

Frank Eller

Visual C# 2005

Grundlagen, Programmier Techniken, Datenbanken



ADDISON-WESLEY

An imprint of Pearson Education

München • Boston • San Francisco • Harlow, England
Don Mills, Ontario • Sydney • Mexico City
Madrid • Amsterdam

Inhalt

	Vorwort	19
I	Einführung	23
I	Das .NET Framework	25
1.1	Warum .NET?	25
1.1.1	Die .NET-Strategie	25
1.1.2	.NET und Java	26
1.2	Vorteile von .NET	28
1.3	Der Aufbau des .NET Frameworks	31
1.3.1	Übersicht	32
1.3.2	Die Common Language Runtime	32
1.3.3	Die Klassenbibliothek	34
1.3.4	Die Benutzerschnittstelle	35
1.3.5	Common Language Specification und Common Type System	36
1.3.6	Der Intermediate Language Code (IL-Code)	37
1.3.7	Der Global Assembly Cache (GAC)	38
1.3.8	Strong Names	39
1.4	Neuerungen in .NET 2.0	40
1.4.1	Neues im Compiler	40
1.4.2	Neues in C# bzw. dem .NET Framework	41
1.4.3	Neuerungen in Windows.Forms	42
2	Erste Schritte	45
2.1	Hello World (Konsole)	45
2.1.1	Das erste Programm	45
2.1.2	Erweiterung des Programms	49
2.2	Hello World (Windows-Version)	52
2.2.1	Projektauswahl	52
2.2.2	Entwurf der Oberfläche	54
2.2.3	Einfügen von Code	56
2.2.4	Quelltext-Dateien	58
3	Das Visual Studio 2005	61
3.1	Einführung	61
3.1.1	Übersicht	62
3.1.2	Systemvoraussetzungen und Versionen	63
3.2	Wichtige Fenster der Entwicklungsumgebung	64
3.2.1	Der Projektmappen-Explorer	66
3.2.2	Die Toolbox	69
3.2.3	Das Eigenschafts-/Ereignisfenster	69
3.2.4	Die Projekteigenschaften	70

3.3	Der visuelle Designer	71
3.4	Der Editor	72
3.4.1	Anpassung des Editors	72
3.4.2	IntelliSense	74
3.4.3	Smarttags	74
3.4.4	Änderungen innerhalb einer Sitzung	74
3.4.5	Refactoring	75
3.5	Tools und Hilfsmittel	75
3.5.1	Das Klassendiagramm	76
3.5.2	Das Objekttestcenter	76
3.5.3	Code Snippets (Codeausschnitte)	78
3.6	Fazit	78

II Grundlagen 79

4 Datentypen 81

4.1	Werte- und Referenztypen	81
4.1.1	Unterschiede zwischen Werte- und Referenztypen	81
4.2	Integrierte Datentypen	83
4.2.1	Der Datentyp bool	84
4.2.2	Der Datentyp char	85
4.2.3	Numerische Datentypen	87
4.2.4	Der Datentyp string	88
4.2.5	Nullable Types	88
4.3	Variablen	91
4.3.1	Deklaration und Initialisierung	91
4.3.2	Bezeichner	92
4.3.3	Gültigkeitsbereich	93
4.3.4	Konstanten	94
4.4	Konvertierungen und Boxing	94
4.4.1	Implizite und explizite Konvertierung	95
4.4.2	Boxing und Unboxing	96
4.4.3	Typumwandlung	97
4.5	Arrays	98
4.5.1	Eindimensionale Arrays	99
4.5.2	Mehrdimensionale Arrays	100
4.5.3	Ungleichförmige Arrays	101
4.5.4	Arbeiten mit Arrays	102
4.5.5	Syntaxzusammenfassung	105
4.6	Aufzählungstypen (enum)	106
4.6.1	Grundlagen	106
4.6.2	Deklaration und Anwendung	107
4.6.3	Bitfelder	109
4.6.4	Arbeiten mit Aufzählungen	111
4.6.5	Syntaxzusammenfassung	113

