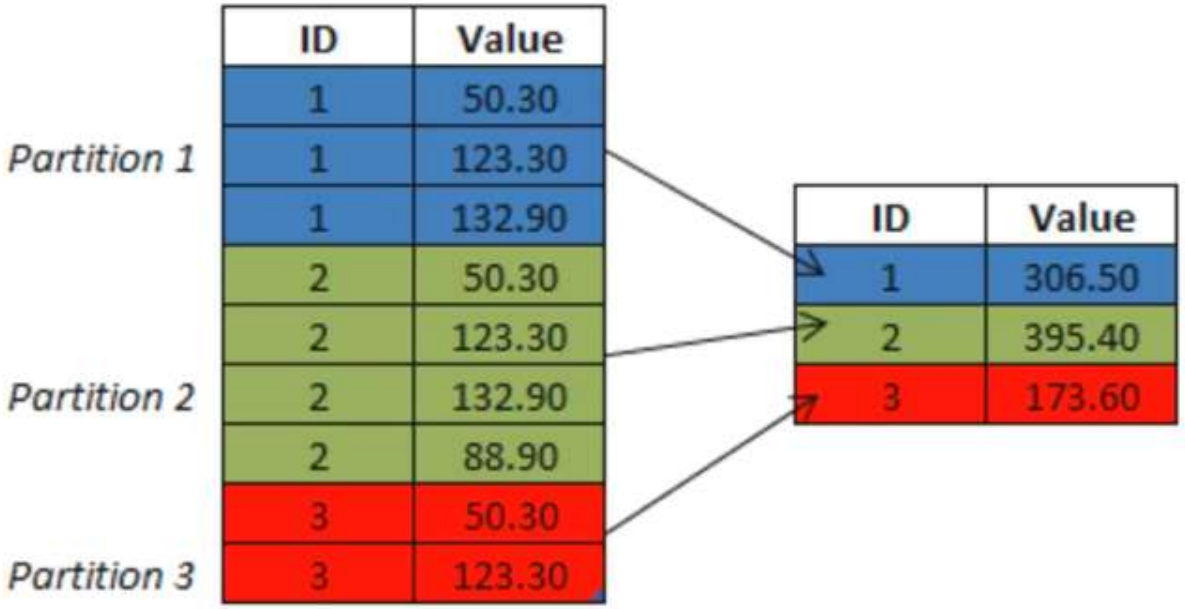
Python for Data Science

Groupby

Python for Data Science

- Groupby allows you to group together rows based off of a column and perform an aggregate function on them.



Python for Data Science

Let’s learn groupby with pandas!

Pandas - Groupby

In [84]:
```
import numpy as np
import pandas as pd
```

In [127]:
```
data = {'Company':['GOOG','GOOG','MSFT','MSFT','FB','FB'],
        'Person':['Sam','Charlie','Amy','Vanessa','Carl','Sarah'],
        'Sales':[200,120,340,124,243,350]}
```

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help Python [conda env:py35]

```
        'Person':['Sam','Charlie','Amy','Vanessa','Carl','Sarah'],
        'Sales':[200,120,340,124,243,350]}
```

In [128]:
```
df = pd.DataFrame(data)
```

In [129]:
```
df
```

Out[129]:

	Company	Person	Sales
0	GOOG	Sam	200
1	GOOG	Charlie	120
2	MSFT	Amy	340
3	MSFT	Vanessa	124
4	FB	Carl	243
5	FB	Sarah	350

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

5	FB	Sarah	350

In [131]:
```
byComp = df.groupby('Company')
```

In [133]:
```
byComp.mean()
```

Out[133]:

	Sales
Company	
FB	296.5
GOOG	160.0
MSFT	232.0

Automatically ignores any other non-number column.

In [ ]:

MSFT	232.0

In [134]:
```
byComp.sum()
```

Out[134]:

	Sales
Company	
FB	593
GOOG	320
MSFT	464

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help

Python [conda env:py35]

In [135]:
```
byComp.std()
```

Out[135]:

	Sales
Company	
FB	75.660426
GOOG	56.568542
MSFT	152.735065

In [ ]:

Company	
FB	75.660426
GOOG	56.568542
MSFT	152.735065

In [136]:
```
byComp.sum()
```

Out[136]:

	Sales
Company	
FB	593
GOOG	320
MSFT	464

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

Out[135]:

	Sales
Company	
FB	75.660426
GOOG	56.568542
MSFT	152.735065

In [137]:
```
byComp.sum().loc['FB']
```

Out[137]:
```
Sales    593
Name: FB, dtype: int64
```

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help Python [conda env:py35]

FB	75.660426
GOOG	56.568542
MSFT	152.735065

In [137]:
```
byComp.sum().loc['FB']
```

Out[137]:
```
Sales    593
Name: FB, dtype: int64
```

Everything in one line

In [138]:
```
df.groupby('Company').sum().loc['FB']
```

Out[138]:
```
Sales    593
Name: FB, dtype: int64
```

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

In [137]:
```
byComp.sum().loc['FB']
```

Out[137]:
```
Sales    593
Name: FB, dtype: int64
```

In [139]:
```
df.groupby('Company').count()
```

Out[139]:

	Person	Sales
Company		
FB	2	2
GOOG	2	2
MSFT	2	2

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

In [137]:
```
byComp.sum().loc['FB']
```

Out[137]:
```
Sales    593
Name: FB, dtype: int64
```

In [140]:
```
df.groupby('Company').max()
```

Out[140]:

	Person	Sales
Company		
FB	Saran	350
GOOG	Sam	200
MSFT	Vanessa	340

Max returns latest end of the alphabet.

In [ ]:

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

In [137]:
```
byComp.sum().loc['FB']
```

Out[137]:
```
Sales    593
Name: FB, dtype: int64
```

In [141]:
```
df.groupby('Company').min()
```

Out[141]:

	Person	Sales
Company		
FB	Carl	243
GOOG	Charlie	120
MSFT	Amy	124

Min returns latest begining of the alphabet.

In [ ]:

MSFT	Amy	124

In [142]:
```
df.groupby('Company').describe()
```

Out[142]:

		Sales
Company		
FB	**count**	2.000000
	mean	296.500000
	std	75.660426
	min	243.000000
	25%	269.750000
	50%	296.500000
	75%	323.250000
	max	350.000000

Pandas

localhost:8888/notebooks/Pandas.ipynb

File Edit View Insert Cell Kernel Widgets Help | Python [conda env:py35]

Out[141]:

	Person	Sales
Company		
FB	Carl	243
GOOG	Charlie	120
MSFT	Amy	124

In [143]:
```
df.groupby('Company').describe().transpose()
```

Out[143]:

Company	**FB**								**GOOG**			
	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**	**count**	**mean**	**...**	**75%**
Sales	2.0	296.5	75.660426	243.0	269.75	296.5	323.25	350.0	2.0	160.0	...	180

1 rows × 24 columns

In [ ]:

FB	Carl	243
GOOG	Charlie	120
MSFT	Amy	124

In [144]:
```
df.groupby('Company').describe().transpose()['FB']
```

Out[144]:

	count	mean	std	min	25%	50%	75%	max
Sales	2.0	296.5	75.660426	243.0	269.75	296.5	323.25	350.0

In [ ]: