

Introduction to Pandas

(Relational Data)

By Gideon & Sandra



“aimed at getting you to
kickass in AI”





No! this class is not about the Pandas
bear.

Sorry to disappoint! [Source](#)

What is Pandas?

Think of Pandas as an Excel sheet, but a next level Excel sheet with more features and flexibility than Excel. pandas is a python library used for data manipulation and analysis. It is designed to make working with “**relational**” or “**labeled**” data both easy and intuitive.

Excel

	A	B	C	D
1	Product	Sales	Date	City
2	Bananas	\$ 121.00	2019-06-13	Atlanta
3	Bananas	\$ 236.00	2019-10-20	Atlanta
4	Apples	\$ 981.00	2019-03-12	Atlanta
5	Bread	\$ 996.00	2019-07-28	New York City
6	Broccoli	\$ 790.00	2019-10-22	New York City

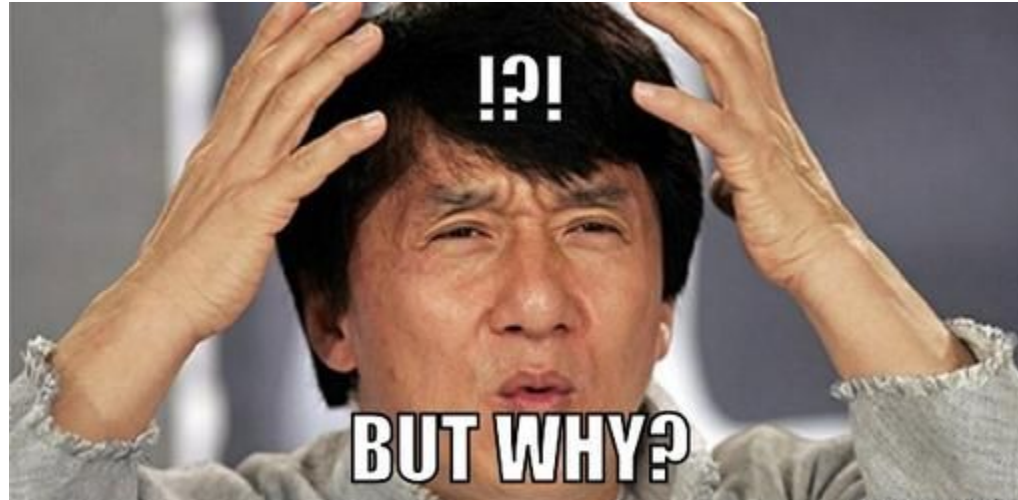
Python with Pandas

	Product	Sales	Date	City
0	Bananas	121	2019-06-13	Atlanta
1	Bananas	236	2019-10-20	Atlanta
2	Apples	981	2019-03-12	Atlanta
3	Bread	996	2019-07-28	New York City
4	Broccoli	790	2019-10-22	New York City

Pretty similar *
*but also kind of different



Why Pandas? Why so Famous?



[Source](#)

Why Pandas? Why so Famous?

- Open Source
- Easy to Learn
- Great Community
- Easy to Analyze and pre-process data in it
- Built-in Data Visualization
- Built-in support for CSV, SQL, HTML, JSON, pickle, excel, and a lot more
- It is built on top of Numpy
- and a lot more





What can I use Pandas for?



What can I use Pandas for?

- Data cleansing
- Data fill
- Data normalization
- Data visualization
- Statistical analysis
- Data inspection
- Loading and saving data
- And much more

