

Inhaltsverzeichnis

	Einleitung	13
	Einführung	13
	An wen sich dieses Buch richtet	13
	Wegweiser für Ein- und Umsteiger	14
	Konventionen und Features in diesem Buch	15
	Konventionen	15
	Weitere Features	16
	Systemvoraussetzungen	16
	Die Beispieldateien installieren und verwenden	17
	Die Beispieldateien installieren	17
	Die Übungsdateien verwenden	17
	Korrekturen, Kommentare und Hilfe	21
	Hilfe zu diesem Buch und der zugehörigen CD	22
	Hilfe zum Produkt Visual Studio 2008	22
Teil A	Einführung in Microsoft Visual C# und Microsoft Visual Studio 2008	23
1	Willkommen bei C#	25
	Das erste Programm mit der Visual Studio 2008-Entwicklungs- umgebung erstellen	26
	Ihr erstes C#-Programm schreiben	31
	Namespaces verwenden	36
	Eine grafische Anwendung erstellen	40
	Schnellübersicht	51
2	Mit Variablen, Operatoren und Ausdrücken arbeiten	53
	Anweisungen verstehen	54
	Bezeichner verwenden	54
	Schlüsselwörter identifizieren	55
	Variablen verwenden	55
	Variablen benennen	56
	Variablen deklarieren	56
	Mit primitiven Datentypen arbeiten	57
	Werte primitiver Datentypen anzeigen	58
	Arithmetische Operatoren verwenden	63
	Operatoren und Typen	63
	Arithmetische Operatoren untersuchen	64
	Die Operatorrangfolge steuern	67

4 Inhaltsverzeichnis

	Assoziativität und die Auswertung von Ausdrücken	68
	Assoziativität und der Zuweisungsoperator	68
	Variablen inkrementieren und dekrementieren	69
	Implizit typisierte lokale Variablen deklarieren.	70
	Schnellübersicht.	72
3	Methoden schreiben und Gültigkeitsbereiche anwenden	73
	Methoden deklarieren	74
	Die Syntax zur Deklaration von Methoden	74
	Rückkehranweisungen schreiben	75
	Methoden aufrufen.	77
	Die Syntax zum Aufruf einer Methode spezifizieren	77
	Gültigkeitsbereiche anwenden	80
	Lokale Gültigkeitsbereiche definieren	80
	Gültigkeitsbereiche auf Klassenebene definieren.	81
	Methoden überladen	81
	Methoden schreiben.	82
	Schnellübersicht.	91
4	Entscheidungsanweisungen verwenden	93
	Boolesche Variablen deklarieren.	94
	Boolesche Operatoren verwenden.	94
	Gleichheit und relationale Operatoren verstehen	94
	Bedingte logische Operatoren verstehen	95
	Operatorvorrang und Assoziativität im Überblick	96
	Entscheidungen mit <i>if</i> -Anweisungen treffen.	97
	Die Syntax der <i>if</i> -Anweisung verstehen	97
	Anweisungen in Blöcken gruppieren	98
	<i>if</i> -Anweisungen verschachteln.	99
	<i>switch</i> -Anweisungen verwenden	104
	Die Syntax der <i>switch</i> -Anweisung verstehen.	105
	Regeln für die <i>switch</i> -Anweisung befolgen.	106
	Schnellübersicht.	110
5	Verbundzuweisungen und Schleifen verwenden	111
	Verbundzuweisungsoperatoren verwenden	112
	Schleifen mit <i>while</i> -Anweisungen schreiben.	113
	Schleifen mit <i>for</i> -Anweisungen schreiben	117
	Den Gültigkeitsbereich von <i>for</i> -Anweisungen verstehen	119
	Schleifen mit <i>do</i> -Anweisungen schreiben	119
	Schnellübersicht.	128
6	Fehler und Ausnahmen behandeln	129
	Fehler beherrschen	130
	Code probieren und Ausnahmen abfangen	130
	Eine Ausnahme behandeln	131
	Unbehandelte Ausnahmen	131

