

# Inhaltsverzeichnis

<b>Vorwort</b>	<b>51</b>
 <b>Teil I: Grundlagen</b>	
<b>1 Einstieg in Visual Studio 2012</b>	<b>57</b>
1.1 Die Installation von Visual Studio 2012	57
1.1.1 Überblick über die Produktpalette	57
1.1.2 Anforderungen an Hard- und Software	59
1.2 Unser allererstes C#-Programm	60
1.2.1 Vorbereitungen	60
1.2.2 Quellcode schreiben	62
1.2.3 Programm kompilieren und testen	63
1.2.4 Einige Erläuterungen zum Quellcode	63
1.2.5 Konsolenanwendungen sind langweilig	64
1.3 Die Windows-Philosophie	65
1.3.1 Mensch-Rechner-Dialog	65
1.3.2 Objekt- und ereignisorientierte Programmierung	65
1.3.3 Programmieren mit Visual Studio 2012	67
1.4 Die Entwicklungsumgebung Visual Studio 2012	68
1.4.1 Neues Projekt	68
1.4.2 Die wichtigsten Fenster	69
1.5 Microsofts .NET-Technologie	72
1.5.1 Zur Geschichte von .NET	72
1.5.2 .NET-Features und Begriffe	75
1.6 Wichtige Neuigkeiten in Visual Studio 2012	82
1.6.1 Die neue Visual Studio 2012 Entwicklungsumgebung	82
1.6.2 Neuheiten im .NET Framework 4.5	85
1.6.3 C# 5.0 – Sprache und Compiler	87

1.7	Praxisbeispiele .....	87
1.7.1	Windows Forms-Anwendung für Einsteiger .....	87
1.7.2	Windows-Anwendung für fortgeschrittene Einsteiger .....	92
<b>2</b>	<b>Grundlagen der Sprache C# .....</b>	<b>101</b>
2.1	Grundbegriffe .....	101
2.1.1	Anweisungen .....	101
2.1.2	Bezeichner .....	102
2.1.3	Schlüsselwörter .....	103
2.1.4	Kommentare .....	103
2.2	Datentypen, Variablen und Konstanten .....	104
2.2.1	Fundamentale Typen .....	104
2.2.2	Werttypen versus Verweistypen .....	105
2.2.3	Benennung von Variablen .....	106
2.2.4	Deklaration von Variablen .....	106
2.2.5	Typsuffixe .....	108
2.2.6	Zeichen und Zeichenketten .....	109
2.2.7	object-Datentyp .....	111
2.2.8	Konstanten deklarieren .....	111
2.2.9	Nullable Types .....	112
2.2.10	Typinferenz .....	113
2.2.11	Gültigkeitsbereiche und Sichtbarkeit .....	114
2.3	Konvertieren von Datentypen .....	114
2.3.1	Implizite und explizite Konvertierung .....	114
2.3.2	Welcher Datentyp passt zu welchem? .....	116
2.3.3	Konvertieren von string .....	117
2.3.4	Die Convert-Klasse .....	119
2.3.5	Die Parse-Methode .....	119
2.3.6	Boxing und Unboxing .....	120
2.4	Operatoren .....	121
2.4.1	Arithmetische Operatoren .....	122
2.4.2	Zuweisungsoperatoren .....	123
2.4.3	Logische Operatoren .....	124
2.4.4	Rangfolge der Operatoren .....	127
2.5	Kontrollstrukturen .....	128
2.5.1	Verzweigungsbefehle .....	128
2.5.2	Schleifenanweisungen .....	131

2.6	Benutzerdefinierte Datentypen .....	134
2.6.1	Enumerationen .....	134
2.6.2	Strukturen .....	135
2.7	Nutzerdefinierte Methoden .....	137
2.7.1	Methoden mit Rückgabewert .....	138
2.7.2	Methoden ohne Rückgabewert .....	139
2.7.3	Parameterübergabe mit ref .....	140
2.7.4	Parameterübergabe mit out .....	141
2.7.5	Methodenüberladung .....	142
2.7.6	Optionale Parameter .....	143
2.7.7	Benannte Parameter .....	144
2.8	Praxisbeispiele .....	145
2.8.1	Vom PAP zur Konsolenanwendung .....	145
2.8.2	Ein Konsolen- in ein Windows-Programm verwandeln .....	147
2.8.3	Schleifenanweisungen verstehen .....	149
2.8.4	Benutzerdefinierte Methoden überladen .....	151
<b>3</b>	<b>OOP-Konzepte .....</b>	<b>155</b>
3.1	Kleine Einführung in die OOP .....	155
3.1.1	Historische Entwicklung .....	155
3.1.2	Grundbegriffe der OOP .....	157
3.1.3	Sichtbarkeit von Klassen und ihren Mitgliedern .....	159
3.1.4	Allgemeiner Aufbau einer Klasse .....	160
3.1.5	Das Erzeugen eines Objekts .....	161
3.1.6	Einführungsbeispiel .....	164
3.2	Eigenschaften .....	169
3.2.1	Eigenschaften mit Zugriffsmethoden kapseln .....	169
3.2.2	Berechnete Eigenschaften .....	171
3.2.3	Lese-/Schreibschutz .....	173
3.2.4	Property-Accessoren .....	174
3.2.5	Statische Felder/Eigenschaften .....	174
3.2.6	Einfache Eigenschaften automatisch implementieren .....	177
3.3	Methoden .....	178
3.3.1	Öffentliche und private Methoden .....	178
3.3.2	Überladene Methoden .....	179
3.3.3	Statische Methoden .....	179

