**Data Visualization with Matplotlib**

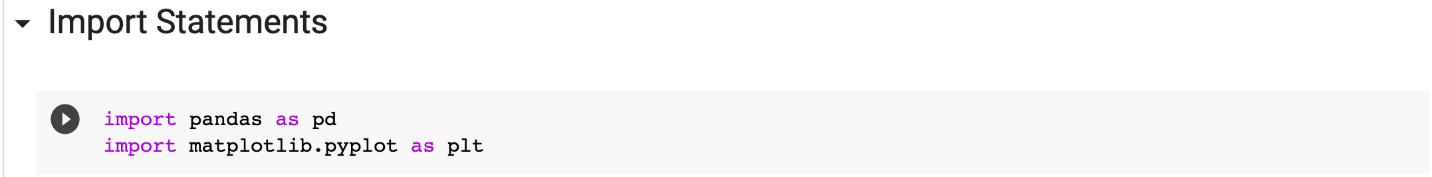
**Matplotlib**

To create our first charts, we're going to use a library called [Matplotlib](https://matplotlib.org/). There are many different libraries in Python to help us create charts and graphs. Matplotlib is an incredibly popular one and it works beautifully in combination with Pandas, so let's check it out.

First, we have to import Matplotlib.

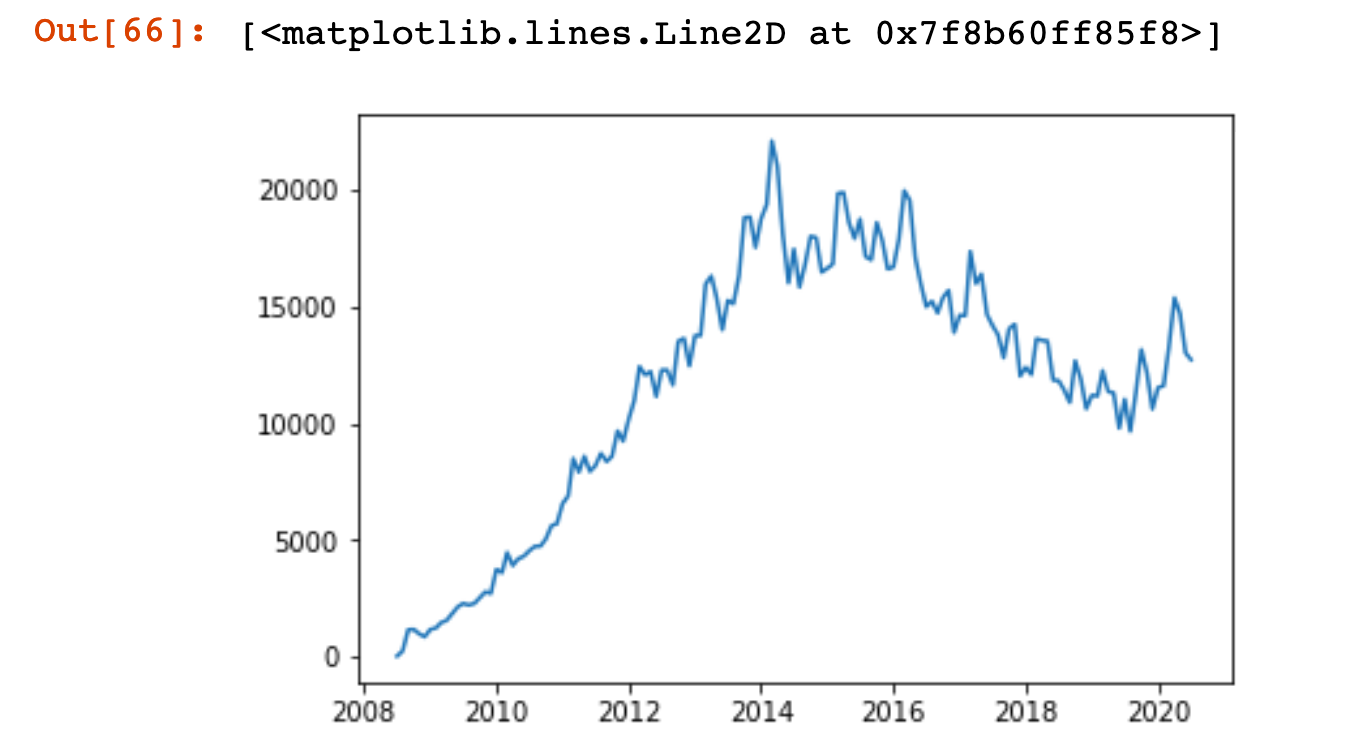
1. import matplotlib.pyplot as plt

Let's do this at the top:



**Mini Challenge**

You can actually show a line chart for the popularity of a programming language using only a single line of code. Can you use the [.plot() documentation](https://matplotlib.org/3.2.1/api/_as_gen/matplotlib.pyplot.plot.html#matplotlib.pyplot.plot) to figure out how to do this? Try and plot the popularity of the Java programming language. Here's what you're aiming for:



**Solution**

All you need to do is supply the values for the horizontal axis (the x-values) and the vertical axis (the y-values) for the chart. The x-values are our dates and the y-values are the number of posts. We can supply these values to the .plot() function by position like so:

1. plt.plot(reshaped\_df.index, reshaped\_df.java)

or like so if you prefer the square bracket notation.

