Problem set 6: Getting, cleaning, and checking your data

For this problem set, you will submit a single python file (.py), but we will not be able to run your python file, since it will reference data that you are using for your final project (data that is on your own computer only). So that we may read and understand what you have done, please add comments throughout your python file that:

1. Explain what your code does (you don't have to explain every line, just provide general explanations for blocks of code like "in this code I'm checking for outliers or missing data")
2. Explain what you found through running your code, and how that changed your decisions going forward. (For example: "This variable has some very large outlier values. I want to restrict my data to a reasonable range of values on this variable, which I do in the following line of code.")

The problems set is divided into three parts.

Part A. Getting your data.

If you have multiple data sources or files, explain this and repeat the following steps for each data source.

1. If the data you're using comes from a website and you obtain it through a flat file download, explain in a comment what website you use, and then write the code to read in the flat file.
2. If the data comes from web scraping or an API, provide the code you used and discuss how it worked.
    a. Ultimately (after testing and debugging on small samples) you should run your web scraping or API pinging code *once*, saving the obtained data somehow (e.g. as a flat file or files with e.g. .to_csv, or somehow as pickles). That way you can re-run your analysis code repeatedly, "refreshing" your data using files saved to your computer rather than waiting for the web scraping or API code to work (and pinging websites over and over). Show the code you used to save the data.
    b. In this case, I recommend that you put the code that gets the data in one script like 1-get-data.py, and then your analysis code in a second script like 2-analysis.py. *However, for the submission for this pset, put all your code in one file* (just copy and paste from your multiple files if needed).
3. If the data you are using comes from a different source or method please explain.

Part B. Cleaning and manipulating your data into tables for your preliminary goal.

1. Explain your (preliminary) analysis plan for your data, that is, answer the question: what do you want to get out of this data? Examples:

- "This is data on trading flows between countries for various wines. I want to see if there are significant trends downward or upward in exports for different countries."
- "This is data on movies, with their IMDB data on genres, and number of reviews, from 2010 through 2023. I want to describe which genres have been growing faster than others in terms of total number of reviews and rank of "most popular movie by number of reviews"."

- “This is data on real estate transactions in NYC for 2007, 2008, 2009, and 2010, during the housing crash. I want to describe which kind of residential real estate had the largest percentage fall in value over this period: small detached homes, walk-up rental buildings, elevator rental buildings, condos, or coops?”

The questions or objectives for your use of the dataset will guide how you want to clean the data. These don’t have to be set in stone right now, but I want you to have some general objective in mind. Moreover, because you are scripting, in later analysis for problem sets or for your final project submission, you can revise your code here and re-run with various changes (most likely you will have some changes, like adding more variables).

2. Describe the table or tables that you want out of your data in order to achieve your preliminary analysis goal.

   a. For example, for the movies question, I want a table by year-genre combination with columns
      i. [year] [genre] [total number of reviews]

      I also want a table by year only
      ii. [year] [most popular movie tconst] [genre of most popular movie]
   b. If your data already comes in the format you want to do analysis without any cleaning, then say so. You are very lucky, that almost never happens.

3. Write the code needed to take the data you loaded in and produce the table or tables that you want for your preliminary analysis plan. By the end of this part of your script you should have a dataframe or a few dataframes that are appropriate for your analysis. (Again, most likely you will revisit and revise this code later when you are actually doing the analysis and realize you need to change the data cleaning “upstream.”)

Part C. Checking your data.

From the dataframes you produced in Part B, check for

1. Distributions
   a. use .value_counts() and .hist() to explore variable distributions
   b. Do the distributions make sense? Any surprises/concerns?
2. Missing or ridiculous values (e.g. negative numbers that should be positive)
   a. Come up with a plan for dealing with these values. Could be as simple as: “I’m going to drop these rows.” If your plan is simple, write code to do it.
3. Outliers (numbers very far out in the distribution).
   a. Discuss whether and how you will remove these.

Also discuss and implement any other data checks that come to mind.