

Study questions for Lecture 6 - Answers

1. Do the Pattern book exercises on pages 69, 75

a) Page 69 – answer in the book

b) Page 75 – answer in the book

2. With respect to the principle "favor composition over inheritance" what are the problems presented by the use of the Java Observer/Observable framework?

There are several issues. One is that the O/O framework requires the class that event publisher to subclass from Observer, and since Java only allows single inheritance you will have used up your one shot on kind of booking function.

What you really want is a kind of helper object to keep track of subscribers, and to assist in their notification. These could be done with a helper Observable class. The publisher object, when created, could be given such an object as a constructor parameter. We would have to define our own methods in the publisher for calling the Observable object methods.

The current Observable could not be used in this way, because some of the methods you would need to call directly are not public, and you can only get at them by inheritance.

3. What design decisions are you making when you build the ISD's for your system as in exercise 1?

There are several issues. One is that you are naming and identifying the objects that are instantiations of the subsystem Interfaces. These are the things that communicate with each other at the top level. Also, if you examine the interactions between actors and the GUI/Interface object, you will find the screens that have to be defined and have a preliminary knowledge of their contents.