

Auf einen Blick

Vorwort	19
Kapitel 1 Einleitung	21
Kapitel 2 Die erste C#-Anwendung	33
Kapitel 3 Visual C# 2010 Express Edition	47
Kapitel 4 Sprachgrundlagen	67
Kapitel 5 Objektorientierte Sprachgrundlagen	103
Kapitel 6 Namespaces	143
Kapitel 7 Vererbung	153
Kapitel 8 Interfaces	179
Kapitel 9 Aufzählungstypen	191
Kapitel 10 Arrays	197
Kapitel 11 Zeichenkettenverarbeitung	207
Kapitel 12 Eigenschaften und Indexer	223
Kapitel 13 Delegaten und Ereignisbehandlung	233
Kapitel 14 Exception-Behandlung	249
Kapitel 15 Attribute	263
Kapitel 16 Auflistungen	273
Kapitel 17 Generische Typen	293
Kapitel 18 Dateien, Verzeichnisse und Streams	305

Kapitel 19	Assemblies	333
Kapitel 20	Anwendungsweitergabe	349
Kapitel 21	Konsolenanwendungen	357
Kapitel 22	Windows-Anwendungen	367
Kapitel 23	Dialoge und Menüs	407
Kapitel 24	Grafikprogrammierung	425
Kapitel 25	Steuerelemente erstellen	447
Kapitel 26	Dokumentationskommentare	461
Kapitel 27	Threads	469
Kapitel 28	Netzwerkanwendungen	485
Kapitel 29	Reflection	503
Kapitel 30	XML	513
Kapitel 31	LINQ	529
Kapitel 32	MSBuild	543
	Stichwortverzeichnis	557

Inhaltsverzeichnis

Vorwort	19
Kapitel 1 Einleitung	21
1.1 Das .NET Framework	21
1.1.1 Die Common Language Runtime (CLR)	22
1.1.2 Dynamic Language Runtime	24
1.2 Common Type System (CTS)	24
1.2.1 Common Language Specification (CLS)	25
1.2.2 Microsoft Intermediate Language	25
1.2.3 Just In Time-Compiler	26
1.2.4 Speicherverwaltung	26
1.2.5 Die Klassenbibliothek	27
1.2.6 Die Sprache C#	28
1.3 Die Installation des .NET Frameworks	29
1.4 Interessante Webseiten	30
1.5 Übungsaufgaben	31
Kapitel 2 Die erste C#-Anwendung	33
2.1 Aufbau einer C#-Anwendung	33
2.1.1 Erstellen des Sourcecodes	34
2.1.2 Verwendung des C#-Compilers	35
2.1.3 Ausführen der Anwendung	37
2.2 Namespaces	37
2.2.1 Namespaces einbinden	38
2.2.2 Namespaces deklarieren	38
2.3 Assemblies	39
2.4 Beispielanwendungen verwenden	41
2.4.1 Kommandozeile	41
2.4.2 MSBuild	42
2.4.3 Visual Studio	42
2.5 Ein- und Ausgabe von Daten	42
2.5.1 Ausgabe von Daten	42
2.5.2 Einlesen von Daten	43
2.5.3 Einlesen von Zahlenwerten	44
2.6 Übungsaufgaben	45

Kapitel 3 Visual C# 2010 Express Edition	47
3.1 Einführung	47
3.1.1 Allgemeines	47
3.1.2 Versionitis	47
3.1.3 Installation	48
3.2 Der erste Start	50
3.2.1 Die Startseite und die Hilfe konfigurieren	50
3.2.2 Eine Konsolenanwendung erstellen ..	51
3.2.3 Grundfunktionalität des Visual Studio	54
3.2.4 Projektverwaltung	55
3.2.5 Das Visual Studio konfigurieren	57
3.3 Verwendung des Text-Editors	58
3.4 Windows-Anwendungen	60
3.4.1 Eine Windows-Anwendung erstellen ..	60
3.5 Hilfe verwenden	62
3.6 Übungsaufgaben	65
Kapitel 4 Sprachgrundlagen	67
4.1 Basiselemente	67
4.1.1 Einführung	67
4.1.2 Reservierte Wörter und Kontext-schlüsselwörter	67
4.1.3 Bezeichner	68
4.1.4 Kommentare	69
4.1.5 Anweisungen und Anweisungsblöcke ..	71
4.1.6 Gültigkeitsbereiche	71
4.1.7 Allgemeiner Programmaufbau	72
4.2 Variablen und Konstanten	72
4.2.1 Variablen	72
4.2.2 Konstanten	76
4.3 Datentypen	77
4.3.1 Wert- und Referenztypen	77
4.3.2 Zeichenketten	80
4.3.3 Explizite und implizite Datentypkon-vertierung	81
4.3.4 Überlaufprüfung mit checked und un-checked	81
4.3.5 Boxing und Unboxing	82
4.4 Ausdrücke und Operatoren	84
4.4.1 Operatoren	84
4.4.2 Ausdrücke	85
4.5 Kontrollstrukturen	86
4.5.1 if-else-Anweisung	86
4.5.2 Bedingungsoperator	87