1. plt.plot(reshaped\_df.index, reshaped\_df['java'])

**Styling the Chart**

Let's look at a couple of methods that will help us style our chart:

.figure() - allows us to resize our chart

.xticks() - configures our x-axis

.yticks() - configures our y-axis

.xlabel() - add text to the x-axis

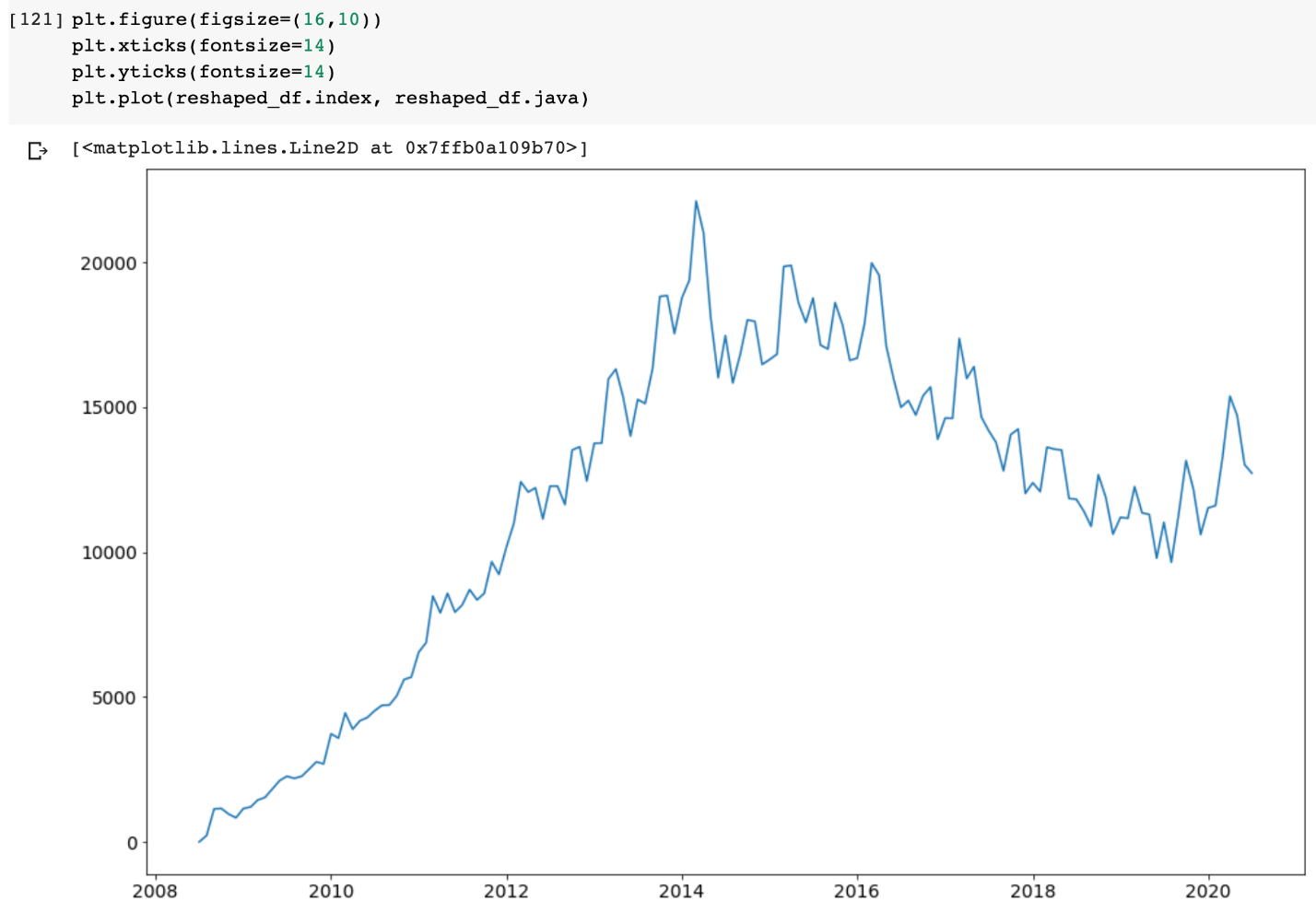
.ylabel() - add text to the y-axis

.ylim() - allows us to set a lower and upper bound

To make our chart larger we can provide a width (16) and a height (10) as the figsize of the figure.

