loc['Store1','Cost']
```

```
Store1    22.5
Store1     2.5
Name: Cost, dtype: float64
```

```
df
```

| | Name | Item Purchased | Cost |
|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 |
| **Store1** | Kevin | Kitty Litter | 2.5 |
| **Store2** | Vinod | Bird Seed | 5.0 |

```
df.T.loc['Cost']
```

```
Store1    22.5
Store1     2.5
Store2       5
Name: Cost, dtype: object
```

저장이 완료되었습니다.

```
Store1    22.5
Store1     2.5
Store2     5.0
Name: Cost, dtype: float64
```

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

| | Name | Cost |
|---|---|---|
| **Store1** | Chris | 22.5 |
| **Store1** | Kevin | 2.5 |
| **Store2** | Vinod | 5.0 |

```
df.drop('Store1')
```

| | Name | Item Purchased | Cost |
|---|---|---|---|
| **Store2** | Vinod | Bird Seed | 5.0 |

```
df_copy=df.copy()
```

```
df_copy
```

| | Name | Item Purchased | Cost |
|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 |
| **Store1** | Kevin | Kitty Litter | 2.5 |
| **Store2** | Vinod | Bird Seed | 5.0 |

```
df_copy=df_copy.drop('Store1')
```

```
df_copy
```

| | Name | Item Purchased | Cost |
|---|---|---|---|
| **Store2** | Vinod | Bird Seed | 5.0 |

```
del df_copy['Name']
```

```
df_copy
```

| | Item Purchased | Cost |
|---|---|---|
| **Store2** | Bird Seed | 5.0 |

저장이 완료되었습니다.

```
df
```

| | Name | Item Purchased | Cost | Location |
|---|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 | None |
| **Store1** | Kevin | Kitty Litter | 2.5 | None |
| **Store2** | Vinod | Bird Seed | 5.0 | None |

```
df.loc['Store1']
```

| | Name | Item Purchased | Cost | Location |
|---|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 | None |
| **Store1** | Kevin | Kitty Litter | 2.5 | None |

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

| | Name | Cost |
|---|---|---|
| **Store1** | Chris | 22.5 |
| **Store1** | Kevin | 2.5 |
| **Store2** | Vinod | 5.0 |

```
# Dataframe Indexing and Loading
```

```
df
```

| | Name | Item Purchased | Cost | Location |
|---|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 | None |
| **Store1** | Kevin | Kitty Litter | 2.5 | None |
| **Store2** | Vinod | Bird Seed | 5.0 | None |

```
costs=df['Cost']
costs
```

```
Store1    22.5
Store1     2.5
Store2     5.0
Name: Cost, dtype: float64
```

저장이 완료되었습니다.

```
df
```

| | Name | Item Purchased | Cost | Location |
|---|---|---|---|---|
| **Store1** | Chris | Dog Food | 24.5 | None |
| **Store1** | Kevin | Kitty Litter | 4.5 | None |
| **Store2** | Vinod | Bird Seed | 7.0 | None |

```
from google.colab import drive
drive.mount('/content/gdrive')
!cat '/content/gdrive/My Drive/MyFile/olympics.csv'
```