3.4	Ereignisse .....	181
3.4.1	Ereignis hinzufügen .....	182
3.4.2	Ereignis verwenden .....	185
3.5	Arbeiten mit Konstruktor und Destruktor .....	188
3.5.1	Konstruktor und Objektinitialisierer .....	188
3.5.2	Destruktor und Garbage Collector .....	191
3.5.3	Mit using den Lebenszyklus des Objekts kapseln .....	194
3.5.4	Verzögerte Initialisierung .....	194
3.6	Vererbung und Polymorphie .....	195
3.6.1	Klassendiagramm .....	195
3.6.2	Method-Overriding .....	197
3.6.3	Klassen implementieren .....	197
3.6.4	Implementieren der Objekte .....	200
3.6.5	Ausblenden von Mitgliedern durch Vererbung .....	202
3.6.6	Allgemeine Hinweise und Regeln zur Vererbung .....	204
3.6.7	Polymorphes Verhalten .....	205
3.6.8	Die Rolle von System.Object .....	208
3.7	Spezielle Klassen .....	209
3.7.1	Abstrakte Klassen .....	209
3.7.2	Versiegelte Klassen .....	210
3.7.3	Partielle Klassen .....	211
3.7.4	Statische Klassen .....	212
3.8	Schnittstellen (Interfaces) .....	213
3.8.1	Definition einer Schnittstelle .....	213
3.8.2	Implementieren einer Schnittstelle .....	214
3.8.3	Abfragen, ob Schnittstelle vorhanden ist .....	215
3.8.4	Mehrere Schnittstellen implementieren .....	215
3.8.5	Schnittstellenprogrammierung ist ein weites Feld .....	215
3.9	Praxisbeispiele .....	216
3.9.1	Eigenschaften sinnvoll kapseln .....	216
3.9.2	Eine statische Klasse anwenden .....	219
3.9.3	Vom fetten zum schlanken Client .....	221
3.9.4	Schnittstellenvererbung verstehen .....	232
<b>4</b>	<b>Arrays, Strings, Funktionen .....</b>	<b>239</b>
4.1	Datenfelder (Arrays) .....	239
4.1.1	Array deklarieren .....	239
4.1.2	Array instanziiieren .....	240

4.1.3	Array initialisieren .....	240
4.1.4	Zugriff auf Array-Elemente .....	241
4.1.5	Zugriff mittels Schleife .....	242
4.1.6	Mehrdimensionale Arrays .....	243
4.1.7	Zuweisen von Arrays .....	245
4.1.8	Arrays aus Strukturvariablen .....	246
4.1.9	Löschen und Umdimensionieren von Arrays .....	247
4.1.10	Eigenschaften und Methoden von Arrays .....	248
4.1.11	Übergabe von Arrays .....	250
4.2	Verarbeiten von Zeichenketten .....	251
4.2.1	Zuweisen von Strings .....	251
4.2.2	Eigenschaften und Methoden von String-Variablen .....	252
4.2.3	Wichtige Methoden der String-Klasse .....	254
4.2.4	Die StringBuilder-Klasse .....	256
4.3	Reguläre Ausdrücke .....	258
4.3.1	Wozu werden reguläre Ausdrücke verwendet? .....	259
4.3.2	Eine kleine Einführung .....	259
4.3.3	Wichtige Methoden/Eigenschaften der Klasse Regex .....	260
4.3.4	Kompilierte reguläre Ausdrücke .....	262
4.3.5	RegexOptions-Enumeration .....	263
4.3.6	Metazeichen (Escape-Zeichen) .....	263
4.3.7	Zeichenmengen (Character Sets) .....	264
4.3.8	Quantifizierer .....	266
4.3.9	Zero-Width Assertions .....	267
4.3.10	Gruppen .....	270
4.3.11	Text ersetzen .....	271
4.3.12	Text splitten .....	272
4.4	Datums- und Zeitberechnungen .....	273
4.4.1	Die DateTime-Struktur .....	273
4.4.2	Wichtige Eigenschaften von DateTime-Variablen .....	274
4.4.3	Wichtige Methoden von DateTime-Variablen .....	275
4.4.4	Wichtige Mitglieder der DateTime-Struktur .....	275
4.4.5	Konvertieren von Datumstrings in DateTime-Werte .....	276
4.4.6	Die TimeSpan-Struktur .....	277
4.5	Mathematische Funktionen .....	278
4.5.1	Überblick .....	278
4.5.2	Zahlen runden .....	279
4.5.3	Winkel umrechnen .....	279

4.5.4	Potenz- und Wurzeloperationen .....	279
4.5.5	Logarithmus und Exponentialfunktionen .....	280
4.5.6	Zufallszahlen erzeugen .....	280
4.6	Zahlen- und Datumsformatierungen .....	281
4.6.1	Anwenden der ToString-Methode .....	281
4.6.2	Anwenden der Format-Methode .....	283
4.7	Praxisbeispiele .....	285
4.7.1	Zeichenketten verarbeiten .....	285
4.7.2	Zeichenketten mit StringBuilder addieren .....	288
4.7.3	Reguläre Ausdrücke testen .....	291
4.7.4	Methodenaufrufe mit Array-Parametern .....	293
<b>5</b>	<b>Weitere Sprachfeatures .....</b>	<b>297</b>
5.1	Namespaces (Namensräume) .....	297
5.1.1	Ein kleiner Überblick .....	297
5.1.2	Einen eigenen Namespace einrichten .....	298
5.1.3	Die using-Anweisung .....	299
5.1.4	Namespace Alias .....	300
5.1.5	Namespace Alias Qualifizierer .....	300
5.2	Operatorenüberladung .....	301
5.2.1	Syntaxregeln .....	301
5.2.2	Praktische Anwendung .....	302
5.3	Collections (Auflistungen) .....	303
5.3.1	Die Schnittstelle IEnumerable .....	303
5.3.2	ArrayList .....	305
5.3.3	Hashtable .....	307
5.3.4	Indexer .....	307
5.4	Generics .....	309
5.4.1	Klassische Vorgehensweise .....	310
5.4.2	Generics bieten Typsicherheit .....	311
5.4.3	Generische Methoden .....	312
5.4.4	Iteratoren .....	313
5.5	Generische Collections .....	314
5.5.1	List-Collection statt ArrayList .....	314
5.5.2	Vorteile generischer Collections .....	315
5.5.3	Constraints .....	315