5	Ablaufsteuerung	115
5.1	Verzweigungen	115
5.1.1	Die if-Anweisung	115
5.1.2	Die switch-Anweisung	117
5.1.3	Die bedingte Zuweisung (tenärer Operator)	118
5.1.4	Die goto-Anweisung	119
5.2	Schleifen	120
5.2.1	Die for-Schleife	120
5.2.2	Die while-Schleife	121
5.2.3	Die do-Schleife	122
5.2.4	Die foreach-Schleife	123
5.3	Operatoren	123
5.3.1	Arithmetische Operatoren	124
5.3.2	Vergleichsoperatoren	125
5.3.3	Logische Operatoren	125
5.3.4	Bitweise Operatoren	126
5.3.5	Zusammengesetzte Operatoren	126
5.3.6	Sonstige Operatoren	127
6	Klassen und Objekte	129
6.1	Grundlagen der Objektorientierung	129
6.1.1	Abstraktion	129
6.1.2	Kapselung	130
6.1.3	Vererbung	131
6.1.4	Polymorphie	132
6.1.5	Aggregation	132
6.2	Gliederung einer Anwendung: Namespaces	133
6.2.1	Deklaration	134
6.2.2	Aufteilung der Deklaration	135
6.3	Klassen	136
6.3.1	Klassendeklaration	136
6.3.2	Felder (Instanzvariablen)	137
6.3.3	Methoden	139
6.3.4	Konstruktoren und Destruktoren	147
6.3.5	Eigenschaften (Properties)	150
6.4	Statische Klasselemente	155
6.4.1	Private Konstruktoren	156
6.4.2	Statische Konstruktoren	157
6.4.3	Statische Klassen	158
6.5	Modifizierer	158
6.6	Operatorenüberladung	161
6.6.1	Überladen mathematischer Operatoren	161
6.6.2	Überladen der Konvertierungsoperatoren	162
6.7	Partielle Klassen	163

6.8	Strukturen (struct)	164
6.8.1	Deklaration	164
6.8.2	Nullable Structs	166
7	Vererbung und Polymorphie	167
7.1	Vererbung	167
7.1.1	Ableiten von Klassen	167
7.1.2	Gemeinsame Methoden aller Klassen	170
7.1.3	Virtuelle Methoden	171
7.2	Polymorphie	173
7.3	Abstrakte, versiegelte und verschachtelte Klassen	177
7.3.1	Versiegelte Klassen	177
7.3.2	Abstrakte Klassen	178
7.3.3	Verschachtelte Klassen	178
8	Delegates und Events	181
8.1	Grundlagen zu Delegates	181
8.2	Verwenden von Delegates	182
8.2.1	Eine Sortierroutine	182
8.2.2	Ein Delegate – mehrere Methoden	188
8.2.3	Anonyme Methoden	189
8.3	Ereignisse (Events)	192
8.3.1	Deklaration	192
8.3.2	Ereignisse implementieren und verwenden	194
9	Interfaces	199
9.1	Grundlagen	199
9.1.1	Deklaration	200
9.1.2	Implementierung	201
9.1.3	Interface explizit verwenden	206
9.1.4	Nicht implementierte Methoden	207
9.2	Die Interfaces IComparer und IComparable	208
9.2.1	Deklaration	208
9.2.2	Verwendung von IComparer und IComparable	209
9.3	Das Interface IDisposable	212
10	Attribute	215
10.1	Grundlagen	215
10.1.1	Verwendung	216
10.1.2	Parameter	216
10.2	Eigene Attribute erstellen	217
10.2.1	Verwendung festlegen	218
10.2.2	Attributparameter	218
10.2.3	Ermitteln des Attributs	219

