**Smoothing out Time-Series Data**

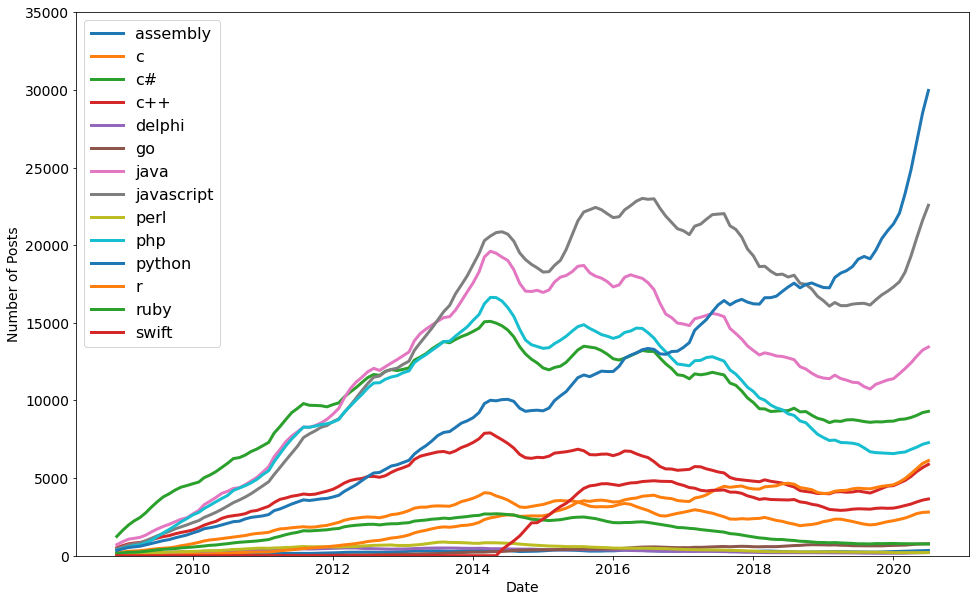
Looking at our chart we see that time-series data can be quite noisy, with a lot of up and down spikes. This can sometimes make it difficult to see what's going on.

A useful technique to make a trend apparent is to smooth out the observations by taking an average. By averaging say, 6 or 12 observations we can construct something called the rolling mean. Essentially we calculate the average in a window of time and move it forward by one observation at a time.

Since this is such a common technique, Pandas actually two handy methods already built-in: [rolling()](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.rolling.html) and [mean()](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.core.window.rolling.Rolling.mean.html). We can chain these two methods up to create a DataFrame made up of the averaged observations.

1. # The window is number of observations that are averaged
2. roll\_df = reshaped\_df.rolling(window=6).mean()
4. plt.figure(figsize=(16,10))
5. plt.xticks(fontsize=14)
6. plt.yticks(fontsize=14)
7. plt.xlabel('Date', fontsize=14)
8. plt.ylabel('Number of Posts', fontsize=14)
9. plt.ylim(0, 35000)
11. # plot the roll\_df instead
12. for column in roll\_df.columns:
13. plt.plot(roll\_df.index, roll\_df[column],
14. linewidth=3, label=roll\_df[column].name)
16. plt.legend(fontsize=16)

Now our chart looks something like this:



Play with the window argument (use 3 or 12) and see how the chart changes!