

Inhalt

Vorwort	17
Was dieses Buch erreichen will	17
Für wen dieses Buch geeignet ist	18
Was Sie nicht in diesem Buch finden werden	19
Wie Sie dieses Buch lesen sollten	19
Website zum Buch	20
I Die .NET-Klassenbibliothek	21
1.1 Einführung	21
1.1.1 Funktionsumfang	22
1.1.2 Implementierung	23
1.1.3 Grundkonzept der FCL	23
1.1.4 Hinweise zu den Vorab-Versionen	25
1.1.5 Nutzbarkeit der FCL von Unmanaged Clients	26
1.1.6 Verfügbarkeit	27
1.2 Typen in .NET	27
1.2.1 Klassen	28
1.2.2 Schnittstelle (Interface)	31
1.2.3 Strukturen (Werteklassen)	32
1.2.4 Aufzählungstypen (Enumerationen)	33
1.2.5 Typnamen und Namespaces	34
1.2.6 Typhierarchien	36
1.3 Verwendung von FCL-Typen	38
1.3.1 Assembly-Referenz	38
1.3.2 Importieren von Namespaces	38
1.3.3 Instanziierung	40
1.3.4 Vererbung von Klassen	40
1.3.5 Sichtbarkeit und Zugriff	41
1.4 Überblick über die FCL-Namespaces	42
1.5 Implementierung der FCL	47
1.6 Original-Dokumentation der FCL	54
1.6.1 Position der FCL-Dokumentation	54
1.6.2 Aufbau der FCL-Dokumentation	55
1.6.3 Weitere Dokumentation	57

1.7	Werkzeuge	57
1.7.1	Windows Class Viewer (WinCV.exe)	57
1.7.2	Objektbrowser	58
1.7.3	Class Browser	60
1.7.4	Reflector	60
1.7.5	Intermediation Language Disassembler	61
1.8	Standardisierung	62
2	Visual Basic .NET (VB 7.0)	65
2.1	Änderungen von VB 6.0 zu VB.NET (VB 7.0)	66
2.1.1	Integration in das .NET Framework	67
2.1.2	Verbesserte Objektorientierung	68
2.1.3	Veränderung bei den Sprachkonstrukten	69
2.1.4	Neue Konstrukte	71
2.1.5	Bewertung	72
2.2	Entwicklung von VB.NET-Programmen mit dem Kommandozeilen-Compiler	73
2.2.1	Erstellung einer Hello-World-Anwendung	73
2.2.2	Entwicklung einer DOS-Anwendung	75
2.2.3	Nutzung von FCL-Klassen	76
2.3	Nutzung der Visual Studio .NET-Entwicklungsumgebung	79
2.3.1	Visual Studio .NET-Überblick	79
2.3.2	Projektmappen und Projekte	83
2.3.3	Ausgaben in der VS.NET-Entwicklungsumgebung	89
2.3.4	Erstellung einer Anwendung in VS.NET	90
3	C# vs. Visual Basic .NET	95
3.1	Grundlagen	95
3.1.1	C# ist case-sensitive	96
3.1.2	Anweisungen und Zeilen	97
3.1.3	Kommentare	98
3.1.4	Codeblöcke	101
3.1.5	Option-Statements	102
3.1.6	Schlüsselwörter	102
3.1.7	Operatoren	105
3.2	Deklarationen	109
3.2.1	Klassen und Module	109
3.2.2	Variablen und Felder	110
3.2.3	Methoden	111
3.2.4	Eigenschaften	112
3.2.5	Konstruktoren und Destruktoren	114
3.3	Programmiertechniken	117
3.3.1	Schleifen	117
3.3.2	Verzweigungen	122
3.3.3	Casting	126
3.3.4	With-Anweisung	127
3.3.5	Arrays	128
3.4	Unterschiede in der Entwicklungsumgebung	130

