

Inhalt

Vorwort 15

1	Das .NET-Konzept 17
1.1	Ein Wort zu diesem Buch 17
1.2	Das Entwicklerdilemma 22
1.2.1	Das .NET-Konzept 23
1.2.2	Einarbeitungszeit 24
1.3	Das Sprachenkonzept 26
1.3.1	NET-Anwendungsentwicklung 26
1.3.2	Die Common Language Specification 27
1.3.3	Das Common Type System 29
1.4	Das .NET-Framework 31
1.4.1	Die Common Language Runtime 31
1.4.2	Die .NET-Klassenbibliothek 32
1.5	Assemblies 36
2	Die Entwicklungsumgebung 39
2.1	Anmerkungen 39
2.2	Hard- und Softwareanforderungen 39
2.3	Die Installation 41
2.4	Die Entwicklungsumgebung des VS .NET 45
2.4.1	Die VB .NET-Vorlagetypen 46
2.4.2	Die Oberfläche der Entwicklungsumgebung 48
2.4.3	Der Code-Editor (Text-Editor) 48
2.4.4	Der Projektmappen-Explorer 51
2.4.5	Die Klassenansicht 51
2.4.6	Das Eigenschaftsfenster 52
2.4.7	Die Werkzeugsammlung (Toolbox) 53
2.4.8	Der Server-Explorer 54
2.4.9	»Dynamische Hilfe« und »Suchen« 55
2.4.10	Das Fenster »Inhalt« 56
2.5	Das Entwickeln einer Konsolenanwendung 57
2.5.1	Zum Abschluss 60
3	Grundlagen der Sprachsyntax 63
3.1	Konsolenanwendungen 63
3.1.1	Allgemeine Anmerkungen 63
3.1.2	Der Projekttyp Konsolenanwendung 64

3.1.3	Die Vorlage »Konsolenanwendung«	65
3.1.4	Zusammenfassung	68
3.2	Variablen und Datentypen	69
3.2.1	Explizite und implizite Variablendeklaration	69
3.2.2	Die Variablendeklaration	77
3.2.3	Die nativen Datentypen	81
3.2.4	Sichtbarkeit und Lebensdauer	93
3.2.5	Module in der Entwicklungsumgebung	98
3.2.6	Initialisierung von Variablen	104
3.2.7	Datentypkonvertierung	105
3.2.8	Typkennzeichen	111
3.2.9	Konstanten	113
3.2.10	Ein- und Ausgabemethoden der Klasse Console	113
3.2.11	Zusammenfassung	119
3.3	Datenfelder (Arrays)	121
3.3.1	Eindimensionale Arrays	121
3.3.2	Mehrdimensionale Arrays	125
3.3.3	Ändern der Arraykapazität	126
3.3.4	Initialisierung der Arrayelemente	128
3.3.5	Bestimmung der Arrayobergrenze	130
3.3.6	Zusammenfassung	132
3.4	Operatoren	133
3.4.1	Arithmetische Operatoren	133
3.4.2	Relationale Operatoren	137
3.4.3	Logische Operatoren	139
3.4.4	Zuweisungsoperatoren	143
3.4.5	Verkettungsoperatoren	145
3.4.6	Operatorprioritäten	145
3.4.7	Bitweise Operationen	146
3.4.8	Zusammenfassung	151
3.5	Kontrollstrukturen	152
3.5.1	Die If-Anweisung	152
3.5.2	Select-Case-Anweisung	156
3.5.3	Einzeilige Entscheidungsanweisungen	159
3.5.4	Zusammenfassung	164
3.6	Programmschleifen	165
3.6.1	Die For...Next-Schleife	165
3.6.2	Do-Schleifen	169
3.6.3	Die While-Schleife	173
3.6.4	Variablendeklaration in Anweisungsblöcken	174
3.6.5	Zusammenfassung	175
3.7	Funktionen und Prozeduren	176
3.7.1	Prozeduren	177
3.7.2	Funktionen	180
3.7.3	Prozedur- und Funktionsaufrufe	184

- 3.7.4 Die Parameterliste 185
- 3.7.5 Zusammenfassung 204

4 Klassen und Objekte (Teil 1) 205

- 4.1 Einführung in die Objektorientierung 205**
 - 4.1.1 Klassen und Objekte 206
 - 4.1.2 Das objektorientierte Paradigma 208
 - 4.1.3 Vorteile der objektorientierten Programmierung 212
 - 4.1.4 Zusammenfassung 213
- 4.2 Die Klassendefinition 214**
 - 4.2.1 Zugriffsmodifizierer einer Klasse 216
 - 4.2.2 Projektmappen-Explorer und Klassenansicht 220
- 4.3 Die Deklaration von Objektvariablen 223**
 - 4.3.1 Frühe und späte Bindung 225
 - 4.3.2 Lebensdauer und Sichtbarkeit von Objektvariablen 226
 - 4.3.3 Zerstören einer Objektreferenz 228
 - 4.3.4 Mehrere Referenzen auf ein Objekt 229
 - 4.3.5 Den Typ einer Objektreferenz ermitteln 230
 - 4.3.6 Typvergleiche von Objektreferenzen 231
 - 4.3.7 Referenzvergleiche 233
 - 4.3.8 Das Clonen von Objekten 239
 - 4.3.9 Deklaration von Objekt-Arrays 242
 - 4.3.10 Zusammenfassung 245
- 4.4 Objekteigenschaften 247**
 - 4.4.1 Datenkapselung 248
 - 4.4.2 Ergänzung der Klasse Circle 254
 - 4.4.3 Lese- und schreibgeschützte Eigenschaften 256
 - 4.4.4 Die Parameterliste einer Property-Prozedur 258
 - 4.4.5 Standardeigenschaften 259
 - 4.4.6 Das With...End With-Statement 261
- 4.5 Objektmethoden 263**
 - 4.5.1 Methodenüberladung 264
 - 4.5.2 Aufruf überladener Methoden mit impliziter Konvertierung 266
 - 4.5.3 Objektreferenzen als Übergabeparameter 268
 - 4.5.4 Der Methodenzugriff auf private Daten 269
 - 4.5.5 Methode oder Eigenschaft? 271
 - 4.5.6 Das Schlüsselwort »Me« 272
 - 4.5.7 Die Trennung von Daten und Code 274
 - 4.5.8 Der aktuelle Stand der Klasse Circle 275
 - 4.5.9 Zusammenfassung 277
- 4.6 Konstruktoren und Destruktoren 278**
 - 4.6.1 Konstruktoren 278
 - 4.6.2 Der Destruktor – Der Finalizer 282
 - 4.6.3 Der aktuelle Stand der Klasse Circle 292
 - 4.6.4 Zusammenfassung 294

5	Klassen und Objekte (Teil 2) 295
5.1	Statische Klassenkomponenten 295
5.1.1	Zugriff auf statische Komponenten 297
5.1.2	Statische Klassenvariablen 298
5.1.3	Klassenspezifische Methoden 300
5.1.4	Statische Methoden in der Klasse Circle 302
5.1.5	Statische Klasseninitialisierer 306
5.1.6	Konstanten als Sonderform der Klassenvariablen 307
5.1.7	Standardmodule als Sonderform von Klassen 308
5.1.8	Der aktuelle Stand der Klasse Circle 309
5.1.9	Zusammenfassung 313
5.2	Ereignisse eines Objekts 314
5.2.1	Ergänzung eines Ereignisses in der Klasse Circle 315
5.2.2	Die Behandlung eines Ereignisses im Ereignisempfänger 317
5.2.3	Ereignisse mit Übergabeparametern 321
5.2.4	Die Handles-Klausel 323
5.2.5	Zusammenfassung 328
5.3	Strukturen – eine Sonderform der Klassen 329
5.3.1	Die Definition einer Struktur 329
5.3.2	Variablen vom Typ einer Struktur 332
5.3.3	Die anwendungsübergreifende Sichtbarkeit 335
5.3.4	Unterscheidungsmerkmale Klasse – Struktur 336
5.3.5	Zusammenfassung 338
5.4	Enum-Auflistungen 339
5.4.1	Wertzuweisung an Enum-Mitglieder 340
5.5	Referenz- und Wertetypen 341
5.5.1	Die Boxing-Konvertierung 343
5.5.2	Die Unboxing-Konvertierung 345
5.5.3	Zusammenfassung 346
5.6	Namensbereiche (Namespaces) 347
5.6.1	Zugriff auf Namespaces 349
5.6.2	Das Imports-Statement 350
5.6.3	Namespaces und Aliasnamen 353
5.6.4	Standardmäßig importierte Namespaces 353
5.6.5	Das Erstellen von Namespaces 355
5.6.6	Eingebettete Namespaces 356
5.6.7	Der aktuelle Stand der Klasse Circle 358
5.6.8	Zusammenfassung 362
6	Vererbung und Polymorphie 363
6.1	Die Grundlagen der Vererbung 363
6.1.1	Die Ableitung einer Klasse 365
6.1.2	Klassen, die nicht vererben können 369
6.2	Konstruktoren in Subklassen 370

6.2.1	Allgemeines	370
6.2.2	Die Konstruktoren der Klasse GraphicCircle	370
6.2.3	Der Zugriffsmodifizierer Protected	371
6.2.4	Konstruktorverkettung	372
6.2.5	Die Konstruktoren der Klasse GraphicCircle	377
6.2.6	Finalizer-Verkettung	377
6.2.7	Alle Zugriffsmodifizierer auf einen Blick	379
6.2.8	Zusammenfassung	381
6.3	Methodenergänzung in einer Subklasse	382
6.3.1	Die Windows-Testanwendung des CircleApplication-Projekts	383
6.3.2	Erläuterung des vorläufigen Programmcodes der Draw-Methode	387
6.3.3	Die endgültige Implementierung der Draw-Methode	392
6.4	Die Methoden einer abgeleiteten Klasse	394
6.4.1	Das Überschreiben von Basisklassenmethoden mit Overloads	394
6.4.2	Überladen der Basisklassenmethoden	398
6.4.3	Das Schlüsselwort Shadows	399
6.4.4	Die Vererbung statischer Mitglieder	401
6.5	Aggregation	403
6.5.1	Beispielanwendung Aggregation	406
6.6	Typumwandlung von Objektvariablen	411
6.6.1	Die implizite Typumwandlung von Objektreferenzen	411
6.6.2	Die explizite Typumwandlung von Objektreferenzen	413
6.6.3	Zusammenfassung	417
6.7	Abstrakte Klassen und Methoden	418
6.7.1	Objektbestimmung mittels Polymorphie	421
6.7.2	Polymorphe Methoden	425
6.7.3	Das Schlüsselwort "MyClass" für Sonderfälle	427
6.8	Die Erweiterung Klassenhierarchie der CircleApplication	430
6.8.1	Zusammenfassung	445
7	Schnittstellen und Delegates	447
7.1	Einführung in die Schnittstellen	447
7.1.1	Schnittstellendeklaration	448
7.1.2	Schnittstellenimplementierung	449
7.1.3	Abstrakte Klassen vs. Schnittstellen	458
7.1.4	Zusammenfassung	467
7.2	Delegates	468
7.2.1	Problembeschreibung	468
7.2.2	Ein erster Lösungsansatz	469
7.2.3	Einfache Delegates	471
7.2.4	Delegates als flexible Lösung	475
7.2.5	Multicast-Delegates	478
7.2.6	Allgemeine Anmerkungen zu Delegates	480
7.2.7	Delegates zur synchronen und asynchronen Benachrichtigung	483

- 7.2.8 Delegates und Events 488
- 7.2.9 Der Sonderfall eines Events in einer Struktur 494
- 7.2.10 EventArgs und EventHandler 494
- 7.2.11 Zusammenfassung 500

8 Fehlerbehandlung 501

- 8.1 Allgemeines 501
- 8.2 Laufzeitfehler behandeln 503
- 8.3 Unstrukturierte Fehlerbehandlung 506
 - 8.3.1 Das Err-Objekt 507
 - 8.3.2 Weitere Möglichkeiten 512
 - 8.3.3 Zusammenfassung 514
- 8.4 Strukturierte Fehlerbehandlung 515
 - 8.4.1 Try-Catch-Anweisung 515
 - 8.4.2 Die Finally-Anweisung 520
 - 8.4.3 Das Weiterleiten von Ausnahmen 521
 - 8.4.4 Die Hierarchie der Exceptions 528
 - 8.4.5 Benutzerdefinierte Exceptions 532
 - 8.4.6 Die benutzerdefinierte Filterung der Ausnahmebehandlung 536
 - 8.4.7 Zusammenfassung 538

