

Aufgabe 02a: Observer

Lesen Sie zur Vertiefung des Observer-Patterns das entsprechende Kapitel in [1] aufmerksam durch. Implementieren (und erweitern) Sie die Weather-App aus der Vorlesung in C#.

- Verwenden Sie das klassische Observer-Pattern.
- Implementieren Sie den push-Ansatz mit einem Event-Objekt.
- Implementieren Sie Subject und Observer selbständig. Verwenden Sie nicht die Klassen aus der C#-Library.
- Sehen Sie eine Möglichkeit vor, Temperatur, Luftfeuchtigkeit und –druck zu manipulieren.
- Implementieren Sie zwei unterschiedliche (sehr simple) ConcreteObservers (z.B. ein Observer zeigt die Temperatur, der andere den Luftdruck an).
- Achten Sie auf eine Trennung der Komponenten im Sinne des MVC-Patterns

Abgabe: git „02a_Observer“.

Aufgabe 02b: Delegates

Lesen Sie den Artikel [2]. Ändern Sie die Abgabe aus Aufgabe 02a so um, dass Sie anstatt des klassischen Observer-Patterns Delegates&Events verwenden (löschen Sie nichtbenötigten Code).

- Halten Sie sich an die Namenskonventionen („Event Pattern“) [2] und verwenden Sie eine eigene Implementierung für EventArgs.

Abgabe: git „02b_Delegates“.

[1] Gamma et al. „Design Patterns. - Elements of Reusable Object-Oriented Software.“

[2] Doug Purdy, Jeffrey Richter. Exploring the Observer Design Pattern:

<http://msdn.microsoft.com/en-us/library/ee817669.aspx>

Arbeiten Sie in 2er-Teams!

Deadline: So., 09.10., 16:00