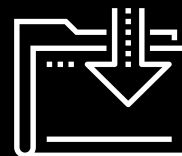




Meet Pandas

3/28/2023

FinTech
Lesson 3.x



Objectives



Describe the benefits of Pandas over spreadsheets to manipulate data on financial use cases.



Explain what a DataFrame is and how it differs from a series.



Create DataFrames from CSV files and use basic commands to manipulate them.



Clean data using built-in commands of DataFrames.



Manipulate data using DataFrame



Some Pandas functions



Create basic data visualizations with Pandas' built-in plotting functions.

Why Pandas?

The Pain of Using Spreadsheets

Spreadsheets are great, but they can become a pain when you are dealing with complex data:

- Calculations are often not reproducible.
- Data can be overwritten in the spreadsheet.
- Data cleaning may overwrite the original data.
- Sharing spreadsheets is difficult.
- Combining data from multiple spreadsheets is difficult.
- Spreadsheets often demonstrate poor performance.
- Large datasets are not handled well.

65421 stop	incidence	minnes	ota	Francis Teller	5 months
464654 ok	paid	Chicag		Mike Michaelson	8 months
7824 ok	paid	minnes	n	James Kowalscky	4 months
		Chicag		James Turner	7 months
		Chicag		Emma Smith	3 months
		Chicag		Bryce Teller	7 months
		minnes	ota	John van Persie	8 months
		wiscon:	in	Jordan Tate	5 months
		Chicag		Mindy Spencer	8 months
		michig:	n	Michael Jones	4 months
		Chicag		Terry Flanagan	5 months
		wiscon:	in	Thomas Tursen	7 months
		minnes	ota	Treapwodd Mint	3 months
		Chicag		Tim Berenger	6 months
		michig:	n	Jonas Stone	8 months
		Chicag		Tobby Rapaport	6 months
		Chicag		Peter Bayega	5 months
		minnes	ota	Javier Ortiz	7 months
		Chicag		James Rodrigues	8 months
		michig:	n	Timmy O'Flanagan	3 months
		minnes	ota	Mike Mcfly	4 months
		Chicag		Jeremiah Tully	6 months
		Chicag		Clemence Sanchez	8 months
		michig:	n	Timmy Richard Lee	6 months



The Origins of Pandas

- [Pandas](#) is one of the most powerful open source libraries in Python for analyzing and manipulating data.
- This library was born on 2008 at [AQR Capital](#) when [Wes McKinney](#) was looking for a solution to offer a high-performance and flexible tool to perform quantitative analysis on financial data.
- Etymology: panel data structures

Why Pandas is Great

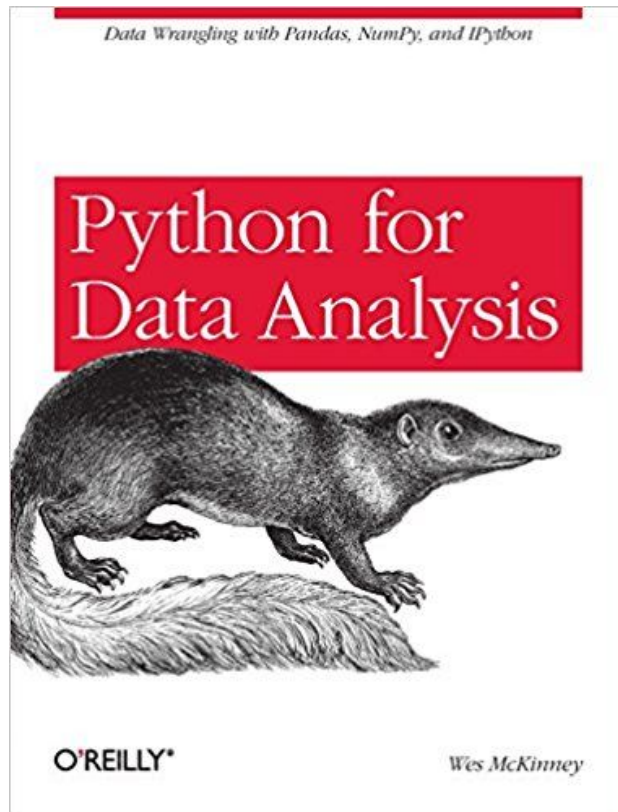
- Python + Pandas = the perfect combination for small experiments or for implementing large-scale production systems to analyze data and make smarter decisions.
- High-performance data structures:
 - Series (1D labeled vectors)
 - DataFrame (2D structures similar to spreadsheets)
- Built-in time series functionality, which is a must for financial and quants analysis



