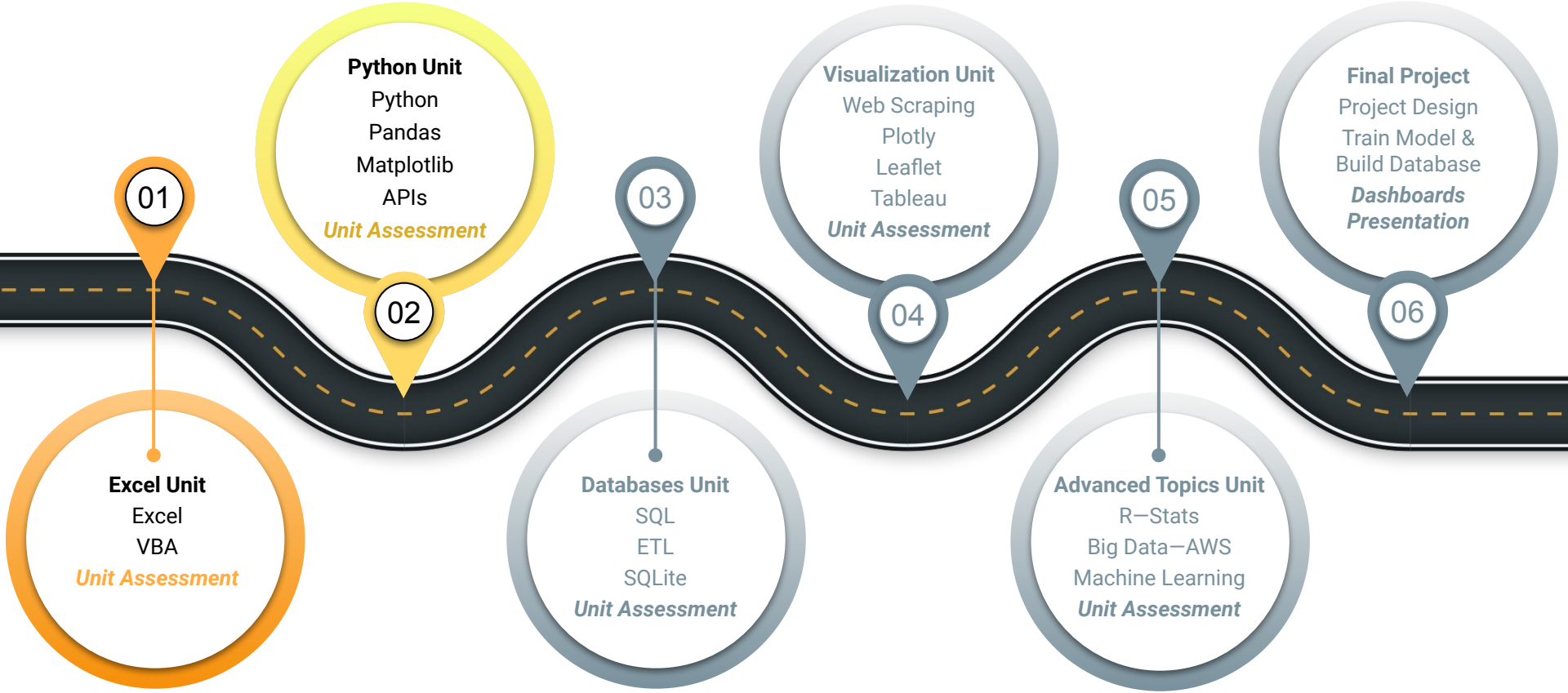
Module 5

This Week: Matplotlib

The Big Picture



This Week: Matplotlib

By the end of this week, you'll know how to:

- Create line, bar, scatter, bubble, pie, and box-and-whisker plots using Matplotlib
- Add and modify features of Matplotlib charts
- Add error bars to line and bar charts
- Determine mean, median, and mode using Pandas, NumPy, and SciPy statistics

This Week's Challenge

Create a summary DataFrame of ride-sharing data by city type and a multiple-line graph showing weekly fares for each city type.