5.6	Das Prinzip der Delegates .....	315
5.6.1	Delegates sind Methodenzeiger .....	316
5.6.2	Einen Delegate-Typ deklarieren .....	316
5.6.3	Ein Delegate-Objekt erzeugen .....	316
5.6.4	Delegates vereinfacht instanziiieren .....	318
5.6.5	Anonyme Methoden .....	319
5.6.6	Lambda-Ausdrücke .....	320
5.6.7	Lambda-Ausdrücke in der Task Parallel Library .....	322
5.7	Dynamische Programmierung .....	324
5.7.1	Wozu dynamische Programmierung? .....	324
5.7.2	Das Prinzip der dynamischen Programmierung .....	324
5.7.3	Optionale Parameter sind hilfreich .....	327
5.7.4	Kovarianz und Kontravarianz .....	328
5.8	Weitere Datentypen .....	328
5.8.1	BigInteger .....	328
5.8.2	Complex .....	331
5.8.3	Tuple<> .....	331
5.8.4	SortedSet<> .....	332
5.9	Praxisbeispiele .....	333
5.9.1	ArrayList versus generische List .....	333
5.9.2	Generische IEnumerable-Interfaces implementieren .....	337
5.9.3	Delegates, anonyme Methoden, Lambda Expressions .....	340
5.9.4	Dynamischer Zugriff auf COM Interop .....	344
<b>6</b>	<b>Einführung in LINQ .....</b>	<b>347</b>
6.1	LINQ-Grundlagen .....	347
6.1.1	Die LINQ-Architektur .....	347
6.1.2	Anonyme Typen .....	348
6.1.3	Erweiterungsmethoden .....	350
6.2	Abfragen mit LINQ to Objects .....	351
6.2.1	Grundlegendes zur LINQ-Syntax .....	351
6.2.2	Zwei alternative Schreibweisen von LINQ Abfragen .....	352
6.2.3	Übersicht der wichtigsten Abfrage-Operatoren .....	353
6.3	LINQ-Abfragen im Detail .....	354
6.3.1	Die Projektionsoperatoren Select und SelectMany .....	355
6.3.2	Der Restriktionsoperator Where .....	356
6.3.3	Die Sortierungsoperatoren OrderBy und ThenBy .....	357
6.3.4	Der Gruppierungsoperator GroupBy .....	358

6.3.5	Verknüpfen mit Join .....	361
6.3.6	Aggregat-Operatoren .....	361
6.3.7	Verzögertes Ausführen von LINQ-Abfragen .....	363
6.3.8	Konvertierungsmethoden .....	364
6.3.9	Abfragen mit PLINQ .....	364
6.4	Praxisbeispiele .....	367
6.4.1	Die Syntax von LINQ-Abfragen verstehen .....	367
6.4.2	Aggregat-Abfragen mit LINQ .....	370

## Teil II: Technologien

<b>7</b>	<b>Zugriff auf das Dateisystem .....</b>	<b>375</b>
7.1	Grundlagen .....	375
7.1.1	Klassen für den Zugriff auf das Dateisystem .....	376
7.1.2	Statische versus Instanzen-Klasse .....	376
7.2	Übersichten .....	377
7.2.1	Methoden der Directory-Klasse .....	377
7.2.2	Methoden eines DirectoryInfo-Objekts .....	378
7.2.3	Eigenschaften eines DirectoryInfo-Objekts .....	378
7.2.4	Methoden der File-Klasse .....	378
7.2.5	Methoden eines FileInfo-Objekts .....	379
7.2.6	Eigenschaften eines FileInfo-Objekts .....	380
7.3	Operationen auf Verzeichnisebene .....	380
7.3.1	Existenz eines Verzeichnisses/einer Datei feststellen .....	380
7.3.2	Verzeichnisse erzeugen und löschen .....	381
7.3.3	Verzeichnisse verschieben und umbenennen .....	381
7.3.4	Aktuelles Verzeichnis bestimmen .....	382
7.3.5	Unterverzeichnisse ermitteln .....	382
7.3.6	Alle Laufwerke ermitteln .....	382
7.3.7	Dateien kopieren und verschieben .....	383
7.3.8	Dateien umbenennen .....	384
7.3.9	Dateiattribute feststellen .....	384
7.3.10	Verzeichnis einer Datei ermitteln .....	386
7.3.11	Alle im Verzeichnis enthaltenen Dateien ermitteln .....	386
7.3.12	Dateien und Unterverzeichnisse ermitteln .....	386



7.4	Zugriffsberechtigungen .....	387
7.4.1	ACL und ACE .....	387
7.4.2	SetAccessControl-Methode .....	388
7.4.3	Zugriffsrechte anzeigen .....	388
7.5	Weitere wichtige Klassen .....	389
7.5.1	Die Path-Klasse .....	389
7.5.2	Die Klasse FileSystemWatcher .....	390
7.6	Datei- und Verzeichnisdialoge .....	391
7.6.1	OpenFileDialog und SaveFileDialog .....	392
7.6.2	FolderBrowserDialog .....	393
7.7	Praxisbeispiele .....	394
7.7.1	Infos über Verzeichnisse und Dateien gewinnen .....	394
7.7.2	Eine Verzeichnisstruktur in die TreeView einlesen .....	398
7.7.3	Mit LINQ und RegEx Verzeichnisbäume durchsuchen .....	400
<b>8</b>	<b>Dateien lesen und schreiben .....</b>	<b>405</b>
8.1	Grundprinzip der Datenpersistenz .....	405
8.1.1	Dateien und Streams .....	405
8.1.2	Die wichtigsten Klassen .....	406
8.1.3	Erzeugen eines Streams .....	407
8.2	Dateiparameter .....	407
8.2.1	FileAccess .....	407
8.2.2	FileMode .....	407
8.2.3	FileShare .....	408
8.3	Textdateien .....	408
8.3.1	Eine Textdatei beschreiben bzw. neu anlegen .....	408
8.3.2	Eine Textdatei lesen .....	410
8.4	Binärdateien .....	411
8.4.1	Lese-/Schreibzugriff .....	411
8.4.2	Die Methoden ReadAllBytes und WriteAllBytes .....	412
8.4.3	Erzeugen von BinaryReader/BinaryWriter .....	412
8.5	Sequenzielle Dateien .....	413
8.5.1	Lesen und schreiben von strukturierten Daten .....	413
8.5.2	Serialisieren von Objekten .....	414
8.6	Dateien verschlüsseln und komprimieren .....	415
8.6.1	Das Methodenpärchen Encrypt/Decrypt .....	415
8.6.2	Verschlüsseln unter Vista/Windows 7/Windows 8 .....	415

