

Assignment - 4

1. What design patterns would you consider and where in the code (justify why)?
 - The design patterns which are considered to input are Creator Pattern, Information Expert Pattern, Low Coupling Pattern and High Cohesion Pattern. The different pattern are used in different parts of the code like :
 - Creator Pattern
We have made a class named CMeeting which is used to check if the meeting times are valid. To do the same we have made an instance of CTime which has hour and minute as attributes which in turn helps to validate the input.
 - Information Expert Pattern
A class named CTime is made to keep up the information about hour and minute of the meeting. Hence the information about the time i.e hour and minutes is provided by CTime.
 - Low Coupling Pattern
Different classes with different responsibilities has been made to avoid dependency on one other. For example, A class named CSchedule is made, which maintains the schedule and makes sure that the meeting don't overlap.
 - High Cohesion Pattern
Classes like CMeeting, CTime and CSchedule are made which are given specific responsibilities as per the requirements.
2. Implement 4 JUnit Test cases (and obviously the unit you are testing).
 - We have made 4 JUnit Test cases which are listed as below :
 - CMeetingTest class tests whether the starting time and ending time are valid and it also tests one meeting don't last more than one hour.
 - CScheduleTest class tests that two meeting times don't overlap.
 - CScheduleTest class tests that total number of hours in meeting are not more than 8 hours.
 - CTimeTest class tests that the inputted times are valid
3. Code to pass each test case.
 - The code for the test has been attached and will be uploaded along with the other required material.
4. Demonstrate how the automated testing works.
 - Automated testing has been implemented and underneath, the screenshot has attached :

