Project 1: Data analysis

Overview

- For this project, you will **analyze one** of the three Kaggle datasets provided for this project.
    - Kaggle is a platform for data scientists and machine learning engineers to find and publish datasets, explore and build models, and participate in competitions to solve data science problems.
    - More information about the Kaggle platform can be found in the How to Use Kaggle documentation.
- The goal of this project is to combine what you're learning about importing and processing plain text files *and* creating and modifying dictionaries. You will **explore the data and perform calculations** using a Kaggle dataset.
- The program that you create will:
    - Import a Kaggle `.csv` dataset and read it into a list of dictionaries or a nested dictionary.
    - Manipulate the read-in data from the dataset to perform two or more calculations
    - Write the results of your calculations to a plain text file (`.txt` or `.csv`)
- **You can collaborate** with others (in a team of 2 or 3 members) and/or use GenAI. However, this must be reported in your submission. Each group member must make individual submissions, including individual videos.
- **There is a checkpoint for this assignment due on October 6.** This is to ensure that you are making progress on the project.
- **The rubric for this assignment** is attached at the end. Please consult it before submitting your project to ensure you have completed everything.

Tasks

Part 1: Download a dataset

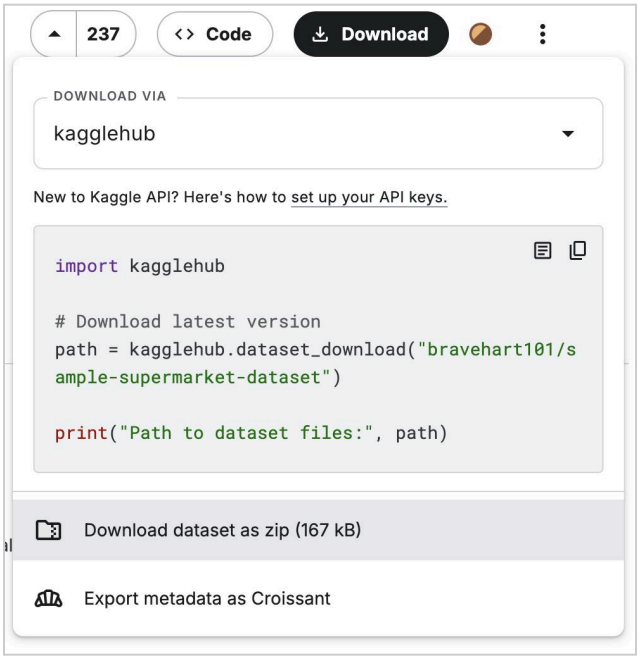
Start by browsing the Kaggle datasets listed below:

- Sample Superstore Dataset: A dataset containing sample superstore data including sales and shipment information. Uploaded to Kaggle by Aman Sharma.
- penguins: A dataset containing detailed information about penguins gathered from Dr. Kristen Gorman and the Palmer Station, Antarctica LTER. Uploaded to Kaggle by Data Science Sean.
- Agriculture Crop Yield: A dataset containing agricultural data based on various factors. Uploaded to Kaggle by Samuel Oti Attakorah.

Explore the dataset on Kaggle and note the variables in each dataset. Recall that in the context of data analysis, a variable refers to a property in the data that you can quantify or measure.

Select a dataset that interests you, and begin to formulate what calculations you want to perform using the variables in the data.

Download a `.zip` file of the dataset using the **Download** button. Extract the files from the zipped file and add the `.csv` file to a new folder where you will work on this project.



Part 2: Read and analyze the .csv file

Using your knowledge of the Python `csv` module, **import the `.csv` file**.This must be done inside a function(s). Analyze the dataset to determine the following:

- A list of the variables in the dataset (or in other words, the names of each column).
- A sample entry (row) in the dataset.

- The number of rows in the dataset.

You don't have to include the above in your write-up. These are here to get you started thinking about the data you have.

Part 3: Decide what to calculate

Decide what calculations you will perform using the data you read from the file.

The calculations you perform should require some processing and analysis of the data. They should reference **at least 3 columns from the dataset**, so plan accordingly (see Part 6).

If you are working alone, you only need to perform two calculations using the dataset. If you are working in a group, each team member must contribute two distinct data calculations. For example, a 2-person team should perform 4 different calculations using the dataset. Each calculation must involve analysis of at least one unique variable/column not used in other team members' calculations.