4.5.3	switch-Anweisung	88
4.5.4	for-Anweisung	89
4.5.5	foreach-Anweisung	91
4.5.6	while- und do-while-Anweisung.....	91
4.5.7	Ablaufsteuerung durch break und continue	93
4.5.8	goto-Anweisung	93
4.6	Direktiven	94
4.6.1	Bedingte Kompilierung	94
4.6.2	Unsicherer Code	98
4.7	Übungsaufgaben	100
Kapitel 5	Objektorientierte Sprachgrundlagen	103
5.1	Klassen und Objekte	103
5.1.1	Klassen	103
5.1.2	Objekte	105
5.1.3	Klassen deklarieren	106
5.2	Objekte erzeugen	108
5.3	Strukturen	111
5.4	Methoden	113
5.4.1	Methoden ohne Parameterübergabe ...	115
5.4.2	Methoden mit Parameterübergabe	116
5.4.3	Methoden überladen	122
5.4.4	Externe Methoden	123
5.5	Konstruktoren und Destruktoren	125
5.5.1	Konstruktoren	125
5.5.2	Objektinitialisierer	128
5.5.3	Destruktoren	129
5.6	Partielle Typdeklarationen	130
5.7	Partielle Methoden	132
5.8	Statische Klassen und Klassenelemente	133
5.9	Erweiterungsmethoden	136
5.10	Überladen von Operatoren	138
5.11	Übungsaufgaben	140
Kapitel 6	Namespaces	143
6.1	Namespaces deklarieren	143
6.1.1	Einführung	143
6.1.2	Verschachtelte Namespaces	145
6.1.3	Syntax von Namespaces	146
6.1.4	Namensgebung	147
6.2	Namespaces einbinden	147
6.2.1	Aliase	149
6.3	Die using-Anweisung.....	151
6.4	Übungsaufgaben	152

Kapitel 7 Vererbung	153
7.1 Grundprinzip	153
7.1.1 Vererbung von Klassen	155
7.1.2 Konstruktorverkettung	156
7.2 Zugriff auf die Basisklasse und das eigene Objekt	158
7.2.1 this	158
7.2.2 base	159
7.3 Zuweisungskompatibilität	159
7.3.1 Typkompatibilität prüfen	160
7.3.2 Sichere Typkonvertierung	160
7.4 Methoden verdecken	162
7.5 Polymorphie	164
7.5.1 Methoden überschreiben	164
7.5.2 Methoden versiegeln	165
7.6 Die Klasse Object	167
7.7 Abstrakte Klassen und Methoden	169
7.8 Versiegelte Klassen	173
7.9 Verschachtelte Klassen	173
7.10 Übungsaufgaben	176
Kapitel 8 Interfaces	179
8.1 Grundprinzip	179
8.2 Interfaces deklarieren	180
8.2.1 Interfaces implementieren	182
8.3 Explizite Interface-Deklaration	185
8.4 Das Interface IDisposable	187
8.5 Übungsaufgaben	190
Kapitel 9 Aufzählungstypen	191
9.1 Basistyp des Aufzählungstypen setzen	194
9.2 Die Klasse Enum	194
9.3 Übungsaufgaben	196
Kapitel 10 Arrays	197
10.1 Eindimensionale Arrays	197
10.2 Mehrdimensionale Arrays	200
10.3 Verzweigte Arrays	201
10.4 Parameterübergabe an Methoden	203
10.5 Statische Methoden der Klasse Array	204
10.6 Übungsaufgaben	206