8.6.3	Verschlüsseln mit der CryptoStream-Klasse .....	416
8.6.4	Dateien komprimieren .....	417
8.7	Memory Mapped Files .....	418
8.7.1	Grundprinzip .....	418
8.7.2	Erzeugen eines MMF .....	419
8.7.3	Erstellen eines Map View .....	420
8.8	Praxisbeispiele .....	421
8.8.1	Auf eine Textdatei zugreifen .....	421
8.8.2	Einen Objektbaum persistent speichern .....	424
8.8.3	Ein Memory Mapped File (MMF) verwenden .....	431
<b>9</b>	<b>Asynchrone Programmierung .....</b>	<b>435</b>
9.1	Übersicht .....	435
9.1.1	Multitasking versus Multithreading .....	436
9.1.2	Deadlocks .....	437
9.1.3	Racing .....	437
9.2	Programmieren mit Threads .....	439
9.2.1	Einführungsbeispiel .....	439
9.2.2	Wichtige Thread-Methoden .....	440
9.2.3	Wichtige Thread-Eigenschaften .....	442
9.2.4	Einsatz der ThreadPool-Klasse .....	443
9.3	Sperrmechanismen .....	445
9.3.1	Threading ohne lock .....	445
9.3.2	Threading mit lock .....	447
9.3.3	Die Monitor-Klasse .....	449
9.3.4	Mutex .....	452
9.3.5	Methoden für die parallele Ausführung sperren .....	454
9.3.6	Semaphore .....	454
9.4	Interaktion mit der Programmoberfläche .....	456
9.4.1	Die Werkzeuge .....	456
9.4.2	Einzelne Steuerelemente mit Invoke aktualisieren .....	456
9.4.3	Mehrere Steuerelemente aktualisieren .....	458
9.4.4	Ist ein Invoke-Aufruf nötig? .....	458
9.4.5	Und was ist mit WPF? .....	459
9.5	Timer-Threads .....	460
9.6	Die BackgroundWorker-Komponente .....	461

9.7	Asynchrone Programmier-Entwurfsmuster .....	464
9.7.1	Kurzübersicht .....	464
9.7.2	Polling .....	465
9.7.3	Callback verwenden .....	467
9.7.4	Callback mit Parameterübergabe verwenden .....	468
9.7.5	Callback mit Zugriff auf die Programm-Oberfläche .....	469
9.8	Asynchroner Aufruf beliebiger Methoden .....	470
9.8.1	Die Beispielklasse .....	470
9.8.2	Asynchroner Aufruf ohne Callback .....	472
9.8.3	Asynchroner Aufruf mit Callback und Anzeigefunktion .....	473
9.8.4	Aufruf mit Rückgabewerten (per Eigenschaft) .....	474
9.8.5	Aufruf mit Rückgabewerten (per EndInvoke) .....	474
9.9	Es geht auch einfacher – async und await .....	475
9.9.1	Der Weg von Synchron zu Asynchron .....	476
9.9.2	Achtung: Fehlerquellen! .....	478
9.9.3	Eigene asynchrone Methoden entwickeln .....	480
9.10	Praxisbeispiele .....	482
9.10.1	Spieltrieb & Multithreading erleben .....	482
9.10.2	Prozess- und Thread-Informationen gewinnen .....	495
9.10.3	Ein externes Programm starten .....	500
<b>10</b>	<b>Die Task Parallel Library .....</b>	<b>503</b>
10.1	Überblick .....	503
10.1.1	Parallel-Programmierung .....	503
10.1.2	Möglichkeiten der TPL .....	506
10.1.3	Der CLR-Threadpool .....	506
10.2	Parallele Verarbeitung mit Parallel.Invoke .....	507
10.2.1	Aufrufvarianten .....	508
10.2.2	Einschränkungen .....	509
10.3	Verwendung von Parallel.For .....	509
10.3.1	Abbrechen der Verarbeitung .....	511
10.3.2	Auswerten des Verarbeitungsstatus .....	512
10.3.3	Und was ist mit anderen Iterator-Schrittweiten? .....	513
10.4	Collections mit Parallel.ForEach verarbeiten .....	513
10.5	Die Task-Klasse .....	514
10.5.1	Einen Task erzeugen .....	514
10.5.2	Den Task starten .....	515
10.5.3	Datenübergabe an den Task .....	517

10.5.4	Wie warte ich auf das Ende des Task? .....	518
10.5.5	Tasks mit Rückgabewerten .....	520
10.5.6	Die Verarbeitung abbrechen .....	523
10.5.7	Fehlerbehandlung .....	526
10.5.8	Weitere Eigenschaften .....	527
10.6	Zugriff auf das User-Interface .....	528
10.6.1	Task-Ende und Zugriff auf die Oberfläche .....	529
10.6.2	Zugriff auf das UI aus dem Task heraus .....	530
10.7	Weitere Datenstrukturen im Überblick .....	532
10.7.1	Threadsichere Collections .....	532
10.7.2	Primitive für die Threadsynchronisation .....	533
10.8	Parallel LINQ (PLINQ) .....	533
10.9	Die Parallel Diagnostic Tools .....	534
10.9.1	Fenster für parallele Aufgaben .....	534
10.9.2	Fenster für parallele Stacks .....	535
10.9.3	IntelliTrace .....	536
10.10	Praxisbeispiel: Spieltrieb – Version 2 .....	536
10.10.1	Aufgabenstellung .....	536
10.10.2	Global-Klasse .....	537
10.10.3	Controller-Klasse .....	538
10.10.4	LKW-Klasse .....	540
10.10.5	Schiff-Klasse .....	541
10.10.6	Oberfläche .....	544
<b>11</b>	<b>Fehlersuche und Behandlung .....</b>	<b>547</b>
11.1	Der Debugger .....	547
11.1.1	Allgemeine Beschreibung .....	547
11.1.2	Die wichtigsten Fenster .....	548
11.1.3	Debugging-Optionen .....	551
11.1.4	Praktisches Debugging am Beispiel .....	553
11.2	Arbeiten mit Debug und Trace .....	557
11.2.1	Wichtige Methoden von Debug und Trace .....	557
11.2.2	Besonderheiten der Trace-Klasse .....	561
11.2.3	TraceListener-Objekte .....	561
11.3	Caller Information .....	564
11.3.1	Attribute .....	564
11.3.2	Anwendung .....	564

