

Inhaltsverzeichnis

	Einführung	21
	Systemvoraussetzungen	21
	So finden Sie den besten Ausgangspunkt in diesem Buch	22
	Korrekturen, Kommentare und Hilfe	24
	Die Begleit-CD installieren und verwenden	25
	Konventionen und Features	34
	Danksagung	35
Teil A	Einführung in Microsoft Visual C# und Visual Studio .NET	37
Kapitel 1	Willkommen bei C#	39
	Das erste Programm mit der Entwicklungsumgebung von Visual Studio .NET erstellen	39
	Eine Konsolenanwendung in Visual Studio .NET erstellen	40
	Schreiben Sie Ihr erstes C#-Programm	43
	Code schreiben mit Hilfe der IntelliSense-Technologie	44
	Die Konsolenanwendung erstellen und ausführen	46
	Namespaces verwenden	49
	Ausgeschriebene Namen ausprobieren	51
	Mit Hilfe von XML und Kommentaren die Dokumentation erstellen	52
	XML-Kommentare in den Programmcode aufnehmen	53
	Eine Windows Forms-Anwendung erstellen	58
	Ein Windows-Projekt in Visual Studio .NET erstellen	58
	Die Benutzerschnittstelle erstellen	59
	Eigenschaften von Steuerelementen verändern	63
	Die Größe des Formulars ändern	65
	Den Code für die OK-Schaltfläche schreiben	66
	Das Windows-Programm starten	66
	Kapitel 1 – Schnellübersicht	68
Kapitel 2	Variablen, Operatoren und Ausdrücke verwenden	69
	Anweisungen verstehen	69
	Bezeichner verwenden	70

	• Schlüsselwörter	70
	• Variablen verwenden	71
	• Variablen benennen	71
	• Variablen deklarieren	72
	• Einfache Datentypen verwenden	73
	• Einfache Datentypen anzeigen	74
	• Einfache Datentypen im Programmcode verwenden	76
	• Arithmetische Operatoren verwenden	78
	• Einsatzmöglichkeiten der Operatoren	79
	• Arithmetische Operatoren in der Praxis	80
	• Der Vorrang der verschiedenen Operatoren	83
	• Assoziativität und die Auswertung von Ausdrücken	84
	• Kapitel 2 – Schnellübersicht	86
Kapitel 3	• Methoden erstellen und Geltungsbereiche verwenden	87
	• Methoden deklarieren	87
	• Die Syntax zur Deklaration von Methoden	88
	• return-Anweisungen erstellen	89
	• Methodendefinitionen ansehen	90
	• Methoden aufrufen	92
	• Die Syntax zum Aufruf einer Methode	92
	• Methodenaufrufe untersuchen	93
	• Gültigkeit von Methoden verstehen	94
	• Lokale Gültigkeit innerhalb einer Methode	94
	• Gültigkeit auf Klassenebene verwenden	95
	• Bezeichner überladen	96
	• Methoden erstellen und aufrufen	97
	• Die weiteren Methoden erstellen	100
	• Die Methoden aufrufen und das Programm starten	102
	• Im Einzelschritt durch die Methoden mit dem Debugger von Visual Studio .NET	103
	• Kapitel 3 – Schnellübersicht	106
Kapitel 4	• Auswahlanweisungen verwenden	107
	• Boolesche Variablen deklarieren	107
	• Boolesche Operatoren verwenden	108
	• Relationale Operatoren verstehen	108
	• Logische Operatoren verstehen	109
	• Zusammenfassung des Vorrangs und der Assoziativität der Operatoren	110
	• if-Anweisungen verwenden	111
	• Die Syntax der if-Anweisung verstehen	111
	• Anweisungen in Blöcken gruppieren	112
	• if-Anweisungen verschachteln	113
	• if-Anweisungen programmieren	114

