Lab 7 Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled Untitled U

**Zeppelin**

# Lab 7

default

READY

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

READY

```
%pyspark
df = pd.DataFrame({'key1': ['a','a','b','b','a'], 'key2' : ['one','two','one','two','one'], '
```

READY

```
%pyspark
df
```

```
      data1     data2 key1 key2
0  1.190083  2.016268    a  one
1  1.046512  1.580925    a  two
2  0.261075 -0.508882    b  one
3 -1.517274  1.170868    b  two
4 -1.096952  0.393116    a  one
```

READY

```
%pyspark
grouped = df['data1'].groupby(df['key1'])
```

READY

```
%pyspark
grouped
```

```
<pandas.core.groupby.SeriesGroupBy object at 0x7ffb5c811510>
```

READY

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

```
key1
a    0.379881
b   -0.628099
Name: data1, dtype: float64
```

READY

```
%pyspark
means = df['data1'].groupby([df['key1'],df['key2']]).mean()
```

Zeppelin

Lab 7

default

READY

```
%pyspark
means
```

```
key1  key2
a     one     0.046566
      two     1.046512
b     one     0.261075
      two    -1.517274
Name: data1, dtype: float64
```

READY

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

```
key2       one       two
key1
a     0.046566  1.046512
b     0.261075 -1.517274
```

READY

```
%pyspark
states = np.array(['Ohio','California','California','Ohio','Ohio'])
years = np.array([2005,2005,2006,2005,2006])
df['data1'].groupby([states,years]).mean()
```

```
California  2005    1.046512
            2006    0.261075
Ohio        2005   -0.163595
            2006   -1.096952
Name: data1, dtype: float64
```

READY

```
%pyspark
df.groupby('key1').mean()
```

```
         data1     data2
key1
a     0.379881  1.330103
b    -0.628099  0.330993
```

READY

```
%pyspark
df.groupby(['key1','key2']).mean()
```

```
              data1     data2
key1 key2
a    one   0.046566  1.204692
     two   1.046512  1.580925
b    one   0.261075 -0.508882
     two  -1.517274  1.170868
```

READY

```
%pyspark
```

```
df.groupby(['key1','key2']).size()
```

```
key1  key2
a     one     2
      two     1
b     one     1
      two     1
dtype: int64
```

READY

```
%pyspark
for name, group in df.groupby('key1'):
    print name
    print group
```

```
a
      data1     data2 key1 key2
0  1.190083  2.016268    a  one
1  1.046512  1.580925    a  two
4 -1.096952  0.393116    a  one
b
      data1     data2 key1 key2
2  0.261075 -0.508882    b  one
3 -1.517274  1.170868    b  two
```

READY

```
%pyspark
for (k1,k2), group in df.groupby(['key1','key2']):
    print k1, k2
    print group
```

```
a one
      data1     data2 key1 key2
0  1.190083  2.016268    a  one
4 -1.096952  0.393116    a  one
a two
      data1     data2 key1 key2
1  1.046512  1.580925    a  two
b one
      data1     data2 key1 key2
2  0.261075 -0.508882    b  one
b two
      data1     data2 key1 key2
3 -1.517274  1.170868    b  two
```

READY

```
%pyspark
pieces = dict(list(df.groupby('key1')))

pieces['b']
```

```
      data1     data2 key1 key2
2  0.261075 -0.508882    b  one
3 -1.517274  1.170868    b  two
```

READY

```
%pyspark
df.dtypes
```

```
data1    float64
data2    float64
key1      object
key2      object
dtype: object
```

READY

```
%pyspark
grouped = df.groupby(df.dtypes, axis=1)
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
0  1.190083  2.016268
1  1.046512  1.580925
2  0.261075 -0.508882
3 -1.517274  1.170868
4 -1.096952  0.393116}
```

READY