TLDR: Use Matplotlib defaults when possible - ***import matplotlib.pyplot as plt***

Tables

When to use:

- to look-up and one-to-one comparisons *
- to examine quant values to find patterns *
- for cases that require more exact number representation than a graph can provide *
- data include multiple sets of quant values using different units of measure (difficult to graph)*
- to combine summary and detail information in one display*

Best Practices

- add table title using a print statement
- use **from IPython.display import display** to keep render properly

Detailed Information about Alice, Bob, and Charlie

	Name	Age	City	Height (cm)	Weight (kg)	Income ($)	Education	Marital Status	Children	Empl
0	Alice	25	New York	165	55	50000	Bachelor's	Single	0	
1	Bob	30	Los Angeles	175	70	60000	Master's	Married	2	
2	Charlie	35	Chicago	180	80	70000	PhD	Single	0	

Font Specifications

Pandas DataFrame Table Font
Style: *inherited from user's browser
Size: *inherited from user's browser
Color: *inherited from user's browser

Matplotlib Table Font
DejaVu Sans (MatplotLib default)
Headers, Size: 10 Bold (MatplotLib default)
Title: Use a print statement
Color: #000000 (default black)

DF Color Specifications

Pandas DF Table Colors - df.plot()
Gridline Color: *inherited from user's browser
Alt Row Color: *inherited from user's browser

Matplotlib Table Colors
Gridline Color: #000000 (default black)
Alt Row Color: #F5F5F5 (default grey)
Text: Color: #000000 (default black)

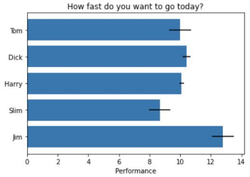
Bar Charts

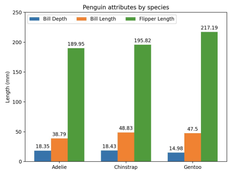
When to use:

- to compare groups
- to compare groups over time
- to show distribution

Best Practices

- always start baseline at 0
- avoid fill patterns *
- fill colors should be equal intensity for data that are equally important *
- use fill colors that are distinct from each other for categorical information *
- only use a bar border to draw attention to a specific bar of data *
- label x and y axes if tick marks are not explicit
- title graph





Font Specifications

Pandas Bar Chart Font
Style: *inherited from user's browser
Size: *inherited from user's browser
Color: *inherited from user's browser

Matplotlib Bar Chart Font
DejaVu Sans (MatplotLib default)
Size: 10 (MatplotLib default)
plt.xlabel - set to 12 (default)
plt ylabel- set to 12 (default)
plt.title - set to 16 (default)
Color: #000000 (default black)

DF Color Specifications

Pandas Bar Chart Colors
Bar Color: *inherited from user's browser
Outline Color: *inherited from user's browser

Matplotlib Bar Chart Colors
Bar Color: #000000 (default black)
Ouline Color: #F5F5F5 (default grey)
Text: Color: #000000 (default black)

Default Colors

#1F77B4
#FF7F0E
#2CA02C
#D62728
#9467BD
#8C564B
#E377C2
#7F7F7F
#BCBD22
#17BECF

Scatterplots and Regression Lines

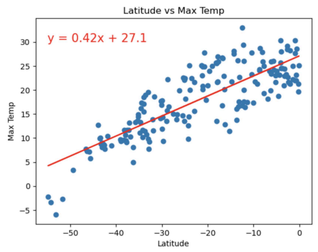
When to use:

- to display a correlation of two paired sets of quantitative data *
- to determine the direction of a correlation relationship *
- use a regression line to further highlight the correlation pattern
- (for displaying correlations between more than 2 quantitative variables, consider using a table.) *

Best Practices

- display r2 value either above or on a visible part of the plot
- use fill colors that are distinct from each other for categorical information
- use circles as data plot shape
- label x and y axes
- title graph

The r^2-value is: 0.6720220238897325



Font Specifications

Pandas Scatterplot
Style: *n/a needs Matplotlib wrapper
Size: *in/a needs Matplotlib wrapper
Color: *n/a needs Matplotlib wrapper

Matplotlib Scatterplot Font
DejaVu Sans (MatplotLib default))
Title Size: 12 (default for ax.set_title())
Axes Label Size: 10 (default)
Color: #000000 (default black)

Color Specifications

Pandas Scatterplot Colors
Gridline Color: *n/a needs Matplotlib wrapper
Alt Row Color: *n/a needs Matplotlib wrapper

Matplotlib Scatterplot Marker/Line
Default Marker Shape: o (circle)
Marker Color: #1F77B4 (default blue)
Regress Line: #FF0000 (default red)
Text: #000000 (default black)

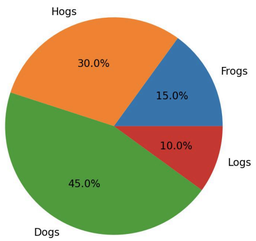
Pie Charts

When to use:

- To visualize proportions or percentages of categories within a whole.
- to represent the composition of a dataset where parts add up to a meaningful total (e.g., 100%).
- To highlight dominant or minor segments within a dataset.
- Best used with a limited number of categories to avoid clutter and maintain readability.

Best Practices:

- limit the number of slices to improve clarity (ideally fewer than 6-8 categories)
- use distinct colors for each slice for easier differentiation.
- avoid using a pie chart if categories are very similar in size—use a bar chart instead for better accuracy.
- label each slice directly with percentages or category names for easy interpretation
- ensure all slices add up to 100% to maintain data accuracy.



Font Specifications

Pandas Pie Charts
Style: *n/a, needs Matplotlib to display
Size: *n/a, needs Matplotlib to display
Color: *n/a, needs Matplotlib to display

Matplotlib Pie Chart Font
DejaVu Sans (MatplotLib default)
Title Size: 16 (MatplotLib default)
Label Size: 12 (default)
Color: #000000 (default black)

DF Color Specifications

Pandas Pie Chart Colors
Slices: *n/a, needs Matplotlib to display

Matplotlib Pie Chart Colors
Slices: (default pallate)
Text: #000000 (default black)

Default Colors

#1F77B4
#FF7F0E
#2CA02C
#D62728
#9467BD
#8C564B
#E377C2
#7F7F7F
#BCBD22
#17BECF

* Few, S. (2012). Show me the numbers: Designing tables and graphs to enlighten. Analytics Press.