	Mehrere <i>catch</i> -Handler verwenden	132
	Mehrere Ausnahmen abfangen	133
	Ganzzahlarithmetik mit <i>checked</i> und <i>unchecked</i>	138
	Geprüfte Anweisungen schreiben	139
	Geprüfte Ausdrücke schreiben	139
	Ausnahmen auslösen	141
	Einen <i>finally</i> -Block verwenden	145
	Schnellübersicht	147
Teil B	Die Grundlagen der Sprache C# verstehen	149
7	Klassen und Objekte erstellen und verwalten	151
	Was ist Klassifizierung?	152
	Was ist Kapselung?	152
	Eine Klasse definieren und verwenden	153
	Den Zugriff kontrollieren	154
	Mit Konstruktoren arbeiten	156
	Konstruktoren überladen	157
	Statische Methoden und Daten verstehen	164
	Ein gemeinsam nutzbares Feld erstellen	165
	Ein statisches Feld mit dem Schlüsselwort <i>const</i> erstellen	166
	Schnellübersicht	171
8	Werte und Verweise verstehen	173
	Werttypvariablen und Klassen kopieren	174
	NULL-Werte und Typen, die NULL-Werte zulassen	179
	Auf NULL festlegbare Typen verwenden	180
	Die Eigenschaften von auf NULL festlegbaren Typen verstehen	180
	Parameter mit <i>ref</i> und <i>out</i> übergeben	181
	Parameter mit <i>ref</i> deklarieren	182
	Parameter mit <i>out</i> deklarieren	183
	Wie der Computerspeicher organisiert ist	185
	Stack und Heap verwenden	186
	Die Klasse <i>System.Object</i>	187
	Boxing	188
	Unboxing	188
	Daten sicher konvertieren	190
	Der Operator <i>is</i>	190
	Der Operator <i>as</i>	191
	Schnellübersicht	193
9	Werttypen mit Enumerationen und Strukturen erstellen	195
	Mit Enumerationen arbeiten	196
	Einen Enumerationstyp deklarieren	196
	Eine Enumeration verwenden	196
	Literalwerte für eine Enumeration wählen	197
	Den zugrunde liegenden Typ für eine Enumeration festlegen	198

	Mit Strukturen arbeiten	201
	Eine Struktur deklarieren	202
	Die Unterschiede zwischen Strukturen und Klassen verstehen	203
	Strukturvariablen deklarieren	205
	Strukturen initialisieren	205
	Strukturvariablen kopieren	207
	Schnellübersicht	211
10	Arrays und Auflistungen verwenden	213
	Was ist ein Array?	214
	Arrayvariablen deklarieren	214
	Arrayinstanzen erzeugen	214
	Arrayvariablen initialisieren	215
	Ein implizit typisiertes Array erstellen	216
	Auf einzelne Elemente des Arrays zugreifen	217
	Ein Array durchlaufen	218
	Arrays kopieren	219
	Was sind Auflistungsklassen?	220
	Die Auflistungsklasse <i>ArrayList</i>	222
	Die Auflistungsklasse <i>Queue</i>	224
	Die Auflistungsklasse <i>Stack</i>	225
	Die Auflistungsklasse <i>Hashtable</i>	226
	Die Klasse <i>SortedList</i>	227
	Initialisierer für Auflistungen verwenden	228
	Arrays und Auflistungen im Vergleich	228
	Mit den Auflistungsklassen Karten spielen	229
	Schnellübersicht	235
11	Parameterarrays verstehen	237
	Arrayargumente verwenden	238
	Parameterarrays deklarieren	239
	Parameterarrays vom Typ <i>object</i> verwenden	241
	Parameterarrays verwenden	242
	Schnellübersicht	245
12	Vererbung richtig einsetzen	247
	Was ist Vererbung?	248
	Vererbung einsetzen	248
	Basisklassen und abgeleitete Klassen	249
	Die Konstruktoren der Basisklasse aufrufen	250
	Klassen zuweisen	251
	Methoden und das Schlüsselwort <i>new</i>	253
	Virtuelle Methoden deklarieren	254
	Überschreibungsmethoden	255
	Geschützter Zugriff	258
	Erweiterungsmethoden verstehen	264
	Schnellübersicht	268

13	Schnittstellen erstellen und abstrakte Klassen deklarieren	269
	Schnittstellen verstehen	270
	Schnittstellensyntax	271
	Schnittstelleneinschränkungen	271
	Eine Schnittstelle implementieren	272
	Auf eine Klasse über ihre Schnittstelle verweisen	273
	Mit mehreren Schnittstellen arbeiten	274
	Abstrakte Klassen	274
	Abstrakte Methoden	276
	Versiegelte Klassen	276
	Versiegelte Methoden	277
	Ein erweiterbares Framework implementieren	277
	Schlüsselwortkombinationen im Überblick	286
	Schnellübersicht	287
14	Garbage Collection und Ressourcenverwaltung einsetzen	289
	Der Lebenszyklus von Objekten	290
	Destruktoren erstellen	291
	Warum wird der Garbage Collector verwendet?	292
	Wie funktioniert der Garbage Collector?	294
	Empfehlungen	294
	Ressourcenverwaltung	295
	Freigabemethoden	295
	Ausnahmesichere Freigabe von Ressourcen	295
	Die <i>using</i> -Anweisung	296
	Die <i>Dispose</i> -Methode aus einem Destruktor aufrufen	298
	Den Programmcode ausnahmesicher machen	299
	Schnellübersicht	302
Teil C	Komponenten erstellen	305
15	Eigenschaften implementieren, um auf Attribute zuzugreifen	307
	Kapselung mithilfe von Methoden implementieren	308
	Was sind Eigenschaften?	310
	Eigenschaften verwenden	311
	Schreibgeschützte Eigenschaften	312
	Lesegeschützte Eigenschaften	312
	Zugriffsmodifizierer bei Eigenschaften	313
	Einschränkungen von Eigenschaften verstehen	314
	Eigenschaften richtig verwenden	315
	Schnittstelleneigenschaften deklarieren	316
	Eigenschaften in einer Windows-Anwendung einsetzen	317
	Automatische Eigenschaften generieren	319
	Objekte mithilfe von Eigenschaften initialisieren	320
	Schnellübersicht	324