10.3	Beispiel: Ein Todo-Attribut	219
10.3.1	Deklaration der Attributklasse	219
10.3.2	Auswertung der Attribute	221
II	Generics	225
11.1	Grundlagen zu Generics	225
11.1.1	Deklaration	225
11.1.2	Beispiel: ein generischer Stack	226
11.1.3	Der Standardwert eines generischen Typs	227
11.2	Constraints	229
11.2.1	Mögliche Bedingungen	230
11.2.2	Erweitern des Beispiels	231
11.3	Vererbung mit Generics	233
11.3.1	Konkrete Klassen mit generischer Basisklasse	234
11.3.2	Generische Klassen mit generischer Basisklasse	234
11.4	Generische Methoden	235
III	Grundlegende Programmieretechniken	237
12	Arbeiten mit Datentypen	239
12.1	Zahlen	239
12.1.1	Notation	239
12.1.2	Rundungsfehler	239
12.1.3	Division durch Null und der Wert unendlich	241
12.1.4	Arithmetische Funktionen	241
12.1.5	Zahlen runden und andere Funktionen	242
12.1.6	Zufallszahlen	243
12.2	Strings	244
12.2.1	Grundlagen	244
12.2.2	Verketteten von Strings	246
12.2.3	Zugriff auf Zeichenketten	247
12.2.4	Vergleiche von Zeichenketten	251
12.2.5	Die Klasse StringBuilder	252
12.2.6	Unicode	255
12.2.7	Syntaxzusammenfassung	257
12.3	Datum und Zeit	262
12.3.1	Die Struktur DateTime	262
12.3.2	Die Struktur TimeSpan	263
12.3.3	Arbeiten mit Datum und Zeit	265
12.3.4	Zeitmessungen - die Klasse Stopwatch	266
12.3.5	Syntaxzusammenfassung	267
12.4	Formatierungsmethoden in .NET	271
12.4.1	Grundlagen	271
12.4.2	Zahlen formatieren	272
12.4.3	Datum und Zeit formatieren	278

13	Collections	281
13.1	Grundlagen	281
13.2	Die Listenklassen aus System.Collections	282
13.2.1	Übersicht	282
13.2.2	Übersicht über die verwendeten Interfaces	285
13.3	Grundlegende Programmier Techniken	286
13.3.1	Listenelemente löschen	286
13.3.2	Sortieren von Listen	289
13.3.3	Suchen in einer ArrayList	294
13.3.4	Queue und Stack verwenden	295
13.3.5	Datenaustausch zwischen Listen	298
13.4	Eigene Listenklassen erstellen	300
13.4.1	Eine neue Art von Eigenschaft: der Indexer	300
13.4.2	Implementierung der foreach-Schleife	302
13.4.3	Beispielprogramm Bücherliste	303
13.5	Syntaxzusammenfassung	308
13.5.1	Interfaces	308
13.5.2	Klassen	310
13.6	Generische Listenklassen	312
13.6.1	Verwendung generischer Listenklassen	313
13.6.2	Geschwindigkeitsvergleich Generics zu normal	315
14	Dateien und Verzeichnisse	317
14.1	Grundlagen	317
14.1.1	Streams	317
14.1.2	Klassen von System.IO	318
14.2	Verzeichnisse und Dateien	320
14.2.1	Datei- und Verzeichnisinformationen	320
14.2.2	Ermittlung von Dateien in einem Verzeichnis	323
14.2.3	Manipulation von Dateien und Verzeichnissen	327
14.2.4	Verzeichnisse, Dateien und Laufwerke ermitteln	332
14.2.5	Datei- und Verzeichnisnamen bearbeiten	335
14.2.6	Beispielanwendung Backup	336
14.2.7	Beispielanwendung Synchronisieren	343
14.2.8	Syntaxzusammenfassung	346
14.3	Dialoge für Verzeichnisse und Dateien	354
14.3.1	Der Dialog zum Öffnen einer Datei	354
14.3.2	Der Dialog zum Speichern einer Datei	355
14.3.3	Der Dialog zur Verzeichnisauswahl	355
14.3.4	Syntaxzusammenfassung	356
14.4	Textdateien	357
14.4.1	Kodierung von Textdateien	357
14.4.2	Textdateien lesen (mit StreamReader)	358
14.4.3	Textdateien schreiben (StreamWriter)	365
14.4.4	Beispielprogramm – Textdatei erstellen und lesen	367

