Hands-on lab

Delegates and events

September 2017

Create Console programs for the following exercises:

Exercise 1

Create a class *LightBulb*, with a *bool* property *IsOn* and a *string* property *Color*. Now create a class called *Switch* with a bool property *IsOn* and a method *Flip* which changes its state. Use only delegates to let a *LightBulb* instance subscribe to a *Switch* instance, so that the *LightBulb* turns on using the *Switch*.

Exercise 2

Based on the previous exercise, use multiple switches, so that one of the switches could be flipped to put the light on/off.

Exercise 3

Based on exercise 1 and 2, use multiple lightbulbs which can all be controlled with one or more switches.

Exercise 4

Rewrite exercise 1 to use events. Use the proper conventions for passing arguments!

Exercise 5 (Challenging)

Create a TemperatureSensor class that measures temperature and raises an event every second, with a new temperature reading. Implement this using a thread-timer. Generate a random double between 0.0 and 40.0. Now create 4 subscriber classes:

- AverageTemperatureCollector: collects all readings and prints a rolling average;
- MinimumTemperatureCollector: collects all readings and prints the minimum value;
- MaximumTemperatureCollector: collects all readings and prints the maximum value;
- TresholdTemperatureCollector: collects all readings and raises an event when a certain threshold is reached (e.g. hotter than 35.0 degrees).

Exercise 5

Proof using any of the examples in the previous exercises, that a delegate return value is actually the last method from the invocation list.