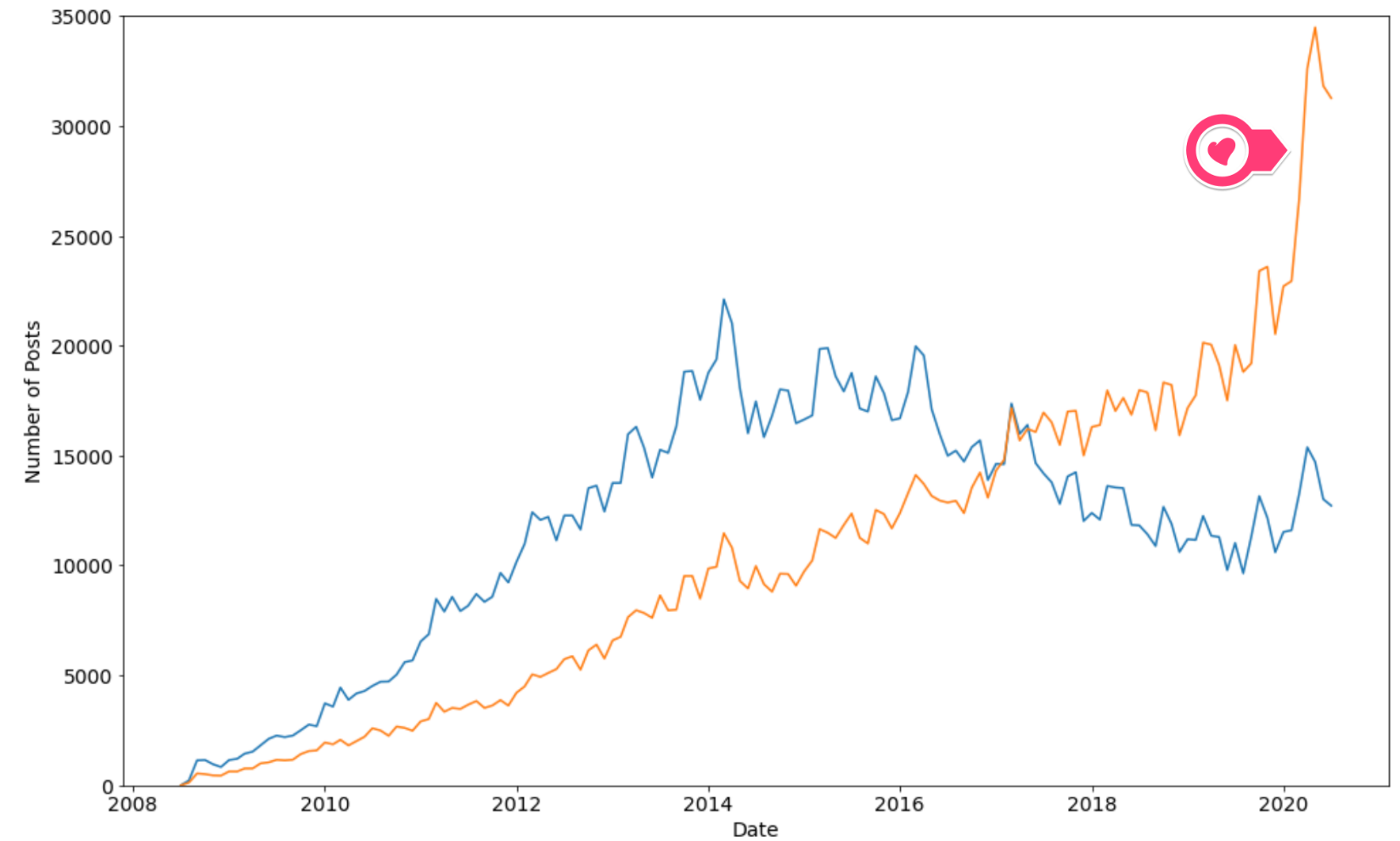
**Multi-Line Charts with Matplotib**

**Solution: Two Line Charts Next to Each Other**

The trick is simply calling the .plot() method twice. That's all there is to it! =)