14.4.5	Beispielprogramm – Textkodierung ändern	369
14.4.6	Zeichenketten lesen und schreiben (StringReader und StringWriter)	370
14.4.7	Syntaxzusammenfassung	371
14.5	Binäre Dateien	375
14.5.1	Die Klasse FileStream	376
14.5.2	Beispielprogramm – Dateien zerteilen	378
14.5.3	Gleichzeitiger Zugriff auf eine Datei	385
14.5.4	Die Klasse BufferedStream	388
14.5.5	MemoryStream (Streams im Arbeitsspeicher)	392
14.5.6	Dateien komprimieren	393
14.5.7	BinaryReader und -Writer (Variablen binär speichern)	395
14.5.8	Beispielprogramm – unterschiedliche Daten schreiben und lesen	397
14.5.9	Syntaxzusammenfassung	402
14.6	Asynchroner Dateizugriff	405
14.6.1	Verwendung eines asynchronen Streams	406
14.6.2	Asynchrones Schreiben mit Callback	409
14.7	Verzeichnisse überwachen	411
14.7.1	FileSystemWatcher verwenden	411
14.7.2	Verzeichnisüberwachung mit Logdatei	413
14.8	Serialisierung	416
14.8.1	Grundlagen	416
14.8.2	Serialisieren mit BinaryFormatter und SoapFormatter	416
14.8.3	Angepasste Serialisierung	422
14.8.4	Die Klasse XmlSerializer	424
15	Multithreading	429
15.1	Grundlagen	429
15.1.1	Preemptives Multitasking	429
15.1.2	Multithreading-Modelle	430
15.1.3	Wozu Multithreading?	430
15.2	Arbeiten mit Threads	431
15.2.1	Die Klasse Thread	431
15.2.2	Beispielanwendung	436
15.2.3	Syntaxzusammenfassung	439
15.3	Synchronisation	440
15.3.1	Wozu synchronisieren?	440
15.3.2	Die Klasse Monitor	441
15.3.3	Die Anweisung lock()	448
15.4	Die Komponente BackgroundWorker	449
15.4.1	Methoden und Ereignisse	449
15.4.2	Beispielapplikation: Fibonacci	450
16	Fehlersuche und Fehlerabsicherung	455
16.1	Fehlerabsicherung	455
16.1.1	Abfangen von Exceptions	456
16.1.2	try-catch-Syntax	457

16.1.3	Eigenschaften und Methoden der Klasse Exception	463
16.1.4	Eigene Exception-Klassen	463
16.2	Fehlersuche (Debugging)	466
16.2.1	Grundlagen	466
16.2.2	Fehlersuche in der Entwicklungsumgebung	468
16.2.3	Ausgaben der Klasse Debug	473
16.2.4	Syntaxzusammenfassung	475

IV Windows-Programmierung 477

17 Einführung in Windows.Forms 479

17.1	Einführung	479
17.1.1	Interaktion zwischen Code und Designer	480
17.1.2	Arbeiten mit dem Visual Studio	483
17.2	Arbeiten mit Formularen	486
17.2.1	Eigenschaften von Formularen	486
17.2.2	Grundlegende Vorgehensweisen	490

18 Standard-Steuerelemente 501

18.1	Überblick	501
18.1.1	.NET-Steuerelemente	501
18.1.2	COM und .NET	502
18.1.3	Steuerelemente in Toolbox einfügen	503
18.2	Gemeinsame Member der Steuerelemente	504
18.2.1	Aussehen	505
18.2.2	Größe, Position und Layout	508
18.2.3	Eingabefokus, Validierung	512
18.2.4	Sonstiges	514
18.2.5	Syntaxzusammenfassung	516
18.3	Buttons	520
18.3.1	Das Steuerelement Button	520
18.3.2	Das Steuerelement CheckBox	522
18.3.3	Das Steuerelement RadioButton	524
18.4	Steuerelemente für Text	525
18.4.1	Das Steuerelement Label	525
18.4.2	LinkLabel	526
18.4.3	Das Steuerelement TextBox	528
18.4.4	Das Steuerelement MaskedTextBox	535
18.4.5	Das Steuerelement RichTextBox	537
18.5	Steuerelemente für Grafik	540
18.5.1	Das Steuerelement PictureBox	540
18.5.2	Das Steuerelement ImageList	541
18.6	Listen	542
18.6.1	ListBox	542
18.6.2	CheckedListBox	557

