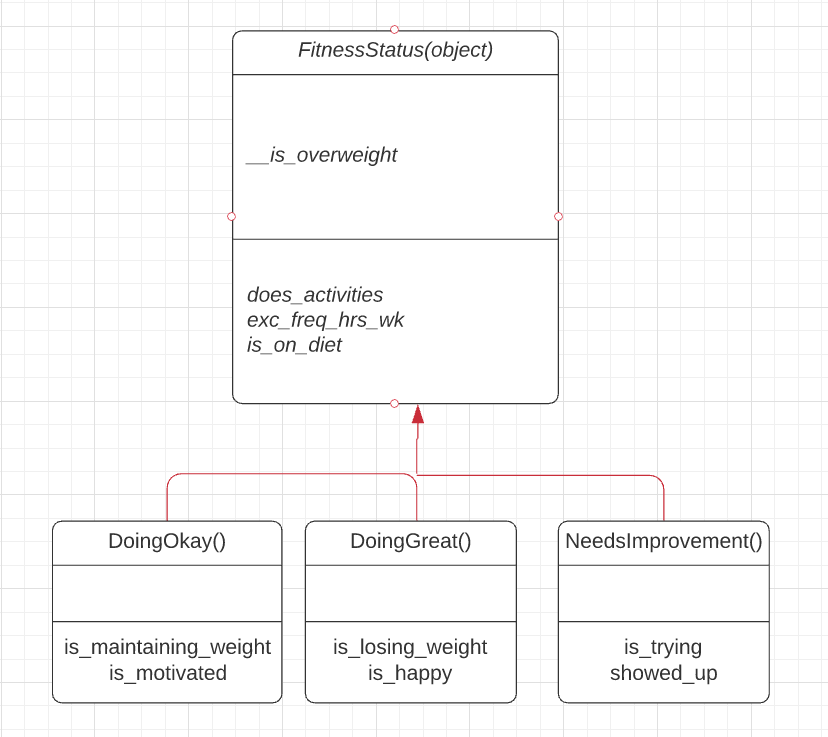
BMI Calculator Program

James B. Hardy

MET CS\_546 Information Structures in Python

Boston University Metropolitan College

Basic Class Model:



Abstract / Readme:

**Classes\_final\_project – this file holds the class definitions and unit tests.**

**Final\_Project\_Main – this file holds the main program logic, and imports the classes from the above file.**

**Final\_project\_output – this file will yield the BMI in a clean and readable format upon completion of the program. (you should not need to touch it for the program to work, but open it up after to see your results!)**

**DIRECTIONS TO RUN THE PROGRAM:**

1. **Open the Classes\_final\_project file, and the final\_project\_main file.**
2. **Run the code in the final\_project\_main file.**
3. **Input the requested information. (height, weight, activity level).**

**The program will output to the terminal some information and comments based on your input, and it will also write the results of your BMI calculation into the output file for a clean result; or in a savable format should the user wish to create their own history document.**

**This project satisfies the following criteria:** (numbers are lines code can be found for reference – unless otherwise noted; lines will be in main program).

* **Container type:-------------------------------------------------------- 38 -44**
* **Iteration type: for, while: --------------------------------------------46-50**
* **Conditionals:----------------------------------------------------57-62, 87-92**
* **Try blocks: ----------------------------------------------------------------72-76** 
  + **------------------------------------------------------(class file: 35-39)**
* **User-defined functions: ------------------------------------------------55-79**
* **Input and/or output file: final\_project\_output.txt (output) – main program writes to file at**

**97-102, and properly closes the file at line 102;( in line with best practices handling files).**

* **User defined class: (class file: 11-73) 3 classes; 1 parent; 2 children**
  + **1 private and 2 public self attributes: -------------------------------(class file: 13-16)**
  + **> 1 private and 1 public method; takes args, returns values, used:**
  + **------------------------------------------------------------(class file) priv:12, 28, 40 pub:33-41**
* **\_\_init()\_\_ method with > 1 arg: ---------------------------------(class file 12)**
* **\_\_repr()\_\_ method:-------------------------------------- (class file 18, 40, 57, 72)**
* **Unit tests: found in class file; so as not to muddy the output. All methods work as expected. Unit tests can be found------------ (class file: 23, 44, 47, 48, 49, 62, 78-91)**
* **-----------------------------------------------------assert statements: main 32, 33**

*-Thanks for all your help this semester Jos.*

*-James Hardy*