Similar to homeworks, at the top of the python file include your name, student ID, email, and the collaborators (other students and/or genAI tools). Also indicate which student created which functions.

Example Calculations:

- The average value of a numerical variable in a specific category
  Example: What is the average amount of rainfall (`Rainfall_mm`) in the `West`?
- The percentage of a dataset that contains a specific value for a categorical variable
  Example: What percentage of the crops measured in the `West` is `Wheat`?
- The percentage of a numerical variable that is greater or less than a value
  Example: What percentage of the `Barley` harvest in the `West` has a higher yield than `5` tons per hectare?
- A new calculation on a numerical variable for each category in a categorical variable
  Example: What is the average amount of rainfall (`Rainfall_mm`) in each region?

Part 4: Decide on the output file format

In addition to performing a calculation, your program must write the results to a file.

In discussion and lecture activities, you learn how to write information to plain text files like `.txt` and `.csv` files. Consider how structured or unstructured your results are and what would be the best format for presenting your results. If you created a new calculation for a category, it might be best presented in a `.csv` file. If you created an analysis on average values, a `.txt` file might present your results better.

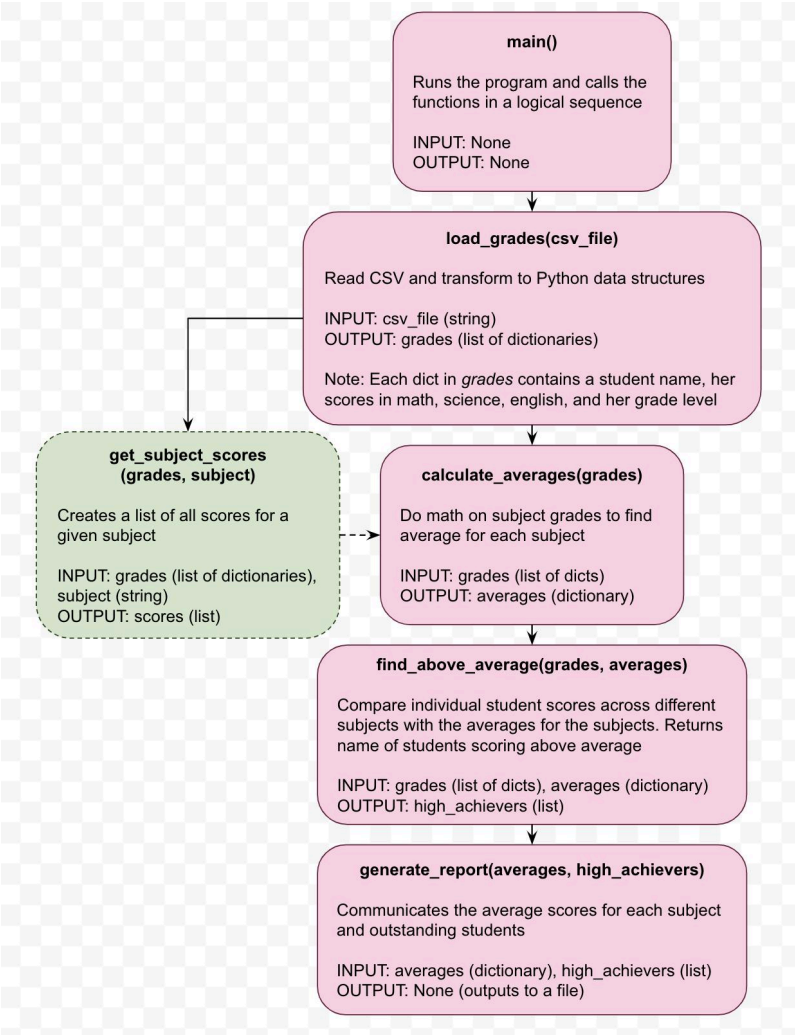
Part 5: Problem decomposition

Decompose your problem into separate functions. Break down the overall calculation into small enough tasks so that AI tools can help you write the code for those tasks. Note that some of the functions you write will and should call other functions. You must also **use dictionaries** in your program.

