

# SHE LOVES DATA

Intermediate Python - Day 2  
September 2020 | Virtual Workshop



# INTRODUCTION - INSTRUCTOR

The graphic features a red background with a blue circular logo containing the text "Intermediate Python". Below the logo, it says "WITH SARAH NOORAVI SR. FINANCIAL ANALYST SNAP INC.". A photo of Sarah Nooravi, a woman with long dark hair, wearing an orange top and a light blue cardigan, is centered in a blue-bordered frame. At the bottom, it shows "Sep 19, 11 am - 1:30 pm PDT" and logos for Zoom, Meiro, and she loves data.

- B.S. Math/Econ
- M.S. Mechanical Engineering
- Marketing Analytics
- Python for 4+ years
- Programming for much longer
  - Matlab & R

# INTRODUCTION - MENTORS



Sarah Nooravi



Julia Rossi



Robyn Goldberg



Jennifer Siwu



Dalya Manatova



Anushka Singh



Adriana Chavez



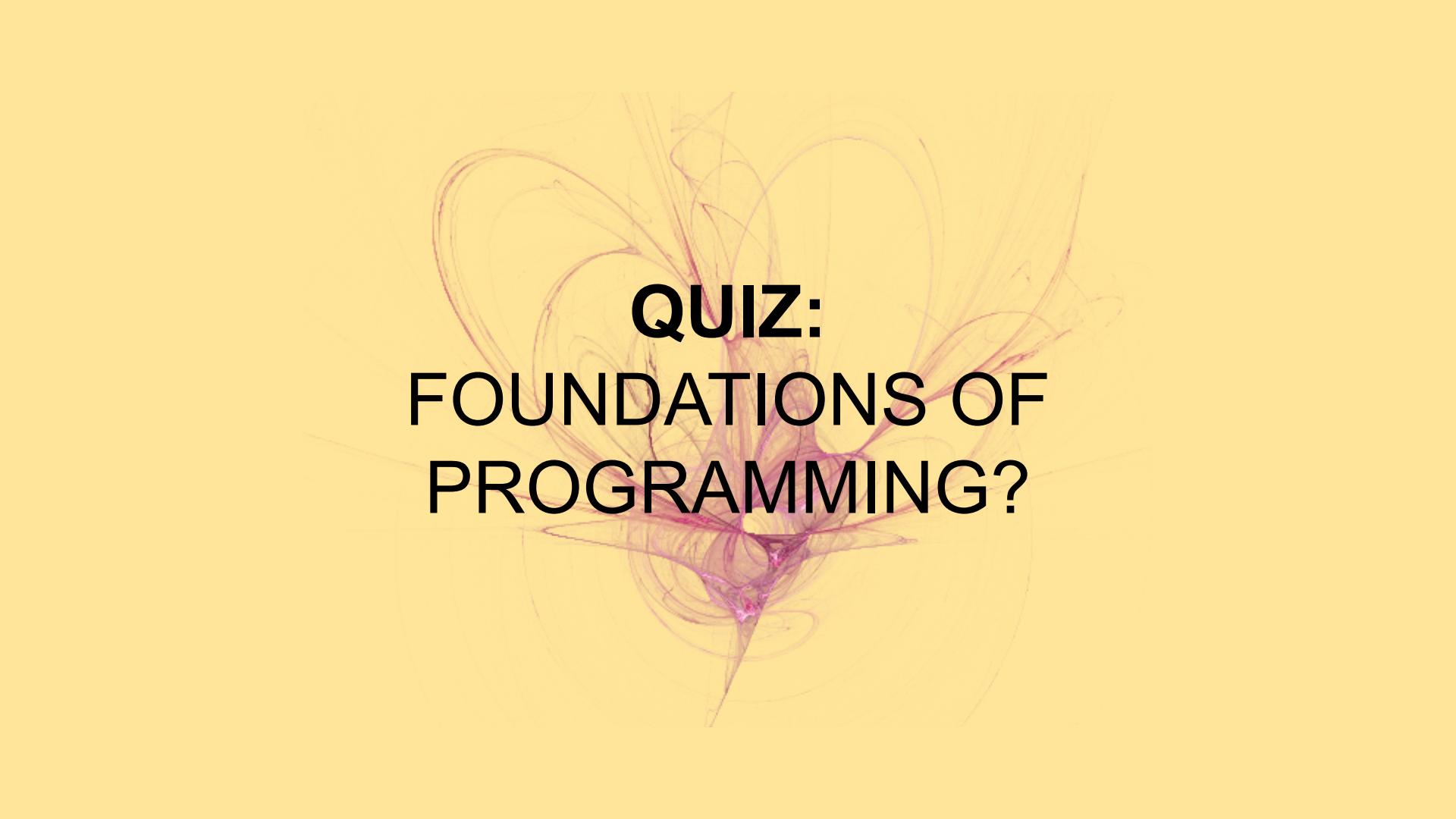
Shibani Rai



# AGENDA

- Quick Review
- Introduction to Python Libraries
- Breakout Session 1 (Intros & Setup)
- Live Coding
- Breakout Session 2 (Series)
- Live Coding
- Hands-On Project!





