

# Inhalt

## Vorwort 23

<b>1</b>	<b>Das .NET-Konzept 25</b>
1.1	Ein Wort zu diesem Buch 25
1.2	Das Entwicklerdilemma 28
1.2.1	Das .NET-Konzept 29
1.2.2	Einarbeitungszeit 30
1.3	Das Sprachenkonzept 30
1.3.1	.NET-Anwendungsentwicklung 30
1.3.2	Die Common Language Specification 32
1.3.3	Das Common Type System 33
1.4	Das .NET-Framework 34
1.4.1	Die Common Language Runtime 34
1.4.2	Die .NET-Klassenbibliothek 35
1.5	Assemblies 38
<b>2</b>	<b>Die Entwicklungsumgebung 41</b>
2.1	Anmerkungen 41
2.2	Hard- und Softwareanforderungen 41
2.3	Die Installation 42
2.4	Die Entwicklungsumgebung von VS.NET 46
2.4.1	Die VisualC#-Vorlagetypen 47
2.4.2	Die Oberfläche der Entwicklungsumgebung 48
2.4.3	Der Codeeditor (Texteditor) 49
2.4.4	Der Projektmappen-Explorer 51
2.4.5	Die Klassenansicht 52
2.4.6	Das Eigenschaftsfenster 53
2.4.7	Die Werkzeugsammlung (Toolbox) 53
2.4.8	Der Server-Explorer 53
2.4.9	Dynamische Hilfe und Suchen 55
2.4.10	Das Fenster Inhalt 56
<b>3</b>	<b>Grundlagen der Sprache C# 57</b>
3.1	Konsolenanwendungen 57
3.1.1	Allgemeine Anmerkungen 57
3.1.2	Ein erstes Konsolenprogramm 57

<b>3.2</b>	<b>Grundlagen der C#-Syntax</b>	<b>60</b>
3.2.1	Das Semikolon	60
3.2.2	Anweisungs- und Gliederungsblöcke	61
3.2.3	Kommentare	62
3.2.4	Die Groß- und Kleinschreibung	64
3.2.5	Die Struktur einer Konsolenanwendung	65
<b>3.3</b>	<b>Variablen und Datentypen</b>	<b>68</b>
3.3.1	Variablendeclaration	68
3.3.2	Der Variablenbezeichner	69
3.3.3	Der Zugriff auf eine Variable	70
3.3.4	Ein- und Ausgabemethoden der Klasse »Console«	71
3.3.5	Die einfachen Datentypen	77
3.3.6	Typkonvertierung	86
<b>3.4</b>	<b>Operatoren</b>	<b>94</b>
3.4.1	Arithmetische Operatoren	95
3.4.2	Boolesche Operatoren	96
3.4.3	Bitweise Operatoren	100
3.4.4	Zuweisungsoperatoren	104
3.4.5	Stringverkettung	104
3.4.6	Sonstige Operatoren	105
3.4.7	Operator-Vorrangregeln	105
<b>3.5</b>	<b>Datenfelder (Arrays)</b>	<b>106</b>
3.5.1	Die Deklaration und Initialisierung eines Arrays	106
3.5.2	Der Zugriff auf die Array-Elemente	108
3.5.3	Speicherabbild eines Arrays	109
3.5.4	Mehrdimensionale Arrays	112
3.5.5	Festlegen der Array-Größe zur Laufzeit	113
3.5.6	Bestimmung der Array-Obergrenze	115
3.5.7	Die Gesamtanzahl der Array-Elemente	116
3.5.8	Verzweigte Arrays	116
<b>3.6</b>	<b>Kontrollstrukturen</b>	<b>118</b>
3.6.1	Die if-Anweisung	118
3.6.2	Das switch-Statement	124
<b>3.7</b>	<b>Programmschleifen</b>	<b>129</b>
3.7.1	Die for-Schleife	129
3.7.2	Die foreach-Schleife	143
3.7.3	Die do- und die while-Schleife	144
<b>4</b>	<b>Klassen und Objekte (Teil 1)</b>	<b>151</b>
<b>4.1</b>	<b>Einführung in die Objektorientierung</b>	<b>151</b>
4.1.1	Das objektorientierte Paradigma	153
4.1.2	Vorteile der objektorientierten Programmierung	155
4.1.3	Zusammenfassung	156
<b>4.2</b>	<b>Die Klassendefinition</b>	<b>157</b>

4.2.1	Zugriffsmodifizierer einer Klasse	160
4.2.2	Projektmappen-Explorer und Klassenansicht	164
4.3	<b>Die Deklaration von Objektvariablen</b>	166
4.3.1	Prüfen auf Initialisierung	168
4.3.2	Die Deklaration vom Typ »Object«	169
4.3.3	Zerstörung einer Objektreferenz	171
4.3.4	Mehrere Referenzen auf ein Objekt	172
4.3.5	Den Typ einer Objektreferenz ermitteln	173
4.3.6	Typvergleiche von Objektreferenzen	174
4.3.7	Referenzvergleiche	175
4.3.8	Das Klonen von Objekten	182
4.3.9	Zusammenfassung	185
4.4	<b>Objektmethoden</b>	186
4.4.1	Der Aufruf einer Methode	188
4.4.2	Rückgabewert einer Methode	191
4.4.3	Variablen in einer Methode	193
4.4.4	Zugriffsmodifizierer einer Methode	196
4.4.5	Die Parameterliste	197
4.4.6	Methodenüberladung	209
4.4.7	Aufruf überladener Methoden mit impliziter Konvertierung	212
4.4.8	Objektreferenzen als Übergabeparameter	213
4.4.9	Zusammenfassung	214
4.5	<b>Objekteigenschaften</b>	215
4.5.1	Datenkapselung	218
4.5.2	Ergänzung der Klasse »Circle«	222
4.5.3	Lese- und schreibgeschützte Eigenschaften	223
4.5.4	Konstanten	224
4.5.5	Methode oder Eigenschaft?	225
4.5.6	Die Trennung von Daten und Code	227
4.5.7	Der Zugriff auf private Daten	227
4.5.8	Zusammenfassung	228
4.6	<b>Konstruktoren</b>	229
4.6.1	Die Konstruktoren in der Klasse »Circle«	229
4.6.2	Die Konstruktoraufrufe	231
4.6.3	Definition von Konstruktoren	232
4.6.4	internal-Konstruktoren	233
4.6.5	private-Konstruktoren	233
4.6.6	Konstruktorkettung in einer Klasse	233
4.6.7	Zusammenfassung der Klasse »Circle«	235
4.6.8	Zusammenfassung	237
4.7	<b>Der Destruktor</b>	237
4.7.1	Die Arbeitsweise von Destruktoren	237
4.7.2	Der Garbage Collector	239
4.7.3	Die Finalize-Methode	240
4.7.4	Der Destruktor	240
4.7.5	Das Zerstören eines Objekts	241

4.7.6	Die Dispose-Methode	241
4.7.7	Der Garbage Collector in Aktion	245
4.7.8	Zusammenfassung	246
<b>4.8</b>	<b>Assistenten der Entwicklungsumgebung</b>	<b>247</b>
4.8.1	Der Methoden-Assistent	247
4.8.2	Der Eigenschafts-Assistent	248
4.8.3	Der Feld-Assistant	249

## **5 Klassen und Objekte (Teil 2)** 251

<b>5.1</b>	<b>Statische Klassenkomponenten</b>	<b>251</b>
5.1.1	Begrifflichkeiten	251
5.1.2	Die Realisierung eines Objektzählers	252
5.1.3	Zugriff auf statische Komponenten	253
5.1.4	Statische Klassenvariablen	254
5.1.5	Klassenspezifische Methoden	256
5.1.6	Statische Methoden in der Klasse »Circle«	257
5.1.7	Statische Klasseninitialisierer	261
5.1.8	Konstanten als Klassenvariablen	264
5.1.9	Der aktuelle Stand der Klasse »Circle«	264
5.1.10	Zusammenfassung	267
<b>5.2</b>	<b>Delegate</b>	<b>268</b>
5.2.1	Einführung in das Prinzip der Delegate	268
5.2.2	Zusammenfassung der Arbeitsschritte	272
<b>5.3</b>	<b>Ereignisse eines Objekts</b>	<b>273</b>
5.3.1	Ergänzung eines Ereignisses in einer Ereignisquelle	274
5.3.2	Die Behandlung eines Ereignisses im Ereignisempfänger	276
5.3.3	Ereignisse mit Überabeparameter	278
5.3.4	Ignorieren der Ereignisauslösung	280
5.3.5	Zusammenfassung	281
<b>5.4</b>	<b>Strukturen – eine Sonderform der Klassen</b>	<b>282</b>
5.4.1	Die Definition einer Struktur	282
5.4.2	Variablendeclaration vom Typ einer Struktur	285
5.4.3	Eine Struktur als Array	286
5.4.4	Die anwendungsübergreifende Sichtbarkeit	287
5.4.5	Weitere Merkmale einer Struktur	289
5.4.6	Änderung der Klasse »Circle«	291
5.4.7	Zusammenfassung	293
<b>5.5</b>	<b>Enumerationen (Aufzählungen)</b>	<b>293</b>
5.5.1	Wertzuweisung an enum-Mitglieder	294
5.5.2	Alle Mitglieder einer Aufzählung durchlaufen	295
<b>5.6</b>	<b>Referenz- und Wertetypen</b>	<b>297</b>
5.6.1	Die Boxing-Konvertierung	298
5.6.2	Die Unboxing-Konvertierung	301
5.6.3	Zusammenfassung	301