	••• switch-Anweisungen verwenden	117
	••• Die Syntax der switch-Anweisung verstehen	118
	••• Regeln für den Einsatz der switch-Anweisung	119
	••• switch-Anweisungen programmieren	121
	••• Kapitel 4 – Schnellübersicht	123
Kapitel 5	Wiederholungen mit Schleifen programmieren	125
	••• Verbundzuweisungsoperatoren verwenden	125
	••• Variablen in- und dekrementieren	127
	••• while-Anweisungen erstellen	128
	••• for-Anweisungen erstellen	130
	••• Geltungsbereich und die for-Anweisung	131
	••• do-Anweisungen erstellen	132
	••• Anzeigen von Schleifen-Anweisungen	133
	••• Eine while-Schleife programmieren	135
	••• Eine Schleife mit do programmieren	139
	••• Im Einzelschrittmodus durch die do-Schleife	143
	••• Kapitel 5 – Schnellübersicht	148
Kapitel 6	Fehler und Ausnahmen behandeln	149
	••• Mit Fehlern fertig werden	149
	••• Versuchen, ob's klappt; Fehler behandeln, wenn nicht	151
	••• Mehrere catch-Handler verwenden	152
	••• Einen allgemeinen catch-Handler erstellen	153
	••• Eine try/catch-Anweisung erstellen	154
	••• Ganzzahlberechnungen mit checked und unchecked	157
	••• checked und unchecked in Anweisungen	158
	••• checked und unchecked in Ausdrücken	159
	••• checked in der Praxis	159
	••• Ausnahmen erzeugen	163
	••• Eigene Ausnahmen erzeugen	164
	••• Behandeln Sie die eigene Ausnahme	167
	••• Einen finally-Block erstellen	169
	••• Kapitel 6 – Schnellübersicht	172
Teil B	Die Grundlagen der Programmiersprache C# verstehen	173
Kapitel 7	Objekte und Klassen erzeugen und verwalten	175
	••• Was ist Klassifizierung?	175
	••• Was ist Kapselung?	176
	••• Den Zugriff kontrollieren	177
	••• Konstruktoren und das Schlüsselwort new	179
	••• Konstruktoren überladen	181

	••• Konstruktoren schreiben und Objekte erzeugen	182
	••• Instanzmethoden erstellen und aufrufen	187
	••• Statische Methoden und Daten verstehen	190
	••• Ein gemeinsam benutzbares Feld erstellen	191
	••• Ein statisches Feld mit dem Schlüsselwort const erstellen	192
	••• Statische Member erstellen und statische Methoden aufrufen	193
	••• Kapitel 7 – Schnellübersicht	196
Kapitel 8	••• Werte und Referenzen verstehen	199
	••• Variablen des Typs int und Klassen kopieren	199
	••• Wert- und Verweisparameter verwenden	201
	••• Parameterübergabe mit ref und out	204
	••• Parameter mit ref deklarieren	205
	••• Parameter mit out deklarieren	206
	••• ref-Parameter verwenden	207
	••• Was sind der Stack und der Heap?	208
	••• Den Stack und den Heap verwenden	208
	••• Was verbirgt sich hinter System.Object?	210
	••• Boxing	211
	••• Unboxing	212
	••• Kapitel 8 – Schnellübersicht	215
Kapitel 9	••• Werttypen mit Enumerationen und ••• Strukturen erstellen	217
	••• Enumerationen verwenden	217
	••• Einen Enumerationstypen deklarieren	218
	••• Die Literalwerte für eine Enumeration wählen	218
	••• Den zugrunde liegenden Typ für eine Enumeration festlegen	219
	••• Eine Enumeration verwenden	219
	••• Enumerationstypen erstellen und verwenden	221
	••• Mit Strukturtypen arbeiten	222
	••• Die Struktur deklarieren	223
	••• Die Unterschiede zwischen struct und Class verstehen	223
	••• Strukturvariablen deklarieren	226
	••• Strukturen und die definitive Zuweisung	227
	••• Die Konstruktoren einer Struktur aufrufen	227
	••• Strukturvariablen kopieren	229
	••• Die Typäquivalenz verstehen	230
	••• Einen Strukturtypen erstellen und verwenden	232
	••• Kapitel 9 – Schnellübersicht	236
Kapitel 10	••• Arrays und Auflistungen verwenden	237
	••• Was ist ein Array?	237
	••• Arrayvariablen deklarieren	237