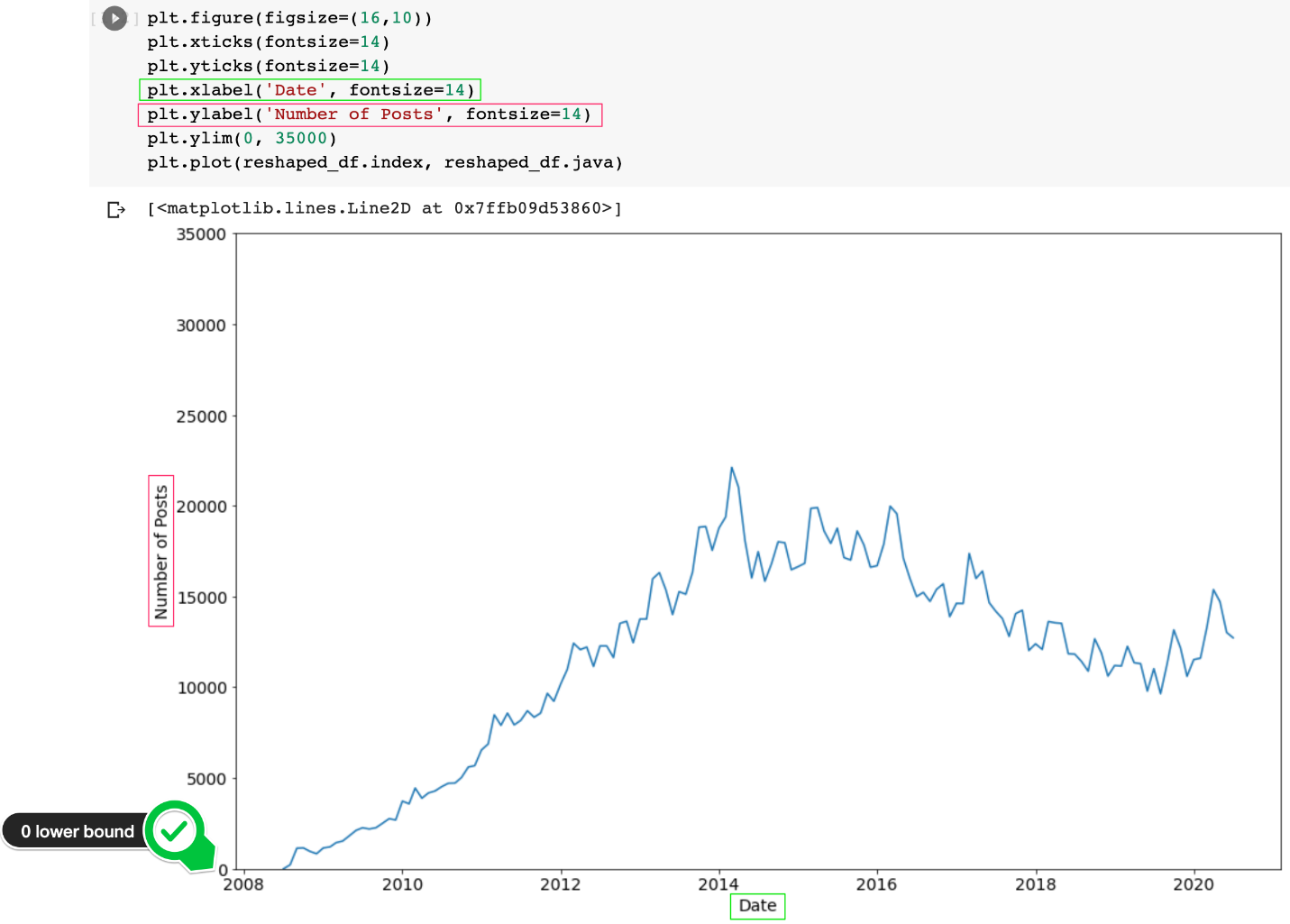
1. plt.figure(figsize=(16,10))
2. plt.plot(reshaped\_df.index, reshaped\_df.java)

This will make our chart easier to see. But when we increase the size of the chart, we should also increase the fontsize of the ticks on our axes so that they remain easy to read:



Now we can add labels. Also, we're never going to get less than 0 posts, so let's set a lower limit of 0 for the y-axis with .ylim().

1. plt.xlabel('Date', fontsize=14)
2. plt.ylabel('Number of Posts', fontsize=14)
3. plt.ylim(0, 35000)



Challenge

Now that you've successfully created and styled your chart, can you figure out how to plot both Java and Python next to each other? The result should look something like this:

