

Inhaltsverzeichnis

1. Einleitung	7
1.1 Über den Autor	7
1.2 Warum C#?	8
1.3 Die Historie von C#	9
1.4 Der .NET-Framework, CIL, die CLI und die CLR	12
2. Visual Studio: die professionelle Entwicklungsumgebung für C#	15
2.1 Die Installation von Visual Studio	15
2.2 Ein Projekt in Visual Studio erstellen	19
3. Das erste Programm in C#	26
3.1 Der Aufbau einer Konsolenanwendung	26
3.2 Aufrufargumente verarbeiten	28
3.3 Kommentare erleichtern das Leben	30
3.4 Übungsaufgabe: Eine Erweiterung für „Hello World!“	32
4. Variablen und ihre Typen	35
4.1 Variablen in der Programmierung: Fast so wie in der Mathematik	35
4.2 Variablen, Typen und Operatoren	36
4.3 Variablen in einem C#-Programm verwenden	46
4.4 Benutzereingaben in Variablen speichern	48
4.5 Konvertieren von Variablen	50
4.6 Übungsaufgabe: Rechnen mit Variablen	52
5. Verzweigungen zur Steuerung des Programmablaufs	55
5.1 Die einfache, bedingte Verzweigung	55
5.2 Vergleichsoperatoren und logische Operatoren	56
5.3 Geschachtelte bedingte Verzweigungen	59
5.4 Switch-Case: Die Mehrfachverzweigung	60
5.5 Die bedingte Zuweisung	67
5.6 Übungsaufgabe: Programmieren mit Verzweigungen	70
6. Programmteile wiederholen mit Schleifen	76
6.1 Die while-Schleife: Erst prüfen, dann arbeiten	76
6.2 Die do-while-Schleife: Erst arbeiten, dann prüfen	78
6.3 Die for-Schleife: Eine feste Anzahl von Wiederholungen	82
6.4 Die foreach-Schleife läuft über alles	84
6.5 Feintuning von Schleifen mit break und continue	87
6.6 Übungsaufgabe: Programmsteuerung mit Schleifen	93
7. Strukturierte Daten in C#	101
7.1 Arrays: Eine Variable für viele Werte	101
7.2 Mehrdimensionale Arrays	103
7.3 Typisierte Listen: Arrays mit Komfort	104
7.4 Das Dictionary, die elegante Listenverwaltung	107
7.5 Das Tuple: Mehrere verschiedene Variablen in einer Struktur	109
7.6 Übungsaufgaben: Arbeiten mit Arrays, Listen, Dictionaries und Tuples	111

