Basic Premise

- Making plots is a very repetive: draw this line, add these colored points, then add these, etc.
- Instead of re-using the same code over and over, `ggplot` implements them using a high-level but very expressive API.
- The result is less time spent creating your charts, and more time interpreting what they mean.

- `ggplot` is not a good fit for people trying to make highly customized data visualizations.
- While you can make some very intricate, great looking plots, `ggplot` sacrafices highly customization in favor of generall doing "what you'd expect".

Data

- ggplot has a symbiotic relationship with pandas.
- If you're planning on using ggplot, it's best to keep your data in DataFrames.
- Think of a DataFrame as a tabular data object.
- For example, let's look at the diamonds dataset which ships with ggplot.

```
from ggplot import *
diamonds.head()
```

	carat	cut	color	clarity	depth	table	price	x	y	z
0	0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
1	0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
2	0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
3	0.29	Premium	I	VS2	62.4	58	334	4.20	4.23	2.63
4	0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75

Aesthetics

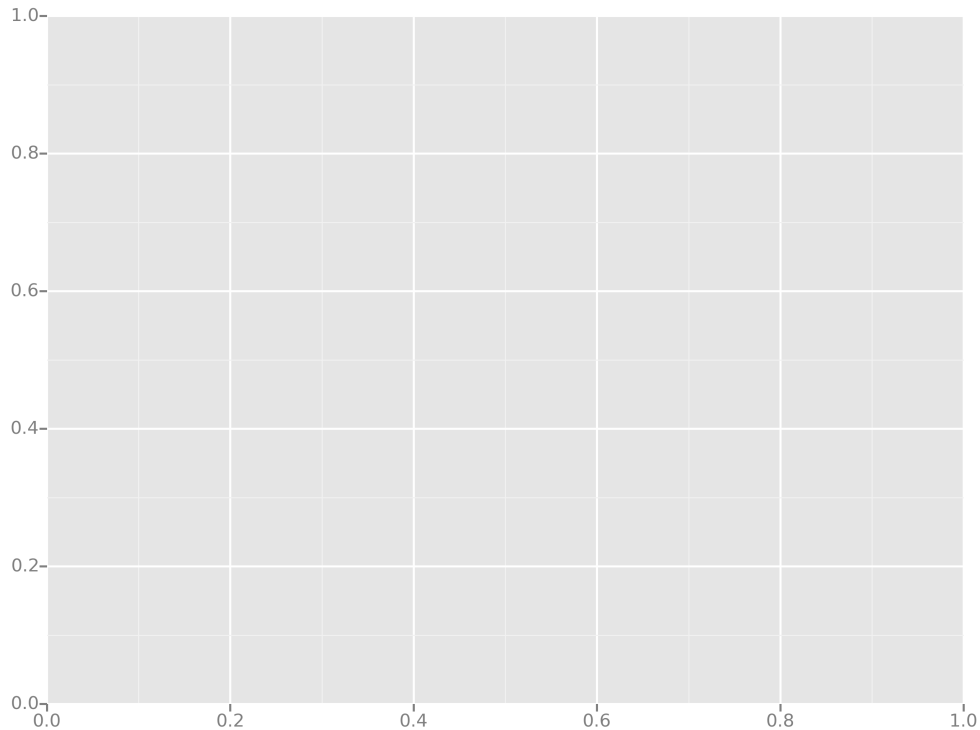
- Aesthetics describe how your data will relate to your plots.
- Some common aesthetics are: x, y, and color.
- Aesthetics are specific to the type of plot (or layer) you're adding to your visual.
- For example, a scatterplot (`geom_point`) and a line (`geom_line`) will share x and y, but only a line chart has a linetype aesthetic.