4	Namespace System	133
4.1	Klasse System.Object	133
4.1.1	GetType()	134
4.1.2	ToString()	135
4.1.3	GetHashCode()	136
4.1.4	Equals()	136
4.1.5	ReferenceEquals()	137
4.1.6	Beispiel zur Anwendung der Klasse System.Object	137
4.1.7	MemberwiseClone()	140
4.2	Klassen für Basis-Datentypen	144
4.2.1	Übersicht	144
4.2.2	System.Void	146
4.2.3	Operationen der Basisdatentyp-Klassen	146
4.2.4	System.String	148
4.3	Weitere Datentypen	156
4.3.1	System.Guid	157
4.3.2	System.Uri und System.UriBuilder	159
4.3.3	System.Version	163
4.4	Klasse System.Console	164
4.5	Klasse System.BitConverter	167
4.6	Klasse System.Convert	169
4.7	Klasse System.Math	172
4.8	Klasse System.Random	172
4.9	Klasse System.GC	174
4.10	Klasse System.Environment	179
4.11	Weitere Klassen im Namespace System	185
5	Namespace System.Collections	187
5.1	Klassen in System.Collections	187
5.1.1	Die Klasse ArrayList	189
5.1.2	Die Klassen Stack und Queue	197
5.1.3	Die Klasse BitArray	201
5.1.4	Die Klasse Hashtable	204
5.2	Eigene Collections erzeugen	207
5.3	Spezialisierte Collections	220
6	Namespace System.IO	223
6.1	Zugriff auf die Struktur des Dateisystems	223
6.1.1	Objektmodell	224
6.1.2	Verzeichnisse auflisten	225
6.1.3	Verzeichnisse und Dateien suchen	226
6.1.4	Verzeichnisse anlegen und löschen	229
6.1.5	Dateien anlegen und löschen	232
6.1.6	Kopieren und Verschieben	233
6.1.7	Informationen über das Dateisystem	234
6.1.8	Arbeit mit Pfadangaben	235
6.2	Zugriff auf Dateiinhalte	237

6.2.1	Readers, Writers und Streams	237
6.2.2	Textdatei schreiben	239
6.2.3	Textdatei lesen	241
6.2.4	Binärdatei schreiben	242
6.2.5	Binärdatei lesen	243
6.3	Dateisystem überwachen	244
7	Namespaces System.Text und System.Text.RegularExpressions	251
7.1	Die Klasse StringBuilder	251
7.1.1	Eigenschaften und Methoden	252
7.1.2	Erzeugen eines Zufalls-Strings	253
7.1.3	Mehrfaches Einfügen von Strings	255
7.2	Reguläre Ausdrücke	257
7.2.1	Übersicht	257
7.2.2	Suchausdrücke (Patterns)	260
7.2.3	Die Klasse Regex	261
8	Namespaces System.Globalization und System.Resources	271
8.1	Lokalisierung	271
8.1.1	Ressourcendateien	272
8.1.2	Fehler vermeiden	272
8.2	Kalenderklassen	274
8.2.1	Umrechnen von Datumsangaben	277
8.3	Lokalisierung von Programmen	279
8.3.1	Die Klasse CultureInfo	280
8.3.2	Lokalisierung mit Visual Studio .NET	287
8.4	Zugriff auf Ressourcendateien	293
8.4.1	Zusammenhänge	293
8.4.2	Die Klasse ResourceManager	295
8.4.3	Eine Anwendung mit lokalisierten Strings	295
9	Der Namespace System.Xml	299
9.1	XmlReader/XmlWriter	300
9.1.1	Xml-Daten aus Dateien lesen	302
9.1.2	XML validieren	311
9.1.3	XML-Daten schreiben	318
9.2	XmlDocument	324
9.2.1	Auslesen der Daten	325
9.2.2	Daten in XmlDocument ändern bzw. hinzufügen	332
9.3	XPath	341
9.3.1	XPathNodeIterator	343
9.3.2	XPathNavigator	343
9.4	XslTransform	345

