ACS_567 HWK1

attis01 - Nikhil Attili

Quick Summary of the Python Food Nutrition App Project

1. Data Encapsulation Class:

Plan: Make a class to hold data with fields, a constructor, and methods to use these fields.
Execution: Built the **FoodData** class with a constructor. It worked out just like I planned.

2. Manager Class (Singleton):

Plan: Set up a singleton class to handle the data. It should have methods for adding, changing, deleting, and analyzing data.
Execution: Got the **DataManager** class done. It follows the singleton idea and works with the CSV file well. I stuck to the plan but tweaked things a bit for handling files.

3. Driver Class:

Plan: Create a class to run the console app, using menus for users to choose options.
Execution: Made the Application class which lets users navigate through menus. This went just as I thought it would and was pretty simple to implement.

4. Console Application:

Plan: Build a console app that uses all these classes to do different things.
Execution: The console app is up and running in the **if __name__ == "__main__":** part. It combines user inputs, data handling, and analysis just like I wanted and was also pretty simple to implement.

5. Documentation:

Plan: Use standard ways to document the code, like Javadoc or pydoc.
Execution: I've added comments and **pydocs** for more clarity and provided the basic understanding of the code. In my opinion I can add more comments for a bit more clarity.

6. GitHub Repository:

Plan: Keep the code on GitHub to track changes.

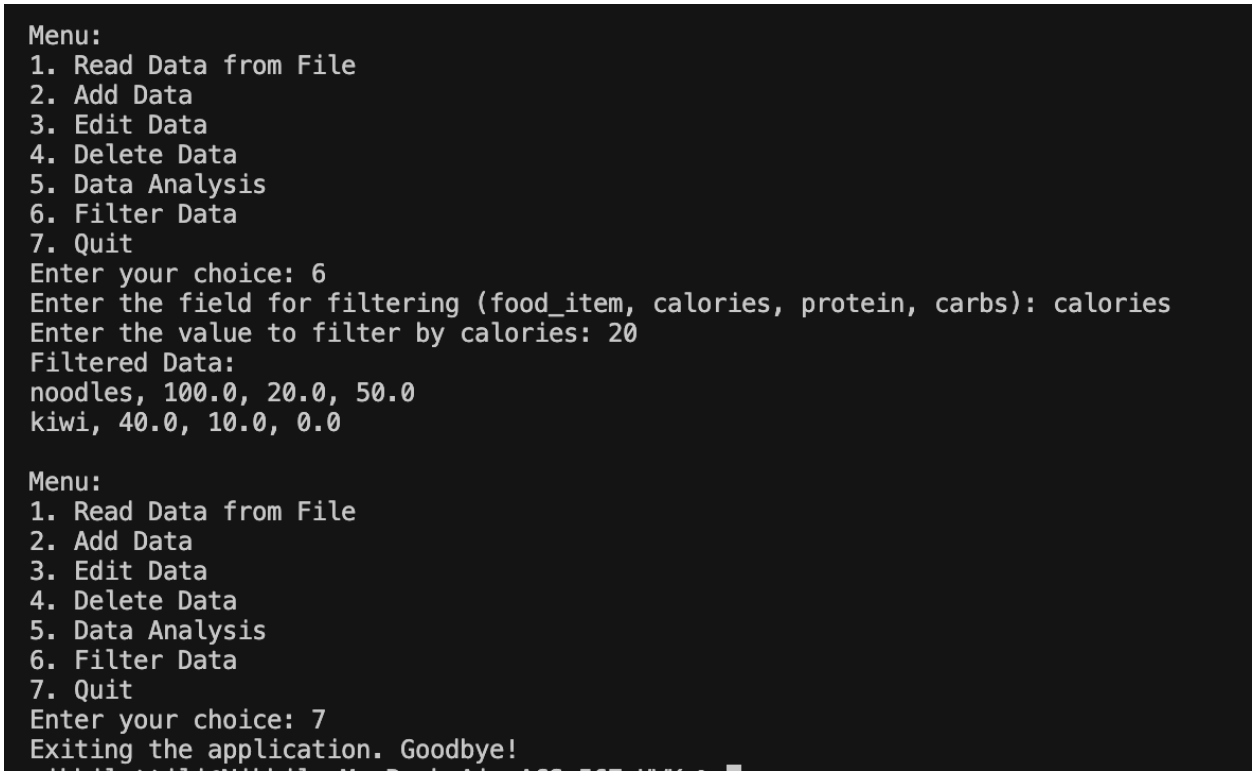
Execution: The GitHub repo is up to date and access given to MP and the TAs.

Reflection:

Overall, I did what I set out to do: Built the classes, the menu system, and the features I wanted. Changing the plan a bit, especially for how I handle files, was a smart move. Looking ahead, I could've made the app even better with more checks for errors, validating inputs, adding tests and leveling up the documentation. The project's foundation was solid, and the changes I made along the way really helped make the final product better.

Screenshots:

Read data:



Data analysis:

```
Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 5
Enter the field for analysis (calories, protein, carbs): calories
Mean calories: 70.0
Median calories: 70.0
```

Delete data:

```
Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 4
Enter the index to delete: 1
Data deleted successfully.

Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 1
Loaded Data:
noodles, 100.0, 20.0, 50.0
```

Edit data:

```
Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 3
Enter the index to edit: 1
Enter Food Item: pizza
Enter Calories: 100
Enter Protein: 20
Enter Carbs: 40
Data edited successfully.

Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 1
Loaded Data:
noodles, 100.0, 20.0, 50.0
pizza, 100.0, 20.0, 40.0
```

Add data:

```
Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 2
Enter Food Item: pizza
Enter Calories: 200
Enter Protein: 20
Enter Carbs: 40
Data added successfully.
```

```
Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 1
Loaded Data:
noodles, 100.0, 20.0, 50.0
```

Duplicate data:

```
Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 2
Enter Food Item: pizza
Enter Calories: 10
Enter Protein: 10
Enter Carbs: 10
Duplicate data. Data not added.
```

Filter data:

```
Menu:
1. Read Data from File
2. Add Data
3. Edit Data
4. Delete Data
5. Data Analysis
6. Filter Data
7. Quit
Enter your choice: 6
Enter the field for filtering (food_item, calories, protein, carbs): food_item
Enter the value to filter by food_item: pizza
Filtered Data:
pizza, 10.0, 10.0, 10.0
```