`Purpose: Reading and writing objects to a file (persistence)

Topic: To Your Health company would like the ability to store their patient’s BMI data to a file for reuse later. To accomplish this, we will use a process called serialization, which basically converts an object to a stream of bytes and writes it to a file. When reading objects from a file, we will use de-serialization, which converts the stream of bytes back into an object.

You will use your Project 2 Part 2 project as your starting point for Project 3.

* BMI class:
  + Add patient first name and last name as new fields
  + Modify your constructors, getters and setters to accommodate the first and last name
  + In setters, validate for empty string and set first name to “John” and last name to “Doe
* BMI program:
  + Create and instantiate a static List that will be used to hold the BMI objects as the user enters them
  + In your initialization process, you will need to de-serialize the objects from the input file into the list
    - Create and open an input file stream
    - Loop through the file, reading each object, de-serialize it and add it to the list
    - Close the file when done
    - Your BMI object list is now ready to add new patients to it
  + For Menu option 1 or 2 (Metric or Standard):
    - In addition to asking and validating for option, weight and height, ask for and validate for first and last name not empty
    - After user successfully enters patient name, weight and height, and the BMI is displayed, ask the user if they would like to add the patient to the file. If yes, add the BMI object to the list of objects.
  + Add another Menu option for displaying the patients, and display patient data as follows:

Name Height Weight Type BMI Status

Doe, John 70 170 Standard 24.22 Normal

Doe, Jane 170 70 Metric 24.2 Normal

* + - This report will be created by looping through the list of BMI objects, similar to using an array of objects in previous examples. Each object in the list will have access to the data and methods that you need to formulate the report.
    - Below the report, pause the screen and allow return back to Menu
  + At Exit, we need to take the objects in the list and serialize them back to the file. The file will be over-written with the objects in the current list.
    - Create and open an output file stream
    - Loop through the list of BMI objects, serializing each object to the file
    - Close the file when done
    - Your BMI file is now ready when your application is ran again, and the process would start over

**Project is worth 35 points.**