	••• Arrayinstanzen erzeugen	238
	••• Arrayvariablen initialisieren	239
	••• Auf einzelne Elemente des Arrays zugreifen	240
	••• Mit einer Schleife auf das Array zugreifen	241
	••• Arrays kopieren	243
	••• Die Klasse System.Array verwenden	245
	••• Was sind Klassen vom Typ Auflistung (collections)?	246
	••• Die Klasse ArrayList	248
	••• Die Klasse Queue	250
	••• Die Klasse Stack	251
	••• Die Klasse SortedList	252
	••• Arrays und Auflistungen im Vergleich	254
	••• Mit den Auflistungsklassen Karten spielen	254
	••• Die Karten geben	254
	••• Die Karten mischen	256
	••• Die Karten auf den Talon zurücklegen	258
	••• Kapitel 10 – Schnellübersicht	260
Kapitel 11	••• Parameterarrays verstehen	263
	••• Überladene Methoden erstellen	263
	••• Arrayargumente verwenden	264
	••• Parameterarrays deklarieren	265
	••• Parameterarrays vom Typ objekt verwenden	269
	••• Parameterarrays verwenden	271
	••• Eine Methode erstellen, die ein Parameterarray besitzt	271
	••• Eine Methode mit einem Parameterarray aufrufen	274
	••• Die Methode Main verwenden	275
	••• Kapitel 11 – Schnellübersicht	278
Kapitel 12	••• Vererbung richtig einsetzen	279
	••• Was ist Vererbung?	279
	••• Die grundlegende Syntax verstehen	280
	••• Basisklasse und abgeleitete Klassen	280
	••• Die Konstruktoren der Basisklasse aufrufen	281
	••• Methoden und das Schlüsselwort new	282
	••• Polymorphie – Virtuelle Methoden	284
	••• Überschreibungsmethoden	285
	••• Den Zugriff mit protected verändern	286
	••• Schnittstellen definieren	287
	••• Syntax	288
	••• Einschränkungen	288
	••• Eine Schnittstelle implementieren	289
	••• Abstrakte Klassen	291
	••• Abstrakte Methoden	294
	••• Versiegelte Klassen	295

	Versiegelte Methoden	295
	Eine Vererbungshierarchie erweitern	296
	Machen Sie sich mit der Vererbungshierarchie vertraut	296
	Die Klasse ColorSyntaxVisitor erstellen	299
	Mehrfachvererbung	302
	Syntax	303
	Explizite Schnittstellenimplementierung	303
	Die verschiedenen Schlüsselwort-Kombinationen im Überblick	305
	Kapitel 12 – Schnellübersicht	307
Kapitel 13	Den Garbage Collector und die Ressourcenverwaltung einsetzen	309
	Garbage Collection	309
	Werte und Objekte vergleichen	309
	Der Lebenszyklus von Objekten	310
	Destruktoren erstellen	311
	Warum wird der Garbage Collector verwendet?	312
	Wie funktioniert der Garbage Collector?	314
	Empfehlungen	314
	Ressourcenverwaltung	315
	Das Dispose-Entwurfsmuster	315
	Ausnahmesichere Bereinigung von Ressourcen	316
	Die using-Anweisung	317
	Den Code an IDisposable anpassen	319
	Eine Bereinigungsmethode vom Destruktor aus aufrufen	320
	Den Programmcode ausnahmesicher machen	322
	Eine using-Anweisung erstellen	322
	Kapitel 13 – Schnellübersicht	326
Teil C	Komponenten erstellen	329
Kapitel 14	Eigenschaften implementieren, um auf Attribute zuzugreifen	331
	Felder und Methoden vergleichen	331
	Was sind Eigenschaften?	333
	get-Accessoren	334
	set-Accessoren	335
	Lese- und Schreibzugriff in einem Ausdruck	335
	Schreibgeschützte Eigenschaften	335
	Lesegeschützte Eigenschaften	336
	Einschränkungen von Eigenschaften verstehen	336
	Statische Eigenschaften verwenden	337
	Eigenschaften in Schnittstellen deklarieren	340
	Eigenschaften in Windows-Anwendungen verwenden	341

