Class_8_Capstone Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitl

Zeppelin

Class_8_Capstone

default

FINISHED

```
%pyspark
import pandas as pd
import numpy as np
```

Took 45 sec. Last updated by anonymous at March 09 2017, 11:29:34 PM.

FINISHED

```
%pyspark
df = pd.DataFrame({'Key1':['a','a','b','b','a'],
                   'Key2':['one','two','one','two','one'],
                   'data1':np.random.randn(5),
                   'data2':np.random.randn(5)
})
```

Took 28 sec. Last updated by anonymous at March 09 2017, 11:29:34 PM.

FINISHED

```
%pyspark
df
```

```
  Key1 Key2     data1     data2
0    a  one -0.685101 -1.112390
1    a  two  0.951374  1.871486
2    b  one  1.337829  1.187213
3    b  two  0.005543 -0.820762
4    a  one -0.331941  0.084290
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:29:39 PM.

FINISHED

```
%pyspark
grouped = df['data1'].groupby(df['Key1'])
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:29:43 PM.

FINISHED

```
%pyspark
grouped
```

```
<pandas.core.groupby.SeriesGroupBy object at 0x000000000600F400>
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:29:45 PM.

FINISHED

```
%pyspark
grouped.mean()
```

```
Key1
a   -0.021889
b    0.671686
Name: data1, dtype: float64
```

Took 1 sec. Last updated by anonymous at March 09 2017, 11:29:49 PM.

Class_8_Capstone

Class_8_Capstone Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitl

Zeppelin

Class_8_Capstone

default

```
%pyspark
means = df['data1'].groupby([df['Key1'],df['Key2']]).mean()
```

FINISHED

Took 0 sec. Last updated by anonymous at March 09 2017, 11:29:52 PM.

```
%pyspark
means
```

FINISHED

```
Key1  Key2
a     one    -0.508521
      two     0.951374
b     one     1.337829
      two     0.005543
Name: data1, dtype: float64
```

Took 1 sec. Last updated by anonymous at March 09 2017, 11:29:56 PM.

```
%pyspark
means.unstack()
```

FINISHED

```
Key2       one       two
Key1
a    -0.508521  0.951374
b     1.337829  0.005543
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:29:59 PM.

```
%pyspark
states = np.array(['ohio','california','california','ohio','ohio'])
```

FINISHED

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:02 PM.

```
%pyspark
years = np.array([2005,2005,2006,2005,2006])
```

FINISHED

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:07 PM.

```
%pyspark
df['data1'].groupby([states,years]).mean()
```

FINISHED

```
california  2005    0.951374
            2006    1.337829
ohio        2005   -0.339779
            2006   -0.331941
Name: data1, dtype: float64
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:09 PM.

```
%pyspark
df.groupby('Key1').mean()
```

FINISHED

Class_8_Capstone

Class_8_Capstone Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitl

Zeppelin

Class_8_Capstone

default

```
Key1
a    -0.021889  0.281129
b    -0.567196  0.083225
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:14 PM.

FINISHED

```
%pyspark
df.groupby(['Key1','Key2']).mean()
```

```
              data1     data2
Key1 Key2
a    one  -0.508521 -0.514050
     two   0.951374  1.871486
b    one   1.337829  1.187213
     two   0.005543 -0.820762
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:16 PM.

FINISHED

```
%pyspark
df.groupby(['Key1','Key2']).size()
```

```
Key1  Key2
a     one     2
      two     1
b     one     1
      two     1
dtype: int64
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:19 PM.

FINISHED

```
%pyspark
for name,group in df.groupby('Key1'):
    print name
    print group
```

```
a
  Key1 Key2     data1     data2
0    a  one -0.685101 -1.112390
1    a  two  0.951374  1.871486
4    a  one -0.331941  0.084290
b
  Key1 Key2     data1     data2
2    b  one  1.337829  1.187213
3    b  two  0.005543 -0.820762
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:23 PM.

FINISHED

```
%pyspark
for (K1,K2),group in df.groupby(['Key1','Key2']):
    print K1,K2
    print group
```

```
a one
```

Class_8_Capstone

Class_8_Capstone | Untitled | Untitled | Untitled | Untitled | Untitled | Untitled | Untitled | Untitled | Untitled | Untitl

Zeppelin

Class_8_Capstone

default

```
  Key1 Key2     data1     data2
4    a  one -0.331941  0.08429
a two
  Key1 Key2     data1     data2
1    a  two  0.951374  1.871486
b one
  Key1 Key2     data1     data2
2    b  one  1.337829  1.187213
b two
  Key1 Key2     data1     data2
3    b  two  0.005543 -0.820762
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:27 PM.

FINISHED

```
%pyspark
pieces = dict(list(df.groupby('Key1')))
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:30 PM.

FINISHED

```
%pyspark
pieces['b']
```

```
  Key1 Key2     data1     data2
2    b  one  1.337829  1.187213
3    b  two  0.005543 -0.820762
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:34 PM.

FINISHED

```
%pyspark
df.dtypes
```

```
Key1      object
Key2      object
data1    float64
data2    float64
dtype: object
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:37 PM.

FINISHED

```
%pyspark
grouped = df.groupby(df.dtypes,axis=1)
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:40 PM.

FINISHED

```
%pyspark
dict(list(grouped))
```

```
{dtype('O'):   Key1 Key2
0    a  one
1    a  two
2    b  one
3    b  two
4    a  one, dtype('float64'):      data1     data2
```

Class_8_Capstone

Class_8_Capstone Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitl

Zeppelin

Class_8_Capstone

default

Took 0 sec. Last updated by anonymous at March 09 2017, 11:30:43 PM.

