Project Name: **Coffee Machine**

Description: This project is about making the best cup of coffee in town! Your design should employ Object Oriented Design Concepts whenever needed. The following guidelines help you specify the system's main objects and their relationships:

Great Coffee starts with great beans, there are two main types of beans, Arabica, Robusta, or a blend.
The machine is filled with water and beans, the user then selects the type of Coffee he wants, that can be:

1. Espresso, which can be:
    a. Single
    b. double
2. Americano, which can be:
    a. Single
    b. Double

Espresso single shot uses 7g of ground coffee and 30 ml of water, while double uses 14g of ground coffee and 60 ml of water. Americano use 7g of ground coffee and 170ml of water for single and 14g of ground coffee and 220ml for double.

After selecting the Coffee choice, the user can hit the start button and Coffee grinder will start grinding the beans, then the brewer, using hot water and pressure, will brew the Coffee and then pour it in the cup.
The size of the grind is important to the taste of the Coffee, so, the Coffee grinder is adjustable to give you the best taste.
The Coffee machine can tell you a few useful information; the total amount of Caffeine the Coffee contains and calories. The Coffee machine has a water and beans capacity, so it will stop when it's out of any. It also needs cleaning, after serving a specific number of cups, an indicator is displayed asking for emptying the drip tray and waste.
Note: Calculating the amount of caffeine a cup of coffee contains, depends on the type of beans-blend and amount of coffee used to make the coffee.

<u>You are not limited to this description; you may add any object/attribute you find necessary.</u>

Project Deliverables

Basic:

- Use UML Diagrams to represent the system's main Classes and its relationships.
- Build the basic Classes and methods stubs that represent the System.
- Provide a getInfo( ) method for all classes. The method should return a String value describing the object's state.

Intermediate:

- Add Coffee beans and water to the machine, if the volume exceeds the machine capacity, the machine will throw an exception.

- After selecting the Coffee choice, and griding settings and when the start button is pushed, the machine will check if it has enough water and coffee beans, if not it will throw an Exception. The machine will grind the beans, brew, pour coffee in cup, and after that:
    - The machine can tell the total calories the cup has.
    - The machine can tell the amount of caffeine the cup has.

- Each time the machine serves a cup of coffee, the remaining amount of water and beans must be calculated.
- After specific number of cups served, the machine will indicate the need to clean the water tray and waste.
- At this point the system is supposed to be working using standard input/output (Console application)

Advanced:

- Design a user-friendly interface for the System.
- Log each time the machine makes coffee, you must have a Logger interface that declares the log operations which will be used to inject the actual logger to objects.
- Save the System objects' data permanently into files.
- Load the saved data from the files and create the corresponding objects to hold these data.
- +Store the Objects' data in a Database (use MySQL).
- +Load the Data from the database and create the corresponding Objects.