	Eigenschaften verwenden	342
	Kapitel 14 – Schnellübersicht	347
Kapitel 15	Indexer verwenden	349
	Was ist ein Indexer?	349
	Ein Beispiel, das keinen Indexer verwendet	349
	Das gleiche Beispiel mit Indexern	351
	get-Accessoren	353
	set-Accessoren	353
	Einen schreibgeschützten Indexer erstellen	354
	Einen lesegeschützten Indexer erstellen	355
	Indexer und Methoden im Vergleich	356
	Indexer und Arrays im Vergleich	356
	Indexer und Eigenschaften im Vergleich	357
	Indexer in Schnittstellen	357
	Indexer in einer Windows-Anwendung verwenden	358
	Machen Sie sich mit der Anwendung vertraut	358
	Die Indexer erstellen	359
	Die Indexer aufrufen	361
	Die Anwendung starten	362
	Kapitel 15 – Schnellübersicht	364
Kapitel 16	Delegaten und Ereignisse	367
	Delegatdeklarationen und Instanzen verwenden	367
	Ein Beispiel, das keinen Delegat verwendet	368
	Das gleiche Beispiel mit einem Delegat	370
	Einen Delegat deklarieren	370
	Einen Delegaten aufrufen	371
	Eine Delegatinstanz erzeugen	373
	Einen Delegaten verwenden	375
	Benachrichtigungen mit Ereignissen realisieren	378
	Ein Ereignis deklarieren	378
	Ein Ereignis abonnieren	379
	Das Abonnement eines Ereignisses kündigen	379
	Ein Ereignis aufrufen	380
	GUI-Ereignisse verstehen	381
	Ereignisse verwenden	383
	Die Anwendung mit der digitalen Uhr überarbeiten	383
	Kapitel 16 – Schnellübersicht	386
Kapitel 17	Überladung von Operatoren	389
	Mit Operatoren arbeiten	389
	Operatoren verstehen	390
	Einschränkungen bei der Implementierung von Operatoren	390

••• Überladene Operatoren	391
••• Symmetrische Operatoren erzeugen	393
••• Zusammengesetzte Zuweisungen verstehen	395
••• Inkrement- und Dekrementoperator deklarieren	395
••• Operatoren paarweise verwenden	397
••• Einen Operator implementieren	397
••• Die Überladungen für den Operator + erstellen	398
••• Operatoren zur Konvertierung deklarieren	401
••• Die eingebauten Konvertierungen	401
••• Benutzerdefinierte Konvertierungsoperatoren deklarieren	402
••• Symmetrische Operatoren erzeugen, die 2te	403
••• Einen impliziten Konvertierungsoperator ergänzen	404
••• Den Konvertierungsoperator programmieren	404
••• Kapitel 17 – Schnellübersicht	406

Teil D

••• **Windows-Anwendungen erstellen** **407**

Kapitel 18

••• Einführung in Windows Forms	409
••• Die Anwendung erstellen	410
••• Eine Windows Forms-Anwendung erstellen	410
••• Das Projekt Middleshire Bell Ringers Association erstellen	411
••• Die Eigenschaften des Formulars einstellen	412
••• Die Anwendung kompilieren	413
••• Die Eigenschaften eines Windows-Formulars	414
••• Eigenschaften im Programmcode setzen	416
••• Steuerelemente in das Formular aufnehmen	418
••• Windows Forms-Steuerelemente verwenden	419
••• Die Windows Forms-Steuerelemente in das Formular einfügen	420
••• Eigenschaften der Steuerelemente einstellen	421
••• Eigenschaften dynamisch verändern	424
••• Die Methode Reset erstellen	424
••• Die ComboBox füllen	425
••• Das aktuelle Datum einstellen	425
••• Die CheckBox initialisieren	425
••• Die Gruppe der runden Optionsfelder initialisieren	426
••• Die ListBox füllen	426
••• Die Methode Reset aufrufen	426
••• Die Anwendung kompilieren und testen	427
••• Ereignisse in Windows Forms	427
••• Ereignisse in Windows Forms bearbeiten	428
••• Das Click-Event der Schaltfläche Zurücksetzen bearbeiten	428
••• Das Click-Event der Schaltfläche Hinzufügen bearbeiten	429
••• Das Ereignis Closing des Formulars bearbeiten	430
••• Die Anwendung starten	431
••• Kapitel 18 – Schnellübersicht	433

