DSCI 510 Final Project

Part – 1 (Feedback submission 1) :

1. **Objectives:**
    a. Find three (3) data sets on the web that are of interest to you.
        - One must require "scraping" (i.e. not available via external API)
        - One must be available via external public API. (You should be able to access it without a ton of trouble)
        - The third can be an API, scraped, or a database

    NOTE: Sites must be such that they require automation to extract data; If you can just cut-and-paste the data, it's not appropriate.
    b. Describe in a paragraph what analysis or presentation can be expected from the combined data.

2. **Result expectations:**
    a. A word document including the proper links to the data.
    b. Description of how the student plan to use the datasets and the type of analysis/questions student would like to answer based on their data.

Part – 2 (Feedback submission 2):

1. **Objectives:**
    a. Scraping of the data and show sample data.
    b. Use the API to gather the data and show sample data.
    c. Model data into ORM, SQL schema, Data Frame, CSV Files, etc and create diagram to show how the data joins.
    d. Describe maintainability/extensibility of model.

2. **Result expectations:**
    a. Submission should contain static dataset files, .py scraping file containing command line functionalities, a drawing, and a README text file explaining how to run the code.
    b. There should be these three ways of running the file:
        1. scraper.py --scrape
            - This will scrape the data but return only 5 entries of each dataset.
        2. scraper.py --static <path_to_dataset>
            - This will return the static dataset scraped from the web and stored in database or CSV file
        3. scraper.py
            - Return the complete scraped datasets.
            -

Part – 3 (Final submission):

1. **Objectives:**
    a. Analyze/Draw conclusions from data.
    b. How does the whole combined data system work?
    c. What facts the data tells you?
    d. Graphs, insights, analysis, etc.

2. **Results expectations:**
    - Submission should contain .py files (.py), database files containing datasets, project description (.pdf), and README file (.txt) to understand how to run your code.
    - For the .py files, command-line arguments that are needed:
        - There should be a .py file that executes data scraping and performs complete analysis. This file can take one argument:
            - --static
            1. This opens the stored database and performs analysis on stored data
            - If no arguments are passed, scrape the complete data, store in a database, then perform analysis on the database (most recently scraped data).

    Note:

    - Students are encouraged to organize all scraping code and analysis code in only one .py file which should be able to be executed in command line. But if they cannot figure out how to manage all the code in one .py file, it'll be fine to include scraping code and some simple analysis code in one scraper.py file, which should be able to be executed in command line.
    - Student can include other complicated code like generating pictures or applying models in other .py files, which don't have to be able to run in the command line.
    - For the database files submitted for the final project, they can be .db files as well as other static dataset files like .csv, .json, .xlsx, etc.
    - For the project description document, these should be included:
        - Motivation surrounding project topic
        - Brief description of data sources
        - Analysis performed
        - Conclusions drawn

Sample Projects: https://drive.google.com/drive/folders/158q0zDPXgu-zsgs5YssE4-NtR_pnCyIx?usp=sharing

Note: The final project is divided into three submissions in which part 1 is not graded.

You can use any source of data on the web.

Here are some sites that list available datasets/APIs if you need inspiration:

https://datasetsearch.research.google.com/

https://github.com/awesomedata/awesome-public-datasets

https://www.programmableweb.com/category/all/apis