10	Namespace System.Data (ADO.NET)	349
10.1	Architektur	350
10.2	Objektmodell	352
10.3	Hinweis zu den Beispielen	356
10.4	Daten-Verbindungen	357
10.5	Daten lesen mit dem OleDbDataReader	360
10.6	DataSets	361
10.6.1	DataTables	362
10.6.2	Datenadapter	365
10.6.3	Lesezugriff auf Daten	366
10.7	Daten ändern mit DataSets	373
10.7.1	Befehle generieren mit dem CommandBuilder	374
10.7.2	Datensätze ändern	376
10.7.3	Datensätze anfügen	378
10.7.4	Datensätze löschen	380
10.8	Daten ändern mit OleDbCommand	382
10.9	Verknüpfung von Tabellen in einem DataSet	384
10.10	Dynamische Tabellen im Speicher	388
10.11	ADO.NET und XML	390
10.11.1	Ausgabe eines DataSets als XML	390
10.11.2	XmlDataDocument	393
10.11.3	XML in DataSet einlesen	396
10.12	Weitere Möglichkeiten von ADO.NET	399
11	Namespaces System.Runtime.Serialization und System.Xml.Serialization	401
11.1	Serialisierer in .NET	402
11.2	Binärer Serialisierer	403
11.2.1	Serialisieren in Binärdatei	404
11.2.2	Serialisieren in Byte-Folge oder Datenbank	405
11.3	SOAP-Serialisierer	409
11.4	XML-Serialisierer	410
11.4.1	XML-Serialisierung	411
11.4.2	XML-Deserialisierung	413
11.4.3	Anwendungsbeispiel	414
11.5	Serialisierung eines Objektmodells	416
11.6	Änderungen an serialisierten Objekten	421
11.7	Benutzerdefinierte (De-)Serialisierung	422
11.7.1	Benutzerdefinierte Serialisierung für BinaryFormatter und SOAPFormatter	422
11.7.2	Benutzerdefinierte Serialisierung für den XmlSerializer	426
12	Namespace System.ServiceProcess	429
12.1	Architektur von Windows-Diensten	429
12.2	Die Klasse ServiceBase	431
12.2.1	Eigenschaften und Methoden	431
12.2.2	Die Dienst-Hauptmethode	433

12.3	Die Klassen ServiceProcessInstaller und ServiceInstaller	434
12.3.1	Beispiel: Ein einfacher Dienst	436
12.3.2	Das Setup-Projekt für den Dienst	442
12.4	Die Klasse ServiceController	444
12.4.1	Eigenschaften und Methoden	444
12.4.2	Dienste auflisten	446
13	Namespace Microsoft.Win32	449
13.1	Die Klasse SystemEvents	449
13.1.1	Standardereignisse	449
13.2	Zugriff auf die Registry	453
13.2.1	Die Klasse Registry	454
13.2.2	Die Klasse RegistryKey	455
14	Namespace System.Diagnostics	469
14.1	Arbeit mit den Ereignisprotokollen	469
14.1.1	Überblick über die Ereignisprotokoll-Klassen	470
14.1.2	Auslesen eines Ereignisprotokolls	470
14.1.3	Liste aller Ereignisprotokolle	472
14.1.4	Anlegen eines neuen Ereignisprotokolls	473
14.1.5	Löschen eines Ereignisprotokolls	474
14.1.6	Schreiben in das Ereignisprotokoll	475
14.1.7	Überwachung eines Ereignisprotokolls	476
14.2	Arbeit mit Prozessen	477
14.2.1	Überblick über die FCL-Klassen	478
14.2.2	Liste der Prozesse	480
14.2.3	Liste der Module	483
14.2.4	Liste der Threads	485
14.2.5	Prozesse starten	486
14.2.6	Prozesse beenden	487
14.2.7	Nicht reagierende Prozesse beenden	489
14.3	Informationen über Programmdateien	489
14.4	Arbeit mit Leistungsindikatoren	490
14.4.1	Grundlagen	491
14.4.2	Überblick über die FCL-Klassen	492
14.4.3	Lesen eines Leistungsindikators	493
14.4.4	Anlegen und Löschen eines Leistungsindikators	494
14.4.5	Schreiben von Werten in einen Leistungsindikator	496
15	Namespace System.DirectoryServices	501
15.1	Architektur	501
15.2	Objektmodell	503
15.2.1	Allgemeine Klassen	503
15.2.2	Klassen für die Ausführung von Suchanfragen	505
15.2.3	Vergleich zwischen .NET-ADSI und COM-ADSI	506

