Unit:
Analysis, Design and Implementation

Assignment title:
Weight Tracker

March 2018

Important notes

- Please refer to the Assignment Presentation Requirements for advice on how to set out your assignment. These can be found on the NCC Education *Campus*. Click on Policies and Advice in the left-hand menu and look under the Advice section.
- You must read the NCC Education documents ‘What is Academic Misconduct? Guidance for Candidates’ and ‘Avoiding Plagiarism and Collusion: Guidance for Candidates’ and ensure that you acknowledge all the sources that you use in your work. These documents are available on *Campus*. Click on Policies and Advice in the left-hand menu and look under the Policies section.
- You **must** complete the **‘Statement and Confirmation of Own Work’**. The form is available on *Campus*. Click on Policies and Advice in the left-hand menu and look under the Policies section.
- Please make a note of the recommended word count. You could lose marks if you write 10% more or less than this.
- You must submit a paper copy and digital copy (on disk or similarly acceptable medium). Media containing viruses, or media that cannot be run directly, will result in a fail grade being awarded for this assessment.
- All electronic media will be checked for plagiarism.

Scenario

Losing weight is difficult, but it’s made less so by paying careful attention to progress. Studies show that keeping a daily diary of caloric intake and minutes spent exercising can help explain patterns of fluctuating weight observed on a daily basis. By charting this against a daily record of weight and waist-size, motivation can be maintained over the long term.

However, doing this requires a considerable amount of day to day book-keeping. A local weight watcher group has contacted you to see if you’d be able to design and implement a system capable of tracking this information and providing useful analysis for users. It would involve the following elements:

- Permit the user to enter a name, gender and age that is stored for them throughout the course of the tracking.
- Allow users to enter a name and calorie value for every individual meal they consume. When a user has entered a particular kind of meal, it should be stored in the system so that if it is entered again there is no need to provide the calorie information. Meals should also allow for a portion size in grams to be recorded.
    - As part of this, the application will calculate a ‘calories per gram’ number that is used to work out how many calories a meal contained purely on the basis of portion size.
- Allow the user to enter a type of exercise, and how many minutes they undertook of that exercise. This too should come with a calorie value.
    - The application will calculate a ‘calories burned per minute’ value which is then used to calculate calorie burn after the first entry has been made.
- Allow the user to enter a weight reading at two points in the day – morning and evening.
    - The weight for the day is the average of the values that were entered. If no value is entered for the day, it is the average between the last and the next values entered.
- Allow the user to enter a waist size at two points in the day – morning and evening.
    - As with the weight, the size for the day is the average of the two values that were entered. If no value is entered for the day, it is the average between the last and next values entered.
- Allow the user to set an average activity level for the day, which is set between inactive, moderately active, and active,

At any time, the user can ask for an analysis of their progress over a set period of time, which is provided in the following forms:

- A calculated average weight loss or gain during the period, calculated by checking the current day’s weight against the last day’s and summing together each daily change.
- The amount of weight change since the beginning of the recorded period and the end.
- A line graph showing the daily weight as it has been recorded, along with a **trend line**.
- A set of values outputted for each day which shows the caloric intake for the day, the caloric burn for the day, and the difference between them.

- An estimate of the calorie intake required to **maintain** the day's weight. For men, this is calculated as 5x the weight in kilograms (kgs) for an inactive day, 6x the weight in kilograms (kgs) for moderately active, and 7.5x the weight in kilograms (kgs) for active. For women, the values are 4x, 5x and 6x respectively.

Your application then needs to provide the following functionality:

- Allows users to add and modify their name, age and gender.
- Allows for users to add, modify and delete meals.
- Allows for users to add, modify and delete exercises.
- Allows the user to record waist size and weight on a 2x daily basis.
- Calculates the caloric value per gram, and per minute, for meals and exercises.
- Calculates the weight loss for a specified period based on entered data.
- Calculates the total weight change between the start and end of the specific period.
- Generates a line chart that shows the weight each day of the specified period.
    - And generates a trend line for that graph.
- Show the values for caloric intake, burn and difference for each day.
- Calculation of calorie intake necessary to maintain.

Your solution will consist of a class diagram, a use-case diagram, and an activity diagram for the process of calculating caloric intake to maintain and comparing it against the caloric burn, intake and difference. This should be done taking into account a period of time specified by the user. You should also submit the completed program code in Java.

Task 1 – 26 Marks

Candidate class list and Diagrams

26 Marks are available for providing an appropriate list of candidate classes, along with the supporting diagrams. The candidate class list should incorporate justifications and discussion as to why each class was selected for inclusion, and how its relationship to other classes was derived. The class diagram should show attributes, operations, scope and relationship of classes to each other.

Please note, you may supplement the attributes listed in the scenario (above) with others that are appropriate. You may wish to discuss this with your lecturer, but credit will still be awarded for reasonable modifications.

The marks for the task are broken down as follows: (1) 10 marks for the Candidate class list; (2) 6 marks for additional classes; and (3) 10 marks for the Class diagram showing associations (with multiplicity), operations and attributes.

Task 2 – 25 Marks

Activity diagram

25 Marks are available for the creation of the appropriate activity diagram. The activity diagram should incorporate the classes involved in a user generating a review of their caloric deficit within a specific period of time. Here, neatness of the flow of logic is important.

The marks for the task are broken down as follows: (1) 20 marks for functionality; and (2) 5 marks for Class ownership.

Task 3 – **8 Marks**

Use case diagrams

8 Marks are available for the provision of suitable use-case diagrams. The use case diagram should incorporate each of the user activities indicated in the brief

Task 4 – **15 Marks**

Code architecture

15 Marks are available for a code architecture that shows an appropriate level of coupling and cohesion, along with the necessary amount of inheritance and encapsulation to express the system.

The marks for the task are broken down as follows: (1) 5 marks for handling the requirements via inheritance and polymorphism; (2) 5 marks for handling user input; and (3) 5 marks for handling output.

Task 5 – **26 Marks**

System implementations

26 marks are available for implementing the system as described and providing the completed Java code.

Submission requirements

- Your program must be submitted as a zip file of the full project.
  - Whatever IDE you use, it should be possible to open and run the project directly from the extracted archive.
- Diagrams and materials associated with the tasks above should be presented in a word-processed document.
- All references and citations must use the Harvard Style.

Candidate checklist

Please use the following checklist to ensure that your work is ready for submission.

Have you read the NCC Education documents ‘What is Academic Misconduct? Guidance for Candidates’ and ‘Avoiding Plagiarism and Collusion: Guidance for Candidates’ and ensured that you have acknowledged all the sources that you have used in your work? □

Have you completed the ‘Statement and Confirmation of Own Work’ form and attached it to your assignment? **You must do this.** □

Have you ensured that your work has not gone over or under the recommended word count by more than 10%? □

Have you ensured that your work does not contain viruses and can be run directly? □