

Übungen L6

Die Abgabe dieser Übungen muss bis am Sonntag vor der übernächsten Vorlesung erfolgen.

Die Resultate sollen in einem ZIP File für alle Übungen dieser Woche in Moodle hochgeladen werden. Bitte das zip File mit dem eigenen Namen versehen.

Bei Abgabe von Sourcecode ist Grundbedingung, dass der Code kompilierbar ist. Nicht kompilierbare Ergebnisse werden nicht weiter validiert und als falsch gewertet.

0. Nachbereitung der Lektionen

- Lesen Sie Kapitel 11 & 19 im Lehrbuch und schauen Sie sich die Slides an.

1. Event Handling

Öffnen Sie das Beispiel aus dem Buch Chapter11\CarEventsWithLambdas.

Im Beispiel wird ein Fahrzeug beschleunigt, bis es explodiert. Wenn die Geschwindigkeit zu hoch wird, wird ein Event gefeuert, welcher meldet dass das Auto explodiert ist. Davor gibt es noch einen Event zur Warnung, dass es bald explodieren wird.

Teilaufgabe 1

Passen Sie den Aufruf in der Klasse **Program.cs** an. Das Auto soll nur bis 90 beschleunigen und danach wieder auf 0 abbremsen.

Teilaufgabe 2

Bauen Sie in die Klasse **Car** einen weiteren Event namens **SlowedDown** ein. Der Event soll gefeuert werden, wenn die Geschwindigkeit kleiner als eine Variable **slowSpeed** wird. Die Variable **slowSpeed** soll bei der Instanzierung des **Cars** gesetzt werden können. Wenn der Event gefeuert wird, geben sie die Meldung „Speed is slow“ in den **CarEventArgs** mit.

Teilaufgabe 3

Passen Sie die Klasse **Program.cs** an, damit die Variable **slowSpeed** auf 50 gesetzt wird. Registrieren sie den Event **SlowedDown** mit einer Lambda Expression und schauen Sie sich den Output auf der Konsole an.

Programmierung mit dem .NET Framework

Teilaufgabe 4

Erstellen Sie in der Klasse **Car.cs** drei Hilfsmethoden um die Events auszulösen. Benennen Sie diese mit: **OnExploded**, **OnAboutToBlow**, **OnSlowedDown**.

Befolgen Sie die Best Practices aus der Vorlesung und passen Sie die Aufrufe an.

Abgabe -> Komplette Solution im zip file in Moodle hochladen. Bezeichnung: L6U1.cs

2. Parallele Programmierung

Öffnen Sie das Beispiel aus dem Buch Chapter11\AsyncCallbackDelegate. Führen Sie die Applikation aus und schauen Sie sich den Source Code an.

Teilaufgabe 1

Fügen Sie neuen Code am Ende der **Main()** Methode hinzu. Machen Sie einen asynchronen Aufruf, indem Sie **BeginInvoke** und **EndInvoke** auf dem **b** delegate aufrufen. Geben Sie den Rückgabewert auf der Konsole aus.

Teilaufgabe 2

Fügen Sie neuen Code am Ende der **Main()** Methode hinzu. Machen Sie einen asynchronen Aufruf, indem Sie Die Klasse **System.Threading.Tasks.Task** benützen. Überlegen Sie sich, wie sie nach der Ausführung den Rückgabewert erhalten und geben Sie diesen auf der Konsole aus.

Teilaufgabe 3

Fügen Sie neuen Code am Ende der **Main()** Methode hinzu. Machen Sie einen asynchronen Aufruf, indem Sie die Klasse **System.Threading.Tasks.Parallel** benützen und rufen Sie die Methode **Add()** 3 mal parallel auf. Benutzen Sie dafür **Lambda Expressions**. (Der Rückgabewert kann in dieser Teilaufgabe ignoriert werden.

Abgabe -> Codedatei (*.cs) im zip file in Moodle hochladen. Bezeichnung: L6U2.cs

Das vollständige zip file sollte folgende Dateien enthalten:

- L6U1.zip
- L6U2.cs