

## # 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
```

```
animals=['Tigers ','Bear ',None]
pd.Series(animals)
```

```
↳ 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
   dtype: object
```

```
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'
```

```

sports={5: 'Bhutan',
        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

```
s.iloc[0]
```

```
↳ '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
   4      916
dtype: int64
```

```
%%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

```

```

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

```

```
cricket_countries
```

```

Cricket      Australia
Cricket      Pakistan
Cricket      England
dtype: object

```

```
all_countries=s.append(cricket_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
<b>Store1</b>	Chris	Dog Food	22.5
<b>Store1</b>	Kevin	Kitty Litter	2.5
<b>Store2</b>	Vinod	Bird Seed	5.0

```
df.loc['Store2']
```

```


```

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
<b>Store1</b>	Chris	Dog Food	22.5
<b>Store1</b>	Kevin	Kitty Litter	2.5

```
df.loc['Store1']['Cost']# 22.5
```

```
df.loc['Store1']['Cost']#미요플
```

```
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
```

```
df['Cost']
```

```
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
<b>Store1</b>	Chris		Dog Food	22.5
<b>Store1</b>	Kevin		Kitty Litter	2.5
<b>Store2</b>	Vinod		Bird Seed	5.0

df\_copy=df\_copy.drop('Store1')

df\_copy

↗

	Name	Item	Purchased	Cost
<b>Store2</b>	Vinod		Bird Seed	5.0

del df\_copy['Name']

df\_copy

↗

	Item	Purchased	Cost
<b>Store2</b>		Bird Seed	5.0

df['Location']=None

df

↗

	Name	Item	Purchased	Cost	Location
<b>Store1</b>	Chris		Dog Food	22.5	None
<b>Store1</b>	Kevin		Kitty Litter	2.5	None
<b>Store2</b>	Vinod		Bird Seed	5.0	None

