Zeppelin Notebook -

from pandas import Series, DataFrame

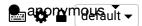
Lab 021617

import pandas as pd

%pyspark



Q



FINISHED ▷ ※ 圓 �

```
obj = Series([4,7,-5,3])
 obj
 obj.values
 obj.index
RangeIndex(start=0, stop=4, step=1)
                                                                          FINISHED ▷ 光 圓 墩
%pyspark
 obj2 = Series([4, 7, -5, 3], index=['d', 'b', 'a', 'c'])
 obj2.index
 obj2['a']
 obj2['d'] = 6
 obj2[['c', 'a', 'd']]
 obj2
d
     6
     7
b
    -5
а
     3
C
dtype: int64
```

```
d 12
b 14
a -10
c 6
dtype: int64
```

%pyspark

obj2 * 2

obj2[obj2 > 0]

FINISHED ▷ 光 圓 ��

```
import numpy as np
 np.exp(obj2)
 'b' in obj2
 'e' in obj2
 sdata = {'Ohio': 35000, 'Texas': 71000, 'Oregon': 16000, 'Utah': 5000}
 obj3 = Series(sdata)
 obj3
Ohio
          35000
0regon
          16000
Texas
          71000
           5000
Utah
dtype: int64
```

```
%pyspark
states = ['California', 'Ohio', 'Oregon', 'Texas']
obj4 = Series(sdata, index=states)
obj4

California NaN
Ohio 35000.0
Oregon 16000.0
Texas 71000.0
```

%pyspark
pd.isnull(obj4)
pd.notnull(obj4)
obj4.isnull()
print(obj3)
print(obj4)
print(obj3 + obj4)
obj4.name = 'population'
obj4.index.name = 'state'
print(obj4)

dtype: float64

FINISHED ▷ 💥 🗉 🕸

```
Ohio
          35000
0regon
          16000
Texas
          71000
           5000
Utah
dtype: int64
state
California
                  NaN
Ohio
              35000.0
              16000.0
0regon
Texas
              71000.0
Name: population, dtype: float64
California
                   NaN
0hio
               70000.0
               32000.0
0regon
Texas
              142000.0
Utah
                   NaN
dtype: float64
c+n+0
```

```
%pyspark
                                                                        FINISHED ▷ 光 圓 贷
 obj.index = ['Bob', 'Steve', 'Jeff', 'Ryan']
 print(obj)
Bob
        4
        7
Steve
Jeff
        -5
Ryan
        3
dtype: int64
        state year
  pop
0
  1.5
         Ohio 2000
  1.7
1
         Ohio 2001
2
 3.6
         Ohio 2002
3 2.4 Nevada 2001
4 2.9 Nevada 2002
array([[2000, 'Ohio', 1.5, nan],
       [2001, 'Ohio', 1.7, -1.2],
       [2002, 'Ohio', 3.6, nan],
       [2001, 'Nevada', 2.4, -1.5],
       [2002, 'Nevada', 2.9, -1.7]], dtype=object)
```

```
### Simples ###
```

```
print(frame2)
 frame2.columns
 frame2['state']
 frame2.year
 frame2.ix['three']
 frame2['debt'] = 16.5
 frame2
 frame2['debt'] = np.arange(5.)
 print(frame2)
 val = Series([-1.2, -1.5, -1.7], index=['two', 'four', 'five'])
 frame2['debt'] = val
 frame2
 frame2['eastern'] = frame2.state == 'Ohio'
 frame2
 del frame2['eastern']
 frame2.columns
 pop = {'Nevada': {2001: 2.4, 2002: 2.9},
        'Ohio': {2000: 1.5, 2001: 1.7, 2002: 3.6}}
 frame3 = DataFrame(pop)
 frame3
 frame3.T
 pdata = {'Ohio': frame3['Ohio'][:-1],
         'Nevada': frame3['Nevada'][:2]}
 DataFrame(pdata)
 frame3.index.name = 'year'; frame3.columns.name = 'state'
 frame3
 frame3.values
frama? valuas
                                                                                         1
        state year
   pop
  1.5
          Ohio 2000
0
  1.7
          Ohio 2001
1
2
  3.6
          Ohio 2002
3 2.4 Nevada 2001
  2.9 Nevada 2002
      year
             state pop debt
       2000
               Ohio 1.5 NaN
one
       2001
               Ohio 1.7 NaN
two
               Ohio 3.6 NaN
three
      2002
four
            Nevada 2.4
       2001
                          NaN
five
       2002
            Nevada 2.9 NaN
      year
             state pop
                         debt
       2000
               Ohio 1.5
                           0.0
one
       2001
               Ohio 1.7
                           1.0
two
      2002
               Ohio 3.6
                           2.0
three
four
       2001 Nevada 2.4
                           3.0
fiva
       つかかつ
                    2 a
                           1 A
            Novada
```