11.4	Fehlerbehandlung .....	565
11.4.1	Anweisungen zur Fehlerbehandlung .....	565
11.4.2	try-catch .....	566
11.4.3	try-finally .....	571
11.4.4	Das Standardverhalten bei Ausnahmen festlegen .....	573
11.4.5	Die Exception-Klasse .....	574
11.4.6	Fehler/Ausnahmen auslösen .....	575
11.4.7	Eigene Fehlerklassen .....	575
11.4.8	Exceptionhandling zur Entwurfszeit .....	577
11.4.9	Code Contracts .....	578

## Teil III: WPF-Anwendungen

<b>12</b>	<b>Einführung in WPF .....</b>	<b>581</b>
12.1	Neues aus der Geräteküche .....	582
12.1.1	Silverlight .....	582
12.1.2	WPF .....	582
12.2	Einführung .....	583
12.2.1	Was kann eine WPF-Anwendung? .....	583
12.2.2	Die eXtensible Application Markup Language .....	585
12.2.3	Verbinden von XAML und C#-Code .....	589
12.2.4	Zielplattformen .....	595
12.2.5	Applikationstypen .....	596
12.2.6	Vor- und Nachteile von WPF-Anwendungen .....	597
12.2.7	Weitere Dateien im Überblick .....	597
12.3	Alles beginnt mit dem Layout .....	600
12.3.1	Allgemeines zum Layout .....	600
12.3.2	Positionieren von Steuerelementen .....	602
12.3.3	Canvas .....	606
12.3.4	StackPanel .....	606
12.3.5	DockPanel .....	608
12.3.6	WrapPanel .....	610
12.3.7	UniformGrid .....	610
12.3.8	Grid .....	612
12.3.9	ViewBox .....	617
12.3.10	TextBlock .....	618

12.4	Das WPF-Programm .....	621
12.4.1	Die App-Klasse .....	621
12.4.2	Das Startobjekt festlegen .....	622
12.4.3	Kommandozeilenparameter verarbeiten .....	623
12.4.4	Die Anwendung beenden .....	624
12.4.5	Auswerten von Anwendungsereignissen .....	624
12.5	Die Window-Klasse .....	625
12.5.1	Position und Größe festlegen .....	625
12.5.2	Rahmen und Beschriftung .....	626
12.5.3	Das Fenster-Icon ändern .....	626
12.5.4	Anzeige weiterer Fenster .....	627
12.5.5	Transparenz .....	627
12.5.6	Abstand zum Inhalt festlegen .....	628
12.5.7	Fenster ohne Fokus anzeigen .....	628
12.5.8	Ereignisfolge bei Fenstern .....	629
12.5.9	Ein paar Worte zur Schriftdarstellung .....	629
12.5.10	Ein paar Worte zur Darstellung von Controls .....	632
12.5.11	Wird mein Fenster komplett mit WPF gerendert? .....	634
<b>13</b>	<b>Übersicht WPF-Controls .....</b>	<b>635</b>
13.1	Allgemeingültige Eigenschaften .....	635
13.2	Label .....	637
13.3	Button, RepeatButton, ToggleButton .....	638
13.3.1	Schaltflächen für modale Dialoge .....	638
13.3.2	Schaltflächen mit Grafik .....	639
13.4	TextBox, PasswordBox .....	640
13.4.1	TextBox .....	640
13.4.2	PasswordBox .....	642
13.5	CheckBox .....	643
13.6	RadioButton .....	645
13.7	ListBox, ComboBox .....	646
13.7.1	ListBox .....	646
13.7.2	ComboBox .....	649
13.7.3	Den Content formatieren .....	651
13.8	Image .....	652
13.8.1	Grafik per XAML zuweisen .....	652
13.8.2	Grafik zur Laufzeit zuweisen .....	653
13.8.3	Bild aus Datei laden .....	654

13.8.4	Die Grafiskalierung beeinflussen .....	655
13.9	MediaElement .....	656
13.10	Slider, ScrollBar .....	658
13.10.1	Slider .....	658
13.10.2	ScrollBar .....	659
13.11	ScrollView .....	660
13.12	Menu, ContextMenu .....	661
13.12.1	Menu .....	661
13.12.2	Tastenkürzel .....	662
13.12.3	Grafiken .....	663
13.12.4	Weitere Möglichkeiten .....	664
13.12.5	ContextMenu .....	665
13.13	ToolBar .....	666
13.14	StatusBar, ProgressBar .....	669
13.14.1	StatusBar .....	669
13.14.2	ProgressBar .....	671
13.15	Border, GroupBox, BulletDecorator .....	672
13.15.1	Border .....	672
13.15.2	GroupBox .....	673
13.15.3	BulletDecorator .....	674
13.16	RichTextBox .....	676
13.16.1	Verwendung und Anzeige von vordefiniertem Text .....	676
13.16.2	Neues Dokument zur Laufzeit erzeugen .....	678
13.16.3	Sichern von Dokumenten .....	678
13.16.4	Laden von Dokumenten .....	680
13.16.5	Texte per Code einfügen/modifizieren .....	681
13.16.6	Texte formatieren .....	682
13.16.7	EditingCommands .....	683
13.16.8	Grafiken/Objekte einfügen .....	684
13.16.9	Rechtschreibkontrolle .....	686
13.17	FlowDocumentPageViewer & Co. ....	686
13.17.1	FlowDocumentPageViewer .....	686
13.17.2	FlowDocumentReader .....	687
13.17.3	FlowDocumentScrollView .....	687
13.18	FlowDocument .....	687
13.18.1	FlowDocument per XAML beschreiben .....	688
13.18.2	FlowDocument per Code erstellen .....	690
13.19	DocumentViewer .....	691