```
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/
0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
,№ Summer,01 !,02 !,03 !,Total,№ Winter,01 !,02 !,03 !,Total,№ Games,01 !,02 !,03 !,Combin
Afghanistan (AFG),13,0,0,2,2,0,0,0,0,0,13,0,0,2,2
Algeria (ALG),12,5,2,8,15,3,0,0,0,0,15,5,2,8,15
Argentina (ARG),23,18,24,28,70,18,0,0,0,0,41,18,24,28,70
Armenia (ARM),5,1,2,9,12,6,0,0,0,0,11,1,2,9,12
Australasia (ANZ) [ANZ],2,3,4,5,12,0,0,0,0,0,2,3,4,5,12
Australia (AUS) [AUS] [Z],25,139,152,177,468,18,5,3,4,12,43,144,155,181,480
Austria (AUT),26,18,33,35,86,22,59,78,81,218,48,77,111,116,304
Azerbaijan (AZE),5,6,5,15,26,5,0,0,0,0,10,6,5,15,26
Bahamas (BAH),15,5,2,5,12,0,0,0,0,0,15,5,2,5,12
Bahrain (BRN),8,0,0,1,1,0,0,0,0,0,8,0,0,1,1
Barbados (BAR) [BAR],11,0,0,1,1,0,0,0,0,0,11,0,0,1,1
Belarus (BLR),5,12,24,39,75,6,6,4,5,15,11,18,28,44,90
Belgium (BEL),25,37,52,53,142,20,1,1,3,5,45,38,53,56,147
Bermuda (BER),17,0,0,1,1,7,0,0,0,0,24,0,0,1,1
Bohemia (BOH) [BOH] [Z],3,0,1,3,4,0,0,0,0,0,3,0,1,3,4
Botswana (BOT),9,0,1,0,1,0,0,0,0,0,9,0,1,0,1
Brazil (BRA),21,23,30,55,108,7,0,0,0,0,28,23,30,55,108
British West Indies (BWI) [BWI],1,0,0,2,2,0,0,0,0,0,1,0,0,2,2
Bulgaria (BUL) [H],19,51,85,78,214,19,1,2,3,6,38,52,87,81,220
Burundi (BDI),5,1,0,0,1,0,0,0,0,0,5,1,0,0,1
Cameroon (CMR),13,3,1,1,5,1,0,0,0,0,14,3,1,1,5
Canada (CAN),25,59,99,121,279,22,62,56,52,170,47,121,155,173,449
Chile (CHI) [I],22,2,7,4,13,16,0,0,0,0,38,2,7,4,13
China (CHN) [CHN],9,201,146,126,473,10,12,22,19,53,19,213,168,145,526
Colombia (COL),18,2,6,11,19,1,0,0,0,0,19,2,6,11,19
Costa Rica (CRC),14,1,1,2,4,6,0,0,0,0,20,1,1,2,4
Ivory Coast (CIV) [CIV],13,0,1,0,1,0,0,0,0,0,12,0,1,0,1
                                                    11,13,10,13,11,34
                                                    ,0,0,0,19,72,67,70,209
Cyprus (CYP),9,0,1,0,1,10,0,0,0,0,19,0,1,0,1
Czech Republic (CZE) [CZE],5,14,15,15,44,6,7,9,8,24,11,21,24,23,68
Czechoslovakia (TCH) [TCH],16,49,49,45,143,16,2,8,15,25,32,51,57,60,168
Denmark (DEN) [Z],26,43,68,68,179,13,0,1,0,1,39,43,69,68,180
Djibouti (DJI) [B],7,0,0,1,1,0,0,0,0,0,7,0,0,1,1
Dominican Republic (DOM),13,3,2,1,6,0,0,0,0,0,13,3,2,1,6
Ecuador (ECU),13,1,1,0,2,0,0,0,0,0,13,1,1,0,2
Egypt (EGY) [EGY] [Z],21,7,9,10,26,1,0,0,0,0,22,7,9,10,26
Eritrea (ERI),4,0,0,1,1,0,0,0,0,0,4,0,0,1,1
Estonia (EST),11,9,9,15,33,9,4,2,1,7,20,13,11,16,40
Ethiopia (ETH),12,21,7,17,45,2,0,0,0,0,14,21,7,17,45
Finland (FIN),24,101,84,117,302,22,42,62,57,161,46,143,146,174,463
France (FRA) [O] [P] [Z],27,202,223,246,671,22,31,31,47,109,49,233,254,293,780
Gabon (GAB),9,0,1,0,1,0,0,0,0,0,9,0,1,0,1
Georgia (GEO),5,6,5,14,25,6,0,0,0,0,11,6,5,14,25
Germany (GER) [GER] [Z],15,174,182,217,573,11,78,78,53,209,26,252,260,270,782
United Team of Germany (EUA) [EUA],3,28,54,36,118,3,8,6,5,19,6,36,60,41,137
East Germany (GDR) [GDR],5,153,129,127,409,6,39,36,35,110,11,192,165,162,519
West Germany (FRG) [FRG],5,56,67,81,204,6,11,15,13,39,11,67,82,94,243
Ghana (GHA) [GHA],13,0,1,3,4,1,0,0,0,0,14,0,1,3,4
Great Britain (GBR) [GBR] [Z],27,236,272,272,780,22,10,4,12,26,49,246,276,284,806
Greece (GRE) [Z],27,30,42,39,111,18,0,0,0,0,45,30,42,39,111
Grenada (GRN),8,1,0,0,1,0,0,0,0,0,8,1,0,0,1
Guatemala (GUA),13,0,1,0,1,1,0,0,0,0,14,0,1,0,1
Guyana (GUY) [GUY],16,0,0,1,1,0,0,0,0,0,16,0,0,1,1
Haiti (HAI) [J],14,0,1,1,2,0,0,0,0,0,14,0,1,1,2
```

저장이 완료되었습니다. ×

