

### **Aufgabe 1 (Thema: Beobachtermuster)**

Hinweis: Erstellen Sie die Lösung der folgenden Aufgabe ohne die Verwendung von Delegaten

Wir modellieren ein System zur Benachrichtigung von Anlegern (Subscriber) des amerikanischen Dow Jones (DJ) Aktienindex. Ein DJ-Informer (Publisher) informiert seine Anleger, falls der DJ an einem Tag um mehr als 3 Prozent gefallen ist. DJ-Informer enthält das Datenelement DJStartwert zu Börseneröffnung sowie den aktuellen DJ-Wert (wird als Eingabe des Benutzers simuliert).

Legen Sie in Main einen DJ-Informer und 2 Anleger an. Anschliessend melden sich die Anleger beim DJ-Informer an, dass sie benachrichtigt werden wollen. Setzen Sie anschliessend in Main den DJ-Startwert auf 18500 und geben Sie den aktuellen DJ-Wert in einer Schleife in Main durch den Benutzer ein, wobei die Eingabe von 1 die Schleife beendet.

Es wird unterstellt, dass keine falschen Eingaben erfolgen (Hinweis: der DJ-Wert bewegt sich zur Zeit zwischen 28000 und 31000.)

Verwenden Sie das Konstrukt Property.

### **Aufgabe 2**

Erstellen Sie die Lösung von Aufgabe 1 unter Verwendung von Delegaten.

### **Aufgabe 3**

Erstellen Sie eine Dll mit der Programmiersprache Visual Basic in der die mathematische Funktion  $x$  hoch  $y$  implementiert ist. Die Dll ist für Maschinenbau-Ingenieure die kein C# aber Visual Basic kennen.

Die Funktion erwartet, dass  $x \geq 0$  und  $y > 0$  ist. Falls  $x < 0$  soll die Ausnahme FalscheBasis und falls  $y \leq 0$  die Ausnahme FalscherExponent geworfen werden. Falls die Berechnung zum arithmetischen Überlauf führt soll eine System.OverflowException geworfen werden. Die Ausnahmen FalscheBasis und FalscherExponent sollen von der Klasse ExponentialException abgeleitet werden. Testen Sie die Dll mit einer C#-Konsolenanwendung. Erproben Sie das Abfangen der Basis- und der abgeleiteten Exceptionklassen.

### **Aufgabe 4 (Thema: Interface)**

a) Welche Ausgabe erzeugt folgendes Programm? Wieso?

```
struct A
{
    public int i;
    public void set(int ii){i=ii;}
}
class T
{
```

```
static void Main(string[] args)
{
    A a = new A();
    a.i=11;
    Console.WriteLine(a.i);
    object o = a;
    ((A)o).set(12);
    // Zeile einfügen!!! Nur 1 Befehl
    Console.WriteLine(((A)o).i);
    Console.Read();
}
}
```

b) Ergänzen (nur Zusatz nichts streichen) sie den Programmcode und ändern Sie Main an der Stelle „// Zeile einfügen!!!“ so, dass bei der Ausgabe Console.WriteLine(((A)o).i); 12 erzeugt wird.