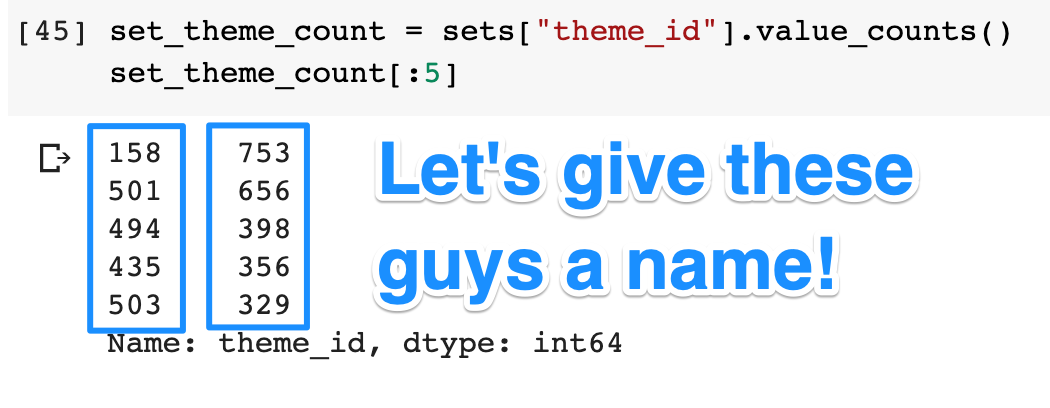
**How to Merge DataFrames and Create Bar Charts**

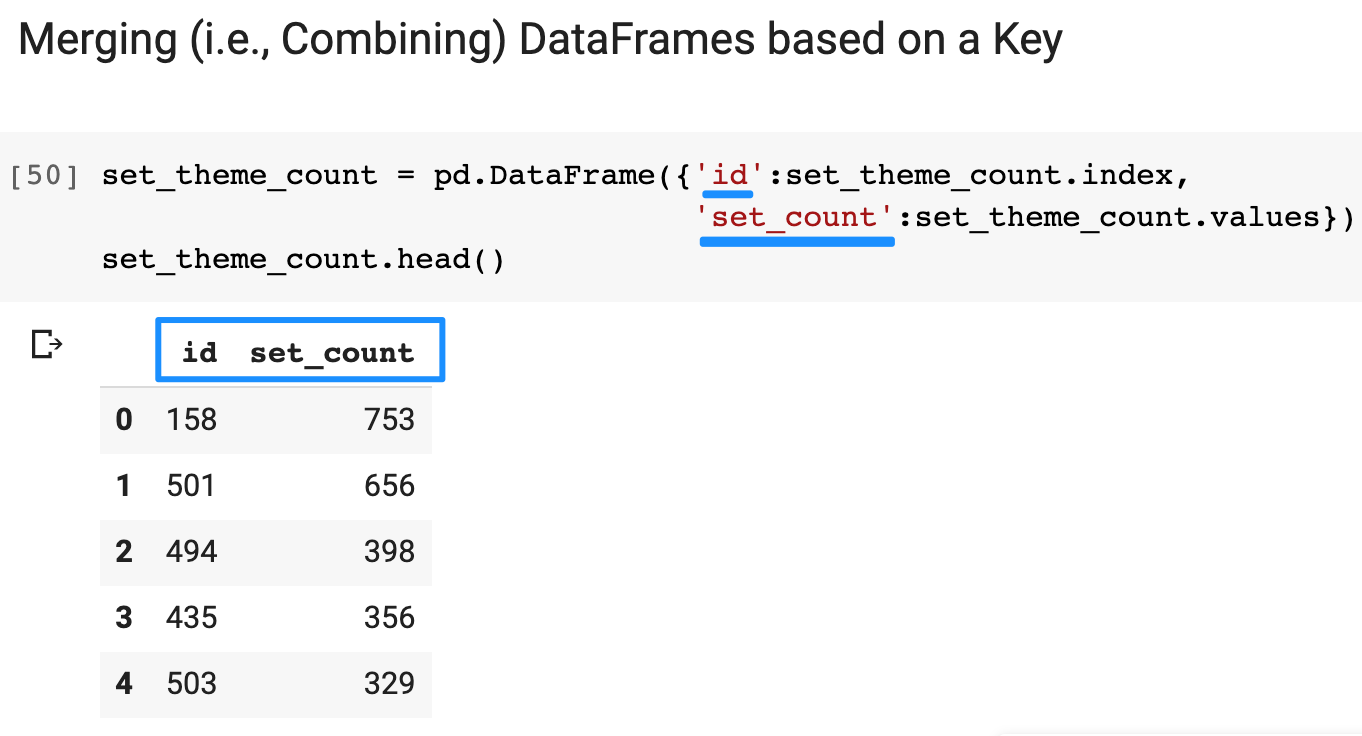
Wouldn't it be nice if we could combine our data on theme names with the number sets per theme?