```
Hong Kong (HKG) [HKG],15,1,1,1,3,4,0,0,0,0,19,1,1,1,3
Hungary (HUN),25,167,144,165,476,22,0,2,4,6,47,167,146,169,482
Iceland (ISL),19,0,2,2,4,17,0,0,0,0,36,0,2,2,4
India (IND) [F],23,9,6,11,26,9,0,0,0,0,32,9,6,11,26
Indonesia (INA),14,6,10,11,27,0,0,0,0,0,14,6,10,11,27
Iran (IRI) [K],15,15,20,25,60,10,0,0,0,0,25,15,20,25,60
Iraq (IRQ),13,0,0,1,1,0,0,0,0,0,13,0,0,1,1
Ireland (IRL),20,9,8,12,29,6,0,0,0,0,26,9,8,12,29
Israel (ISR),15,1,1,5,7,6,0,0,0,0,21,1,1,5,7
Italy (ITA) [M] [S],26,198,166,185,549,22,37,34,43,114,48,235,200,228,663
Jamaica (JAM) [JAM],16,17,30,20,67,7,0,0,0,0,23,17,30,20,67
Japan (JPN),21,130,126,142,398,20,10,17,18,45,41,140,143,160,443
Kazakhstan (KAZ),5,16,17,19,52,6,1,3,3,7,11,17,20,22,59
Kenya (KEN),13,25,32,29,86,3,0,0,0,0,16,25,32,29,86
North Korea (PRK),9,14,12,21,47,8,0,1,1,2,17,14,13,22,49
South Korea (KOR),16,81,82,80,243,17,26,17,10,53,33,107,99,90,296
Kuwait (KUW),12,0,0,2,2,0,0,0,0,0,12,0,0,2,2
Kyrgyzstan (KGZ),5,0,1,2,3,6,0,0,0,0,11,0,1,2,3
Latvia (LAT),10,3,11,5,19,10,0,4,3,7,20,3,15,8,26
Lebanon (LIB),16,0,2,2,4,16,0,0,0,0,32,0,2,2,4
Liechtenstein (LIE),16,0,0,0,0,18,2,2,5,9,34,2,2,5,9
Lithuania (LTU),8,6,5,10,21,8,0,0,0,0,16,6,5,10,21
Luxembourg (LUX) [O],22,1,1,0,2,8,0,2,0,2,30,1,3,0,4
Macedonia (MKD),5,0,0,1,1,5,0,0,0,0,10,0,0,1,1
Malaysia (MAS) [MAS],12,0,3,3,6,0,0,0,0,0,12,0,3,3,6
Mauritius (MRI),8,0,0,1,1,0,0,0,0,0,8,0,0,1,1
Mexico (MEX),22,13,21,28,62,8,0,0,0,0,30,13,21,28,62
Moldova (MDA),5,0,2,5,7,6,0,0,0,0,11,0,2,5,7
Mongolia (MGL),12,2,9,13,24,13,0,0,0,0,25,2,9,13,24
Montenegro (MNE) 2,0,1,0,1,2,0,0,0,0,4,0,1,0,1
                                                ,0,19,6,5,11,22
                                                ,0,9,1,0,1,2
Namibia (NAM),6,0,4,0,4,0,0,0,0,0,6,0,4,0,4
Netherlands (NED) [Z],25,77,85,104,266,20,37,38,35,110,45,114,123,139,376
Netherlands Antilles (AHO) [AHO] [I],13,0,1,0,1,2,0,0,0,0,15,0,1,0,1
New Zealand (NZL) [NZL],22,42,18,39,99,15,0,1,0,1,37,42,19,39,100
Niger (NIG),11,0,0,1,1,0,0,0,0,0,11,0,0,1,1
Nigeria (NGR),15,3,8,12,23,0,0,0,0,0,15,3,8,12,23
Norway (NOR) [Q],24,56,49,43,148,22,118,111,100,329,46,174,160,143,477
Pakistan (PAK),16,3,3,4,10,2,0,0,0,0,18,3,3,4,10
Panama (PAN),16,1,0,2,3,0,0,0,0,0,16,1,0,2,3
Paraguay (PAR),11,0,1,0,1,1,0,0,0,0,12,0,1,0,1
Peru (PER) [L],17,1,3,0,4,2,0,0,0,0,19,1,3,0,4
Philippines (PHI),20,0,2,7,9,4,0,0,0,0,24,0,2,7,9
Poland (POL),20,64,82,125,271,22,6,7,7,20,42,70,89,132,291
Portugal (POR),23,4,8,11,23,7,0,0,0,0,30,4,8,11,23
Puerto Rico (PUR),17,0,2,6,8,6,0,0,0,0,23,0,2,6,8
Qatar (QAT),8,0,0,4,4,0,0,0,0,0,8,0,0,4,4
Romania (ROU),20,88,94,119,301,20,0,0,1,1,40,88,94,120,302
Russia (RUS) [RUS],5,132,121,142,395,6,49,40,35,124,11,181,161,177,519
Russian Empire (RU1) [RU1],3,1,4,3,8,0,0,0,0,0,3,1,4,3,8
Soviet Union (URS) [URS],9,395,319,296,1010,9,78,57,59,194,18,473,376,355,1204
Unified Team (EUN) [EUN],1,45,38,29,112,1,9,6,8,23,2,54,44,37,135
Saudi Arabia (KSA),10,0,1,2,3,0,0,0,0,0,10,0,1,2,3
Senegal (SEN),13,0,1,0,1,5,0,0,0,0,18,0,1,0,1
Serbia (SRB) [SRB],3,1,2,4,7,2,0,0,0,0,5,1,2,4,7
Serbia and Montenegro (SCG) [SCG],3,2,4,3,9,3,0,0,0,0,6,2,4,3,9
Singapore (SIN),15,0,2,2,4,0,0,0,0,0,15,0,2,2,4
```

저장이 완료되었습니다.

