

OOP Coding Assignment – Remote Control Assignment



Waqar Ahmad

13 November, 2015

You may have seen how in the Open Closed Principle lesson the interface to tell doctors, nurses and other hospital staff to perform work was exactly the same. The `performDuties()` method. This allowed for abstraction and extension. All employees in the hospital could be told to perform their jobs whatever their jobs demanded without getting into the details of how to perform them.

I want you to apply a similar strategy with an assignment. If this seems a little challenging keep pressing on and struggling with it if you have to. That's how you'll learn. You can view my solution once you have practiced it on your own. Please don't view my solution before you have tried to complete it on your own.

We're going to model a home entertainment system and the ability for a user to be able to use a single remote control to control multiple digital devices such as a TV or sound system etc. Keep in mind we only want one controller in our living room! It's a nice universal controller capable of connecting to multiple devices. This may call for using the singleton design pattern. We'll be covering that pattern more formally later in the course but I wanted to expose you to it a little early so you're better equipped for the later lesson. Google it if you have to understand how it works.

Here is what you need to do: Create a `RemoteControl` class with 3 methods in it. One for choosing the device to connect the remote control to (make this a static method), and the other 2 methods are for turning that particular device on or off. Create three other classes – `Projector`, `TV`, `SurroundSoundSystem`. All three of these devices should follow a consistent interface so that the same remote can be used to turn the devices on or off. Remember, you want to implement this in such a way that if more devices need to be controlled by the same remote in the future, that can be easily done. That's what the Open Closed Principle is all about. We want to be open for extension but We don't want to keep modifying the internals of the remote control class to be able to support more devices in the future. The last class you can create is the `Person` class. This will be the user of the home entertainment system. You can define a main method in this class to test out your code. Initialize the devices here and have the remote connect to that device to turn it on or off. Have fun with this exercise. Try other devices and add more functionality to the remote control class. Remember the best way to learn is to explore and create new things as we have been doing

in the course so far. Again I want to stress – try this out on your own first, struggle with it if you have to and only then view my solution code.

Good luck!

[Click Here to Download Solution Code](#)