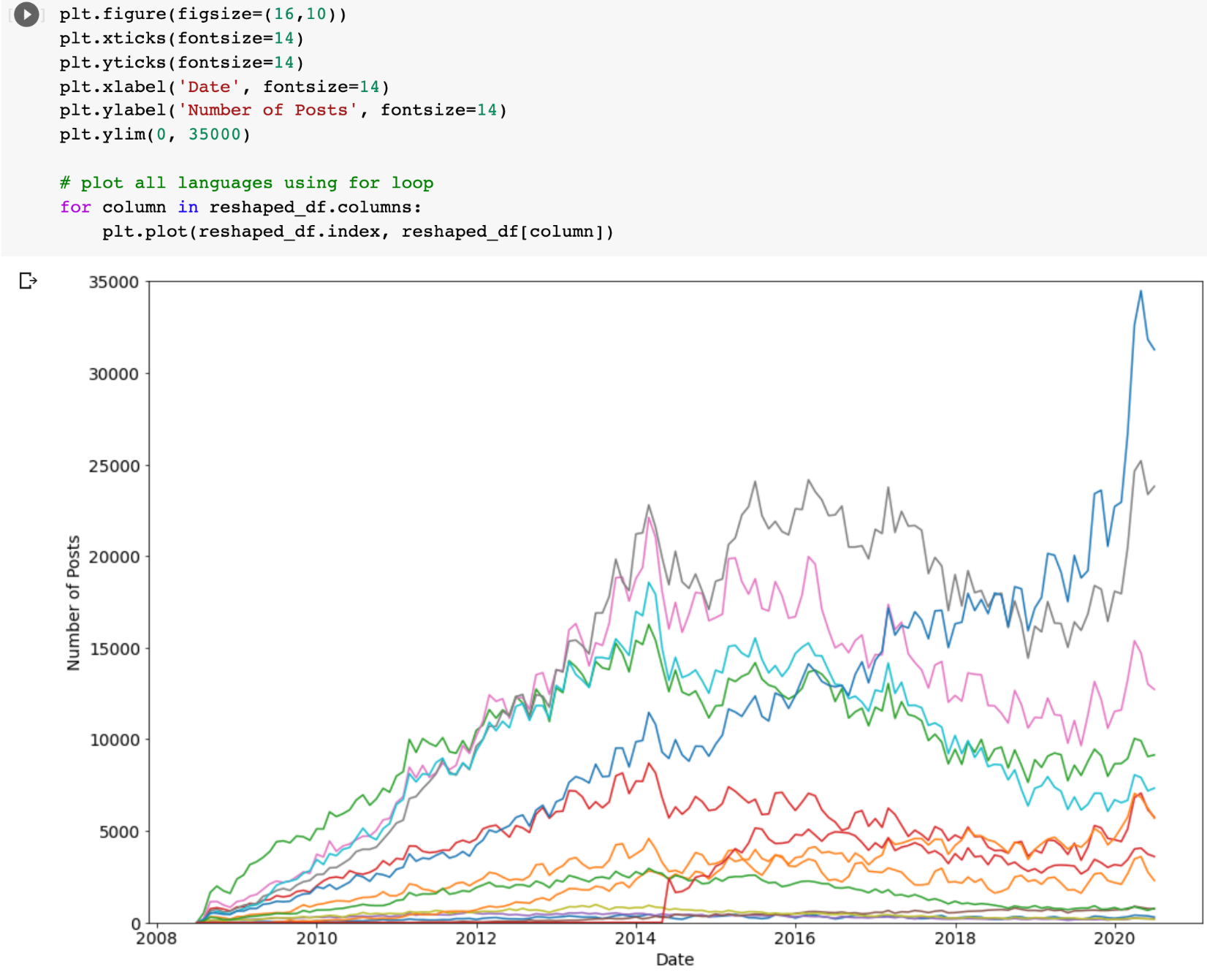
1. plt.figure(figsize=(16,10)) # make chart larger
2. plt.xticks(fontsize=14)
3. plt.yticks(fontsize=14)
4. plt.xlabel('Date', fontsize=14)
5. plt.ylabel('Number of Posts', fontsize=14)
6. plt.ylim(0, 35000)
8. plt.plot(reshaped\_df.index, reshaped\_df.java)
9. plt.plot(reshaped\_df.index, reshaped\_df.python) # Tadah!



But what if we wanted to plot all the programming languages on the same chart? We don't want to type out .plot() a million times, right? We can actually just use a for-loop:

1. for column in reshaped\_df.columns:
2. plt.plot(reshaped\_df.index, reshaped\_df[column])

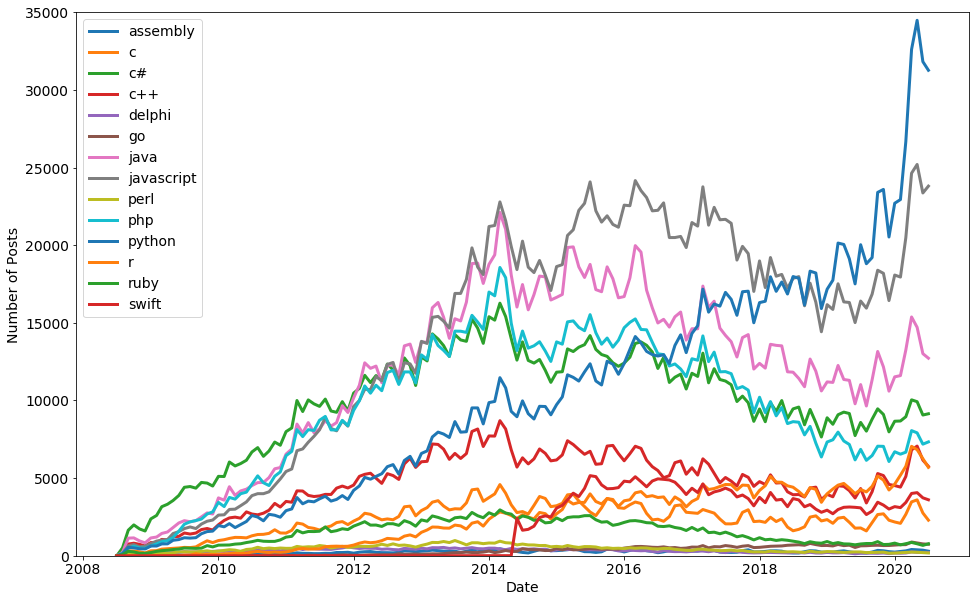
This will allow us to iterate over each column in the DataFrame and plot it on our chart. The final result should look like this:



But wait, which language is which? It's really hard to make out without a legend that tells us which colour corresponds to each language. Let's modify the plotting code to add a label for each line based on the column name (and make the lines thicker at the same time using linewidth). Then let's add a legend to our chart:

1. plt.figure(figsize=(16,10))
2. plt.xticks(fontsize=14)
3. plt.yticks(fontsize=14)
4. plt.xlabel('Date', fontsize=14)
5. plt.ylabel('Number of Posts', fontsize=14)
6. plt.ylim(0, 35000)
8. for column in reshaped\_df.columns:
9. plt.plot(reshaped\_df.index, reshaped\_df[column],
10. linewidth=3, label=reshaped\_df[column].name)
12. plt.legend(fontsize=16)

We should now see something like this:



Looks like Python is the most popular programming language judging by the number of posts on Stack Overflow! Python for the win! =)