9 Multithreading 539

- 9.1 Prozesse, Anwendungsdomänen und Threads 539
 - 9.1.1 Multitasking und virtueller Speicherraum 539
 - 9.1.2 Multithreading 542
 - 9.1.3 Threadzustände und Prioritäten 544
 - 9.1.4 Einsatz von mehreren Threads 545
 - 9.1.5 Anwendungsdomänen 546
 - 9.1.6 Zusammenfassung 550
- 9.2 Die Entwicklung einer Multithread-Anwendung 551
 - 9.2.1 Die Klasse System.Threading.Thread 555
 - 9.2.2 Threadpools 567
 - 9.2.3 Zusammenfassung 568
- 9.3 Die Synchronisation von Threads 569
 - 9.3.1 Unsynchronisierte Threads 569
 - 9.3.2 Der Monitor zur Synchronisation 571
- 9.4 Asynchrone Aufrufe 582
 - 9.4.1 Eine kleine Einführung 582
 - 9.4.2 Asynchroner Methodenaufruf 583
 - 9.4.3 Das Ende des asynchronen Aufrufs 586
 - 9.4.4 Asynchroner Aufruf mit Rückgabewerten 588
 - 9.4.5 Eine Klasse mit asynchronen Methodenaufrufen 591
 - 9.4.6 Zusammenfassung 596

10 Fundamentale Klassen des .NET-Frameworks 597

- 10.1 Allgemeines 597**
- 10.2 Die Klasse Object 599**
 - 10.2.1 Der Konstruktor 599
 - 10.2.2 Die Methoden 599
 - 10.2.3 Zusammenfassung 611
- 10.3 Die Klassen der Wertetypen 612**
 - 10.3.1 Die nativen Datentypen 613
 - 10.3.2 Allgemeine Informationen 616
 - 10.3.3 Ausgabeformatierung 618
 - 10.3.4 Zusammenfassung 622
- 10.4 Die String-Klasse 623**
 - 10.4.1 Das Erzeugen eines Strings 623
 - 10.4.2 Unveränderliche String-Objekte 625
 - 10.4.3 Die Eigenschaften eines String-Objekts 626
 - 10.4.4 Die Methoden der Klasse String 628
 - 10.4.5 Zusammenfassung der Klasse String 643
- 10.5 Die Klasse StringBuilder 645**
 - 10.5.1 Die Konstruktoren der Klasse StringBuilder 646
 - 10.5.2 Die Eigenschaften der Klasse StringBuilder 646
 - 10.5.3 Die Methoden der Klasse StringBuilder 647
 - 10.5.4 Allgemeine Anmerkungen 650
 - 10.5.5 Zusammenfassung 651
- 10.6 Die Klasse Char 652**
- 10.7 Die Klasse DateTime 653**
 - 10.7.1 Die Konstruktoren der Klasse DateTime 654
 - 10.7.2 Die Eigenschaften der Klasse DateTime 656
 - 10.7.3 Die Methoden der Klasse DateTime 658
- 10.8 Die Klasse TimeSpan 662**
 - 10.8.1 Zusammenfassung 666
- 10.9 Die Klasse Array 667**
 - 10.9.1 Das Erzeugen eines Array-Objekts 668
 - 10.9.2 Die Eigenschaften eines Array-Objekts 669
 - 10.9.3 Die Methoden der Klasse Array 670
 - 10.9.4 Zusammenfassung 681
- 10.10 Objektaufstellungen (Collections) 682**
 - 10.10.1 Die Schnittstelle IList 686
 - 10.10.2 Die Schnittstelle IDictionary 696
 - 10.10.3 Die Klassen Queue und Stack 708
 - 10.10.4 Objektaufstellungen im Überblick 711
 - 10.10.5 Zusammenfassung 713

