SI 201: Project 1 - Data Analysis

Overview

In this project, you will analyze a real-world dataset from Kaggle. 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:

- Break your code into functions
- Read a .csv file into a list of dictionaries or a nested dictionary
- Write two functions that perform a calculation on the data (e.g. averages, mode, median, totals) per group member
- Write the results of your calculations to a file (`.txt` or `.csv`)
- Create unit tests

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.
- **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.**

- Similar to homework, you may use GenAI. However, you must report that you used it and how you used it. We recommend using it to help you learn, not just do your work for you. You are responsible for all GenAI code that you use in your project.

Deliverables

1. **Checkpoint** (Due 02/16/26; You will not get feedback on this before the project is due) (20 points):
    - Each group member must submit a file to Canvas with your initial project plan (checkpoint details outlined after Task 5), including:
        i. The names of collaborators
        ii. The name of the dataset being used
        iii. A description of the calculations you will be performing; including which columns are used in each calculation
        iv. A function diagram

2. **Final Submission** (Due 02/20/26) (180 points):
    - Submit a link to your group's Project 1 Github Repository, including:
        i. The output file (either `.txt` or `.csv`)
        ii. All committed code for Project 1 (Tasks 5-6)
        iii. One video explaining your function diagram and one from each group member explaining one of their calculation functions (Task 7)
        iv. A function diagram (updated if needed from the checkpoint version)

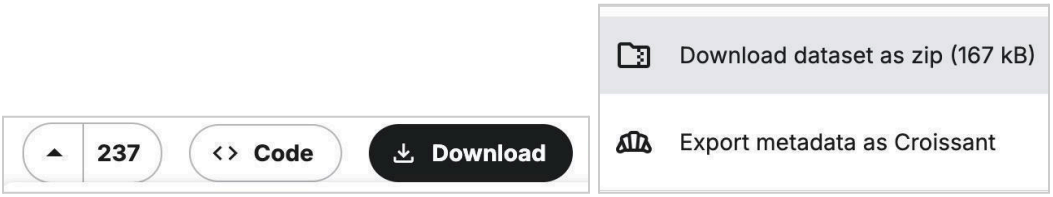
Task Breakdown

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 `.zip` file and add the resulting `.csv` file to the cloned repo for this project.



Task 2: Define Calculations

Read the data from the file and explore the data, i.e. the columns and their values. Then, determine what you would like to calculate from this data.

- Every student in the group must create **two distinct** calculation functions (e.g. averages, totals).
- **Each calculation must use at least three columns of data**

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 3: 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 4: Function Diagram

Break down the work for this project **into separate functions**.

Create a diagram showing:

- Each function's name. Include who is authoring each function
- A short description of what that function does
- Parameters (names and types) and the type of data returned, if any
- Indicate which functions call other functions with an arrow

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 at the end of this document.

Task 5: Test Functions

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

- One test case must test general/usual case
- One test case must test an edge case

For aid with this part, please refer to the Lecture slides that can be found on Canvas under `Files` > `Slides` > `L8 - Unit Testcases`

When writing test cases, **create a subset of your chosen dataset** and add it to your GitHub repository. Please ensure that it meets the following criteria:

- Must have at least 10 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 at the end of this document.

Task 6: Code

Implement the functions from the diagram created in Task 5.

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

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

Task 7: Videos

- Each project team must create one video explaining the final function diagram. Highlight any changes made after the checkpoint.
- Each member of your group must explain one of their calculation functions in an individual video.
- All final videos must be between 1-3 videos in length.

Please add the link to your videos in the GitHub repository after following the instructions in the file `Video_Links.txt`.

