SI 201: Project 1 - Data Analysis

Overview

In this project, you will analyze a real-world dataset from Kaggle to practice skills you are learning in this course, including:

- importing and processing `.csv` files
- working with dictionaries
- designing and organizing your own Python code

Kaggle is a platform for data scientists and machine learning engineers to share datasets. More information about the Kaggle platform can be found in the How to Use Kaggle documentation.

You will choose **one** of the three recommended Kaggle datasets and use Python to:

- Read the dataset into a list of dictionaries or a nested dictionary
- Perform two or more calculations on the data (e.g. averages, mode, median, totals)
- Write the results of your calculations to a file (`.txt` or `.csv`)
- Organize code using multiple functions

Collaboration and Academic Integrity

You may work **individually** or in a **group of up to 3 students**.

- All members of your group must submit the same GitHub URL of the group's repository on Canvas.
- The group's GitHub repository should include each team member's individual video.
- Similar to homework, you may use GenAI. However, this must be reported in your submission and we don't recommend relying on it for writing the majority of your code.
- All collaboration and GenAI use **must be clearly reported** in your submission. You are responsible for all GenAI code that you use in your project.
- **If you choose to collaborate**, please refer to the "Collaborating with GitHub" document that can be found in Canvas under `Files` > `Useful Docs` > `Collab_with_Github.pdf` **before starting.**

Deliverables

1. **Checkpoint** (Due 02/16/26; This checkpoint is for progress only and will not receive feedback):
    - Submit a `.pdf` or `.docx` file to Canvas with your initial project plan (checkpoint details outlined after Task 5), including:
        - Name of dataset being used

- Columns you will be doing calculations with
- Calculations you will be performing
- Function decomposition diagram
- Names of collaborators

2. **Final Submission** (Due 02/20/26)
    - Submit the link to your group's Project 1 Github Repository, including:
        - Result file (either `.txt` or `.csv`)
        - All committed code for Project 1 (Tasks 6 - 7)
        - Each group member's explanation video (Task 8)
        - Function decomposition diagram

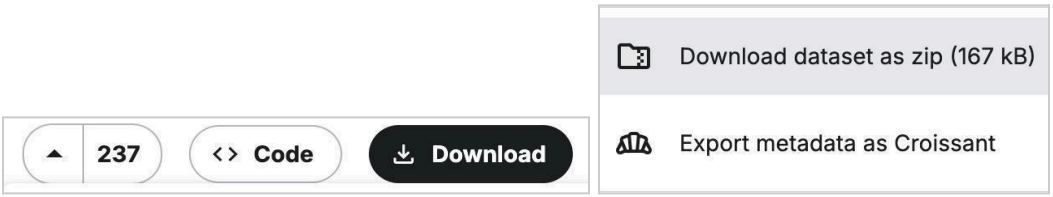
Task Breakdown

Task 1: Choose a dataset

Select one of the following Kaggle datasets:

- 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. 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.

Download a `.zip` file of your chosen 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.



Task 2: Read and Explore the Data

Using the Python `csv` module, **import the `.csv` file**.This must be done inside a function(s).

Analyze the dataset and include the following in the checkpoint write-up:

- 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.

Task 3: Define Calculations

Each student must define **two distinct** calculations (e.g. averages, totals) on the selected dataset.

- **Each calculation must utilize at least three columns of data**
- Group members must contribute their own calculations

Include the identified calculations and the corresponding dataset columns in your checkpoint write-up. Do not implement code in this step.

Example Calculations:

The following calculation definitions are based on the Kaggle IMDB dataset:

- *For each genre, what is the average IMDB rating of movies that have a runtime longer than 120 minutes?* (Columns used: `Genre`, `Runtime`, `IMDB_Rating`)
- *What is the total number of votes for movies released in each decade that have a Meta score above 75?* (Columns used: `Released_Year`, `Meta_score`, `No_of_Votes`)

Task 4: Choose Output Format

Decide whether results are best written to:

- `.csv` (structured/tabular results), or
- `.txt` (summary/narrative output)

If you created a new calculation for a category, it might be best presented in a `.csv` file. If you created an analysis with average values, a `.txt` file might present your results better.

Task 5: Function Decomposition Diagram

Break down your **defined calculations into separate functions**.