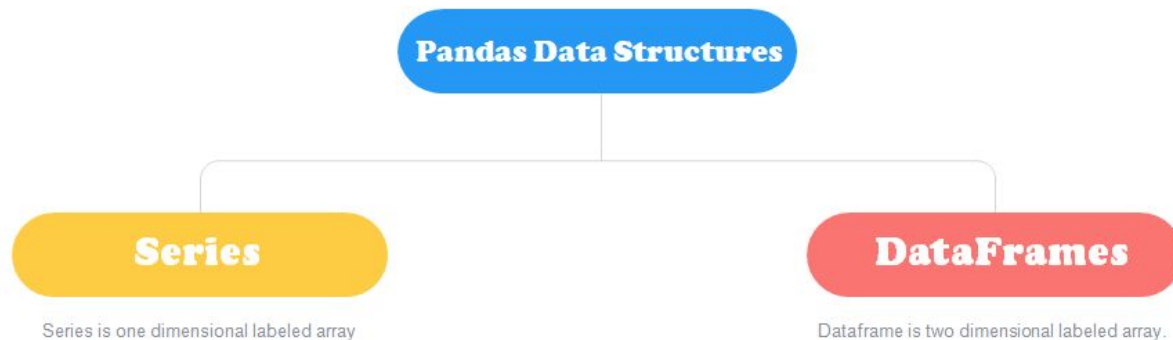
Pandas Data Structures

Pandas is a Python package for data analysis and it supports two main types of Data Structures.

Series and DataFrames!

Series : It is a one dimensional labeled array

DataFrame: It is a two dimensional labeled array



Pandas Series

Pandas Series is the one-dimensional labeled array just like the NumPy Arrays. The difference between these two is that Series is mutable and supports heterogeneous data. So Series is used when you have to create an array with multiple data types.

SERIES

Student

0	Student 1
1	Student 2
2	Student 3

[Source](#)



Creating Pandas Series

```
[2] #Importing Pandas
import pandas as pd
#let's first create example Series
s = pd.Series(["AI", 6, 2.1, False])
print(s)
```

```
0      AI
1        6
2      2.1
3    False
dtype: object
```

```
[4] #By default, index takes values from 0 to n but you can define your own index values. Here's one ways to define index
s = pd.Series(["AI", 6, 2.1, False],index=["String","Integer","Float","Boolean"])
print(s)
```

```
String      AI
Integer      6
Float      2.1
Boolean    False
dtype: object
```

```
▶ #Here's another way to define index
s.index = [1,2,3,4]
print(s)
```

```
1      AI
2        6
3      2.1
4    False
dtype: object
```



Pandas Dataframe

DataFrame is the most commonly used data structure in pandas. DataFrame is a two-dimensional labeled array. It can be thought of as a dictionary of Series structures where both the columns and the rows are indexed, denoted as 'index' in the case of rows and 'columns' in the case of columns.



To Learn more about pandas see this <https://www.w3schools.com/python/pandas/default.asp>



Series + Series = Dataframe



SERIES

Student

0	Student 1
1	Student 2
2	Student 3

+

SERIES

Age

0	18
1	19
2	20

=

DATAFRAME

Student Age

0	Student 1	18
1	Student 2	19
2	Student 3	20

[Source](#)



Creating Dataframes

```
days = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]
df = pd.DataFrame(days)
print(df)
```

```
0
0    Monday
1    Tuesday
2  Wednesday
3   Thursday
4     Friday
5   Saturday
6     Sunday
```

```
[12] #several types of data can be used to create dataframe: Dictionary of 1D ndarrays, lists, dictionaries, or Series structures.
data = {"Staff":["Staff1", "Staff2", "Staff3"],"Age":[31,38,26]}
df = pd.DataFrame(data)
print(df)
```

```
   Staff  Age
0  Staff1   31
1  Staff2   38
2  Staff3   26
```



Creating Dataframes

```
[9] df.index = ["row1","row2","row3"]  
    print(df)
```

	Staff	Age
row1	Staff1	31
row2	Staff2	38
row3	Staff3	26

```
[10] df.columns = ["Name of Staff","Age of Staff"]  
     print(df)
```

	Name of Staff	Age of Staff
row1	Staff1	31
row2	Staff2	38
row3	Staff3	26



Data analysis process





[Source](#)

Hands-on Pandas Series and Dataframe Examples

It's time to Switch to
Jupyter Notebook



Link to Jupyter Notebook



[Pandas Notebook](#)



Thank YOU!

Github: <https://github.com/AISaturdaysLagos>

Twitter: <https://twitter.com/aisaturdaylagos>

Linkedin: <https://www.linkedin.com/company/aisaturdayslagos/>

Instagram: <https://www.instagram.com/aisaturdayslagos/>

Youtube: <https://www.youtube.com/c/AISaturdaysLagos>