11	Dateien und Streams 715
11.1	Allgemeine Einführung 715
11.2	Dateien und Verzeichnisse 717
11.2.1	Die Klasse File 717
11.2.2	Die Klasse System.IO.FileInfo 727
11.2.3	Die Klassen Directory und DirectoryInfo 730
11.2.4	Die Klasse System.IO.Path 735
11.2.5	Zusammenfassung 739
11.3	Die Stream-Klassen 740
11.3.1	Die Klasse Stream 741
11.3.2	Die abgeleiteten Stream-Klassen 743
11.3.3	Die Klasse FileStream 744
11.3.4	Zusammenfassung 754
11.4	Die Reader- und Writer-Klassen 755
11.4.1	Die Klassen TextReader und TextWriter 756
11.4.2	Die Klasse StreamWriter 757
11.4.3	Die Klasse StreamReader 762
11.4.4	Die Klassen StringWriter und StringReader 766
11.4.5	Die Klassen BinaryReader und BinaryWriter 767
11.4.6	Komplexe binäre Dateien 771
11.4.7	Zusammenfassung 779

12	Serialisierung und Attribute 781
12.1	Einführung in die Serialisierung 781
12.1.1	Einfache Serialisierung 782
12.1.2	Benutzergesteuerte Serialisierung 792
12.1.3	Beispiel einer benutzergesteuerten Serialisierung 795
12.1.4	Zusammenfassung 799
12.2	Attribute 800
12.2.1	Was ist ein Attribut? 800
12.2.2	Beispiel eines benutzerdefinierten Attributs 802
12.2.3	Fortsetzung des UserSerializationAttribut-Beispiels 809
12.2.4	Zusammenfassung 816

13	Assemblies und Verteilung 817
13.1	Die COM-Technologie 817
13.1.1	Allgemeines 817
13.1.2	Überblick über die COM-Technologie 819
13.1.3	Die Registrierung von COM-Komponenten 821
13.1.4	Die Problematik mit COM 824
13.2	Das Konzept der Assemblies 826
13.3	Der Inhalt einer Assembly 829

13.3.1	Die Struktur einer Assembly	829
13.3.2	Manifest und Metadaten	831
13.4	Einzeldatei-Assemblies	837
13.5	Mehrdateien-Assemblies	840
13.5.1	Manifest und MSIL-Code in einer Datei	841
13.5.2	Zusammenfassung der Schritte	850
13.5.3	Das Manifest in einer separaten Datei	850
13.6	Das Verteilen von Assemblies	853
13.6.1	Gemeinsame genutzte Assemblies	854
13.6.2	Das Erstellen einer globalen Assembly	857
13.6.3	Eine gemeinsam benutzte Assembly in einer Anwendung	864
13.6.4	Versionierung	864
13.7	Globale Assemblies im praktischen Einsatz	866
13.7.1	Die Entwicklung einer globalen Assembly	866
13.7.2	Der Endanwender von MeiersClientApp	868
13.7.3	Ein zweiter .NET-Anwendungsentwickler	868
13.7.4	Die überarbeitete Version einer globalen Assembly	868
13.7.5	Zwei gleichnamige, versionsverschiedene Komponenten auf einem Rechner	870
13.7.6	Konfiguration einer Anwendung	871
13.7.7	Die Attribute oldVersion und newVersion	874
13.7.8	Die Entwicklung von Konfigurationsdateien	875
13.7.9	Die Beispiele auf der CD	875

14 TCP/IP-Programmierung 877

14.1	Ein paar fundamentale Netzwerkgrundlagen	877
14.1.1	Das WinSock-API	877
14.1.2	TCP, UDP und IP	878
14.1.3	Das Client-Server-Prinzip	880
14.1.4	Zusammenfassung	882
14.2	Netzwerkprogrammierung mit dem .NET-Framework	883
14.2.1	Einfacher Verbindungsaufbau	883
14.2.2	Der Datenaustausch zwischen Client und Server	892
14.2.3	Kommunikation zwischen zwei TCP-Partnern	896
14.2.4	Zusammenfassung	903
14.3	E-Mails verschicken	904
14.3.1	Einleitung	904
14.3.2	Die Anforderungen an das Mail-Programm	905
14.3.3	Der Transport einer E-Mail	906
14.3.4	Struktur einer E-Mail-Nachricht	920
14.4	Ein »anderer« MailClient	925
14.4.1	Fazit	928

Über den Autor 931

Die Microrep GmbH – Grenzenlose Weiterbildung 933

Index 935

›