After identifying functions and their logical relationships, design a diagram showing:

- Each function's name and description,
- Parameters and return values
- How functions interact

You may use any diagramming tool (digital or hand-drawn, but it *must* be legible).

For an example of a good diagram, please refer to the appendix for this task at the end of this document.

Task 6: Test Functions

Before writing the code for your calculation functions, write **four test cases per calculation function**.

- Two test cases must test general/usual cases
- Two test cases must target edge cases

For aid with this part, please refer to Discussion 5 slides that can be found on Canvas under `Files` > `Discussions` > `Discussion 5` > `Discussion5_Slides`.

For writing test cases, **use a subset of your chosen dataset** and ensure that it meets the following criteria:

- Must have at least 15 rows (excluding header row)
- Must use the same format, values, and headers as the original dataset
- Must have diverse data values

For an example of a good subset of data, please refer to the appendix for this task at the end of this document.

Task 7: Code

Implement the functions from the diagram created in Task 5.

Each function should be clearly defined and target a specific task. Make sure calculation functions utilize **at least 3 database columns**. Verify that you **have at least one output function** that writes results to a `.txt` or `.csv` file.

*You will receive points for the first **four commits**, but it is best practice to commit often, especially after making any notable changes.*

Task 8: Video

Explain your project and your challenges in a 1 to 3 minute video. In the video, tell us:

- How you broke down the calculations into functions
- Use the diagram to explain how your program is structured
- The challenges you faced while working on the project

If you are working in a group, each member must create individual videos. *Please upload all videos to a Google Drive folder and share its link in the file titled* `Video_Links.py`.

Rubric

Checkpoint Submission (Tasks 1 - 5): 50 points

- Task 1: Name of dataset being used *(10 pts)*
- Task 2: Columns you will be doing calculations with *(10 pts)*
- Task 3: Calculations you will be performing *(10 pts)*
- Task 5: Function decomposition diagram *(15 pts)*
- Names of collaborators *(5 pts)*

Final Submission (Tasks 4 - 8): 140pts

- Task 4: Result file (either `.txt` or `.csv`) *(20 pts)*
- Task 5: Function decomposition diagram *(15 pts)*
- Tasks 6, 7: All committed code for Project 1 *(90 pts)*
- Task 8: Each group member's explanation video *(15 pts)*