Rubric

Project Checkpoint (20 points)

Item	Points
Names of collaborators stated	2 points
Name of dataset being used stated	2 points
Short description of each calculation function provided	4 points
Names of columns used for each calculation function included	4 points
Function diagram included	8 points

Dataset Reading and Representation (18 points)

Item	Points
Code correctly opens and reads the downloaded dataset file	8 points
A list of dictionaries or a nested dictionary is used to store data read from the dataset	10 points

Calculations (individually graded) (54 points)

Item	Points
Calculations are performed correctly and produce expected outputs	36 points (18 points per calculation function)
Calculations use at least 3 dataset columns	12 points (6 points per calculation function)
Calculations are meaningful and demonstrate different insights	6 points

Output (18 points)

Item	Points
Output file is generated as either a .csv or .txt when the code is run	6 points
Output includes summary/results for each calculation performed	8 points
Output is well-formatted and understandable	4 points

Testing (individually graded) (45 points)

Item	Points
General/usual test case provided and pass at submission time	12 points (6 points per calculation function)
Edge test case provided and pass at submission time	12 points (6 points per calculation function)
Testing dataset - Subset of original dataset - At least 10 rows of data - Has diverse values - Has same headers/format as the original dataset	12 points (3 points) (3 points) (3 points) (3 points)
Tests are organized and have clear assert statements	9 points

Function Diagram (36 points)

Item	Points
Function names and short functional description for each	10 points
Author(s) of each function indicated	6 points

Input/Output types documented for each function	8 points
Arrows showing which function calls which function	8 points
The function diagram is well organized and understandable	4 points

Videos (9 points)

Item	Points
Function diagram explanation video	3 points
Calculation function explanation video (individually graded)	6 points

Appendix

Example Function 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:

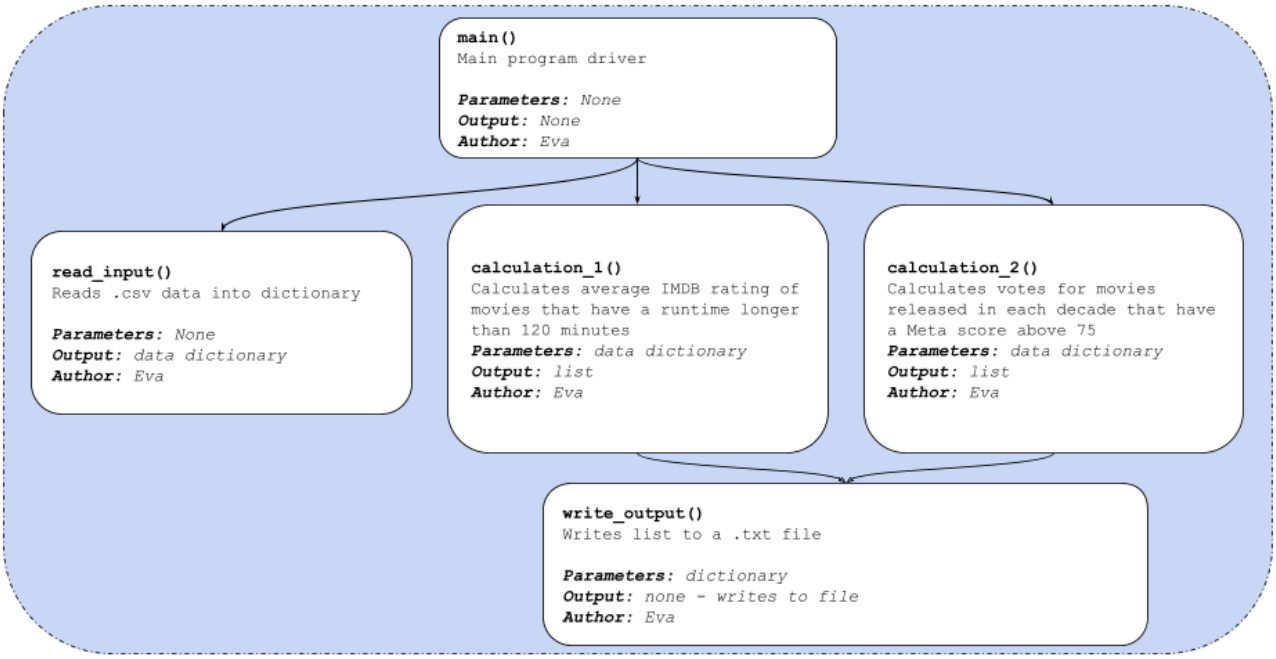


Fig: Function Diagram for One Person team

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!

Example Subset for Testing

For a subset of the Kaggle IMDB dataset, **we chose 10 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`

`[9 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 10 rows (excluding header row).