Kapitel 11 Zeichenkettenverarbeitung	207
11.1 Einführung	207
11.2 Die Klasse String	207
11.2.1 Strings initialisieren	207
11.2.2 Länge einer Zeichenkette ermitteln	209
11.2.3 Zugriff auf einzelne Zeichen	209
11.2.4 Weitere Operationen mit Zeichenketten	209
11.2.5 Zeichenketten formatieren	216
11.3 Die Klasse StringBuilder	220
11.3.1 Anwendungsgebiet	220
11.3.2 Verketten mit Append()	220
11.3.3 Einfügen, Ersetzen oder Entfernen	221
11.4 Übungsaufgaben	221
Kapitel 12 Eigenschaften und Indexer	223
12.1 Eigenschaften	223
12.1.1 Einführung	223
12.1.2 Automatisch implementierte Eigenschaften	226
12.1.3 Asymmetrischer Accessor-Zugriff	226
12.1.4 Eigenschaften in Schnittstellen verwenden	227
12.2 Indexer	228
12.2.1 Einführung	228
12.2.2 Indexer in Schnittstellen verwenden ..	231
12.3 Übungsaufgaben	232
Kapitel 13 Delegaten und Ereignisbehandlung	233
13.1 Delegaten	233
13.1.1 Rückrufmethoden	236
13.2 Multicast-Delegaten	237
13.2.1 Kontra- und Kovarianz	239
13.2.2 Anonyme Methoden	240
13.2.3 Lambda-Ausdrücke	241
13.3 Ereignisse	244
13.3.1 Ereignisse bereitstellen	245
13.4 Übungsaufgaben	248

Kapitel 14 Exception-Behandlung	249
14.1 Einführung	249
14.2 Exceptions behandeln	250
14.2.1 Einsatz von try-catch-Blöcken	252
14.2.2 Verarbeitung von Exceptions	254
14.2.3 Eigenschaften und Methoden von Exceptions	255
14.2.4 Exception-Klassen	256
14.3 Exceptions auslösen	257
14.3.1 Neue Exceptions auslösen	257
14.3.2 Exceptions erneut auslösen	257
14.3.3 Verschachtelte Exceptions	257
14.4 Ressourcenschutz	258
14.5 Eigene Exceptions definieren	259
14.6 Übungsaufgaben	262
Kapitel 15 Attribute	263
15.1 Vordefinierte Attribute	263
15.2 Eigene Attribute definieren	267
15.2.1 Vorgehensweise	267
15.2.2 Verwenden eigener Attribute	269
15.2.3 Attributwerte auslesen	270
15.3 Übungsaufgaben	271
Kapitel 16 Auflistungen	273
16.1 Einführung	273
16.1.1 Anwendungsbeispiele	273
16.1.2 Übersicht der Auflistungsschnittstellen	274
16.2 Allgemeine Auflistungen	275
16.2.1 ArrayList	275
16.2.2 Queue	278
16.2.3 Stack	279
16.2.4 Hashtable	280
16.2.5 SortedList	282
16.2.6 Eigene Sortierungen implementieren ..	283
16.3 Auflistungsinitialisierer	285
16.4 Spezialisierte Auflistungen	285
16.4.1 StringCollection	286
16.4.2 StringDictionary	286
16.4.3 OrderedDictionary	286
16.4.4 NameValueCollection	286
16.4.5 ListDictionary	287
16.4.6 HybridDictionary	287
16.4.7 CollectionsUtil	287