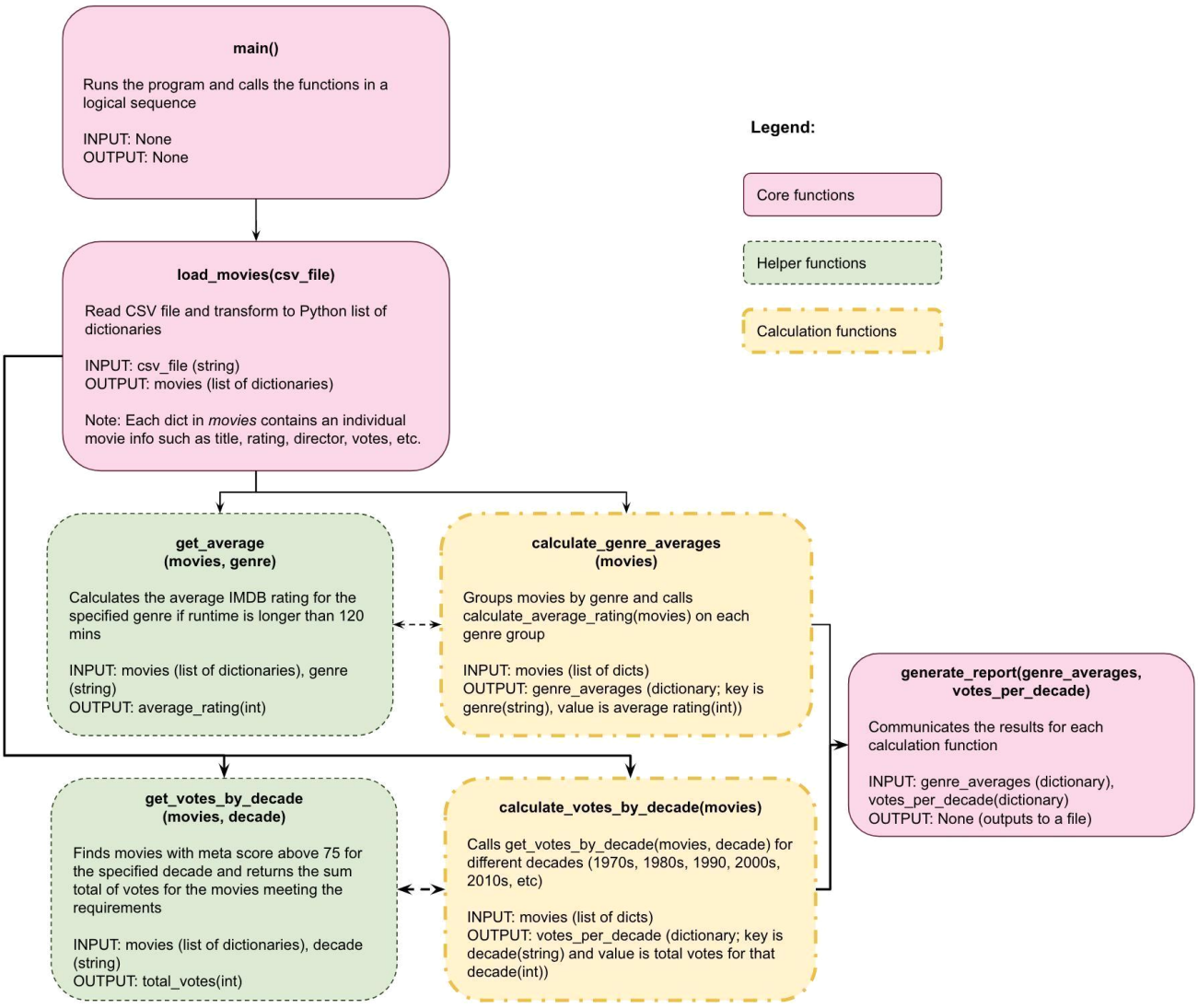
Appendix

Task 5: Example Function Decomposition Diagram

For the following calculations based on the Kaggle IMDB dataset:

- *For each genre, what is the average IMDB rating of movies that have a runtime longer than 120 minutes?* (Columns used: `Genre`, `Runtime`, `IMDB_Rating`)
- *What is the total number of votes for movies released in each decade that have a Meta score above 75?* (Columns used: `Released_Year`, `Meta_score`, `No_of_Votes`)

the corresponding function diagram could look like this:



A larger image can be found in your GitHub repository!

Notice how:

- Each function box tells us the function's purpose, its input and output, and their types
- The arrows tell us how variables are passed between different functions

We color coded our functions above to showcase the different purposes of each function and for improved readability. However, it is not required for you to do the same!

Task 6: Example Subset for Testing

For a subset of the Kaggle IMDB dataset, **we chose 15 rows with diverse directors, decades, and genres** rather than choosing only rows with a specific director or a specific genre.

`Poster_Link,Series_Title,`**`Released_Year`**`,Certificate,Runtime,`**`Genre`**`,IMDB_Rating,Overview,Meta_score,`**`Director`**`,Star1,Star2,Star3,Star4,No_of_Votes,Gross`

- `..., Soorarai Pottru, `**`2020`**`, U, 153 min, `**`Drama`**`, 8.6, ..., (missing), `**`Sudha Kongara`**`, Suriya, Madhavan, Paresh Rawal, Aparna Balamurali, 54995,(missing)`
- `..., Interstellar, `**`2014`**`, UA, 169 min, `**`Adventure`**`, Drama, Sci-Fi, 8.6, ..., 74, `**`Christopher Nolan`**`, Matthew McConaughey, Anne Hathaway, Jessica Chastain, Mackenzie Foy, 1512360, 188020017`
- `..., Cidade de Deus, `**`2002`**`, A, 130 min, `**`Crime`**`, Drama, 8.6, ..., 79, `**`Fernando Meirelles`**`, Kátia Lund, Alexandre Rodrigues, Leandro Firmino, Matheus Nachtergaele, 699256, 7563397`

`[12 more rows]`

The above subset of rows have the same header as the original dataset, a diverse range of values for the movie genres, directors, have missing values (good for testing edge cases), and have 15 rows (excluding header row).