# **QUIZ:** **FOUNDATIONS OF** **PROGRAMMING?**

# FOUNDATIONAL BUILDING BLOCKS

- Variables
- Sequences
- Loops
- If/Then (or Case/When)
- Functions



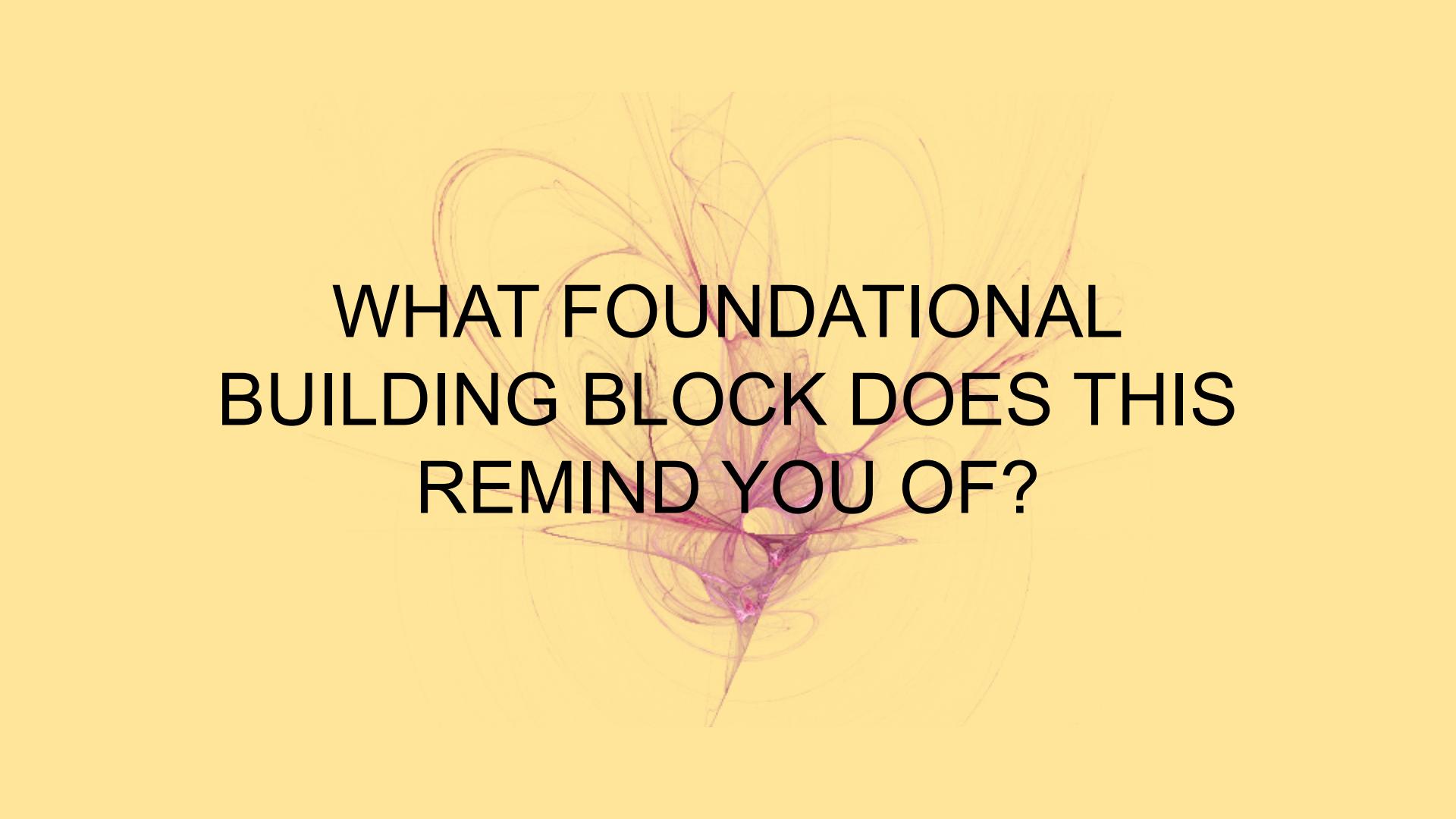
# PYTHON LIBRARIES



## What is a Library?

A library is a collection of pre-combined codes that can be used iteratively to reduce the time required to code. They are particularly useful for accessing the pre-written frequently used codes, instead of writing them from scratch every single time. Similar to the physical libraries, these are a