Module 5

Today’s Agenda

Today’s Agenda

By completing today’s activities, you’ll learn the following skills:

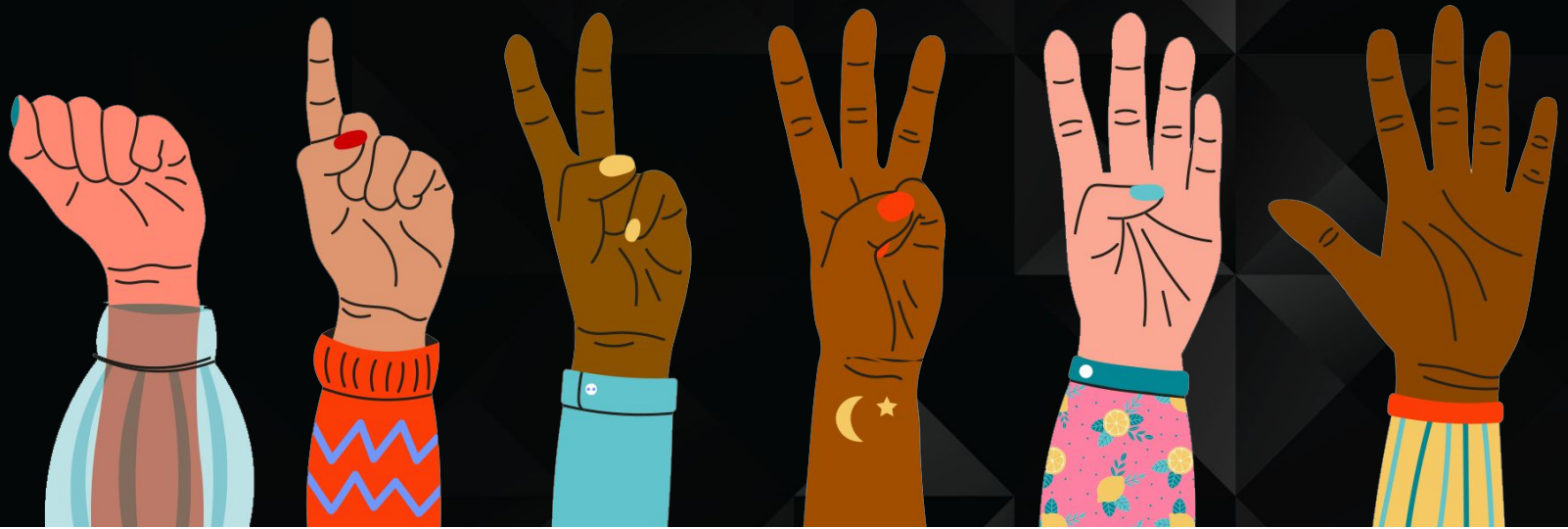
01 Create line, bar, pie, and scatter charts from Pandas DataFrames

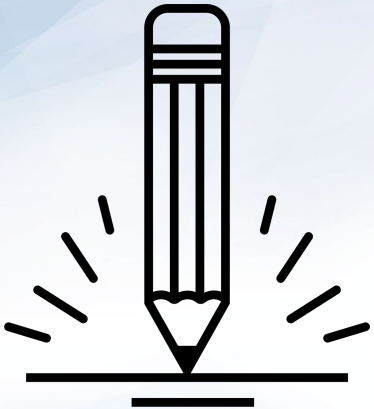
02 Add and modify chart features for readability

Make sure you’ve downloaded any relevant class files!

FIST TO FIVE:

How comfortable do you feel with this topic?





Activity: PyPlot Warmup

Suggested Time:
20 minutes

Plotting Pandas Data

The plots within the previous activity were generated using mock data.

… but we will deal with real-world data more often.

