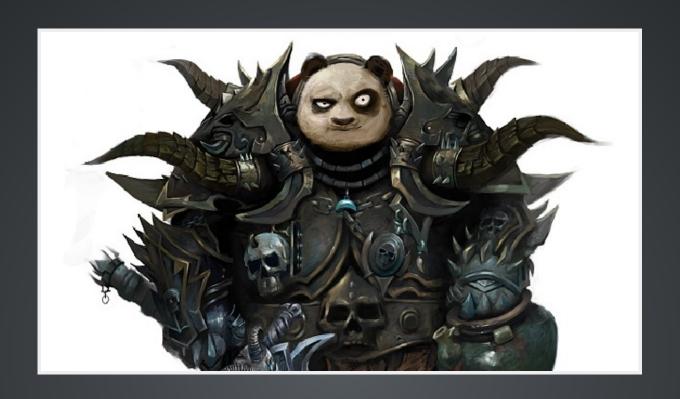


PANDAS A POWERFUL DATA MANIPULATION TOOL

Jason A Myers / @jasonamyers / Emma

WHAT'S SO SPECIAL ABOUT PANDAS?

- 1. Tabular/Matrix
- 2. Data Flexibility
- 3. Data Manipulation
- 4. Time Series



INSTALLATION

```
pip install pandas
pip install pandas as pd
```

PANDAS DATA STRUCTURES

- Series basically an ordered dict that can be named
- Dataframe A labeled two dimensional datatype

SERIES

WHAT DOES IT LOOK LIKE?

```
O Chocolate Chip
Peanut Butter
Ginger Molasses
Oatmeal Raisin
Sugar
Sugar
oreo
dtype: object
```

PROPERTIES

SPECIFYING THE INDEX

INDEXED SERIES

Chocolate Chip	12	
Peanut Butter	10	
Ginger Molasses	8	
Oatmeal Raisin	6	
Sugar	4	
Powder Sugar	2	
dtype: int64		

NAMING THE VALUES AND INDEXES

ACCESSING ELEMENTS

DATAFRAMES

```
df = pd.DataFrame({
    'count': [12, 10, 8, 6, 2, 2, 2],
    'type': ['Chocolate Chip', 'Peanut Butter', 'Ginger Molasses', 'Oatmeal Rowner': ['Jason', 'Jason', 'Jason', 'Jason', 'Jason', 'Marvin']
})
```

	count	owner	type
0	12	Jason	Chocolate Chip
1	10	Jason	Peanut Butter
2	8	Jason	Ginger Molasses
3	6	Jason	Oatmeal Raisin
4	2	Jason	Sugar
5	2	Jason	Powder Sugar
6	2	Marvin	Sugar

ACCESSING COLUMNS

```
>>> df['type']

0    Chocolate Chip
1    Peanut Butter
2    Ginger Molasses
3    Oatmeal Raisin
4         Sugar
5    Powder Sugar
6         Sugar
Name: type, dtype: object
```

ACCESSING ROWS

SLICING ROWS

PIVOTING

GROUPING

```
>>> df.groupby('owner').sum()

count
owner
Jason 40
Marvin 2
```

```
>>> df.groupby(['type','owner']).sum()
                               count
    type
                      owner
    Chocolate Chip Jason
                                  12
    Ginger Molasses Jason
Oatmeal Raisin Jason
                                   8
                                   6
    Peanut Butter
                                  10
                     Jason
    Powder Sugar
                      Jason
    Sugar
                      Jason
                      Marvin
```

RENAMING COLUMNS

PIVOT TABLES

```
>>> pd.pivot_table(df, values='count', index=['type'], columns=['owner'])
      0wner
                       Jason Marvin
      type
      Chocolate Chip
                          12
                                 NaN
      Ginger Molasses
                           8
                                 NaN
      Oatmeal Raisin
                                 NaN
      Peanut Butter
                          10
                                 NaN
      Powder Sugar
                                 NaN
      Sugar
```

JOINING

```
>>> df = pivot_t.join(g_sum)
>>> df.fillna(0, inplace=True)
                         Jason Marvin Total
     type
     Chocolate Chip
                             12
                                               12
    Ginger Molasses
                              8
                                                8
    Oatmeal Raisin
                                                6
                             10
                                        0
                                               10
     Peanut Butter
    Powder Sugar
                                        0
     Sugar
                                                4
```

