Assignment 3: Implementing IoC

Problem Statement:

In this assignment, you are supposed to learn how IoC (Inversion of Control) principle is applied. You are required to build an implementation example to showcase any use-case of *your choice*. The example should clearly demonstrate the principle, with code fragments for both IoC as well as conventional procedural programming for doing the same task. You must also make a small document (preferably less than or equal to 3 pages in length) where you explain the following aspects:

1. The use-case, or the overall purpose that the code solves.
2. The code fragment for traditional method.
3. The code fragment implementing IoC.
4. What advantages do you achieve using IoC.

Please note:
We have purposefully decided to not form a concrete problem statement for you to solve. Coming up with a use-case that can show the advantages of IoC is an important part of the learning.
Please do not discuss the assignment solutions with your friends. Two solutions having the same use-cases will be reviewed as suspected case of plagiarism.

Marking Scheme (tentative):

Roughly 70% of the weightage would go to the document you prepare, i.e. for coming up with a suitable use-case to demonstrate advantages of IoC, as well as your explanations for the contrast in the code fragments. The rest will be for an actual running code sample.

Submission:

Each of you are supposed to submit a Zip file (the file "must be **ZIP ONLY**", we do not guarantee your assignment getting graded if you use any other file format). The file should contain the code fragments, a readme to explain how to run the code, and the document as explained in the problem statement.

The file **MUST** adhere to the following naming convention:
<Your Roll Number>_Assgn3.zip
For example, if your Roll Number is 13111164, your file must be named as "*13111164_Assgn3.zip*".
We may automate some part of Assignment grading, and if your files do not match the above formats, they may not be graded. The submissions are to be made **only** on the mookit instance of the course (http://cs455.cse.iitk.ac.in). We will not accept any submissions over mail.

Deadline:

The deadline for submitting the assignment is **Thursday, Oct 27, 2016, 11:59 PM**.

Demo:

You may be called in to give a demo of the code that you wrote. You may also be asked to explain the use-case in detail.

Clarifications:

Any clarifications regarding the assignment can be sought by dropping a mail to the Course TAs. Please note that all the clarification mails **MUST** be sent by **Sunday, Oct 23, 2016, 11:59 PM**. **No clarification queries**, via mail or in person, will be answered **after that** (meaning you must start with the assignment as early as you can).

Plagiarism:

This is an individual assignment, and you are not supposed to collaborate with any one else. If we find any unusual similarities in codes or use-cases of two or more individuals, the same will be reported as suspected act of plagiarism. If found guilty, you will face severe punishment for the same.