16.5	Bit-Auflistungen	287
16.5.1	BitArray	287
16.5.2	BitVector32	289
16.6	Aufzählbare Klassenelemente	289
16.6.1	Enumeratoren	289
16.6.2	Iteratoren	290
16.6.3	Mehrere Iteratoren	291
16.7	Übungsaufgaben	291
Kapitel 17 Generische Typen		293
17.1	Einführung	293
17.2	Generische Typen	294
17.3	Typparameter einschränken	296
17.4	Generische Delegaten	298
17.5	Generische Auflistungen	299
17.5.1	Einfache Elementlisten mit der Klasse List	301
17.5.2	Sortierte Elementlisten mit der Klasse SortedList	302
17.6	Übungsaufgaben	303
Kapitel 18 Dateien, Verzeichnisse und Streams		305
18.1	Dateien und Verzeichnisse bearbeiten	306
18.1.1	Dateisysteminformationen	306
18.1.2	Umgebungsinformationen	308
18.1.3	Datei- und Verzeichnisoperationen	310
18.2	Dateiinhalte bearbeiten	315
18.2.1	Einfacher Stream-Zugriff	317
18.2.2	Zugriff auf Binärdateien	319
18.2.3	Zugriff auf Textdateien	321
18.2.4	Gepufferter Dateizugriff	323
18.3	Weitere Stream- und Dateifunktionen	324
18.3.1	Verwenden von Speicher-Streams	324
18.3.2	Kompression	326
18.3.3	Registry-Zugriff	326
18.3.4	Überwachung von Datei- und Verzeichnisoperationen	328
18.3.5	Isolierte Datenspeicherung	330
18.4	Übungsaufgaben	331
Kapitel 19 Assemblies		333
19.1	Einführung	333
19.1.1	Aufbau einer Assembly	334
19.1.2	Module	334
19.1.3	Manifest einer Assembly	336
19.1.4	Verwendung des IL-Disassemblers	337

19.2	Klassenbibliotheken	338
19.3	Friend-Assemblies	340
19.4	Assembly konfigurieren	341
19.5	Assemblies im GAC	343
19.6	Assemblies signieren	344
19.6.1	Starke Namen vergeben	345
19.6.2	Assemblies mit Zertifikat signieren ..	347
19.7	Übungsaufgaben	347
Kapitel 20 Anwendungsweitergabe		349
20.1	Einführung	349
20.2	Bereitstellungsvarianten	350
20.2.1	Die .NET-Laufzeitumgebung	351
20.3	XCopy-Bereitstellung	352
20.4	ClickOnce	352
20.4.1	Aufbau einer ClickOnce-Bereitstellung	353
20.4.2	Bereitstellung über das Visual Studio ..	353
20.5	Übungsaufgaben	356
Kapitel 21 Konsolenanwendungen		357
21.1	Die Klasse Console	357
21.1.1	Ein- und Ausgabestreams	357
21.1.2	Tastendruck entgegennehmen	358
21.1.3	Eigenschaften des Konsolenfensters ..	358
21.1.4	Der Cursor im Konsolenfenster	359
21.2	Konsolenanwendungen mit Parametern und Rückgabewert	362
21.3	Anwendungen über Batch-Dateien starten ..	363
21.4	Übungsaufgaben	364
Kapitel 22 Windows-Anwendungen		367
22.1	Einführung	367
22.2	Einfache Windows-Anwendungen	368
22.2.1	GUI ohne den Forms-Designer entwickeln	368
22.2.2	GUI mit dem Forms-Designer entwickeln	370
22.3	Mit Formularen arbeiten	374
22.3.1	Eigenschaften	374
22.3.2	Methoden	375
22.4	Zugriff auf Anwendungseigenschaften	379
22.5	Steuerelemente verwenden	379
22.6	Container	382
22.7	Steuerelemente anordnen	385
22.7.1	Steuerelemente hinzufügen	385
22.7.2	Layoutgestaltung eines Fensters	386

22.8	Ereignisse in Windows-Anwendungen	390
22.9	Verwendung von Komponenten	397
22.9.1	Beispiel Baumstrukturen	397
22.9.2	Beispielanwendung	398
22.10	Übungsaufgaben	403
Kapitel 23 Dialoge und Menüs		407
23.1	Standarddialoge	407
23.2	Hinweisfenster	408
23.3	Benutzerdefinierte Dialogfenster	409
23.4	Menüs	411
23.4.1	Hauptmenüs	411
23.4.2	Kontextmenüs	419
23.5	Symbolleisten	420
23.6	Übungsaufgaben	422
Kapitel 24 Grafikprogrammierung		425
24.1	Einführung	425
24.2	Bilder anzeigen	426
24.3	Grafiken zeichnen	428
24.3.1	Das Graphics-Objekt	428
24.3.2	Zeichenstifte, Farben und Pinsel	430
24.3.3	Zeichenmethoden	433
24.3.4	Text ausgeben	437
24.4	Verschiebung, Drehung, Größenänderung	438
24.5	Darstellung der Grafikobjekte verbessern	443
24.6	Übungsaufgaben	445
Kapitel 25 Steuerelemente erstellen		447
25.1	Einführung	447
25.2	Nichtgrafische Steuerelemente	448
25.2.1	Grafische Steuerelemente	451
25.2.2	Zusammengesetzte Steuerelemente	452
25.2.3	Steuerelemente erweitern	453
25.2.4	Neue Steuerelemente erstellen	453
25.3	Steuerelemente in die Toolbox einbinden	457
25.4	Übungsaufgaben	458
Kapitel 26 Dokumentationskommentare		461
26.1	Einführung	461
26.2	Verwendung der Kommentar-Tags	465
26.3	Verwendung im Visual Studio	466
26.4	Verarbeiten der XML-Datei	467
26.5	Übungsaufgaben	468