collection of reusable resources, which means every library has a root source. This is the foundation behind the numerous open-source libraries available in Python.





WHAT FOUNDATIONAL  
BUILDING BLOCK DOES THIS  
REMIND YOU OF?

# FUNCTIONS



A **function** is a block of code which only runs when it is called. You can pass data, known as parameters, into a function.

A function can return data as a result.

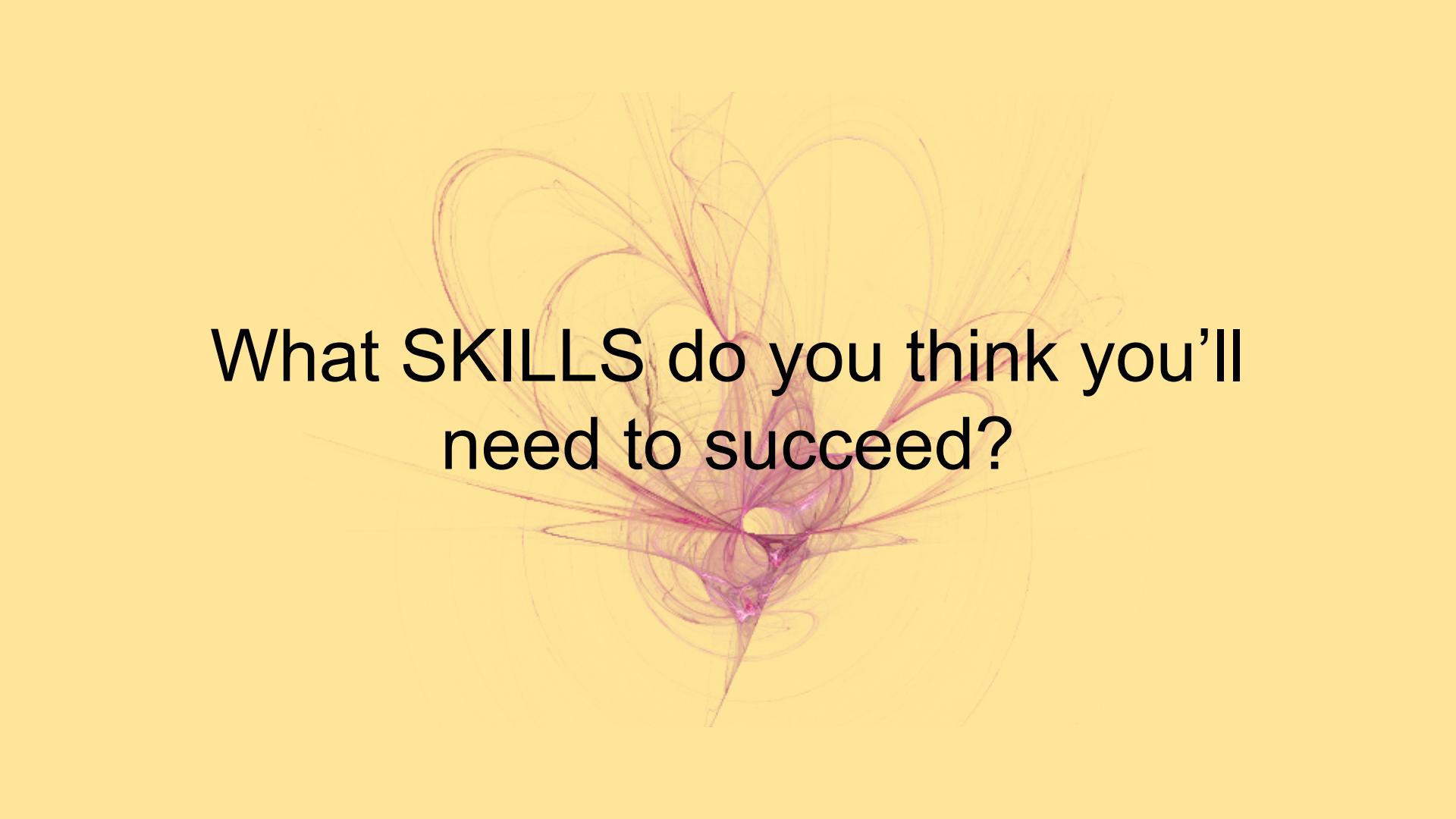
## Example

```
def my_function():
    print("Hello from a function")

my_function()
```