15.3	Überblick über die Programmiermechanismen	506
15.3.1	Objektidentifikation und Bindung	506
15.3.2	Zugriff auf Attribute und Methoden	508
15.3.3	Zugriff auf Container-Objekte	514
15.3.4	Instanzenverwaltung	515
15.4	Active Directory-Programmierung	518
15.4.1	Grundlagen zur Active Directory-Programmierung	518
15.4.2	Testumgebung	523
15.4.3	Ermittlung der Partitionen	524
15.4.4	Informationen über einen Benutzer	525
15.4.5	Benutzer ändern	528
15.4.6	Organisationseinheit anlegen	529
15.4.7	Organisationseinheit löschen	530
15.4.8	Benutzer anlegen	531
15.4.9	Benutzerkennwort setzen	532
15.4.10	Benutzer löschen	533
15.4.11	Benutzer umbenennen	534
15.4.12	Benutzer verschieben	535
15.4.13	Gruppe anlegen	536
15.4.14	Benutzer einer Gruppe hinzufügen	537
15.4.15	Benutzer aus Gruppe entfernen	537
15.4.16	Gruppe löschen	538
15.4.17	Containerinhalt auflisten	539
15.4.18	Computerkonto anlegen	540
15.4.19	Computerkonto löschen	541
15.4.20	Suche im Active Directory	541
15.5	Verzeichnisdienstprogrammierung mit dem WinNT-Provider	545
15.5.1	Grundlagen zum NT4-Verzeichnisdienst	545
15.5.2	Benutzerliste ausgeben	548
15.5.3	Benutzer verwalten	549
15.5.4	Dienste verwalten	550
15.5.5	Dateisystem-Dienst verwalten	552
16	Namespace System.Threading	557
16.1	Funktionsweise von Threads	557
16.2	Threads starten und stoppen	558
16.2.1	Thread und ThreadStart	558
16.2.2	Anhalten und Fortsetzen von Threads	562
16.3	Thread-Prioritäten setzen	572
16.4	Synchronisation	574
16.4.1	Wozu synchronisieren?	574
16.4.2	Die Klasse Monitor	575
16.4.3	Die Anweisung SyncLock	581
16.5	Threadpools	582

17	Namespace System.Windows.Forms	595
17.1	Grundlagen	595
17.1.1	Die Quelldatei des Programms	595
17.1.2	Start des Programms	597
17.1.3	Anordnen von Steuerelementen	598
17.2	Arten von Steuerelementen	599
17.2.1	Controls	599
17.2.2	Nicht-visuelle Steuerelemente	600
17.2.3	Dialoge	600
17.2.4	Übersicht	600
17.3	Datenbindung	602
17.4	Die Klasse Form	605
17.4.1	Eigenschaften	606
17.4.2	Formulare dynamisch erzeugen	609
17.4.3	Zugriff auf Komponenten eines Formulars	611
17.4.4	Zugriff auf Komponenten des Hauptformulars	612
17.4.5	MDI-Applikationen	616
17.5	Arbeiten mit den Steuerelementen	625
17.5.1	Dynamische Erzeugung von Steuerelementen	625
17.5.2	Eingaben kontrollieren	627
17.5.3	Drag&Drop	629
18	Der Namespace System.Drawing	637
18.1	Die Klasse Graphics	638
18.2	Hilfsklassen	640
18.3	Füllwerkzeuge	642
18.3.1	Die Klasse SolidBrush	642
18.3.2	Die Klasse LinearGradientBrush	643
18.3.3	Die Klasse PathGradientBrush	648
18.3.4	Die Klasse TextureBrush	652
18.4	Bitmaps	653
18.4.1	Die Klasse Pen	653
18.4.2	Einfache Pens verwenden	654
18.4.3	Zeichenstift mit Farbverlauf	658
18.5	Arbeiten mit Grafiken	662
18.5.1	Die Klasse Image	662
18.5.2	Die Klasse Bitmap	668
19	Namespace System.Security.Cryptography	679
19.1	Verschlüsselungstechniken	680
19.1.1	Private Schlüssel	680
19.1.2	Öffentliche Schlüssel	681
19.1.3	Hashcodes	681
19.1.4	Digitale Signatur	682
19.2	Hash-Algorithmen	682