Resources for Learning More About Pandas

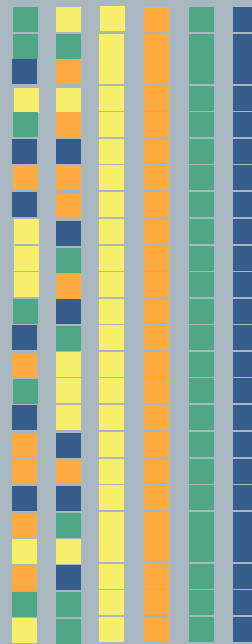
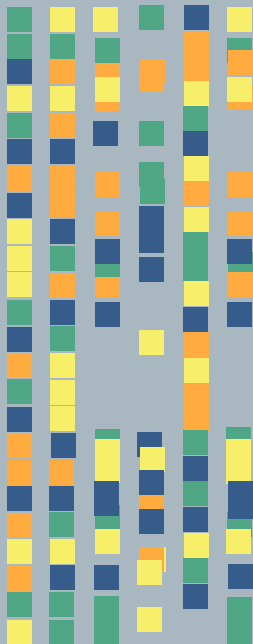
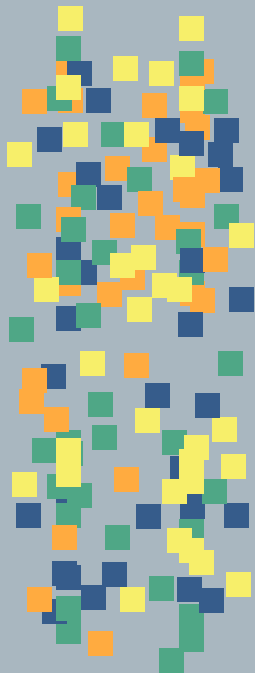
- Official website: <https://pandas.pydata.org/>
- Pandas on GitHub: <http://github.com/pydata/pandas>
- *Python for Data Analysis* by Wes McKinney

Python for Data Analysis
by Wes McKinney
(O'Reilly Media, 2017)



Sorting

Data is not always organized in the best way for analysis. Sometimes, data needs to be cleaned and sorted.



Sorting

The `sort_values` function in Pandas can be used to sort a DataFrame. Sorting data helps improve visual representation of data.

Data can be sorted in either ascending or descending order.

```
sort_values(ascending=True)
```



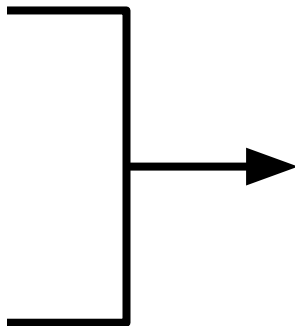
Consider dates: would you rather see dates sorted or randomly listed?

Grouping

A key component of data analysis is grouping data. **Grouping** allows for similar data to be aggregated or manipulated as groups.

Example aggregations that can be done on groups are adding, summing, determining min and max, etc.

Category	Sales
a	1
a	2
b	10
b	9



Category	Sales
a	3
b	19

Grouping

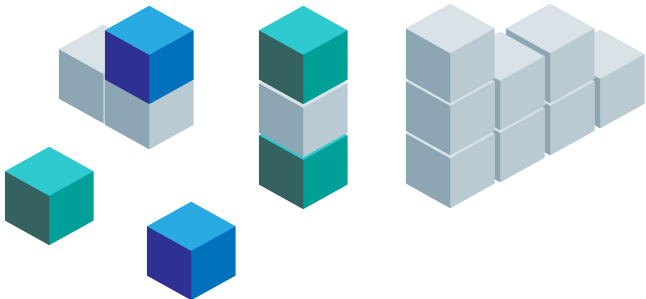
Behind the scenes, the Pandas `groupby` function does the following:



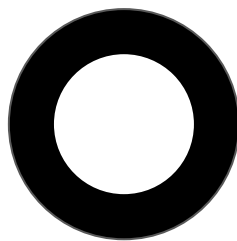
Splits the data into groups based on certain criteria.



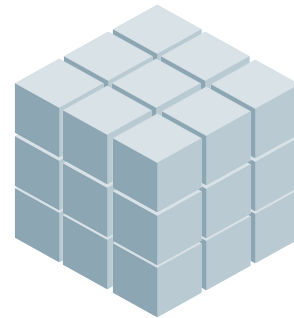
Applies a function to each group independently.



Splitting Data



Applying a Function



Combining Results

Returns Over Time

Returns over time can be calculated using the `pct_change()` function.



Concatenation

Pandas has a `concat` function that can be used to combine DataFrames.

DataFrames can be concatenated so that the records from two DataFrames are combined.

DataFrames can be combined by column so that the columns from one DataFrame are placed adjacent to columns from another DataFrame.



*The
End*