Department of Computer Science & Engineering

BBM104 Introduction to Programming Laboratory

Experiment 2

EMRE HANCI - 21604552

Programming Language   : Java

Submission Date   : 21.03.2018

Due Date   : 11.04.2018

Advisors   : Dr. Gönenç Ercan, Dr. Öner Barut, Dr. Cumhur
Yiğit Özcan, Dr. Ali Seydi Keçeli, R. A. Nebi Yılmaz

-What was the problem?

In this assignment, the expectation from us write Java code that calculates the calories by considering taken and burned calories during the day for the healthy life of the people. When we calculate it, we need three text files;

people.txt

This file include people informations with an informantion template which is show in below. This text file contains up to 50 items.
[person ID] tab **[name]** tab **[gender]** tab **[weight]** tab **[height]** tab **[date of birth]** newline

food.txt

This file include food informations with an informantion template which is show in below. This text file contains up to 100 items.
[food ID] tab **[name of food]** tab **[calorie count]** newline

sport.txt

This file include sport informations with an informantion template which is show in below. This text file contains up to 100 items.
[sport ID] tab **[name of sport]** tab **[calorie burned]** newline

Calculation of daily calorie needs

The daily calorie needs (dailyCalorieNeeds) of people vary by gender, age, height and weight. Therefore, it will be calculated separately for men and women as follows:
Calculation for men = 66 + (13.75 x weight (kg)) + (5 x height (cm)) – (6.8 x age)
Calculation for women = 665 + (9.6 x weight (kg)) + (1.7 x height (cm)) – (4.7 x age)

command.txt

In this file, there are command lines which are the progress the program. There are three type commands.
printList : When this command execute, program show all the people who eat something or did sport.
print(**[Person_ID]**) : When this command execute, program show the information of person whose id given.
Eat something and doing sport ;
[person ID] tab **[food ID]** tab **[number of portions]** newline
[person ID] tab **[sport ID]** tab **[sport duration]** newline
Food id is start with 10.. and Sport id is start with 20.. thats way we can split them according the second argument.

-My solution

For the solve this problem I used three classes and I created object array from the classes. When all the informations upload the object, commands are execute line by line until reach the end of file.

-The explanation of my codes

As I mentioned above I use three classes, one of them person.java in this file there are two methods constructor and search. The constructor of this class load the informations to object array. The purpose of search method is find the index of given person id and return the index of this element. The second class is food.java, there are two methods constructor and search. This purpose of this two methods is same with person class. The last class is sport.java in this class again there are two method and this two methods are same with the previous two class.

The explanation of two special things:
- line = line.replace("ï»¿", "");
In first line of every files there are this chars because of the file type, first examples text files are saved as UTF-8 with BOM.
- printWriter.println("**************");
The purpose of this code is write the stars to file after every line except the last line. For the solve that I write every star lines before the command output lines except the first line because I can not know how many line in command file.

-The algorithm of my program

The algorithm of my codes;
- Object arrays created, the load methods of objects is executed.
- Load methods are creat object and load them to object arrays in main class.
- Command file opened for the read purpose.
- Monitoring.txt file opened for the writing purpose.
- Command file lines read line by line until reach the end of file.
- Program check the line for the remove useless chars.
- If program executing the first line, it does not print the stars line.
- Program check the line lenght
    - If it is longer than one program check the second argument.
      -If it is smaller than 2000, it means a person eat something.
        -Program search the index of given person id.
        -Program search the index of given food id.
        -Program calculate the result of eaten food.
        -Finally if person id is not in the proccessed person arrray, the person id assign to it.
      -If it is greatter than 2000, it means a person did sport.
        -Program search the index of given person id.
        -Program search the index of given food id.
        -Program calculate the result of burned calorie.
        -Finally if person id is not in the proccessed person arrray, the person id assign to it.
    - If it is equal to one item and if it is equal to "printList"
      - Program search all the person one by one and search them eat something or burned_calorie.
      - If this person did one of them program print this person information.
    - If it is not suitable previous two condition, it means program should print the person whose person id given.
      - Print the information of given person id.
- Monitoring.txt file closed.
- Command.txt file closed.