19.3	Texte verschlüsseln	686
19.3.1	Asymmetrische Verschlüsselung	687
19.3.2	Symmetrische Verschlüsselung	694
19.4	Verschlüsselung in der Realität	701
19.4.1	Alice und Bob	701
19.4.2	Die Funktionalität	703
20	Namespace System.Reflection	713
20.1	Das Reflection-API	713
20.2	Die Klasse Type	714
20.2.1	Basisinformationen ermitteln	715
20.2.2	Detailinformationen ermitteln	720
20.3	Die Klassen Assembly und Module	728
20.3.1	Informationen über Assemblies ermitteln	728
20.3.2	Namespaces ermitteln	733
20.4	Code on the fly mit Reflection.Emit	741
20.4.1	Erzeugen einer Assembly	743
20.4.2	Die Klasse TypeBuilder	745
20.4.3	Die Klasse ILGenerator	746
20.4.4	Hello World mit Reflection.Emit	747
21	Namespaces System.Net und System.Net.Sockets	751
21.1	Der Namespace System.Net	751
21.1.1	Die Klasse WebClient	752
21.1.2	Die Klasse WebRequest	755
21.1.3	Die Klasse WebResponse	756
21.2	Der Namespace System.Net.Sockets	762
21.2.1	Die Klasse NetworkStream	763
21.2.2	Die Klasse TcpClient	763
21.3	Hilfsklassen	772
22	Namespace System.Management	775
22.1	Windows Management Instrumentation (WMI)	775
22.2	Programmieren mit dem Meta-Objektmodell	776
22.2.1	Überblick über das Objektmodell	776
22.2.2	Objektbindung	778
22.2.3	Zugriff auf Attribute und Methoden	780
22.2.4	Lesezugriff	781
22.2.5	Schreibzugriff	783
22.2.6	Methodenaufrufe	784
22.2.7	Informationen über WMI-Objekte	785
22.2.8	Auflisten von Objektmengen	788
22.3	Programmieren mit Wrapper-Klassen	789
22.3.1	Erzeugung einer Wrapper-Klasse	789
22.3.2	Verwendung der erzeugten Klasse	791
22.3.3	Verwendung der erzeugten Objektmengen	792

22.4	WQL-Abfragen	794
22.4.1	Datenabfragen	796
22.4.2	Ereignisabfragen	799
22.5	WMI im Visual Studio .NET Server Explorer	802
22.6	Weitere Beispiele	808
22.6.1	Abmelden, Herunterfahren und Neustart	808
22.6.2	Netzwerkverbindungen	810
22.6.3	Speichernutzung	811
22.6.4	ASP.NET-Leistungsdaten	812
A	.NET-Glossar	815
B	Active Directory Service Interface (ADSI)	853
B.1	Verzeichnisdienste	854
B.1.1	Aufbau eines Verzeichnisbaums	855
B.1.2	Verzeichnisdienstschemata	857
B.1.3	Objektidentifikation	858
B.1.4	Meta-Directories	859
B.1.5	Lightweight Directory Access Protocol (LDAP)	859
B.2	ADSI-COM-Komponente	863
B.2.1	Architektur	863
B.2.2	Installation	865
B.2.3	Verfügbare ADSI-Provider	865
B.2.4	ADSI Software Development Kit	866
B.2.5	Meta-Objektmodell	867
B.2.6	Überblick über die ADSI-COM-Standardschnittstellen	870
B.3	ADSI-Werkzeuge	876
B.3.1	Active Directory Service Browser (ADB)	876
B.3.2	ADSI Explorer	877
B.3.3	ADSI Edit	878
B.3.4	ADSI Schema Browser	880
C	Windows Management Instrumentation (WMI)	881
C.1	Überblick über die Managed Objects	882
C.2	Installation und Konfiguration	883
C.3	Architektur	884
C.4	Programmierung	885
C.5	Konzepte	886
C.5.1	Schema	887
C.5.2	Namespaces	888
C.5.3	Lokalisierung	889
C.5.4	Klassen und Objekte	890
C.5.5	Managed Object Format (MOF)	893
C.5.6	Sicherheitsfunktionen	895
C.5.7	Ereignisse	895
C.5.8	WMI Query Language (WQL)	898

C.6	Werkzeuge	898
C.6.1	WMI-Steuerung	899
C.6.2	WMI Object Browser	899
C.6.3	WMI CIM Studio	902
C.6.4	WMI Event Registration Tool	905
C.6.5	WMI Event Viewer	907
C.6.6	VBInstance	908
C.6.7	WMI-Testprogramm	908
C.6.8	MOF Compiler	909
D	Hinweise zum Buch	911
D.1	Sprachliche Konventionen	911
D.2	Hinweise zur Formatierung des Textes	912
D.3	Hinweise zu den Code-Beispielen	913
D.4	Grafische Notation in den Objektdiagrammen	917
D.4.1	Knoten (geometrische Formen)	918
D.4.2	Kanten (Verbindungslinien)	919
E	CD-ROM und Website	923
E.1	Der Inhalt der CD-ROM	923
E.2	Die Website zu diesem Buch	923
F	Abkürzungsverzeichnis	925
G	Literaturverzeichnis	933
G.1	Gedruckte Literatur	933
G.2	Quellen im Internet	934
G.2.1	Links	934
G.2.2	Requests for Comment (RFCs)	936
G.2.3	Newsgroups zu .NET	937
G.2.4	Newsgroups zu speziellen COM-Komponenten	937
	Index	939