```
Slovakia (SVK) [SVK],5,7,9,8,24,6,2,2,1,5,11,9,11,9,29
Slovenia (SLO),6,4,6,9,19,7,2,4,9,15,13,6,10,18,34
South Africa (RSA),18,23,26,27,76,6,0,0,0,0,24,23,26,27,76
Spain (ESP) [Z],22,37,59,35,131,19,1,0,1,2,41,38,59,36,133
Sri Lanka (SRI) [SRI],16,0,2,0,2,0,0,0,0,0,16,0,2,0,2
Sudan (SUD),11,0,1,0,1,0,0,0,0,0,11,0,1,0,1
Suriname (SUR) [E],11,1,0,1,2,0,0,0,0,0,11,1,0,1,2
Sweden (SWE) [Z],26,143,164,176,483,22,50,40,54,144,48,193,204,230,627
Switzerland (SUI),27,47,73,65,185,22,50,40,48,138,49,97,113,113,323
Syria (SYR),12,1,1,1,3,0,0,0,0,0,12,1,1,1,3
Chinese Taipei (TPE) [TPE] [TPE2],13,2,7,12,21,11,0,0,0,0,24,2,7,12,21
Tajikistan (TJK),5,0,1,2,3,4,0,0,0,0,9,0,1,2,3
Tanzania (TAN) [TAN],12,0,2,0,2,0,0,0,0,0,12,0,2,0,2
Thailand (THA),15,7,6,11,24,3,0,0,0,0,18,7,6,11,24
Togo (TOG),9,0,0,1,1,1,0,0,0,0,10,0,0,1,1
Tonga (TGA),8,0,1,0,1,1,0,0,0,0,9,0,1,0,1
Trinidad and Tobago (TRI) [TRI],16,2,5,11,18,3,0,0,0,0,19,2,5,11,18
Tunisia (TUN),13,3,3,4,10,0,0,0,0,0,13,3,3,4,10
Turkey (TUR),21,39,25,24,88,16,0,0,0,0,37,39,25,24,88
Uganda (UGA),14,2,3,2,7,0,0,0,0,0,14,2,3,2,7
Ukraine (UKR),5,33,27,55,115,6,2,1,4,7,11,35,28,59,122
United Arab Emirates (UAE),8,1,0,0,1,0,0,0,0,0,8,1,0,0,1
United States (USA) [P] [Q] [R] [Z],26,976,757,666,2399,22,96,102,84,282,48,1072,859,750,2681
Uruguay (URU),20,2,2,6,10,1,0,0,0,0,21,2,2,6,10
Uzbekistan (UZB),5,5,5,10,20,6,1,0,0,1,11,6,5,10,21
Venezuela (VEN),17,2,2,8,12,4,0,0,0,0,21,2,2,8,12
Vietnam (VIE),14,0,2,0,2,0,0,0,0,0,14,0,2,0,2
Virgin Islands (ISV),11,0,1,0,1,7,0,0,0,0,18,0,1,0,1
Yugoslavia (YUG) [YUG],16,26,29,28,83,14,0,3,1,4,30,26,32,29,87
Independent Olympic Participants (IOP) [IOP],1,0,1,2,3,0,0,0,0,0,1,0,1,2,3
                                              0,0,0,12,0,1,1,2
                                              0,0,0,0,13,3,4,1,8
Mixed Team (ZZX) [ZZX],5,6,5,4,17,0,0,0,0,0,3,8,5,4,17
Totals,27,4809,4775,5130,14714,22,959,958,948,2865,49,5768,5733,6078,17579
```

저장이 완료되었습니다.

