Explanation code first question:

Biggest predictor of co2 output

First step:

Let’s import the necesseary libraries to create this code. I always import more then normal, why?

There are more roads that leads to Rome, sometimes the function you like to use don’t always work on your code. Then it’s always an advantage that you have more options to make your code work like you want it to work.

Second step:

Import the csv files found on the world in data url, i use 8 files in total for this code because you want to have a complete dataframe.

Third step:

Using .os.chdir method used to change the current working directory to specified path. It takes only a single argument as new directory path.

Fourth step:

Python .glob() method returns a list of files or folders that matches the path specified in the pathname argument.

Fifth step:

Now let’s merge all files together using .concat() and .to.csv(), that works great because you don’t need to fill in all the files seperatly and it’s simple.

Sixth step:

Now let’s use .groupby() function to split the data into groups based on criteria.

Seventh step:

Now we need to use .drop\_duplicates(), because that helps us removing duplicates out of our dataframe.

Eight step:

This dataframe is to big fors ure! There are some columns simular to eachother, so we use .drop() function to fix this problem. And if we run the code you see a big difference and it already looks allot more organized.

Nineth step:

The names of this dataframe is to big and messy, so we are going to .rename() them and make it more easy to understand also for the less smart crowd.

Tenth step:

The .fillna() method replaces the NULL values with a specified value, The .fillna() then returns a new Dataframe object unless the inplace inplace parameter is set to True, in that case the .fillna() method does the replacing in the original Dataframe instead. But we don’t use that in this case. Instead of that we use .mean() function so we immediatly calculate the mean(average) of the given data set.

Eleventh step:

Now let’s find the pairwise correlation of all columns in this Dataframe and we use the .corr() function together with the pearson method known as pearson’s r.

Twelfth step:

We alse use kendall method together .corr() to make you see the difference between the both method’s and also a different outcome as result!

Thirteenth step:

Let’s import two more libraries to give us some more options in using different functions.

Fourteenth step:

Let’s go and start plotting the calculation and find positive and negative correlations with different types of plots, the color also help us to find positive and negative correlations looking the first plot and look good you will see that how darker the color is(blue) the more stronger the plus in correlation is in comparing to the negative correlation the color is very light. I use three plots including two heatmaps with.

Fifthteenth step:

I also make some .regplot() examples to show you if some methods actually works or not, there you see that ‘acute care’ and ‘gdp’ is still a growing problem because of the growing population. The .regplot() method is used to plot data and a linear regression model fit. And you see other comparings also have the same issues and the regression line shows that clearly.

Sixteenth step:

I give more examples including a .boxplot(), that can be a Numpy array of python list of Tuple of arrays. It’s also known as Whisker plot!