13.20	Expander, TabControl .....	692
13.20.1	Expander .....	692
13.20.2	TabControl .....	694
13.21	Popup .....	695
13.22	TreeView .....	697
13.23	ListView .....	700
13.24	DataGrid .....	701
13.25	Calendar/DatePicker .....	702
13.26	InkCanvas .....	706
13.26.1	Stift-Parameter definieren .....	706
13.26.2	Die Zeichenmodi .....	707
13.26.3	Inhalte laden und sichern .....	708
13.26.4	Konvertieren in eine Bitmap .....	708
13.26.5	Weitere Eigenschaften .....	709
13.27	Ellipse, Rectangle, Line und Co. ....	710
13.27.1	Ellipse .....	710
13.27.2	Rectangle .....	710
13.27.3	Line .....	711
13.28	Browser .....	711
13.29	Ribbon .....	714
13.29.1	Allgemeine Grundlagen .....	714
13.29.2	Download/Installation .....	715
13.29.3	Erste Schritte .....	716
13.29.4	Registerkarten und Gruppen .....	717
13.29.5	Kontextabhängige Registerkarten .....	718
13.29.6	Einfache Beschriftungen .....	719
13.29.7	Schaltflächen .....	719
13.29.8	Auswahllisten .....	721
13.29.9	Optionsauswahl .....	724
13.29.10	Texteingaben .....	724
13.29.11	Screentips .....	724
13.29.12	Symbolleiste für den Schnellzugriff .....	725
13.29.13	Das RibbonWindow .....	726
13.29.14	Menüs .....	727
13.29.15	Anwendungsmenü .....	729
13.29.16	Alternativen .....	732
13.30	Chart .....	732
13.31	WindowsFormsHost .....	733



<b>14</b>	<b>Wichtige WPF-Techniken</b>	<b>737</b>
14.1	Eigenschaften	737
14.1.1	Abhängige Eigenschaften (Dependency Properties)	737
14.1.2	Angehängte Eigenschaften (Attached Properties)	739
14.2	Einsatz von Ressourcen	739
14.2.1	Was sind eigentlich Ressourcen?	739
14.2.2	Wo können Ressourcen gespeichert werden?	739
14.2.3	Wie definiere ich eine Ressource?	741
14.2.4	Statische und dynamische Ressourcen	742
14.2.5	Wie werden Ressourcen adressiert?	743
14.2.6	System-Ressourcen einbinden	744
14.3	Das WPF-Ereignis-Modell	744
14.3.1	Einführung	744
14.3.2	Routed Events	745
14.3.3	Direkte Events	747
14.4	Verwendung von Commands	748
14.4.1	Einführung zu Commands	748
14.4.2	Verwendung vordefinierter Commands	748
14.4.3	Das Ziel des Commands	750
14.4.4	Welche vordefinierten Commands stehen zur Verfügung?	751
14.4.5	Commands an Ereignismethoden binden	752
14.4.6	Wie kann ich ein Command per Code auslösen?	753
14.4.7	Command-Ausführung verhindern	754
14.5	Das WPF-Style-System	754
14.5.1	Übersicht	754
14.5.2	Benannte Styles	755
14.5.3	Typ-Styles	756
14.5.4	Styles anpassen und vererben	757
14.6	Verwenden von Triggern	760
14.6.1	Eigenschaften-Trigger (Property Triggers)	760
14.6.2	Ereignis-Trigger	762
14.6.3	Daten-Trigger	763
14.7	Einsatz von Templates	764
14.7.1	Neues Template erstellen	764
14.7.2	Template abrufen und verändern	768
14.8	Transformationen, Animationen, StoryBoards	771
14.8.1	Transformationen	771
14.8.2	Animationen mit dem StoryBoard realisieren	777

14.9	Praxisbeispiel .....	781
14.9.1	Arbeiten mit Microsoft Expression Blend .....	781
<b>15</b>	<b>WPF-Datenbindung .....</b>	<b>787</b>
15.1	Grundprinzip .....	787
15.1.1	Bindungsarten .....	788
15.1.2	Wann eigentlich wird die Quelle aktualisiert? .....	789
15.1.3	Geht es auch etwas langsamer? .....	790
15.1.4	Bindung zur Laufzeit realisieren .....	791
15.2	Binden an Objekte .....	793
15.2.1	Objekte im XAML-Code instanziiieren .....	793
15.2.2	Verwenden der Instanz im C#-Quellcode .....	795
15.2.3	Anforderungen an die Quell-Klasse .....	795
15.2.4	Instanziiieren von Objekten per C#-Code .....	797
15.3	Binden von Collections .....	798
15.3.1	Anforderung an die Collection .....	798
15.3.2	Einfache Anzeige .....	799
15.3.3	Navigieren zwischen den Objekten .....	800
15.3.4	Einfache Anzeige in einer ListBox .....	802
15.3.5	DataTemplates zur Anzeigeformatierung .....	803
15.3.6	Mehr zu List- und ComboBox .....	804
15.3.7	Verwendung der ListView .....	806
15.4	Noch einmal zurück zu den Details .....	808
15.4.1	Navigieren in den Daten .....	808
15.4.2	Sortieren .....	810
15.4.3	Filtern .....	810
15.4.4	Live Shaping .....	811
15.5	Anzeige von Datenbankinhalten .....	812
15.5.1	Datenmodell per LINQ to SQL-Designer erzeugen .....	813
15.5.2	Die Programm-Oberfläche .....	814
15.5.3	Der Zugriff auf die Daten .....	815
15.6	Drag & Drop-Datenbindung .....	816
15.6.1	Vorgehensweise .....	816
15.6.2	Weitere Möglichkeiten .....	819
15.7	Formatieren von Werten .....	820
15.7.1	IValueConverter .....	821
15.7.2	BindingBase.StringFormat-Eigenschaft .....	823
15.8	Das DataGrid als Universalwerkzeug .....	825

