

Basic Structure (The Grammar of Graphics)

plotnine is an implementation of the Grammar of Graphics in Python, based on R's ggplot2.

```
from plotnine import *
```

```
(
    ggplot(data, aes(x='col_x',
                    y='col_y', color='col_z'))
    + geom_point()
    + labs(title="My Plot", x="X Axis",
           y="Y Axis")
    + theme_minimal()
)
```

Geoms (Geometric Objects)

Define how data is visually represented.

- `geom_point()`: Scatter plot.
- `geom_line()`: Line plot.
- `geom_bar()`: Bar chart (counts by default).
- `geom_col()`: Bar chart (uses y values directly).
- `geom_histogram()`: Histogram.
- `geom_boxplot()`: Box and whisker plot.
- `geom_violin()`: Violin plot.
- `geom_smooth()`: Add a regression line or Smoother (e.g., `method='lm'`).
- `geom_text()` / `geom_label()`: Add text annotations.
- `geom_tile()`: Heatmaps or grids.

Aesthetics (aes)

Map variables to visual properties.

- `x`, `y`: Position coordinates.
- `color`: Outline/line color.
- `fill`: Background/area color.
- `size`: Size of elements.
- `shape`: Point shapes.
- `alpha`: Transparency level.
- `linetype`: Solid, dashed, etc.

Scales

Control how data values are mapped to aesthetics.

- `scale_x_continuous()`, `scale_y_log10()`.

- `scale_color_manual(values=['red', 'blue'])`.
- `scale_fill_gradient(low='white', high='red')`.
- `scale_color_brewer(type='qual', palette='Set1')`.

Faceting

Split one plot into multiple panels based on a variable.

- `facet_wrap('~variable')`: Multi-panel wrap.
- `facet_grid('var_rows ~ var_cols')`: 2D grid of plots.

Labels and Themes

- `labs()`: Titles and axis labels.
- `theme()`: Fine-tune appearance (e.g., `legend_position='bottom'`).
- Pre-defined Themes: `theme_bw()`, `theme_minimal()`, `theme_classic()`, `theme_dark()`.

Statistical Transformations (stats)

Specify data transformations.

- `stat_count()` (default for `geom_bar`).
- `stat_identity()` (direct values).
- `stat_summary()`: Aggregate values on the fly.

Coordinate Systems

- `coord_flip()`: Swap x and y axes (useful for horizontal bars).
- `coord_fixed()`: maintain aspect ratio.
- `coord_cartesian(xlim=(0, 10))`: Zoom into a specific region.

Pro Tips

Parentheses for Pythonic Chaining

In Python, wrap your ggplot command in parentheses to allow multi-line chaining with `+` without using backslashes.

Integration with Pandas

plotnine expects Pandas DataFrames. It uses the index strings or column names directly in the `aes()` calls.

Saving Plots

Use `.save(filename, width=8, height=6)` method on the plot object or `ggsave(filename, plot)`.