Kapitel 19	Menüs erstellen und verwenden	435
	Richtlinien für Menüs und deren Stil	435
	Menüs erstellen und Menüereignisse bearbeiten	436
	Ein Menü erstellen	436
	Das Datei-Menü erstellen	437
	Eigenschaften für Menübefehle festlegen	439
	Das Menü testen	442
	Weitere Eigenschaften für Menübefehle	442
	Menüereignisse	443
	Ereignisse von Menübefehlen bearbeiten	444
	Testen Sie die Menüereignisse	445
	Popup-Menüs	446
	Popup-Menüs erstellen	446
	Die Popup-Menüs für den Vor- und Nachnamen erstellen	446
	Das Kontextmenü für das Formular erstellen	449
	Kapitel 19 – Schnellübersicht	452
Kapitel 20	Gültigkeitsprüfung durchführen	455
	Daten auf Gültigkeit überprüfen	455
	Die Eigenschaft CausesValidation	456
	Ereignisse bei der Gültigkeitsprüfung	456
	Ein Beispiel – Kundendaten warten	457
	Das Programm untersuchen	457
	Die Anwendung starten	460
	Seien Sie zurückhaltend!	461
	Die Stelle ändern, an der die Gültigkeitsprüfung stattfindet	462
	Die Anwendung erneut testen	464
	Das Steuerelement ErrorProvider verwenden	465
	Das Steuerelement ErrorProvider testen	467
	Kapitel 20 – Schnellübersicht	469
Kapitel 21	Komplexe Steuerelemente einsetzen	471
	Das Explorer-Interface	471
	Trennbalken, angedockte Steuerelemente und Panels	473
	Mit dem Splitter und angedockten Steuerelementen arbeiten	473
	Ein Panel verwenden	476
	Steuerelemente zum Bewegen in Daten	478
	Das TreeView-Steuerelement einsetzen	479
	Das ListView-Steuerelement einsetzen	489
	Kapitel 21 – Schnellübersicht	500

Kapitel 22	MDI, Fenster und Dialogfelder verwenden	501
	Was ist das Multiple Document Interface?	502
	Eine MDI-Anwendung erstellen	502
	Das MDI Hauptfenster erstellen	503
	Ein MDI Dokumentfenster erstellen	504
	Das MDI Dokumentfenster anzeigen	506
	Das Fenster-Menü vollenden	507
	Die Dokumentfenster schließen	509
	Alternativen zu MDI	509
	Dialogfelder erstellen	510
	Modale und nicht modale Dialogfelder anzeigen	510
	Ein Info-Dialogfeld erstellen	511
	Das Info-Dialogfeld anzeigen	513
	Standarddialogfelder verwenden	515
	Das Steuerelement SaveFileDialog verwenden	515
	Das Steuerelement SaveFileDialog einfügen	515
	Das Steuerelement SaveFileDialog nutzen	517
	Kapitel 22 – Schnellübersicht	519
Kapitel 23	GUI-Komponenten erstellen	521
	Benutzerdefinierte Steuerelemente	521
	Das Steuerelement Login erstellen	523
	Die Eigenschaften des Steuerelements einstellen	525
	Daten überprüfen und Ereignisse bereitstellen	528
	Das benutzerdefinierte Steuerelement Login verwenden	530
	Die Testanwendung erstellen	531
	Steuerelemente durch Subclassing anpassen	534
	Das Steuerelement ToggleButton erstellen	534
	Die Eigenschaften des Steuerelements ToggleButton definieren	536
	Das Click-Ereignis überschreiben	538
	Das Steuerelement ToggleButton verwenden	539
	Das Steuerelement in die Toolbox aufnehmen	539
	Das Steuerelement ToggleButton testen	539
	Kapitel 23 – Schnellübersicht	542
Kapitel 24	Forms erstellen und Vererbung verwenden	543
	Forms als Vorlagen	543
	Eine Subklasse eines Formulars erstellen	544
	Die Formularklasse Standard erstellen	545
	Die Eigenschaften der Formular-Klasse Standard festlegen	546
	Die Ereignisse Load und Closing erweitern	546
	Die Formular-Klasse Standard verwenden	551
	Starten Sie die Test-Anwendung	556
	Kapitel 24 – Schnellübersicht	557