Once you've completed the problem decomposition, **create a diagram that represents the functions and their logical relationships**. The diagram should indicate clearly the name of each function, its description, its parameters, its return, and whether it is called from the main program or another function.

Example Diagram:

Assume the question we wanted to answer was, "*What is the average test score for each subject, and which students scored above average?*" For this question, a function decomposition diagram would look like the one below. The pink boxes/boxes with solid lines are core functions whereas the green box/box with dotted line is a helper function:



Note:

There are many tools you can use to create a flowchart. We used Google Drawings. Searching online for "tools for creating a flowchart" will give you several options to explore. You can hand draw too but you must ensure the readability of your flowchart and handwriting.

Checkpoint (Due Monday, October 6)

Please submit the name of the dataset you are using, the columns you will be working with, the calculations you will be performing, the function decomposition diagram, and the names of collaborators (if any) by Monday, October 6.

Part 6: Test Functions

Before you begin writing the code for your calculation functions, please write **four test cases** per function. Two must test general/usual cases and two test edge cases.

Part 7: Coding

Write code for the functions in your diagram! Remember that each function should be clearly defined, target a specific task, and be a reasonable length. Make sure that for your functions that reference the dataset, you are referencing **at least 3 columns** in your function. At the end, you should **have at least one output function** that writes your results into a `.txt` or `.csv` file.

As you work through the functions and debug your program, make sure to make plenty of git commits to prevent data loss and/or track changes with your team members. You will earn points for the first four commits, but we recommend that you have many more than that.

Best practices:

- Be sure to test your code by running it to make sure it works as intended before submission. Include edge cases in your testing functions.
- You may need to break down a function into smaller parts (or helper functions). That is perfectly fine and encouraged!
- If you find yourself repeating code, that's a good indicator that you need a loop or a helper function.

Note on genAI usage:

You're encouraged to use an AI tool to help you write your functions. Remember to critically evaluate the output that the AI tool gives. If the output is not suitable or contains errors, you should reprompt the AI tool or modify the code to meet your needs.

Part 7: Explain your work

A final part of the project is to explain your work and demonstrate your understanding of the program you created and the obstacles you encountered. If you are working in a group, each member must make individual videos and submit the project separately.

You will **share your explanation in a 1 to 2 minute video.** We recommend creating an outline of the points that you want to make.

Your video must include the following:

- How did you break down your problem into functions?
- Explain one of the most complex functions line by line.
- What challenges did you encounter and how did you solve them?

Rubric

Problem Decomposition (40 points)

Criteria	Max Points	Deduction Scenarios
Calculations	20	-20 if there aren't enough calculations (two calculations per member) -10 if a calculation lacks sufficient complexity
Function Diagram	20	-20 if no diagram provided -10 if diagram is unclear, or poorly organized -10 if diagram doesn't show function relationships -10 if descriptions, parameters, and return values are not clearly defined

Code Implementation (100 points)

Criteria	Max Points	Deduction Scenarios
CSV Import & Data Transformation	15	-15 if no function imports a CSV file and outputs either a list of dictionaries or a nested dictionary -10 if CSV data is improperly transformed to Python data types (such as transforming csv numbers into python strings rather than ints)
Calculations *(2 per person)*	30	-15 per missing function -15 for each function that does not use 3 columns from the dataset -15 if a function uses the same variables/columns as another function
File Output Function	15	-15 if no output file created (.txt or .csv) or output file is empty -15 if hardcoded values are used instead of calculated results
Test Functions	40	-10 for each missing general (usual) test case -10 for teach missing edge case

Video Explanation (30 points)

Criteria	Max Points	Deduction Scenarios
Function Decomposition Explanation	10	-10 if doesn't explain how functions work together and what each does at a high level
Complex Function Description	10	-10 if no complex function is explained -5 if the description is not detailed (line by line)
Challenges	10	Explain what challenges you encountered and how you solved them

Other (30 points)

Criteria	Max Points	Deduction Scenarios
Checkpoint	10	-10 if you miss the checkpoint. Report the name of the dataset you are using, the columns you will be working with, the calculations you will be performing, the function decomposition diagram, and the names of collaborators (if any) by Monday, October 6.
Top of code information	20	-20 if missing required information at the top of the file such as who you worked with, who wrote each function, and how you used AI