- Strange formats
- Messy
- Missing data
- Misleading headers

How to Work with Messy Data

Pandas enables us to quickly and easily:

- Rename headers
- Remove missing data
- Convert and clean up column data

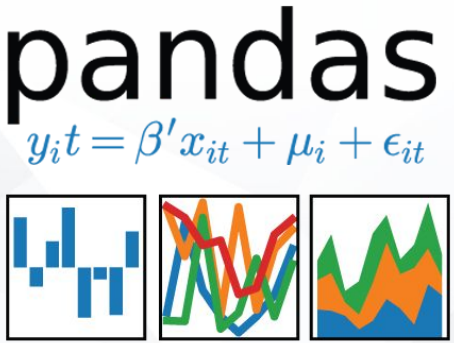
In most cases, we will work with real-world data in Pandas.

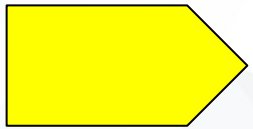
Plotting Pandas Data

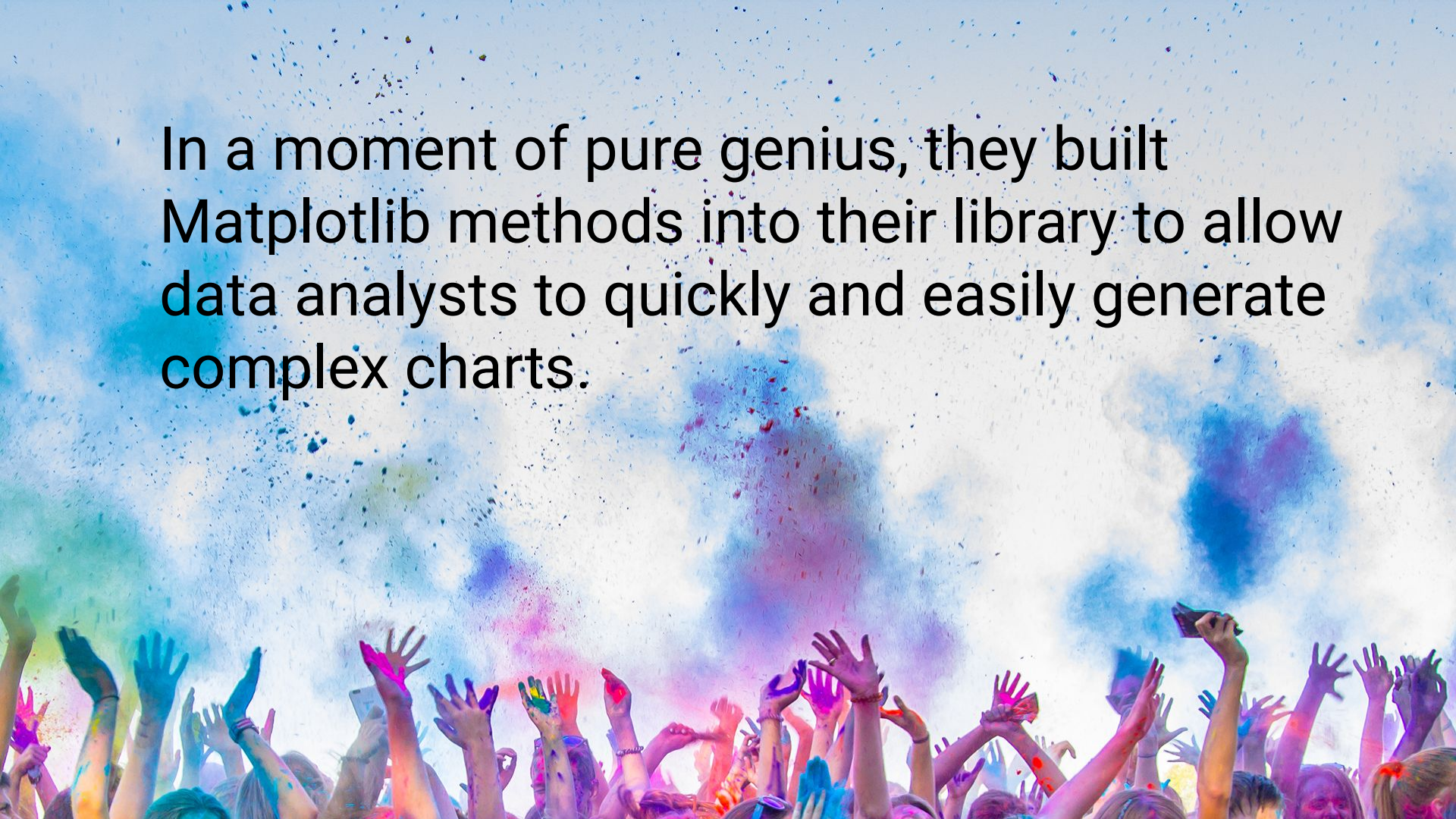
Last week, we learned how to clean up and preprocess data sets using Pandas. Most likely, real-world data that we'll want to analyze and create visualizations will be in a CSV file which will have to be read into a Pandas DataFrame.