Teil E	Daten verwalten	559
Kapitel 25	Datenbanken – früher und heute	561
	Was ist denn eine Datenbank?	561
	Structured Query Language (SQL)	565
	Einen Standard schaffen	567
	Universellen Datenzugriff mit ADO implementieren	568
Kapitel 26	Eine Datenbank verwenden	571
	ADO.NET-Datenbanken verwenden	571
	Die Northwind-Datenbank verwenden	572
	Die Beispieldatenbank installieren	572
	Zugriff auf die Datenbank	573
	Zur Datenbank verbinden	573
	Produktinformationen ansehen	576
	Produktinformationen in einer Windows	
	Forms-Anwendung anzeigen	577
	Code ergänzen, um Änderungen zu speichern	581
	Mit ADO.NET programmieren	585
	Verbindung zur Datenbank herstellen	585
	Eine Abfrage an die Tabelle Orders schicken	587
	Daten abrufen und Bestellungen anzeigen	588
	Die Datenbankverbindung trennen	590
	Null-Werte in Datenbanken behandeln	592
	Kapitel 26 – Schnellübersicht	594
Kapitel 27	Datenbindung und DataSets verwenden	597
	Windows Formular-Steuerelemente und Datenbindung	598
	Einfache Datenbindung verwenden	598
	Datenbindung an die Spalte eines DataSets	600
	Komplexe Datenbindung verwenden	603
	Das Formular erstellen	603
	Eine Datenquelle erstellen und konfigurieren	604
	Die ListBox an das DataSet binden	606
	Nicht verbundene DataSets	607
	Ein nicht verbundenes DataSet erstellen	608
	Die Tabelle Territories untersuchen	608
	Die Anwendung erstellen	608
	Das DataSet erzeugen und die Datenbindung vornehmen	609
	Daten abrufen und Verbindung trennen	611
	Updates bei einem verbindungslosen DataSet handhaben	613
	Die Änderungen validieren	613
	Die Datenbank aktualisieren	616
	Kapitel 27 – Schnellübersicht	619

Kapitel 28	XML verwenden	621
	Warum XML?	621
	Die Ziele von XML	622
	Die Struktur von XML	622
	XML-Schemata	623
	XML als Transportformat und Protokoll	625
	XML APIs und das .NET Framework	625
	Die Timesheet-Anwendung	626
	Ein XML-Schema erzeugen	626
	Das XML-Schema testen	631
	Die Anwendung erstellen, mit der die Aktivitäten aufgezeichnet werden	633
	Das Formular untersuchen	633
	Die Daten in einer XML-Datei speichern	634
	Die Anwendung testen	637
	Das Programm zur Analyse der XML-Daten erstellen	637
	Die XML-Daten validieren	637
	Eine Zusammenfassung der XML-Daten erstellen	640
	Kapitel 28 – Schnellübersicht	643
Kapitel 29	Systemdaten verwalten	645
	Den Server-Explorer verwenden	645
	Mit einem Server verbinden	646
	Den Server untersuchen	648
	Auf Systemdaten zugreifen	653
	Das Windows Ereignisprotokoll verwenden	653
	Einträge im Anwendungsprotokoll erstellen	653
	Das Ereignisprotokoll Anwendung überwachen	657
	Leistungsindikatoren überwachen	659
	Die Anzahl der Ausnahmen in der gemeinsamen Laufzeitschicht überwachen	659
	Die Anzahl der Ausnahmen als Rate ermitteln	663
	Dienste verwalten	666
	Den Status des ASP.NET-Statusdienstes untersuchen	667
	Den ASP.NET-Statusdienst starten und beenden	669
	Kapitel 29 – Schnellübersicht	672
Teil F	Webanwendungen erstellen	675
Kapitel 30	Einführung in ASP.NET	677
	Das Internet als Infrastruktur begreifen	678
	Anfragen an und Antworten von Webservern verstehen	678
	Den Status verwalten	679
	ASP.NET verstehen	680