Layers

- ggplot lets you combine or add different types of visualization components (or layers) together.
- The command ggplot does not actually create any plot, rather it prepares a "blank canvas" for further plotting

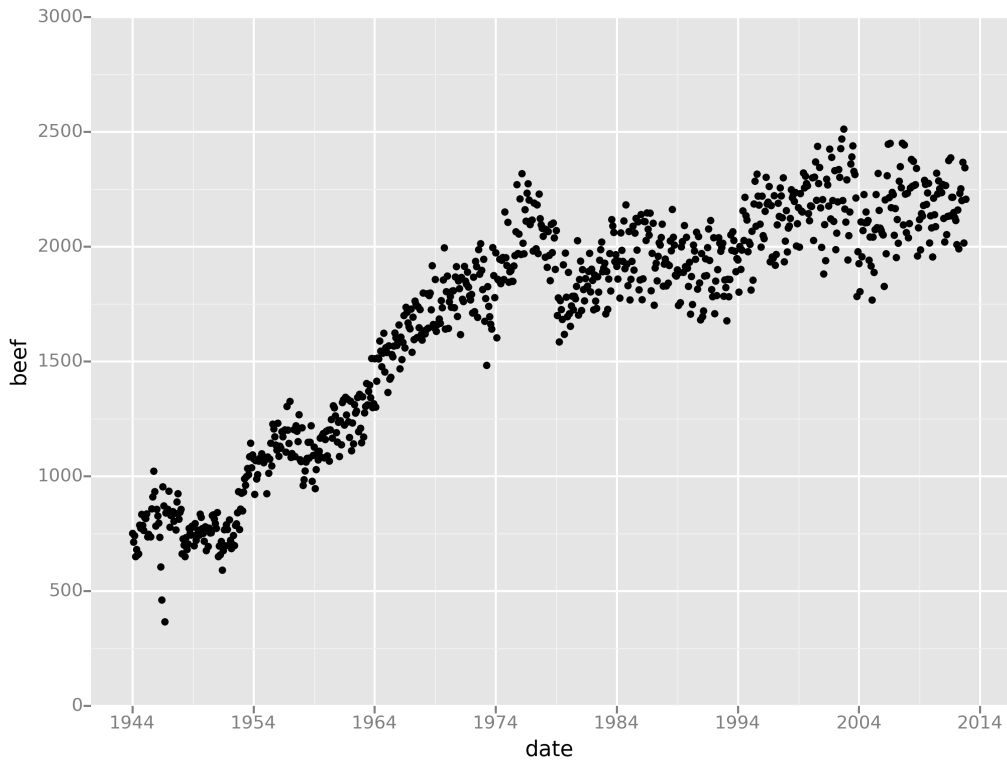
Start with a blank canvas.

```
p = ggplot(aes(x="date", y="beef"), data=meat)
p
```



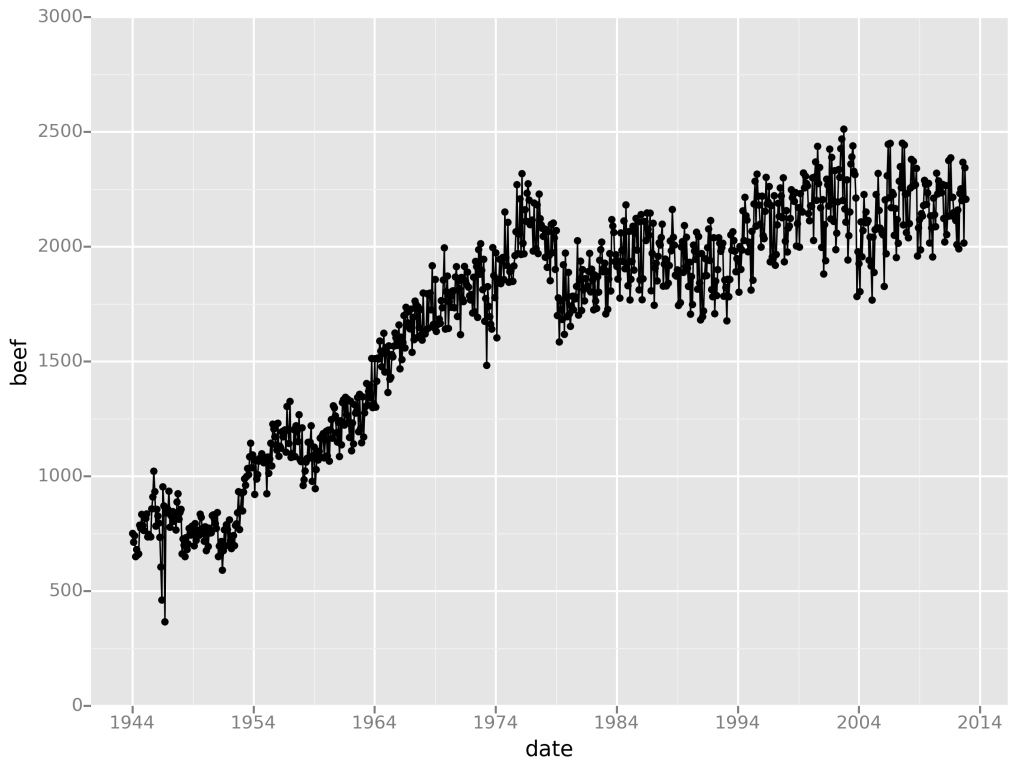
Add some points.

```
p + geom_point()
```