`read_csv()`	Allows us to import a CSV file into a Pandas DataFrame.
`head()`	Allows us to see the first 5 lines of a given DataFrame.

The creators of Pandas realized that most people using Pandas would move on to visualize their plots using Matplotlib.

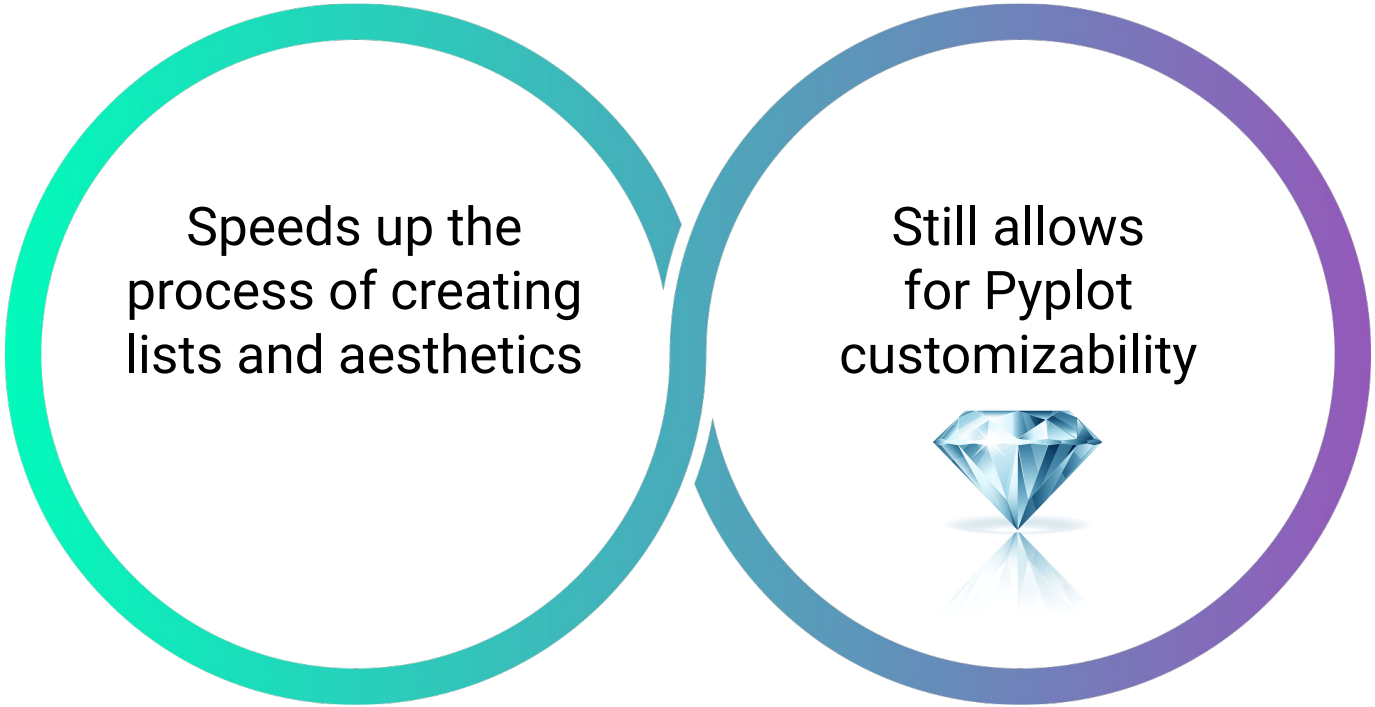


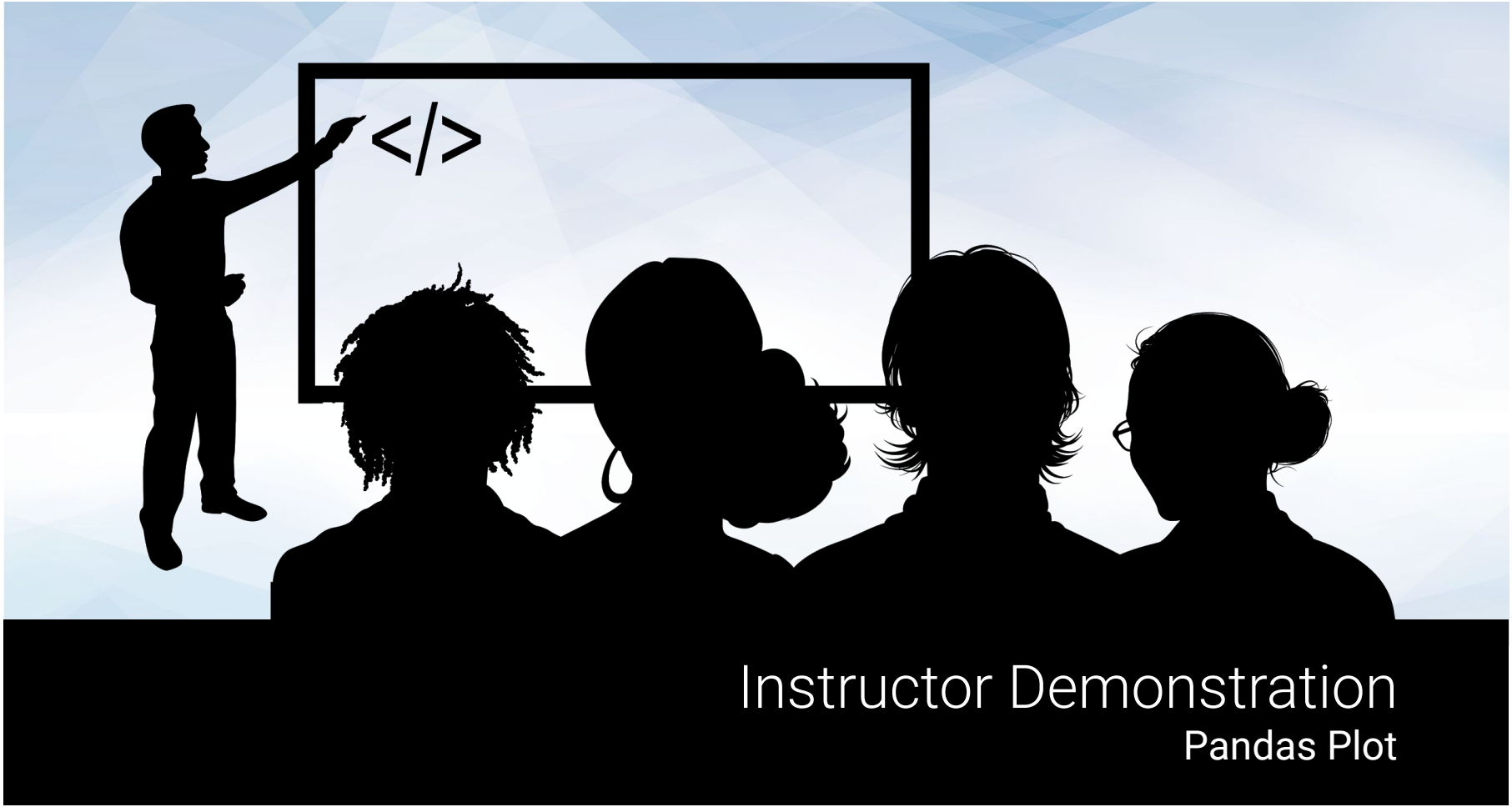



In a moment of pure genius, they built Matplotlib methods into their library to allow data analysts to quickly and easily generate complex charts.