Kapitel 27 Threads	469
27.1 Einfache Threads	470
27.1.1 Datenübergabe an einen Thread	472
27.1.2 Zeitgeber	473
27.1.3 Übersicht über Methoden und Eigenschaften von Threads	473
27.1.4 Thread-Zustände	474
27.1.5 Auf einen Thread warten	476
27.1.6 Threads unterbrechen und fortsetzen ..	477
27.1.7 Hintergrund-Threads	478
27.2 Synchronisation	478
27.2.1 Einfache Sperren	479
27.2.2 Monitore	481
27.2.3 Mutexe	482
27.2.4 Deadlocks	483
27.3 Übungsaufgaben	483
Kapitel 28 Netzwerkanwendungen	485
28.1 Hilfsklassen	485
28.2 Zugriff auf Daten aus dem Netzwerk	488
28.2.1 Die Klasse WebClient	488
28.2.2 Die Klassen WebRequest und WebResponse	490
28.3 Socket-Verbindungen	492
28.3.1 Die Klasse Socket	493
28.3.2 Klassen TcpClient und TcpListener ..	499
28.4 Übungsaufgaben	501
Kapitel 29 Reflection	503
29.1 Einführung	503
29.1.1 Anwendungsbereiche	503
29.1.2 Assemblies analysieren	504
29.2 Assemblies dynamisch laden	507
29.2.1 Dynamisches Laden	508
29.2.2 Über Schnittstellen auf Typen in Assemblies zugreifen	510
29.3 Übungsaufgaben	512
Kapitel 30 XML	513
30.1 Einführung	513
30.2 Streambasierte XML-Verarbeitung	515
30.2.1 XmlReader	515
30.2.2 Validierung	517
30.2.3 XmlWriter	519

30.3	DOM-basierte XML-Verarbeitung	521
30.3.1	XML-DOM-Klassen	521
30.3.2	Lesen und Schreiben mit XmlDocument	523
30.4	Selektion von XML-Daten über XPath	524
30.5	Transformation von XML-Daten mittels XSLT ..	525
30.6	Übungsaufgaben	527
Kapitel 31 LINQ	529
31.1	Einführung	529
31.1.1	Was braucht man für LINQ?	531
31.1.2	Zeitpunkt der Ausführung festlegen ..	533
31.1.3	Generische und nicht generische Auflistungen abfragen	534
31.1.4	Ergebnisse sortieren	536
31.1.5	Abfrage mit einem Filter versehen	536
31.1.6	Ergebnisse gruppieren	537
31.2	LINQ für XML	539
31.3	Übungsaufgaben	541
Kapitel 32 MSBuild	543
32.1	Einführung	543
32.1.1	Aufruf von MSBuild	544
32.1.2	Grundlegender Aufbau einer Projektdatei	544
32.2	Targets	545
32.3	Tasks	546
32.3.1	Bedingungen	547
32.3.2	Fehlerbehandlung	547
32.4	ItemGroups	548
32.4.1	Bedingungen	549
32.4.2	Metadaten	549
32.5	Eigenschaften	549
32.6	Weitere Elemente	551
32.7	Logger und Transformationen	553
32.7.1	Logger	553
32.7.2	Transformationen	553
32.8	Eigene Tasks	553
32.9	Übungsaufgaben	556
Stichwortverzeichnis	557