18.6.3	ComboBox	558
18.6.4	ListView	559
18.6.5	Beispielprogramm	568
18.6.6	TreeView	579
18.6.7	Beispielprogramm: Festplatteninhalt ermitteln	587
18.7	Datum und Zeit	593
18.7.1	MonthCalendar	593
18.7.2	DateTimePicker	596
18.8	Schiebe- und Zustandsbalken, Drehfelder	597
18.8.1	HScrollBar, VScrollBar	597
18.8.2	TrackBar	599
18.8.3	ProgressBar	601
18.8.4	NumericUpDown	601
18.8.5	DomainUpDown	601
18.9	Gruppieren von Steuerelementen	602
18.9.1	Panel	602
18.9.2	GroupBox	604
18.9.3	TabControl (Dialogblätter)	604
18.10	Weitere Steuerelemente	606
18.10.1	SplitContainer	606
18.10.2	TableLayoutPanel	607
18.10.3	FlowLayoutPanel	607
18.10.4	Timer	607
18.10.5	ToolTip	608
18.10.6	HelpProvider	611
18.10.7	ErrorProvider	614
18.10.8	NotifyIcon	617
19	Eigene Steuerelemente erstellen	621
19.1	Grundlagen zu Steuerelementen	621
19.1.1	Arten von Steuerelementen	621
19.1.2	Vorbereitung	622
19.2	Zusammengesetzte Steuerelemente	624
19.2.1	Eine IP-Textbox als Steuerelement	624
19.2.2	Funktionalität hinzufügen	626
19.2.3	Die Bitmap für die Toolbox	634
19.2.4	Testen des Steuerelements	635
19.3	Abgeleitete Steuerelemente	636
19.3.1	Ein erweitertes Panel	636
19.3.2	Funktionalität hinzufügen	638
20	Benutzeroberfläche	647
20.1	Bedienelemente	647
20.1.1	Der Menüeditor	648
20.1.2	Menüeigenschaften einstellen	649
20.1.3	Kontextmenüs	653

20.1.4	Symbolleisten (ToolStrip-Steuererelement)	657
20.1.5	Die Statusleiste (StatusStrip-Steuererelement)	660
20.1.6	Eigene Elemente für den ToolStrip	661
20.2	Standarddialoge	667
20.2.1	Dateien öffnen und speichern	667
20.2.2	Farbauswahl	669
20.2.3	Schriftart auswählen	669
20.3	MDI-Anwendungen	670
20.3.1	Grundlagen	670
20.3.2	MDI-Fenster verwalten	671
20.3.3	Beispiel: Mergen von Menüs und ToolStrip-Komponenten	673
20.4	Programmiertechniken	676
20.4.1	Anzeige eines Splashscreens	676
20.4.2	Arbeiten mit der Zwischenablage	677
20.4.3	Drag&Drop	678
20.4.4	Konfigurationsdateien	686
21	Grafikprogrammierung (GDI+)	691
21.1	Einführung	691
21.1.1	Ein erstes Beispiel	692
21.1.2	Grafik-Container (Form, PictureBox)	695
21.1.3	Dispose für Grafikobjekte	697
21.1.4	Fazit	698
21.2	Elementare Grafikoperationen	699
21.2.1	Linien, Rechtecke, Vielecke, Ellipsen, Kurven (Graphics-Klasse)	699
21.2.2	Farben (Color-Struktur)	712
21.2.3	Linienformen (Pen-Klasse)	714
21.2.4	Füllmuster (Brush-Klassen)	718
21.2.5	Koordinatensysteme und -transformationen	723
21.2.6	Syntaxzusammenfassung	727
21.3	Text ausgeben (Font-Klassen)	731
21.3.1	Einführung	731
21.3.2	TrueType-, OpenType- und Type-1-Schriftformate	733
21.3.3	Schriftarten und -familien	734
21.3.4	Schriftgröße	737
21.3.5	Schriftauszeichnung und Textformatierung	748
21.3.6	Font-Auswahldialog	760
21.3.7	Syntaxzusammenfassung	762
21.4	Bitmaps, Icons und Metafiles	764
21.4.1	Die Klassen Graphics, Image und Bitmap	764
21.4.2	Bitmaps in Formularen darstellen	767
21.4.3	Bitmaps manipulieren	770
21.4.4	Transparente Bitmaps	778
21.4.5	Metafile-Dateien	783
21.4.6	Syntaxzusammenfassung	786
21.5	Fortgeschrittene Programmiertechniken	787