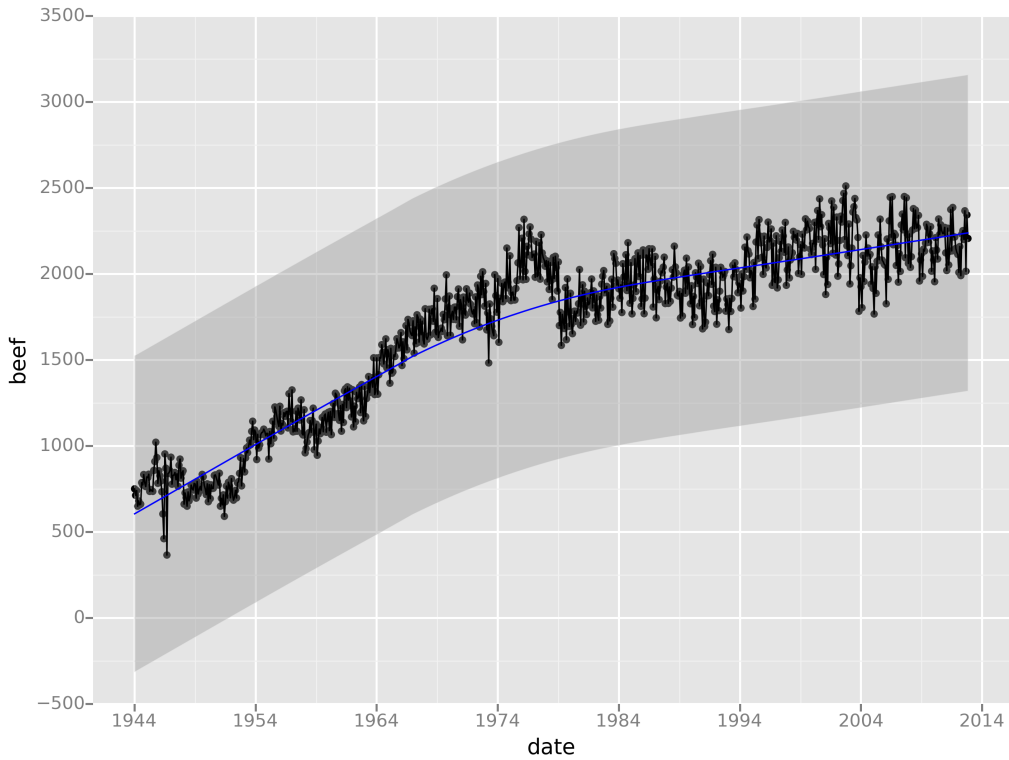
Add a line.

```
p + geom_point() + geom_line()
```



Add a trendline.

```
p + geom_point() + geom_line() +
 stat_smooth(color="blue")
```



Geometric objects (geoms) are the visual representations of (subsets of) observations.

- **Univariate** - single numeric variable
- **Bivariate** - two numeric variable
- **Multivariate** - Multiple variables

Faceting

The faceting approach supported by ggplot2 partitions a plot into a matrix of panels. Each panel shows a different subset of the data. There are two faceting approaches:

- `facet_wrap(~cell)` - univariate: create a 1-d strip of panels, based on one factor, and wrap the strip into a 2-d matrix
- `facet_grid(row~col)` - (usually) bivariate: create a 2-d matrix of panels, based on two factors

Scales

- A scale determines how an attribute of the data is mapped into an aesthetic property of a geom (e.g., the geom's position along the x axis, or a geom's fill color in a color space).