Anaconda



What SKILLS do you think you'll  
need to succeed?

Google

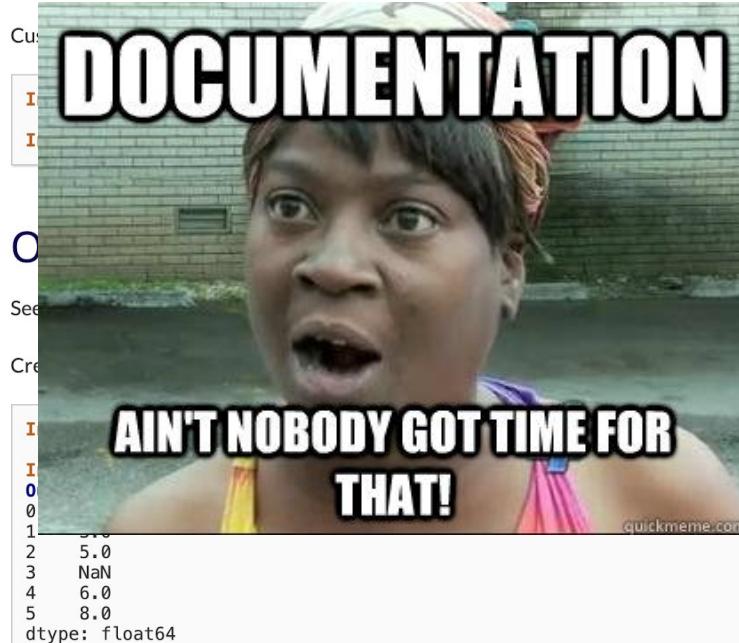
Search the docs ...

## 10 minutes to pandas

- Intro to data structures
- Essential basic functionality
- IO tools (text, CSV, HDF5, ...)
- Indexing and selecting data
- MultIndex / advanced indexing
- Merge, join, concatenate and compare
- Reshaping and pivot tables
- Working with text data
- Working with missing data
- Categorical data
- Nullable integer data type
- Nullable Boolean data type
- Visualization
- Computational tools
- Group by: split-apply-combine
- Time series / date functionality
- Time deltas
- Styling

# 10 minutes to pandas

This is a short introduction to pandas, geared mainly for new users. You can see more complex recipes in the [Cookbook](#).



≡ On this page

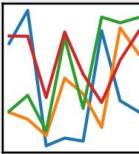
- Object creation
- Viewing data
- Selection
- Missing data
- Operations
- Merge
- Grouping
- Reshaping
- Time series
- Categoricals
- Plotting
- Getting data in/out
- Gotchas

# LIBRARIES FOR THIS WORKSHOP

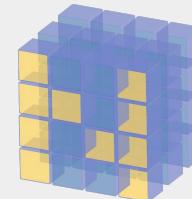
## FOR DATAFRAME MANIPULATION

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



## FOR ARRAY MANIPULATION



NumPy

## FOR VISUALIZATION

seaborn

matplotlib





- **pandas** is an open source, BSD-licensed library providing high-performance, easy-to-use **data structures** and **data analysis tools** for the Python programming language
- Great resource for some common data manipulation functions.

A screenshot of Microsoft Excel showing a PivotTable setup. The PivotTable Field List on the right shows fields: ID, First, Last, and Department. The Row Labels section has 'Department' selected. The Values section has 'Count of First' selected. A red arrow points from this screenshot to the 'Pivot' diagram below.