The Creators of Pandas Are Geniuses!

Pandas creators directly added Matplotlib functionality, which:

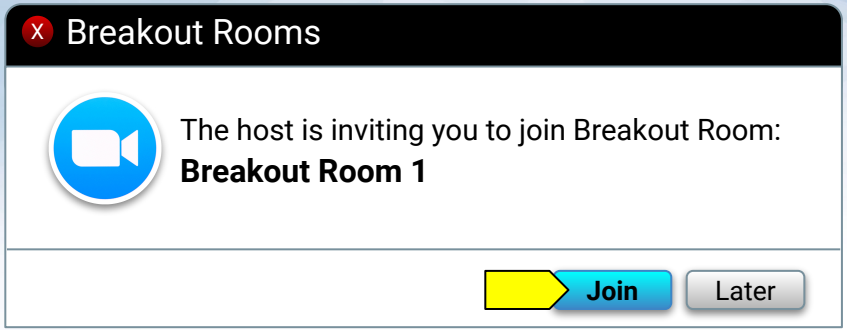




Instructor Demonstration

Pandas Plot

Questions?



Zoom Breakout Room Activity:

Battling Kings

In this activity, you will create a bar chart that visualizes which kings in the Game of Thrones universe have participated in the most battles.

Suggested Time:
15 minutes

Let’s Review

Plotting Groups

How do we group data in Pandas?

The `.groupby()` function allows you to **group Pandas objects based on a common record**.

Grouping and Summarizing in Pandas

The `dataframe.groupby()` function allows us to group data.
Data can be grouped by function or category.

fuel_type	mileage	horsepower	num_doors	num_cup_holder	price
gasoline	389052	302	2	8	54234
gasoline	127148	142	4	4	5032
diesel	23423	350	2	4	43289
gasoline	57482	100	4	10	12739
gasoline	42421	90	2	6	32129
bio	23845	120	4	6	18234
diesel	234712	150	2	10	20502

The output of the function is a GroupBy object.

```
<pandas.core.groupby.groupby.DataFrameGroupBy object at 0x10cde6278>
```

	avg_mile	avg_power	mode_door	mode_cup_holder	avg_price
fuel_type					
gasoline	86245	212	4	6	26533
diesel	101234	275	2	6	30235
bio	69234	140	4	4	42139