	Webanwendungen mit ASP.NET erstellen	682
	Eine ASP.NET-Anwendung erstellen	682
	Die Webanwendung erzeugen	683
	Das Layout für das Web Formular festlegen	683
	Das Web Formular testen	692
	Serversteuerelemente verstehen	692
	Ein Serversteuerelement untersuchen	693
	Eventhandler programmieren	694
	Das Web Formular erneut testen	697
	Kapitel 30 – Schnellübersicht	702
Kapitel 31	Überprüfungssteuerelemente verwenden	703
	Die Prüfung auf dem Client und auf dem Server im Vergleich	703
	Überprüfung auf dem Server	704
	Überprüfung auf dem Client	704
	Clientseitige Überprüfung implementieren	705
	Die RequiredFieldValidator-Steuerelemente hinzufügen	706
	Das RangeValidator-Steuerelement verwenden	708
	Das ValidationSummary-Steuerelement verwenden	710
	Kapitel 31 – Schnellübersicht	714
Kapitel 32	Mit Web-Formularen auf Daten zugreifen	715
	Das Web Forms DataGrid-Steuerelement verwenden	715
	Sicherheitsfeatures implementieren	716
	Forms-basierte Sicherheit verstehen	717
	Forms-basierte Sicherheit implementieren	717
	Daten abfragen	722
	Kundendaten anzeigen	723
	Das DataSet füllen und die Daten anzeigen	726
	Daten auf Anforderung abrufen	728
	Daten bearbeiten	731
	Zeilen löschen	732
	Zeilen aktualisieren	735
	Kapitel 32 – Schnellübersicht	740
Kapitel 33	ASP.NET-Webanwendungen erstellen	743
	Zusätzliche Möglichkeiten des DataGrid-Steuerelements	743
	Die Datentabelle sortieren	743
	Das DataGrid-Steuerelement anbinden und die Daten darstellen	745
	Die Sortierungsfunktionalität hinzufügen	746
	Spaltenvorlagen verwenden	747
	Erzeugen der Anwendung ProductInfo	748
	Das DataGrid anbinden	749

	Die Inhalte der Spalte Discontinued als Kontrollkästchen darstellen	750
	Die Spalten des Steuerelements reorganisieren	752
	Zwischen Forms navigieren	754
	Schaltflächen und Hyperlinks verwenden	755
	Erzeugen der Anwendung ProductCategories	755
	Die Navigationsschaltfläche hinzufügen	757
	Das Web Formular ProductForm.aspx anlegen	759
	Wenn Sie mit dem nächsten Kapitel weitermachen möchten	763
	Wenn Sie Visual Studio .NET nun beenden möchten	763
	Kapitel 33 – Schnellübersicht	764
Kapitel 34	Einen XML-Webdienst erstellen	765
	Was ist ein XML-Webdienst?	765
	Die Rolle von SOAP	767
	Die Web Services Description Language	769
	Den Webdienst ProductsService erstellen	770
	Den Webdienst erzeugen	771
	Die Webmethode HowMuchWillItCost definieren	772
	Die Webmethode testen	774
	Umgang mit komplexen Daten	776
	Die Klasse Product definieren	777
	Die Webmethode GetProductInfo definieren	780
	Kapitel 34 – Schnellübersicht	783
Kapitel 35	Einen Webdienst verwenden	785
	Webdienste, Clients und Proxies	785
	SOAP einsetzen: Der schwierige Weg	786
	SOAP einsetzen: Der einfache Weg	786
	Den Webdienst ProductsService verwenden	787
	Einen Verweis auf den Webdienst einfügen	788
	Eine Webmethode aufrufen	790
	Die Anwendung testen	792
	Eine Webmethode asynchron aufrufen	793
	Eine weitere Client-Anwendung für den Webdienst erstellen	794
	Eine Webmethode asynchron aufrufen	795
	Kapitel 35 – Schnellübersicht	798
	Stichwortverzeichnis	799
	Die Autoren	825