15.8.1	Grundlagen der Anzeige .....	825
15.8.2	UI-Virtualisierung .....	826
15.8.3	Spalten selbst definieren .....	826
15.8.4	Zusatzinformationen in den Zeilen anzeigen .....	828
15.8.5	Vom Betrachten zum Editieren .....	829
15.9	Praxisbeispiele .....	830
15.9.1	Collections in Hintergrundthreads füllen .....	830
15.9.2	Drag & Drop-Bindung bei 1:n-Beziehungen .....	833
<b>16</b>	<b>Druckausgabe mit WPF .....</b>	<b>839</b>
16.1	Grundlagen .....	839
16.1.1	XPS-Dokumente .....	839
16.1.2	System.Printing .....	840
16.1.3	System.Windows.Xps .....	841
16.2	Einfache Druckausgaben mit dem PrintDialog .....	841
16.3	Mehrseitige Druckvorschau-Funktion .....	844
16.3.1	Fix-Dokumente .....	844
16.3.2	Flow-Dokumente .....	850
16.4	Druckerinfos, -auswahl, -konfiguration .....	853
16.4.1	Die installierten Drucker bestimmen .....	854
16.4.2	Den Standarddrucker bestimmen .....	855
16.4.3	Mehr über einzelne Drucker erfahren .....	855
16.4.4	Spezifische Druckeinstellungen vornehmen .....	857
16.4.5	Direkte Druckausgabe .....	859

## Teil IV: Windows Store Apps

<b>17</b>	<b>Erste Schritte in WinRT .....</b>	<b>863</b>
17.1	Grundkonzepte und Begriffe .....	863
17.1.1	Windows Runtime (WinRT) .....	863
17.1.2	Windows Store Apps .....	864
17.1.3	Fast and Fluid .....	865
17.1.4	Process Sandboxing und Contracts .....	866
17.1.5	.NET WinRT-Profil .....	868
17.1.6	Language Projection .....	868
17.1.7	Vollbildmodus .....	870

17.1.8	Windows Store .....	870
17.1.9	Zielpattformen .....	871
17.2	Entwurfsumgebung .....	872
17.2.1	Betriebssystem .....	872
17.2.2	Windows-Simulator .....	873
17.2.3	Remote-Debugging .....	875
17.3	Ein (kleines) Einstiegsbeispiel .....	876
17.3.1	Aufgabenstellung .....	876
17.3.2	Quellcode .....	876
17.3.3	Oberflächenentwurf .....	879
17.3.4	Installation und Test .....	881
17.3.5	Verbesserungen .....	882
17.3.6	Fazit .....	885
17.4	Weitere Details zu WinRT .....	887
17.4.1	Wo ist WinRT einzuordnen? .....	887
17.4.2	Die WinRT-API .....	888
17.4.3	Wichtige WinRT-Namespaces .....	890
17.4.4	Der Unterbau .....	891
17.5	Gedanken zum Thema "WinRT & Tablets" .....	894
17.5.1	Windows 8-Oberfläche versus Desktop .....	894
17.5.2	Tablets und Touchscreens .....	894
17.6	Praxisbeispiel .....	896
17.6.1	WinRT in Desktop-Applikationen nutzen .....	896
<b>18</b>	<b>WinRT-Oberflächen entwerfen .....</b>	<b>899</b>
18.1	Grundkonzepte .....	899
18.1.1	XAML (oder HTML 5) für die Oberfläche .....	900
18.1.2	Die Page, der Frame und das Window .....	901
18.1.3	Das Befehlsdesign .....	902
18.1.4	Die Navigationsdesigns .....	904
18.1.5	Achtung: Fingereingabe! .....	905
18.1.6	Verwendung von Schriftarten .....	906
18.2	Projekttypen und Seitentemplates .....	906
18.2.1	Leere App .....	906
18.2.2	Geteilte App (Split App) .....	908
18.2.3	Raster-App (Grid App) .....	910
18.2.4	Leere Seite (Blank Page) .....	911

18.2.5	Standardseite (Basic Page) .....	912
18.2.6	Ein eigenes Grundlayout erstellen .....	914
18.3	Seitenauswahl und -navigation .....	915
18.3.1	Die Startseite festlegen .....	915
18.3.2	Navigation und Parameterübergabe .....	915
18.3.3	Den Seitenstatus erhalten .....	916
18.4	Die vier App-Ansichten .....	917
18.4.1	Vollbild quer und hochkant .....	917
18.4.2	Angedockt und Füllmodus .....	918
18.4.3	Reagieren auf die Änderung .....	918
18.4.4	Angedockten Modus aktiv beenden .....	922
18.5	Skalieren von Apps .....	922
18.6	Praxisbeispiele .....	924
18.6.1	Seitennavigation und Parameterübergabe .....	924
18.6.2	Auf Ansichtsänderungen reagieren .....	926
18.7	Tipps & Tricks .....	930
18.7.1	Symbole für WinRT-Oberflächen finden .....	930
18.7.2	Wie werde ich das Grufti-Layout schnell los? .....	931
<b>19</b>	<b>Die wichtigsten Controls .....</b>	<b>933</b>
19.1	Einfache WinRT-Controls .....	933
19.1.1	TextBlock, RichTextBlock .....	933
19.1.2	Button, HyperlinkButton, RepeatButton .....	936
19.1.3	CheckBox, RadioButton, ToggleButton, ToggleSwitch .....	938
19.1.4	TextBox, PasswordBox, RichEditBox .....	939
19.1.5	Image .....	943
19.1.6	ScrollBar, Slider, ProgressBar, ProgressRing .....	945
19.1.7	Border, Ellipse, Rectangle .....	946
19.2	Layout-Controls .....	947
19.2.1	Canvas .....	947
19.2.2	StackPanel .....	948
19.2.3	ScrollView .....	948
19.2.4	Grid .....	949
19.2.5	VariableSizedWrapGrid .....	950
19.3	Listendarstellungen .....	951
19.3.1	ComboBox, ListBox .....	951
19.3.2	ListView .....	955