8. Methoden schaffen Ordnung	119
8.1 Wozu benötigt man Methoden?	119
8.2 Einfache Methoden.....	119
8.3 Methoden mit Übergabeparametern.....	120
8.4 Methoden mit Rückgabewerten.....	123
8.5 Methoden überladen	128
8.6 Übungsaufgaben: Programmieren mit Methoden	129
9. Grundlagen der Objektorientierten Programmierung	151
9.1 Was ist Objektorientierte Programmierung?.....	152
9.2 Eine Klasse erstellen	155
9.3 Klassen mit Konstruktoren.....	158
9.4 Eine Instanz einer Klasse erzeugen und verwenden	160
9.5 Zugriffsmodifizierer für Klassen, Eigenschaften und Methoden.....	164
9.6 Properties statt Variablen verwenden	165
9.7 Operatoren überladen.....	174
9.8 Übungsaufgabe: Mit Objekten programmieren.....	178
10. Objektorientierung für Fortgeschrittene	203
10.1 Vererbung: abgeleitete Klassen	203
10.2 Das Interface: Eine definierte Schnittstelle	210
10.3 Der Garbage Collector: Eine automatische Speicherverwaltung	217
10.4 Polymorphie: Virtuelle Methoden und Properties	229
10.5 Abstrakte und versiegelte Klassen	240
10.6 Erweiterungsmethoden	243
10.7 Generische Klassen.....	247
10.8 Generische Methoden.....	272
10.9 Generische Properties.....	275
10.10 Methoden und Variablen: Delegaten verwischen die Unterschiede.....	277
10.11 Übungsaufgabe: Fortgeschrittene Programmierung mit Objekten	282
11. Objekte verarbeiten mit „Linq to Objects“	296
11.1 Was ist Linq?	296
11.2 Ein erstes Beispiel für eine Linq-Abfrage	297
11.3 Daten abfragen mit Linq	300
11.4 Daten konvertieren mit Linq.....	306
11.5 Linq-Methoden verketten	313
11.6 Verschachtelte Linq-Ausdrücke	317
11.7 Wann wird eine Linq Abfrage ausgeführt?	320
11.8 Parallel Verarbeitung mit Linq.....	322
11.9 Übungsaufgabe: Linq verwenden.....	326
12. Fehlerbehandlung mit Exceptions	334
12.1 Was ist eine Exception?	334
12.2 Exceptions abfangen mit try – catch.....	336
12.3 Exceptions kontrolliert auslösen.....	340
12.4 Eigene Exceptions definieren	342
12.5 Übungsaufgabe: Ein Programm mit Fehlerbehandlung erstellen	346
13. Visual Studio reloaded: Funktionalitäten für Fortgeschrittene	352
13.1 Ein weiteres Projekt zur Projektmappe hinzufügen	352
13.2 Eine Klassenbibliothek erstellen	361
13.3 Die eigene Klassenbibliothek verwenden.....	364
13.4 Codenavigation.....	370

13.5 Fehler finden mit dem Debugger.....	384
13.6 Übungsaufgabe: Eine eigene Klassenbibliothek erstellen	391
14. Dateizugriff mit C#	400
14.1 Textdateien lesen und schreiben.....	400
14.2 XML-Dateien verarbeiten.....	411
14.3 Objekte serialisieren und deserialisieren.....	416
14.4 Verzeichnisse erzeugen und durchsuchen.....	423
14.5 Übungsaufgaben: Arbeiten mit Dateien	426
15. Datenbankzugriff mit dem Microsoft SQL-Server	435
15.1 Microsoft SQL-Server installieren	435
15.2 Eine Datenbank erstellen.....	451
15.3 Den Entity Framework zum Projekt hinzufügen.....	453
15.4 Ein Entity-Modell erstellen.....	455
15.5 Lesen, ändern, hinzufügen und löschen von Daten	470
15.6 Übungsaufgaben: Programmierung mit Datenbanken.....	487
16. ASP.NET MVC – Anwendungen fürs Web	502
16.1 Eine neue ASP.NET MVC-Anwendung erstellen.....	502
16.2 Einen Controller und eine View hinzufügen.....	519
16.3 Daten mit dem Controller bereitstellen.....	525
16.4 Daten mit der View anzeigen	531
16.5 Eine eigene View, um Daten zu ändern	533
16.6 Übungsaufgabe: Die Webanwendung erweitern.....	539
17. Grafische Benutzeroberflächen mit WPF	548
17.1 Eine neue WPF-Anwendung erstellen	548
17.2 Was ist XAML?	552
17.3 Wir programmieren einen einfachen Textdateibetrachter	556
17.4 Organisation der Benutzeroberfläche mit dem Grid-Steuerelement	563
17.5 Auswahl einer Datei mit einem OpenFileDialog-Objekt	567
17.6 Auflistung der Dateien mit einem ListBox-Steuerelement	570
17.7 Styling mit Ressourcen	577
17.8 Das MVVM-Entwurfsmuster	587
17.9 Übungsaufgabe: Eine WPF-Anwendung auf MVVM umstellen.....	596
18. Testautomatisierung mit MS-Test	608
18.1 Ein Testprojekt anlegen	608
18.2 Erstellen eines Komponententests	610
18.3 Verwendung des Test-Explorers.....	613
18.4 Abhängigkeiten reduzieren mit Mocks	614
18.5 Übungsaufgabe: Komponententests unter Verwendung von Mocks	620
19. Schlusswort	627
20. Glossar	628
21. Index	633