Let's use the [.merge() method](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.merge.html?highlight=merge#pandas.DataFrame.merge) to combine two separate DataFrames into one. The merge method works on columns with the same **name** in both DataFrames.

Currently, our theme\_ids and our number of sets per theme live inside a Series called set\_theme\_count.



To make sure we have a column with the name id, I'll convert this Pandas Series into a Pandas DataFrame.



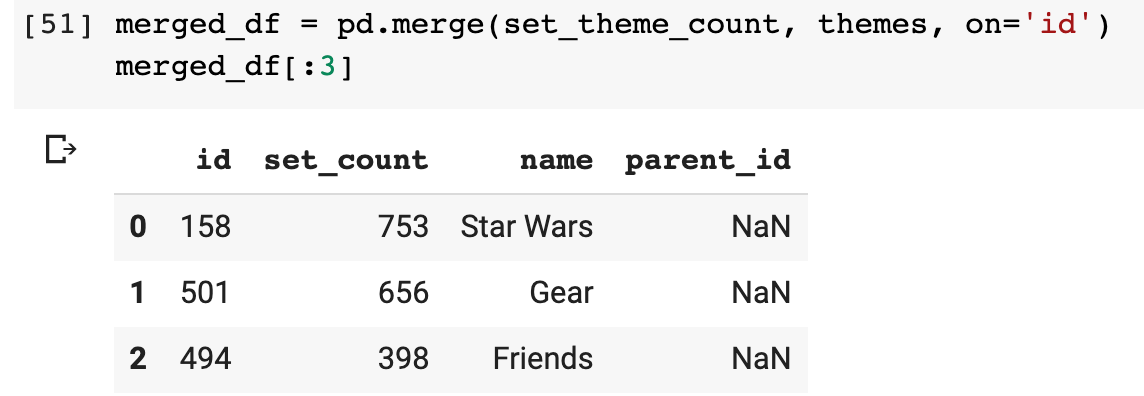
Here I'm providing a dictionary to create the DataFrame. The keys in the dictionary become my column names.

**The Pandas .merge() function**

To [.merge()](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.merge.html) two DataFrame along a particular column, we need to provide our two DataFrames and then the **column name** on which to merge. This is why we set on='id'. Both our set\_theme\_count and our themes DataFrames have a column with this name.

1. merged\_df = pd.merge(set\_theme\_count, themes, on='id')

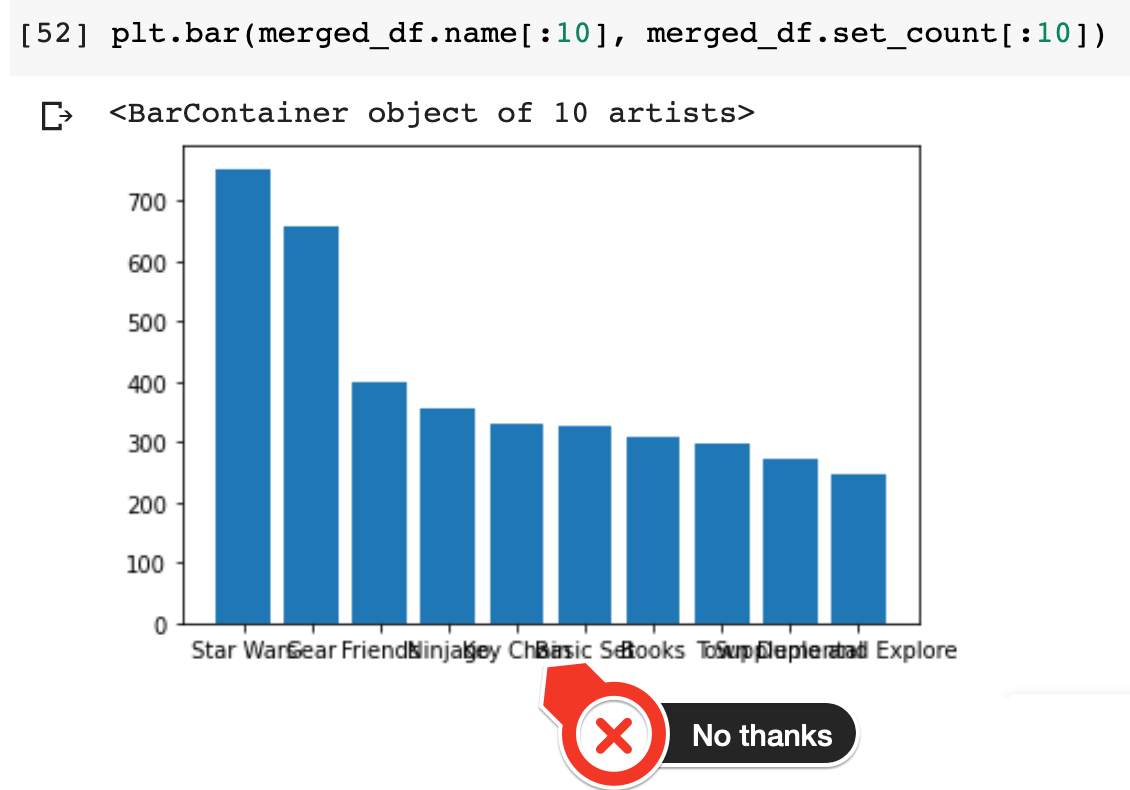
The first 3 rows in our merged DataFrame look like this:



Aha! Star Wars is indeed the theme with the most LEGO sets. Let's plot the top 10 themes on a chart.

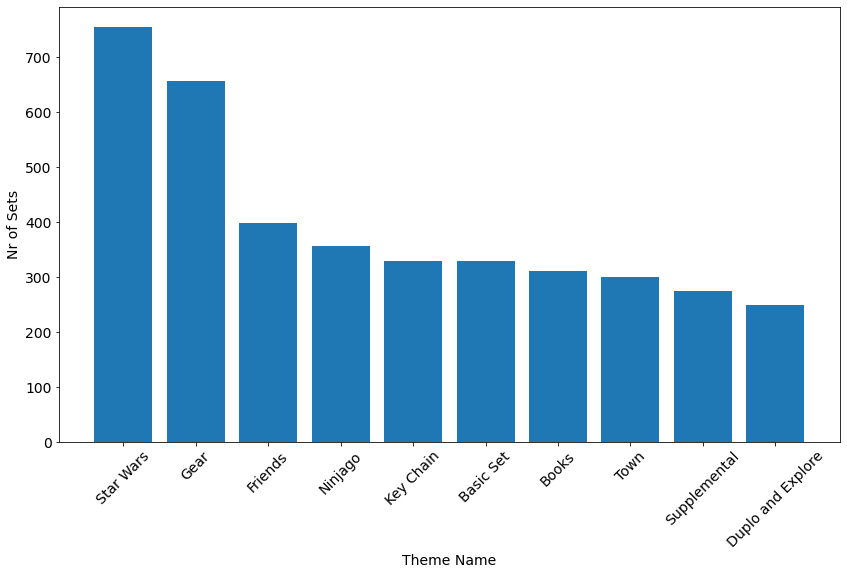
**Creating a Bar Chart**

Matplotlib can create almost any chart imaginable with very few lines of code. Using [.bar()](https://matplotlib.org/3.3.2/api/_as_gen/matplotlib.pyplot.bar.html) we can provide our theme names and the number of sets. This is what we get:



That worked, but it's almost unreadable. 😩 The good thing for us is that we already know how to customize our charts! Here's what we get when we increase the size of our figure, add some labels, and most importantly, rotate the category names on the x-axis so that they don't overlap.

1. plt.figure(figsize=(14,8))
2. plt.xticks(fontsize=14, rotation=45)
3. plt.yticks(fontsize=14)
4. plt.ylabel('Nr of Sets', fontsize=14)
5. plt.xlabel('Theme Name', fontsize=14)
7. plt.bar(merged\_df.name[:10], merged\_df.set\_count[:10])



Niiiiice.😌 So what can we see here? Well, a couple of these themes like Star Wars, Town, or Ninjago are what I would think of when I think of LEGO. However, it looks like LEGO also produces a huge number of ... books and key chains?!?! I guess I'm showing my age here, but it's interesting that the LEGO company seems to produce so much more these days than just plastic bricks. The 'Gear' category itself is huge and includes everything from bags to pencil cases apparently. Has LEGO strayed from its core business or is it successfully diversifying? That we can't answer from our dataset. I'll leave that one up to a business school case study to decide. 🤷‍♀️