```
df = pd.read_csv('/content/gdrive/My Drive/MyFile/olympics.csv')
df.head()
```

```
d.read_csv('/content/gdrive/My Drive/MyFile/olympics.csv',index_col=0, skiprows=1)
```

```
df.head()
```

| | № Summer | 01 ! | 02 ! | 03 ! | Total | № Winter | 01 !.1 | 02 !.1 | 03 !.1 | Total.1 | № Games |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Afghanistan (AFG)** | 13 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 13 |
| **Algeria (ALG)** | 12 | 5 | 2 | 8 | 15 | 3 | 0 | 0 | 0 | 0 | 15 |
| **Argentina (ARG)** | 23 | 18 | 24 | 28 | 70 | 18 | 0 | 0 | 0 | 0 | 41 |
| **Armenia (ARM)** | 5 | 1 | 2 | 9 | 12 | 6 | 0 | 0 | 0 | 0 | 11 |
| **Australasia (ANZ) [ANZ]** | 2 | 3 | 4 | 5 | 12 | 0 | 0 | 0 | 0 | 0 | 2 |

```
df.columns
```

```
Index(['№ Summer', '01 !', '02 !', '03 !', 'Total', '№ Winter', '01 !.1',
       '02 !.1', '03 !.1', 'Total.1', '№ Games', '01 !.2', '02 !.2', '03 !.2',
       'Combined total'],
      dtype='object')
```

```
for col in df.columns:
    if col[:2]=='01':
        df.rename(columns={col:'Gold'+col[4:]}, inplace=True)
    if col[:2]=='02':
        df.rename(columns={col:'Silver'+col[4:]}, inplace=True)
    if col[:2]=='03':
        df.rename(columns={col:'Bronze'+col[4:]}, inplace=True)
    if col[:1]=='№':
        df.rename(columns={col:'#'+col[4:]}, inplace=True)
df.head()
```

저장이 완료되었습니다.

| | # Summer | Gold | Silver | Bronze | Total | # Winter | Gold.1 | Silver.1 | Br |
|---|---|---|---|---|---|---|---|---|---|
| **Afghanistan (AFG)** | 13 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | |
| **Algeria (ALG)** | 12 | 5 | 2 | 8 | 15 | 3 | 0 | 0 | |
| **Argentina (ARG)** | 23 | 18 | 24 | 28 | 70 | 18 | 0 | 0 | |
| **Armenia (ARM)** | 5 | 1 | 2 | 9 | 12 | 6 | 0 | 0 | |
| **Australasia (ANZ) [ANZ]** | 2 | 3 | 4 | 5 | 12 | 0 | 0 | 0 | |

```
#Querying a Dataframe
```

```
df['Gold']>0
```

```
Afghanistan (AFG)                                 False
Algeria (ALG)                                      True
Argentina (ARG)                                    True
Armenia (ARM)                                      True
Australasia (ANZ) [ANZ]                            True
                                                  ...
Independent Olympic Participants (IOP) [IOP]      False
Zambia (ZAM) [ZAM]                                False
Zimbabwe (ZIM) [ZIM]                               True
Mixed team (ZZX) [ZZX]                             True
Totals                                             True
Name: Gold, Length: 147, dtype: bool
```