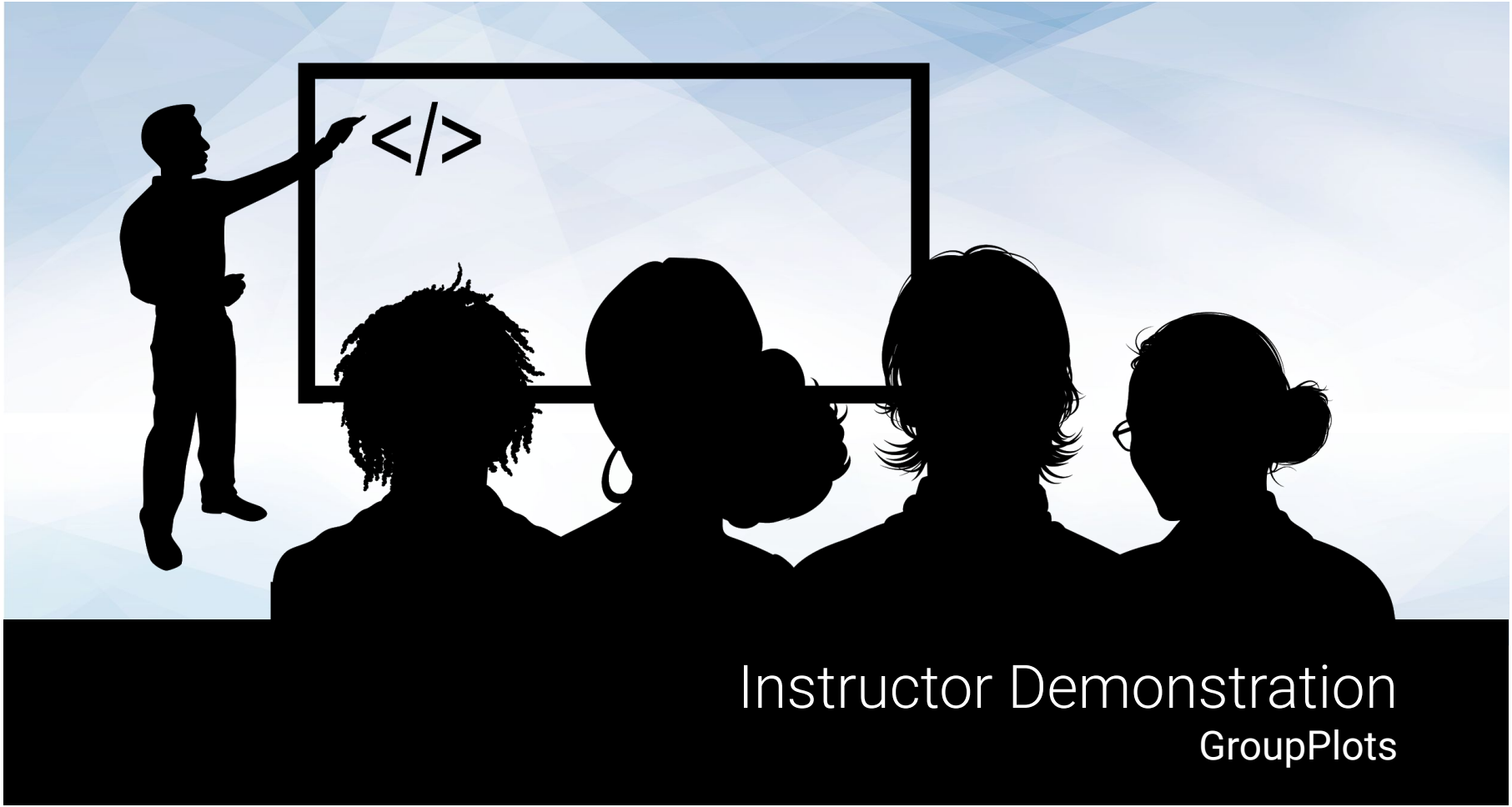
Grouping and Summarizing in Pandas

```
df.groupby('state').mean()
```

Returns a DataFrame from a GroupBy Object

```
states = df.groupby('state')
states = df.groupby['city'].mean()
```

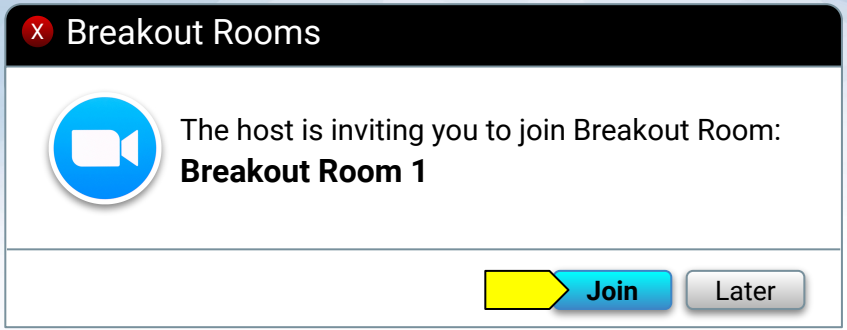
Returns a Series from a GroupBy Object



Instructor Demonstration

GroupPlots

Questions?



Zoom Breakout Room Activity:

Bike Trippin’

In this activity, you will create a pair of charts based on community bike data collected from Seattle.

Suggested Time:
20 Minutes

Let's Review

Plotting Time Series Data with Resample

Challenge:
Plotting Time Series Data with Resample

For the final activity of the day we will create a multiple-line graph to show the number of bike trips for each gender for a selected year from the bike trip data used in the previous activity.

Suggested Time:
25 minutes

Questions?