SM (M) 2017-2018 Lab 4: Design Patterns II

Eclipse
Load in the Lab 4 workspace (available on Moodle) into your local workspace in Eclipse. Use the file -> import command to do this.

Java Generics
The Observer pattern uses and ArrayList, which is an example of generics in Java code.Understanding and using generics properly is an important part of Java programming.

Check this ArrayList API:
http://docs.oracle.com/javase/7/docs/api/index.html?java/util/ArrayList.html

And for reference, the Java Generics tutorial:
http://docs.oracle.com/javase/tutorial/extra/generics/index.html

The constructor in observer.Observable is empty. Using java generics, complete this constructor and test by running the main class.

Interface Polymorphism
Some of the patterns in the workspace use interface polymorphism. Identify which patterns use this, identify which lines of code this occurs on, and how it is crucial to the function of the pattern.

Related Patterns
This week we looked at the Proxy pattern. This pattern is closely related to the façade and adaptor patterns, but each pattern has a slightly different purpose. How are each of these patterns different, and how are they similar?

What is the difference between read only and immutable patterns? Why would you use one over the other?

Anti Patterns
The OOSE Book describes antipatterns for Proxy, Read Only Interface, and the Observer patterns. Be sure you understand these and could identify an antipattern implementation. If you have time, test the antipattern implementation in your own workspace.