



Hacettepe University

Computer Science and Engineering Department

Name and Surname : İdil CAN

Identity Number : 21727011

Course : BBM104

Experiment : Assignment 2

Subject : Basic Java skills and Classes

Data Due : 14th March 2018

Advisors : Nebi Yılmaz

e-mail : idil.can@hacettepe.edu.tr

INDEX OF THE REPORT

Topic.....	Page No
1. <i>Software Using Documentation</i>	3
1.1. <i>Software Usage</i>	3
1.2. <i>Provided Possibilities</i>	3
1.3. <i>Error Messages</i>	3
2. <i>Software Design Notes</i>	3
2.1. <i>Description of the program</i>	3
2.1.1. <i>Problem</i>	3
2.1.2. <i>Solution</i>	3
2.2. <i>Main data structure</i>	4
2.3. <i>Algorithm</i>	4

1. SOFTWARE USING DOCUMENTATION

1.1. Software Usage

This program works as a health app. It takes the user's activities: which food they ate and which sport they did.

The program takes one string from command line: name of the commands' file. There is no other input in execution time. Apart from these, the program opens 3 files: "food.txt", "sport.txt", "person". In that we needed to write the output strings to the file, program opens/creates (to be detailed more below) one more file: "monitoring.txt".

1.2. Provided Possibilities

Program has variety of classes. To make programming more systematic and clear I used n classes: "Person", "PersonManager", "Food", "FoodManager", "Sport", "SportManager", "HelpingFuncs", "CommandsManager". This makes functions more grouped and takes weight from the "Main.java".

1.3. Error Messages

The program does not have any error messages that's been asked but it gives error messages if an error accrues while opening and reading a file.

2. SOFTWARE DESIGN NOTES

2.1. Description of the program

This program is a basic health assistant that holds calorie calculation. Takes the backup information from preloaded files and implements them. Then, according to those files it adds food and sport. User can print all the used person's citation or just one person's.

2.1.1. Problem

This program has a need to use multiple classes in order to having a plain and simple program. Actually, the assignment was given to us in order to we practice on our Java skills and class using.

2.1.2. Solution

Java is an object-oriented language. Since we are not allowed to use inheritance or encapsulation, the program has a lot of classes. Some of these classes has copied functions.

Personally, it made me understand the use of inheritance more and now I know when I will be needed to use inheritance. The same thought is true for encapsulation too.

2.2. Main Data Structure

Program has eight classes (as I mentioned in 1.2. Provided Possibilities). Three of them are structure class and rest of them are function classes.

“Person”, “Food” and “Sport” classes are structure classes that holds the values and the functions that about one of an element of these types’ (ex: name, id... get functions, changing functions...).

Other classes that has a bigger area of work (I mention them as “Manager” classes) has functions as “find”.

“HelpingFuncs” has functions that is being used in multiple classes.

Lastly, “CommandsManager” has a more special role. This role is having the main loop that commands managed inside.

2.4. Algorithm

Code takes one argument from command prompt. It includes the commands that will be executed.

There are three files to be opened: “sport.txt”, “people.txt”, “food.txt”. After opening these code reads them and gives a list of that class.

Right after that, it starts to reading the command file step by step in the main loop.

First, it controls if the file has a next element. If it exists, it starts to implement the main loop.

Loop’s first job is to control if the value is a print or not. If not, it searches the id in an array of mentioned users. When it finds, it won’t do anything at this time but when it doesn’t find it, it adds the person to the array so it makes the “printList” array.

Then, code controls the value if it’s a print or an input. If it’s a print, it finds the id in person list and prints out the properties of the user. Else, it still finds the id. The difference is after finding the person it looks for the next element and finds whether it’s sport or food. Foods changes the taken calories variable and net calories. Sports are only different that it won’t change the taken calories but burned calories.

Then it moves to the next line of command.