CS 471 Term Project: OOD with Design Patterns
(Deadline for Teaming up: 4:00PM 11/7/2017; Project Due: 4:00PM 12/5/2017)

Problem Description

Suppose two new requirements are added to the Monopoly Game studied in the class.

1. Load/store a game simulation. Simulations are stored in an external data storage system. The storage system could be a local storage, an online storage, or any other mechanisms, and each of them may have different interfaces for data access, but should be easily switched from one to another. The only required operations are load and store. When a game starts the user has the option to start a new game as described in the textbook or to load a game. When a game is over the game in stored automatically.
2. Update the information of each turn in the simulation to objects that want to observe the game play.

Your task is to design and implement an OO solution with design patterns for LoadGame, StoreGame, and Update in the requirement described above. At the minimum, you need to consider: how to deal with variations of the external storage interface, the creation of class/object, which class should have only one instance, how updates are sent to other objects, etc. The grade is based on the design's quality and implementation's correctness.

The deliverable submitted to D2L Dropbox should include

1. The design class diagrams, the interaction diagrams, and any other useful diagrams;
2. A working implementation in Java mapped from the design;
3. A report that documents the use of design patterns (which, where, why, etc.) and the detailed responsibilities of each team member.

Logistics

The project is to be done by a team of 2 to 3 students. Everyone needs to participate in design and implementation.

Post the team members' names to the D2L Discussions Forum designated for this purpose once you have at least two members. A student without a team may ask to join a team of two, and the team should not reject the request. A request to join a team is granted on the first-come-first-serve basis. All teams should be formed by 4:00PM November 7th. All the team-forming activities, namely announcements, request and replies, should be done at D2L "Team Info" Discussions Topic.

If a student is not in a team by 4:00PM November 7th, the student is required to do it by themselves and the highest score the student can get is 80% of the project score.

To get started

1. Implement the Monopoly Game studied in the class, including at least the code for iteration 1 (available on pages 380-384) and a driver/client class as the base system.
2. Study the GoF Design Patterns in the textbook and notes.
3. Study the Design Patterns tutorial assigned to CS471 at Lynda.com. (This is actually our Homework Assignment 9.)

4. Appointments with the instructions to discuss the projects or during development are welcome.

Other Development Requirements

1. The design patterns used for this project should include at least Adaptor, Factory, Singleton, and Observer. You are encouraged to use additional design patterns.
2. Design interfaces for the external storage systems. At least two different ones need to be designed and implemented.
3. At least two observers are required in the implementation, where one is text-based and the other GUI based.
4. Use Java's Observable and Observer to implement the observer pattern.

Demonstration

Each group needs schedule with the instructor to give a 15-minute demonstration during the final week. (This is our "final exam"!) The demonstration includes a live demo of running your project, your design, and the implementation (i.e., code).