Reshaping by pivoting DataFrame objects

Pivot

	foo	bar	baz	zoo
0	one	A	1	x
1	one	B	2	y
2	one	C	3	z
3	two	A	4	q
4	two	B	5	w
5	two	C	6	t



```
df.pivot(index='foo',  
        columns='bar',  
        values='baz')
```

	bar	A	B	C
foo				
one	1	2	3	
two	4	5	6	



# HOW TO INSTALL

## Installation

### Working with conda?

pandas is part of the [Anaconda distribution](#) and can be installed with Anaconda or Miniconda:

```
conda install pandas
```

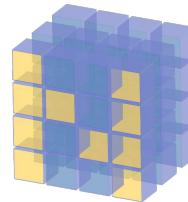
### Working with pip?

pandas can be installed via pip from [PyPI](#).

```
pip install pandas
```

DONE





# NumPy

- It's the universal standard for working with **numerical data in Python**, and it's at the core of the scientific Python and PyData ecosystems.
- The NumPy API is used extensively in Pandas, SciPy, Matplotlib, scikit-learn, scikit-image and most other data science and scientific Python packages.

## What's the difference between a Python list and a NumPy array?

NumPy gives you an enormous range of fast and efficient ways of creating arrays and manipulating numerical data inside them.

While a Python list can contain different data types within a single list, all of the elements in a NumPy array should be homogenous.

The mathematical operations that are meant to be performed on arrays would be extremely inefficient if the arrays weren't homogenous.

### Why use NumPy?

NumPy arrays are faster and more compact than Python lists. An array consumes less memory and is convenient to use. NumPy uses much less memory to store data and it provides a mechanism of specifying the data types. This allows the code to be optimized even further.



# HOW TO INSTALL

## Installing NumPy

To install NumPy, we strongly recommend using a scientific Python distribution. If you're looking for the full instructions for installing NumPy on your operating system, you can [find all of the details here](#).

If you already have Python, you can install NumPy with:

```
conda install numpy
```

or

```
pip install numpy
```

If you don't have Python yet, you might want to consider using [Anaconda](#). It's the easiest way to get started. The good thing about getting this distribution is the fact that you don't need to worry too much about separately installing NumPy or any of the major packages that you'll be using for your data analyses, like pandas, Scikit-Learn, etc.

You can find all of the installation details in the [Installation](#) section at [SciPy](#).

DONE





I WANT TO USE THESE  
PACKAGES..  
HOW DO I START?



You ;)

# SIMPLE IMPORT STATEMENTS

Customarily, we import as follows:

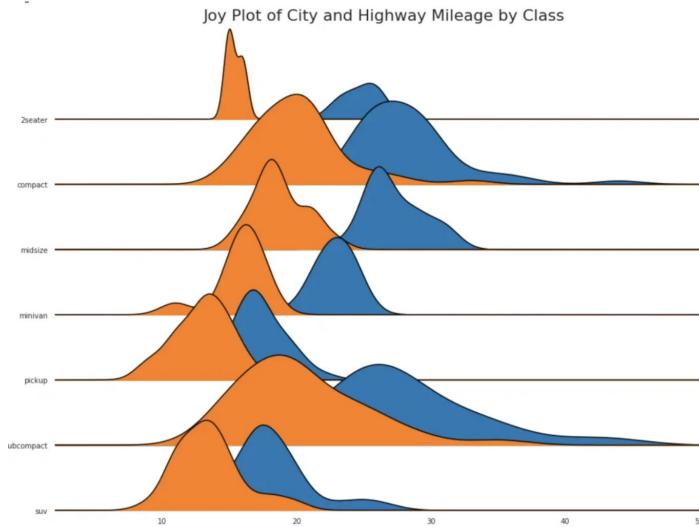
```
In [1]: import numpy as np
```

```
In [2]: import pandas as pd
```