19.3.3	GridView	956
19.3.4	FlipView	959
19.4	Sonstige Controls	961
19.4.1	CaptureElement	961
19.4.2	MediaElement	962
19.4.3	Frame	964
19.4.4	WebView	964
19.4.5	ToolTip	965
19.5	Praxisbeispiele	967
19.5.1	Einen StringFormat-Konverter implementieren	967
19.5.2	Besonderheiten der TextBox kennen lernen	969
19.5.3	Daten in der GridView gruppieren	972
19.5.4	Das SemanticZoom-Control verwenden	977
19.5.5	Die CollectionViewSource verwenden	981
19.5.6	Zusammenspiel ListBox/AppBar	985
19.5.7	Musikwiedergabe im Hintergrund realisieren	988
<b>20</b>	<b>Apps im Detail</b>	<b>993</b>
20.1	Ein Windows Store App-Projekt im Detail	993
20.1.1	Contracts und Extensions	994
20.1.2	AssemblyInfo.cs	995
20.1.3	Verweise	996
20.1.4	App.xaml und App.xaml.cs	997
20.1.5	Package.appxmanifest	998
20.1.6	Application1_TemporaryKey.pfx	1002
20.1.7	MainPage.xaml & MainPage.xaml.es	1002
20.1.8	Datentyp-Konverter/Hilfsklassen	1003
20.1.9	StandardStyles.xaml	1005
20.1.10	Assets/Symbole	1006
20.1.11	Nach dem Kompilieren	1007
20.2	Der Lebenszyklus einer WinRT-App	1007
20.2.1	Möglichkeiten der Aktivierung von Apps	1009
20.2.2	Der Splash Screen	1011
20.2.3	Suspending	1011
20.2.4	Resuming	1012
20.2.5	Beenden von Apps	1013
20.2.6	Die Ausnahmen von der Regel	1014
20.2.7	Debuggen	1014

20.3	Daten speichern und laden .....	1018
20.3.1	Grundsätzliche Überlegungen .....	1018
20.3.2	Worauf und wie kann ich zugreifen? .....	1018
20.3.3	Das AppData-Verzeichnis .....	1018
20.3.4	Das Anwendungs-Installationsverzeichnis .....	1021
20.3.5	Das Downloads-Verzeichnis .....	1022
20.3.6	Sonstige Verzeichnisse .....	1023
20.3.7	Anwendungsdaten lokal sichern und laden .....	1023
20.3.8	Daten in der Cloud ablegen/laden (Roaming) .....	1025
20.3.9	Aufräumen .....	1028
20.3.10	Sensible Informationen speichern .....	1028
20.4	Praxisbeispiele .....	1030
20.4.1	Unterstützung für den Search-Contract bieten .....	1030
20.4.2	Die Auto-Play-Funktion unterstützen .....	1037
20.4.3	Einen zusätzlichen Splash Screen einsetzen .....	1041
20.4.4	Eine Dateiverknüpfung erstellen .....	1043
<b>21</b>	<b>WinRT-Techniken .....</b>	<b>1049</b>
21.1	Arbeiten mit Dateien/Verzeichnissen .....	1049
21.1.1	Verzeichnisinformationen auflisten .....	1049
21.1.2	Unterverzeichnisse auflisten .....	1052
21.1.3	Verzeichnisse erstellen/löschen .....	1053
21.1.4	Dateien auflisten .....	1055
21.1.5	Dateien erstellen/schreiben/lesen .....	1057
21.1.6	Dateien kopieren/umbenennen/löschen .....	1061
21.1.7	Verwenden der Dateipicker .....	1062
21.1.8	StorageFile-/StorageFolder-Objekte speichern .....	1069
21.1.9	Verwenden der Most Recently Used-Liste .....	1071
21.2	Datenaustausch zwischen Apps/Programmen .....	1072
21.2.1	Zwischenablage .....	1073
21.2.2	Teilen von Inhalten .....	1080
21.2.3	Eine App als Freigabeziel verwenden .....	1083
21.2.4	Zugriff auf die Kontaktliste .....	1084
21.3	Spezielle Oberflächenelemente .....	1086
21.3.1	MessageDialog .....	1086
21.3.2	Popup-Benachrichtigungen .....	1089
21.3.3	PopUp/Flyouts .....	1097

21.3.4	Das PopupMenu einsetzen .....	1101
21.3.5	Eine AppBar verwenden .....	1103
21.4	Datenbanken und Windows Store Apps .....	1108
21.4.1	Der Retter in der Not: SQLite! .....	1108
21.4.2	Verwendung/Kurzüberblick .....	1108
21.4.3	Installation .....	1110
21.4.4	Wie kommen wir zu einer neuen Datenbank? .....	1112
21.4.5	Wie werden die Daten manipuliert? .....	1116
21.5	Vertrieb der App .....	1118
21.5.1	Verpacken der App .....	1118
21.5.2	Windows App Certification Kit .....	1120
21.5.3	App-Installation per Skript .....	1122
21.6	Ein Blick auf die App-Schwachstellen .....	1123
21.6.1	Quellcodes im Installationsverzeichnis .....	1123
21.6.2	Zugriff auf den App-Datenordner .....	1125
21.7	Praxisbeispiele .....	1125
21.7.1	Ein Verzeichnis auf Änderungen überwachen .....	1125
21.7.2	Eine App als Freigabeziel verwenden .....	1128
21.7.3	ToastNotifications einfach erzeugen .....	1133

## Anhang

<b>A</b>	<b>Glossar .....</b>	<b>1141</b>
<b>B</b>	<b>Wichtige Dateiextensions .....</b>	<b>1147</b>
	<b>Index .....</b>	<b>1149</b>