16	Indexer verwenden	327
	Was ist ein Indexer?	327
	Ein Beispiel, das keine Indexer verwendet	328
	Das gleiche Beispiel mit Indexern	329
	Accessoren für Indexer verstehen	331
	Indexer und Arrays im Vergleich	332
	Indexer in Schnittstellen	334
	Indexer in einer Windows-Anwendung einsetzen	335
	Schnellübersicht	341
17	Delegaten und Ereignisse	343
	Delegaten deklarieren und verwenden	344
	Szenario: Automatisierte Fabrik	344
	Die Fabrik ohne Delegaten implementieren	345
	Die Fabrik mit einem Delegaten implementieren	345
	Delegaten verwenden	348
	Lambda-Ausdrücke und Delegaten	351
	Einen Methodenadapter erstellen	351
	Einen Lambda-Ausdruck als Adapter verwenden	352
	Die Form von Lambda-Ausdrücken	353
	Benachrichtigungen mit Ereignissen realisieren	355
	Ein Ereignis deklarieren	355
	Ein Ereignis abonnieren	356
	Das Abonnement eines Ereignisses kündigen	357
	Ein Ereignis auslösen	357
	Ereignisse der WPF-Benutzeroberfläche verstehen	357
	Ereignisse verwenden	359
	Schnellübersicht	362
18	Einführung in Generics	365
	Das Problem mit <i>object</i> -Typen	365
	Die Generics-Lösung	367
	Generics und verallgemeinerte Klassen	369
	Generics und Einschränkungen	370
	Eine generische Klasse erstellen	370
	Die Theorie der binären Bäume	370
	Eine Binärbaum-Klasse mit Generics erstellen	373
	Eine generische Methode erstellen	382
	Eine generische Methode für einen binären Baum	383
	Schnellübersicht	386
19	Auflistungen enumerieren	387
	Die Elemente in einer Auflistung enumerieren	387
	Einen Enumerator manuell implementieren	389
	Die Schnittstelle <i>IEnumerable</i> implementieren	393

	Einen Enumerator mit einem Iterator implementieren	395
	Ein einfacher Iterator	396
	Einen Enumerator für die Klasse <i>Tree<TItem></i> mit einem Iterator definieren	397
	Schnellübersicht	400
20	Speicherinterne Daten mit Abfrageausdrücken abrufen	401
	Was ist LINQ?	402
	LINQ in einer C#-Anwendung einsetzen	402
	Daten auswählen	404
	Daten filtern	406
	Daten sortieren, gruppieren und zusammenfassen	407
	Daten verknüpfen	409
	Abfrageoperatoren verwenden	411
	Daten in <i>Tree<TItem></i> -Objekten abfragen	413
	LINQ und verzögerte Auswertung	418
	Schnellübersicht	421
21	Überladen von Operatoren	423
	Operatoren verstehen	424
	Operatoreinschränkungen	424
	Überladene Operatoren	425
	Symmetrische Operatoren erstellen	426
	Verbundzuweisungen verstehen	428
	Inkrement- und Dekrementoperatoren deklarieren	429
	Operatorpaare definieren	431
	Einen Operator implementieren	432
	Konvertierungsoperatoren verstehen	434
	Vordefinierte Konvertierungen bereitstellen	434
	Benutzerdefinierte Konvertierungsoperatoren implementieren	435
	Noch einmal: Symmetrische Operatoren	436
	Einen impliziten Konvertierungsoperator hinzufügen	437
	Schnellübersicht	439
Teil D	Mit Windows-Anwendungen arbeiten	441
22	Einführung in Windows Presentation Foundation	443
	Eine WPF-Anwendung erstellen	444
	Eine WPF-Anwendung erstellen	444
	Steuerelemente in das Formular aufnehmen	459
	WPF-Steuerelemente verwenden	459
	Eigenschaften dynamisch ändern	468
	Ereignisse in einem WPF-Formular behandeln	473
	Ereignisse in Windows Forms verarbeiten	473
	Schnellübersicht	479