21.5.1	Zeichen- und Textqualität	787
21.5.2	Grafikobjekte zusammensetzen (GraphicsPath)	792
21.5.3	Umgang mit Regionen und Clipping	794
21.5.4	Rechteck-Auswahl mit der Maus (Rubberbox)	801
21.5.5	Bitmap-Grafik zwischenspeichern (AutoRedraw)	805
21.5.6	Flimmerfreie Grafik (Double-Buffer-Technik)	817
21.5.7	Scrollbereich für Grafik	818
21.5.8	Einfache Animationseffekte	820
21.5.9	Bitmaps direkt manipulieren	825
22	Drucken	831
22.1	Überblick	831
22.1.1	Limitationen und weitere Werkzeuge zum Drucken	831
22.1.2	Die wichtigsten Klassen und Steuerelemente	832
22.2	Grundlagen	833
22.2.1	Die Komponente PrintDocument	833
22.2.2	Die Dialoge PrintDialog und PageSetupDialog	836
22.2.3	Der Dialog PrintPreviewDialog	840
22.2.4	Druckereigenschaften und Seitenlayout	841
22.2.5	Syntaxzusammenfassung	844
22.3	Beispiel – Mehrseitiger Druck	847
22.4	Eine Textbox zum Drucken	851
22.5	Weitere Programmiertechniken	856
22.5.1	Unterschiedliches Seitenlayout	856
22.5.2	Drucken ohne Status- bzw. Abbruch-Dialog	856
22.6	Eigene Seitenvorschau	857
V	Programmiertechniken	863
23	Lokalisierung von Anwendungen	865
23.1	Eigenschaften von Steuerelementen und Formularen lokalisieren	865
23.1.1	Ressourcendateien	865
23.1.2	Auswertung der Lokalisierungseinstellungen	866
23.1.3	Auswahl der aktuellen Kultur	867
23.1.4	Zusätzliche Zeichenketten in den Lokalisierungsdateien speichern	867
23.2	Beispielprogramm	868
24	Externe Programme steuern (Automation)	873
24.1	Automation mittels COM-Komponenten	873
24.1.1	Verwendung der Klassenbibliothek	873
24.1.2	Beispiel – RichTextBox mit Word ausdrucken	874
24.2	API-Aufrufe (P/Invoke)	878
24.2.1	Grundlagen zu P/Invoke	878
24.2.2	Konvertierungen	880
24.2.3	Aufruf von DLL-Funktionen	881

