tds\_CAN\_clim\_data.py is the file being used to make the “follow along” version of the exercise as detailed at the website: <https://towardsdatascience.com/how-to-group-yearly-data-by-periods-5199a1dba5db>

Where the author says df I am saying dfCAN.

Also using <https://towardsdatascience.com/basic-time-series-manipulation-with-pandas-4432afee64ea> as a follow along to make the time component of the dataframes.

As it relates to my project scbData.py, can I start with the concatenated data (i.e. all the years being considered already strung together after the pd.concat() command has been performed and also after the getattr() command is used to make dfs for all the individual elements and those dfs encompass all the years being considered) or does it make sense to make the dfs for each component (EN)- for each year- prior to concatenating any data for the EN so that each years’ dataframe can be made and then the data attached to said year can be manipulated to attach a time component to that individual year – after which the dfs can be concatenated? ???? !!! ???

Try to use the old code from the datacleaning.py file and see if the set of bridges joined on STRUCNUM and EN to come up with a uniform set of elapsed times between observations of condition states.

I’m back to the drawing board on how to make plots of the data and fit lines/perform regression- because I’m not sure how to make the number of days between observations of the condition states equal. If I am able to eliminate data down to the point that only data from bridge STRUCNUM present in all the years considered also only has condition state data for said STRUCNUM corresponding to the EN that were observed in all years being considered as well, the data would be uniform in its elapsed time between observations- and thusly seem more uniform in the presentation of its trained model.