```
only_gold=df.where(df['Gold']>0)
only_gold.head()
```

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Br |
|---|---|---|---|---|---|---|---|---|---|
| **Afghanistan (AFG)** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | |
| **Algeria (ALG)** | 12.0 | 5.0 | 2.0 | 8.0 | 15.0 | 3.0 | 0.0 | 0.0 | |
| **Argentina (ARG)** | 22.0 | 18.0 | 24.0 | 28.0 | 70.0 | 18.0 | 0.0 | 0.0 | |
| **Armenia (ARM)** | 11.0 | 1.0 | 2.0 | 9.0 | 12.0 | 6.0 | 0.0 | 0.0 | |
| **Australasia (ANZ) [ANZ]** | 2.0 | 3.0 | 4.0 | 5.0 | 12.0 | 0.0 | 0.0 | 0.0 | |

저장이 완료되었습니다.

```
only_gold['Gold'].count()
```

```
100
```

```
df.count()
```

```
№ Summer    147
Gold        147
Silver      147
Bronze      147
```

```
df['Gold'].count()
```

```
147
```

```
only_gold.head()
```

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Br |
|---|---|---|---|---|---|---|---|---|---|
| **Afghanistan (AFG)** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | |
| **Algeria (ALG)** | 12.0 | 5.0 | 2.0 | 8.0 | 15.0 | 3.0 | 0.0 | 0.0 | |
| **Argentina (ARG)** | 23.0 | 18.0 | 24.0 | 28.0 | 70.0 | 18.0 | 0.0 | 0.0 | |
| **Armenia (ARM)** | 5.0 | 1.0 | 2.0 | 9.0 | 12.0 | 6.0 | 0.0 | 0.0 | |
| **Australasia (ANZ) [ANZ]** | 2.0 | 3.0 | 4.0 | 5.0 | 12.0 | 0.0 | 0.0 | 0.0 | |

```
only_gold=only_gold.dropna()# NaN제거
only_gold.head()
```

저장이 완료되었습니다.

| | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Bro |
|---|---|---|---|---|---|---|---|
| **Algeria (ALG)** | 2.0 | 8.0 | 15.0 | 3.0 | 0.0 | 0.0 | |
| **Argentina (ARG)** | 24.0 | 28.0 | 70.0 | 18.0 | 0.0 | 0.0 | |
| **Armenia (ARM)** | 2.0 | 9.0 | 12.0 | 6.0 | 0.0 | 0.0 | |
| **Australasia (ANZ) [ANZ]** | 4.0 | 5.0 | 12.0 | 0.0 | 0.0 | 0.0 | |
| **Australia (AUS) [AUS] [Z]** | 152.0 | 177.0 | 468.0 | 18.0 | 5.0 | 3.0 | |

```
only_gold=df[df['Gold']>0]#boolean Mask
only_gold.head()
```

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Bro |
|---|---|---|---|---|---|---|---|---|---|
| **Algeria (ALG)** | 12 | 5 | 2 | 8 | 15 | 3 | 0 | 0 | |
| **Argentina (ARG)** | 23 | 18 | 24 | 28 | 70 | 18 | 0 | 0 | |
| **Armenia (ARM)** | 5 | 1 | 2 | 9 | 12 | 6 | 0 | 0 | |
| **Australasia (ANZ) [ANZ]** | 2 | 3 | 4 | 5 | 12 | 0 | 0 | 0 | |
| **Australia (AUS) [AUS] [Z]** | 25 | 139 | 152 | 177 | 468 | 18 | 5 | 3 | |

```
both_gold= df[(df['Gold']>0) & (df['Gold.1']>0)]
both_gold.head()
```

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Bronz |
|---|---|---|---|---|---|---|---|---|---|
| **Australia (AUS) [AUS] [Z]** | 25 | 139 | 152 | 177 | 468 | 18 | 5 | 3 | |
| | | | 33 | 35 | 86 | 22 | 59 | 78 | |
| | | | 24 | 39 | 75 | 6 | 6 | 4 | |
| **Belgium (BEL)** | 25 | 37 | 52 | 53 | 142 | 20 | 1 | 1 | |
| **Bulgaria (BUL) [H]** | 19 | 51 | 85 | 78 | 214 | 19 | 1 | 2 | |

저장이 완료되었습니다.