FINISHED

```
%pyspark
df.groupby('Key1')['data1']
```

```
<pandas.core.groupby.SeriesGroupBy object at 0x0000000006087630>
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:31:00 PM.

FINISHED

```
%pyspark
df.groupby('Key1')[['data2']]
```

```
<pandas.core.groupby.DataFrameGroupBy object at 0x000000000629F470>
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:31:27 PM.

FINISHED

```
%pyspark
df['data1'].groupby(df['Key1'])
```

```
<pandas.core.groupby.SeriesGroupBy object at 0x0000000006087668>
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:31:51 PM.

FINISHED

```
%pyspark
df[['data2']].groupby(df['Key1'])
```

```
<pandas.core.groupby.DataFrameGroupBy object at 0x00000000060876D8>
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:32:14 PM.

FINISHED

```
%pyspark
df.groupby(['Key1', 'Key2'])[['data2']].mean()
```

```
             data2
Key1 Key2
a    one  -0.514050
     two   1.871486
b    one   1.187213
     two  -0.820762
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:32:47 PM.

FINISHED

```
%pyspark
s_grouped = df.groupby(['Key1', 'Key2'])['data2']

s_grouped
```

```
<pandas.core.groupby.SeriesGroupBy object at 0x0000000003E876A0>
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:33:15 PM.

Class_8_Capstone

Class_8_Capstone Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitl

Zeppelin

Class_8_Capstone default

```
Key1  Key2
a     one    -0.514555
      two     1.871486
b     one     1.187213
      two    -0.820762
Name: data2, dtype: float64
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:33:42 PM.

FINISHED

```
%pyspark
people = pd.DataFrame(np.random.randn(5, 5),
 columns=['a', 'b', 'c', 'd', 'e'],
 index=['Joe', 'Steve', 'Wes', 'Jim', 'Travis'])
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:34:21 PM.

FINISHED

```
%pyspark
people.ix[2:3, ['b', 'c']] = np.nan # Add a few NA values

people
```

```
               a         b         c         d         e
Joe     0.423312  1.651784  1.658499 -1.604033  1.900242
Steve   2.040601  1.454562  0.555265 -1.526755 -0.405573
Wes    -0.204819       NaN       NaN  1.343713  1.750284
Jim    -0.427145  0.800888 -0.462763  0.597108  1.080201
Travis  0.200884 -0.387432  0.697745  0.516584 -1.654361
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:35:42 PM.

FINISHED

```
%pyspark
mapping = {'a': 'red', 'b': 'red', 'c': 'blue',
 'd': 'blue', 'e': 'red', 'f' : 'orange'}

by_column = people.groupby(mapping, axis=1)

by_column.sum()
```

```
            blue       red
Joe     0.054466  3.975339
Steve  -0.971490  3.089590
Wes     1.343713  1.545466
Jim     0.134345  1.453945
Travis  1.214329 -1.840909
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:36:27 PM.

FINISHED

```
%pyspark
map_series = pd.Series(mapping)

map_series
```

```
a       red
b       red
c      blue
```

Class_8_Capstone

Class_8_Capstone Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitl

Zeppelin

Class_8_Capstone

default

```
d     blue
e      red
f    orange
dtype: object
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:37:04 PM.

FINISHED

```
%pyspark
people.groupby(map_series, axis=1).count()
```

```
        blue  red
Joe        2    3
Steve      2    3
Wes        1    2
Jim        2    3
Travis     2    3
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:37:46 PM.

FINISHED

```
%pyspark
people.groupby(len).sum()
```

```
          a         b         c         d         e
3 -0.208652  2.452673  1.195736  0.336788  4.730728
5  2.040601  1.454562  0.555265 -1.526755 -0.405573
6  0.200884 -0.387432  0.697745  0.516584 -1.654361
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:38:26 PM.

FINISHED

```
%pyspark
key_list = ['one', 'one', 'one', 'two', 'two']
people.groupby([len, key_list]).min()
```

```
              a         b         c         d         e
3 one -0.204819  1.651784  1.658499 -1.604033  1.750284
  two -0.427145  0.800888 -0.462763  0.597108  1.080201
5 one  2.040601  1.454562  0.555265 -1.526755 -0.405573
6 two  0.200884 -0.387432  0.697745  0.516584 -1.654361
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:39:20 PM.

FINISHED

```
%pyspark
columns = pd.MultiIndex.from_arrays([['US', 'US', 'US', 'JP', 'JP'],
 [1, 3, 5, 1, 3]], names=['cty', 'tenor'])
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:39:55 PM.

FINISHED

```
%pyspark
hier_df = pd.DataFrame(np.random.randn(4, 5), columns=columns)

hier_df
```

```
cty          US                            JP
tenor         1         3         5         1         3
0      0.608612 -0.527968  1.540790  0.648186 -1.020264
1      0.000860 -1.005398 -0.698190 -1.419141  0.540718
```

```
2      0.063638 -3.539274  0.333320 -0.109230  0.823649
[illegible] -0.028447 -1.355646  0.506471
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:40:37 PM.

FINISHED

```
%pyspark
hier_df.groupby(level='cty', axis=1).count()
```

```
cty  JP  US
0     2   3
1     2   3
2     2   3
3     2   3
```

Took 0 sec. Last updated by anonymous at March 09 2017, 11:41:22 PM.

READY