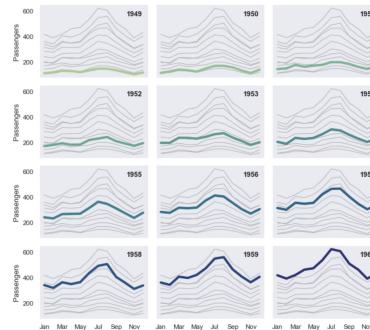
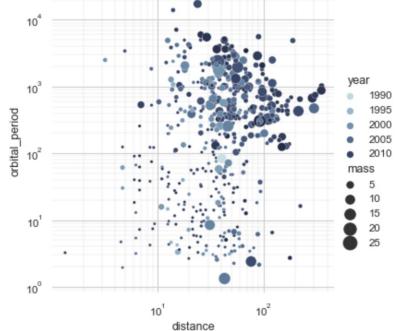
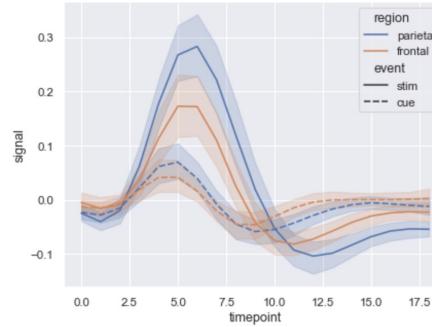
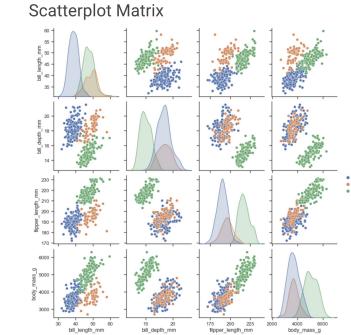
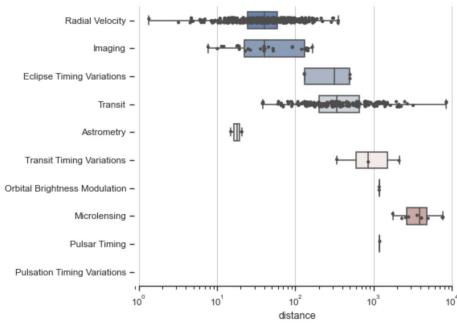
# matplotlib

- Creates static, animated, and interactive visualizations in Python
- For more examples, see their [gallery](#) or some [master plots](#).



# seaborn

- Builds Visuals  
[fancier than `matplotlib` :P]
  - Graphs
  - Barchart
  - Plots



# HOW TO INSTALL

## Installing and getting started

Official releases of seaborn can be installed from PyPI:

```
pip install seaborn
```

DONE

The library is also included as part of the Anaconda distribution:

```
conda install seaborn
```



# INTERESTING PYTHON LIBRARIES

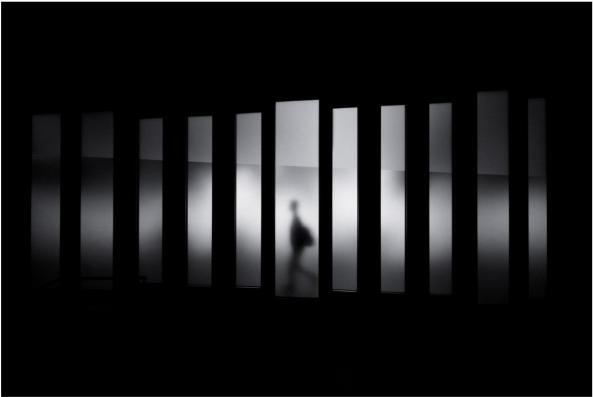


Photo by David Werbrouck on [Unsplash](#)

```
>>> import numpy as np
>>> from bashplotlib.histogram import plot_hist
>>> arr = np.random.normal(size=1000, loc=0, scale=1)
>>> plot_hist(arr, bincount=50)

56|          o
53|         oo
50|        oooo o
47|        oooo o
44|       ooooooo o
41|      o  ooooooo o
38|     ooooooooooo o
35|    ooooooooooooooo
32|   ooooooooooooooo
30|  oooooooooooooooo
27| o  oooooooooooooooo
24|   oooooooooooooooo
21|   o  oooooooooooooooo o
18|   o  oooooooooooooooo o
15|   o  oooooooooooooooo
12|   oooooooooooooooo
9|    oooooooooooooooo
6|    oooooooooooooooo
3|     o  oooooooooooooooo
1| o o o  oooooooooooooooo o
```



Photo by JOSHUA COLEMAN on [Unsplash](#)

5 Interesting Python Libraries [Link](#)



“

# INTRODUCE YOURSELF

1.

Name

2.

Where are you located?

3.

Fun thing you've been  
doing in quarantine