```
len(df[(df['Gold']>0) | (df['Gold.1']>0)])
#하계 혹은 동계 에서 금메달을 딴 나라 갯수
```

```
101
```

```
df[(df['Gold.1']>0) & (df['Gold']==0)]#하계 o 동계 x
```

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Bro |
|---|---|---|---|---|---|---|---|---|---|
| **Liechtenstein (LIE)** | 16 | 0 | 0 | 0 | 0 | 18 | 2 | 2 | |

```
# Indexing DataFrames
```

```
df.head()
```

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Br |
|---|---|---|---|---|---|---|---|---|---|
| **Afghanistan (AFG)** | 13 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | |
| **Algeria (ALG)** | 12 | 5 | 2 | 8 | 15 | 3 | 0 | 0 | |
| **Argentina (ARG)** | 23 | 18 | 24 | 28 | 70 | 18 | 0 | 0 | |
| **Armenia (ARM)** | 5 | 1 | 2 | 9 | 12 | 6 | 0 | 0 | |
| **Australasia (ANZ) [ANZ]** | 2 | 3 | 4 | 5 | 12 | 0 | 0 | 0 | |

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

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Br |
|---|---|---|---|---|---|---|---|---|---|
| **Afghanistan (AFG)** | 13 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | |
| **Algeria (ALG)** | 12 | 5 | 2 | 8 | 15 | 3 | 0 | 0 | |
| **Argentina (ARG)** | 23 | 18 | 24 | 28 | 70 | 18 | 0 | 0 | |
| **Armenia (ARM)** | 5 | 1 | 2 | 9 | 12 | 6 | 0 | 0 | |
| **Australasia (ANZ) [ANZ]** | 2 | 3 | 4 | 5 | 12 | 0 | 0 | 0 | |

저장이 완료되었습니다.

```
df=df.set_index('Combined total')
df.head()
```

| Combined total | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Bronze.1 |
|---|---|---|---|---|---|---|---|---|---|
| **2** | 13 | 0 | 0 | 2 | 2 | 0 | 0 | 0 | 0 |
| **15** | 12 | 5 | 2 | 8 | 15 | 3 | 0 | 0 | 0 |
| **70** | 23 | 18 | 24 | 28 | 70 | 18 | 0 | 0 | 0 |
| **12** | 5 | 1 | 2 | 9 | 12 | 6 | 0 | 0 | 0 |
| **12** | 2 | 3 | 4 | 5 | 12 | 0 | 0 | 0 | 0 |

```
df=df.sort_index()
df
```

| | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Bronze.1 | Combined total |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 13 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | |
| **1** | 13 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | |
| **1** | 11 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | |
| **1** | 9 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | |
| **1** | 16 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **782** | 15 | 174 | 182 | 217 | 573 | 11 | 78 | 78 | 53 | |
| **806** | 27 | 236 | 272 | 272 | 780 | 22 | 10 | 4 | 12 | |
| | | | | 9 | 296 | 1010 | 9 | 78 | 57 | 59 |
| **2681** | 26 | 976 | 757 | 666 | 2399 | 22 | 96 | 102 | 84 | |
| **17579** | 27 | 4809 | 4775 | 5130 | 14714 | 22 | 959 | 958 | 948 | |

147 rows × 15 columns

저장이 완료되었습니다.

```
df.head()
```

```
                  № Summer  Gold  Silver  Bronze  Total  № Winter  Gold.1  Silver.1  Bronze.1  Combined
```

```
df=df.reset_index()
df.head()
```

| | Combined total | № Summer | Gold | Silver | Bronze | Total | № Winter | Gold.1 | Silver.1 | Bronze. |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 13 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | |
| **1** | 1 | 13 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | |
| **2** | 1 | 11 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | |
| **3** | 1 | 9 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | |
| **4** | 1 | 16 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | |

```
#복합 키
purchase_1=pd.Series({'Name':'Chris',
                      'Item Purchased':'Dog Food',
                      'Cost':22.50})
purchase_2=pd.Series({'Name':'Kevin',
                      'Item Purchased':'Kitty Litter',
                      'Cost':2.50})
```

저장이 완료되었습니다.

```
                      'Bird Seed',
df=pd.DataFrame([purchase_1,purchase_2,purchase_3],
                index=['Store1','Store1','Store2'])
df.head()
```

| | Name | Item Purchased | Cost |
|---|---|---|---|
| **Store1** | Chris | Dog Food | 22.5 |
| **Store1** | Kevin | Kitty Litter | 2.5 |
| **Store2** | Vinod | Bird Seed | 5.0 |

```
df['store']= df.index
df
```