

DS670-Class6





default ▾

FINISHED ▶ ⌵ 📖 ⚙

```
%pyspark
from pandas import Series, DataFrame
import numpy as np, pandas as pd
df = DataFrame([[1.4,np.nan],[7.1,-4.5],
                [np.nan,np.nan],[0.75,-1.3]],
                index=['a','b','c','d'],
                columns=['one','two'])
```

Took 0 sec. Last updated by anonymous at February 23 2017, 6:47:42 PM.



FINISHED ▶ ⌵ 📖 ⚙

```
%pyspark
from pandas import Series, DataFrame
import numpy as np, pandas as pd
df
```

```
   one  two
a  1.40 NaN
b  7.10 -4.5
c   NaN NaN
d  0.75 -1.3
```

Took 1 sec. Last updated by anonymous at February 23 2017, 6:57:49 PM.

FINISHED ▶ ⌵ 📖 ⚙

```
%pyspark
from pandas import Series, DataFrame
import numpy as np, pandas as pd
df.sum()
```

```
one    9.25
two   -5.80
dtype: float64
```

Took 0 sec. Last updated by anonymous at February 23 2017, 6:58:13 PM.



FINISHED ▶ ⌵ 📖 ⚙

```
%pyspark
df.sum(axis=1)
```

```
a    1.40
b    2.60
c    0.00
d   -0.55
dtype: float64
```

Took 0 sec. Last updated by anonymous at February 23 2017, 6:59:48 PM.

FINISHED ▶ ⌵ 📖 ⚙

```
%pyspark
df.idxmax()
```

```
one    b
```



DS670-Class6UntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitledUntitled

Zeppelin

Took 0 sec. Last updated by anonymous at February 23 2017, 7:14:41 PM.

DS670-Class6



default

FINISHED

```
%pyspark
df
df.sum()
df.sum(axis=1)
df.mean(axis=1, skipna=False)
df.idxmax()
df.describe()
obj = Series(['a', 'a', 'b', 'c'] * 4)
obj
obj.describe()
```

```
count      16
unique       3
top         a
freq         8
dtype: object
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:16:24 PM.

```
%pyspark
from pandas_datareader import data as web
all_data = {}
```

FINISHED

Took 2 sec. Last updated by anonymous at February 23 2017, 7:22:23 PM.

```
%pyspark
from pandas_datareader import data, wb
all_data = {}
for ticker in ['AAPL', 'IBM', 'MSFT', 'GOOG']:
    all_data[ticker] = web.get_data_yahoo(ticker)
price = DataFrame({tic: data['Adj Close']
                    for tic, data in all_data.items()})
volume = DataFrame({tic: data['Volume']
                    for tic, data in all_data.items()})
```

FINISHED

Took 1 sec. Last updated by anonymous at February 23 2017, 7:25:03 PM.

```
%pyspark
returns = price.pct_change()
returns.tail()
```

FINISHED

	AAPL	GOOG	IBM	MSFT
Date				
2017-02-15	0.003629	-0.001792	0.008605	-0.000619
2017-02-16	-0.001181	0.006325	-0.001376	-0.000155
2017-02-17	0.002734	0.004744	-0.004189	0.001550
2017-02-21	0.007221	0.004335	-0.002269	-0.002012
2017-02-22	0.002999	-0.001082	0.004937	-0.002016

Took 0 sec. Last updated by anonymous at February 23 2017, 7:25:35 PM.

```
%pyspark
```

FINISHED

DS670-Class6

Zeppelin



default ▼

Took 0 sec. Last updated by anonymous at February 23 2017, 7:25:51 PM.

FINISHED ▶ ↻ 📖 ⚙️

0.49515377802280919

1

Took 0 sec. Last updated by anonymous at February 23 2017, 7:26:42 PM.

FINISHED ▶ 🔍 📖 ⚙️

3.5977652563835427e-05

I

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:02 PM.

FINISHED ▶ ↻ 📖 ⚙️

	AAPL	GOOG	IBM	MSFT
AAPL	1.000000	0.409541	0.381549	0.388972
GOOG	0.409541	1.000000	0.402872	0.470820
IBM	0.381549	0.402872	1.000000	0.495154
MSFT	0.388972	0.470820	0.495154	1.000000

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:17 PM.

FINISHED ▶ 🔍 📖 ⚙️

	AAPL	GOOG	IBM	MSFT
AAPL	0.000270	0.000105	0.000075	0.000093
GOOG	0.000105	0.000244	0.000075	0.000107
IBM	0.000075	0.000075	0.000144	0.000086
MSFT	0.000093	0.000107	0.000086	0.000210

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:33 PM.

FINISHED ▶ 🔍 📖 ⚙️

AAPL	0.381549
GOOG	0.402872
IBM	1.000000
MSFT	0.495154

dtype: float64

Zeppelin

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:45 PM.

DS670-Class6

%pyspark

returns.corrwith(volume)

AAPL -0.074323
GOOG -0.009670
IBM -0.194432
MSFT -0.091017
dtype: float64

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:55 PM.



FINISHED



default



READY    