25	Reflection	885
25.1	Grundlagen zu Reflection	885
25.1.1	Grundlegende Eigenschaften und Methoden von Type	885
25.1.2	Relevante Klassen für Reflection	886
25.2	Beispielapplikation: Informationen über die BCL ermitteln	887
25.2.1	Das Hauptformular der Anwendung – der Aufbau	888
25.2.2	Die Klasse AssemblyReflector	889
25.2.3	Die Klasse TypeReflector	896
25.2.4	Das Hauptformular – die Funktionalität	901
25.2.5	Programmstart mit Splashscreen	903
25.3	Beispielprogramm: Daten mittels Reflection ändern	906
25.3.1	Die Klasse zum Instanzieren	907
25.3.2	Das Hauptformular	907
26	Weitergabe von Windows-Programmen (Setup.exe)	911
26.1	Einführung	911
26.2	Installationsprogramm erstellen (Entwicklersicht)	912
26.3	Installation ausführen (Kundensicht)	914
26.4	Installationsprogramm für Spezialaufgaben	916
26.4.1	Grundeinstellungen eines Setup-Projekts	916
26.4.2	Startmenü, Desktop-Icons	919
26.4.3	Benutzeroberfläche des Installationsprogramms	920
26.4.4	Start- und Weitergabebedingungen	923
26.4.5	Dateityp registrieren	925
26.5	ClickOnce	926
26.5.1	ClickOnce Einstellungen	927
26.5.2	ClickOnce-Installation aus Anwendersicht	932
26.6	Signieren eines Projekts	933
26.6.1	Bestandteile einer Signatur	933
26.6.2	Der Signiervorgang	935
26.6.3	Signierte DLL in den GAC einfügen	936
VI	Datenbanken (ADO.NET)	939
27	Grundlagen	941
27.1	Datenbanksysteme	941
27.1.1	Übersicht über gebräuchliche Datenbanksysteme	942
27.2	Einrichten der Datenbankumgebung	944
27.2.1	Installation des SQL Server 2005 (Standard/Express)	944
27.2.2	Die Beispieldatenbank des Buchs	947
27.3	Datenbankwerkzeuge	948
27.3.1	SQL Server Management Studio	948
27.3.2	SQL Management Studio Express	953
27.4	SQL Grundlagen	955
27.4.1	Data Definition Language	955

27.4.2	Data Manipulation Language	957
27.4.3	Data Query Language	959
27.4.4	Stored Procedures	962
27.5	Datenbankaufbau	965
27.5.1	Global Unique Identifiers (GUID)	966
27.5.2	Normalformen	967
28	Überblick über ADO.NET	969
28.1	Grundlegender Datenbankzugriff	969
28.1.1	Namespaces für die Datenbankprovider	969
28.1.2	Die Klasse SqlConnection	970
28.1.3	Datenbankkommandos absetzen (SqlCommand)	974
28.1.4	Daten ermitteln (SqlDataReader)	975
28.1.5	Stored Procedures verwenden	983
28.2	In-Memory-Datenbank: Das DataSet	986
28.2.1	Klassen für das DataSet	987
28.2.2	Laden der Daten: SqlDataAdapter	988
28.2.3	Erzeugen von Kommandos: Der SqlCommandBuilder	990
28.2.4	Speichern der Daten mit SqlDataAdapter	993
28.2.5	Datenspeicher: Die Klasse DataTable	995
28.2.6	Verknüpfen von Tabellen (DataRelation)	998
28.3	Visual Data Tools	1001
28.3.1	Steuerelemente für den Datenzugriff	1002
28.3.2	Die Hilfsmittel des Visual Studio	1002
28.3.3	Das DataGridView im Detail	1009
28.3.4	Weitere Steuerelemente für die Datenbindung	1015
28.3.5	Typisierte DataSets	1017
29	Fortgeschrittene Programmieretechniken	1023
29.1	Metadaten des SQL Server ermitteln	1023
29.1.1	Arten von Metadaten	1023
29.1.2	Ermitteln des Datenbankschemas	1026
29.1.3	Parameter für GetSchema()	1026
29.1.4	Ermitteln der Datenbankinformationen	1027
29.2	Automatische Erstellung von Business-Objekten	1032
29.2.1	Das Hauptformular	1033
29.2.2	Tabellen auswählen	1037
29.2.3	»Konvertieren« der Tabellen in Klassen	1041
29.3	Datenbindung mit Objekten	1049
29.3.1	Grundlagen der Datenbindung an Objekte	1049
29.3.2	Beispielprogramm mit Objekt-Datenbindung	1051
A	Glossar	1059
	Index	1065