10 Inhaltsverzeichnis

23	Mit Menüs und Dialogfeldern arbeiten	481
	Richtlinien für Menüs	482
	Menüs und Menüereignisse	482
	Ein Menü erstellen.....	483
	Menüereignisse behandeln	488
	Kontextmenüs	494
	Kontextmenüs erstellen	495
	Windows-Standarddialogfelder	499
	Die Klasse <i>SaveFileDialog</i> verwenden.....	499
	Schnellübersicht.....	503
24	Gültigkeitsprüfung durchführen	505
	Daten auf Gültigkeit prüfen.....	505
	Strategien für die Gültigkeitsprüfung von Benutzereingaben	506
	Ein Beispiel – Kundendaten verwalten.....	506
	Gültigkeitsprüfung mithilfe von Datenbindung durchführen	507
	Den Zeitpunkt für die Gültigkeitsprüfung ändern	524
	Schnellübersicht.....	529
Teil E	Daten verwalten	531
25	Eine Datenbank verwenden	533
	Eine Datenbank mit ADO.NET abfragen	534
	Die <i>Northwind</i> -Datenbank.....	534
	Die Datenbank erstellen	535
	Bestellinformationen mit ADO.NET abfragen.....	537
	Eine Datenbank mit DLINQ abfragen	546
	Eine Entitätsklasse definieren.....	546
	Eine DLINQ-Abfrage erstellen und ausführen	549
	Verzögertes und sofortiges Abrufen	551
	Tabellen verknüpfen und Beziehungen erstellen	551
	Noch einmal: Verzögertes und sofortiges Abrufen	555
	Eine benutzerdefinierte <i>DataContext</i> -Klasse definieren	556
	Bestellinformationen mit DLINQ abfragen	557
	Schnellübersicht.....	562
26	Daten mit Datenbindung anzeigen und bearbeiten	563
	Datenbindung mit DLINQ verwenden.....	563
	Daten mit DLINQ modifizieren	579
	Vorhandene Daten aktualisieren.....	579
	Kollidierende Aktualisierungen behandeln	580
	Daten hinzufügen und löschen.....	582
	Schnellübersicht.....	591

Teil F	Webanwendungen erstellen	593
27	Einführung in ASP.NET	595
	Das Internet als Infrastruktur verstehen	596
	Webserver – Anforderungen und Antworten verstehen	596
	Den Status verwalten	597
	ASP.NET verstehen	598
	Webanwendungen mit ASP.NET erstellen	600
	Eine ASP.NET-Anwendung erstellen	600
	Serversteuerelemente verstehen	612
	Ein Design erstellen und verwenden	620
	Schnellübersicht	624
28	Überprüfungssteuerelemente von Web Forms verstehen	625
	Überprüfung auf dem Server und auf dem Client im Vergleich	626
	Daten auf dem Webserver überprüfen	626
	Daten im Webbrowser überprüfen	626
	Clientseitige Überprüfung implementieren	628
	Schnellübersicht	636
29	Eine Website sichern und mit Web Forms auf Daten zugreifen	637
	Sicherheit verwalten	638
	Formularauthentifizierung verstehen	638
	Formularauthentifizierung implementieren	639
	Daten abfragen und anzeigen	646
	Das Web Forms- <i>GridView</i> -Steuerelement verstehen	646
	Kundendaten und Bestellverlauf anzeigen	647
	Daten in Seiten anzeigen	652
	Daten bearbeiten	654
	Zeilen über ein <i>GridView</i> -Steuerelement aktualisieren	654
	Zwischen Formularen navigieren	656
	Schnellübersicht	664
30	Einen Webdienst erstellen und verwenden	665
	Was ist ein Webdienst?	666
	Die Rolle von SOAP	666
	Was ist die Web Services Description Language?	667
	Nicht funktionale Anforderungen von Webdiensten	668
	Die Rolle von Windows Communication Foundation	669
	Einen Webdienst erstellen	670
	Den <i>ProductsService</i> -Webdienst erstellen	670
	Webdienste, Clients und Proxys	680
	SOAP sprechen: Der schwierige Weg	680
	SOAP sprechen: Der einfache Weg	680
	Den Webdienst <i>ProductService</i> nutzen	681
	Schnellübersicht	687

Danksagung	689
Stichwortverzeichnis	691
Über den Autor	715