<b>5.7</b>	<b>Namensbereiche (Namespaces) 301</b>
5.7.1	Zugriff auf Namespaces 303
5.7.2	Die using-Direktive 304
5.7.3	Namespaces und Aliasnamen 306
5.7.4	Das Erstellen von Namespaces 307
5.7.5	Zusammenfassung 309
5.7.6	Der aktuelle Stand der Klasse »Circle« 309

## **6 Vererbung und Polymorphie 313**

<b>6.1</b>	<b>Einführung 313</b>
6.1.1	Basisklassen und abgeleitete Klassen 313
6.1.2	Die Ableitung einer Klasse 315
6.1.3	Parameter vom Typ einer Basisklasse 316
6.1.4	Klassen, die nicht vererben können 318
6.1.5	Zusammenfassung 318
<b>6.2</b>	<b>Konstruktoren in Subklassen 319</b>
6.2.1	Allgemeines 319
6.2.2	Die Konstruktoren der Klasse »GraphicCircle« 319
6.2.3	Der Zugriffsmodifizierer »protected« 320
6.2.4	Konstruktorverkettung 321
6.2.5	Destruktor-Verkettung 326
6.2.6	Zusammenfassung 328
<b>6.3</b>	<b>Objektspezifische Methoden einer Subklasse 328</b>
6.3.1	Die Windows-Testanwendung des »CircleApplication«-Projekts 329
6.3.2	Der vorläufige Programmcode der »Draw«-Methode 334
6.3.3	Die endgültige Implementierung der »Draw«-Methode 338
<b>6.4</b>	<b>Die Methoden einer abgeleiteten Klasse 338</b>
6.4.1	Das Verdecken von Basisklassenmethoden mit »new« 339
6.4.2	Der Gültigkeitsbereich eines überschreibenden Klassenmitglieds 341
6.4.3	Der Zugriff auf die überdeckte Methode in der Basisklasse 342
6.4.4	Überladen der Basisklassenmethoden 343
<b>6.5</b>	<b>Ereignisse in der Vererbung 344</b>
<b>6.6</b>	<b>Hat-eine-Beziehungen 347</b>
6.6.1	Weiterleitung einer internen Objektreferenz 349
6.6.2	Die Aggregation 350
6.6.3	Verschachtelte Typdefinition 356
6.6.4	Abschlussbetrachtungen 358
<b>6.7</b>	<b>Typumwandlung von Objektvariablen 359</b>
6.7.1	Die implizite Typumwandlung von Objektreferenzen 359
6.7.2	Die explizite Typumwandlung von Objektreferenzen 361
6.7.3	Zusammenfassung 365
<b>6.8</b>	<b>Abstrakte Klassen und Methoden 365</b>

<b>6.9</b>	<b>Polymorphie 368</b>
6.9.1	Das Aufrufproblem 369
6.9.2	Polymorphe Methoden 370
6.9.3	Inhomogene Mengen 371
6.9.4	Überdecken und Überschreiben geerbter Methoden 375
6.9.5	Überschreiben der Methode »System.Object.ToString« 376
6.9.6	Versiegelte Methoden 376
6.9.7	Zusammenfassung 377
<b>6.10</b>	<b>Die Erweiterung der Klassenhierarchie der »CircleApplication« 377</b>

<b>7</b>	<b>Fortgeschrittene Techniken 389</b>
<b>7.1</b>	<b>Operatorüberladung 389</b>
7.1.1	Einführung 389
7.1.2	Beispiel einer Operatorüberladung 389
7.1.3	Die Syntax der Operatorüberladung 390
7.1.4	Überladen von Gleichheitsoperatoren 392
7.1.5	Benutzerdefinierte Konvertierungen 393
7.1.6	true-/false-Operatoren 401
7.1.7	Zusammenfassung 403
<b>7.2</b>	<b>Indexer 403</b>
7.2.1	Indexer eines beliebigen Typs 405
7.2.2	Überladen von Indexern 406
7.2.3	Indexer als Übergabeargumente 407
7.2.4	Parameterbehaftete Eigenschaften 407
7.2.5	Der Indexer-Assistent 411
7.2.6	Zusammenfassung 412
<b>7.3</b>	<b>Fortgeschrittene Delegat-Techniken 412</b>
7.3.1	Eine Beispielanwendung 412
7.3.2	Multicast-Delegate 417
7.3.3	Delegate zur synchronen und asynchronen Benachrichtigung 420
7.3.4	Der Unterschied zwischen einem Ereignis und einem Delegaten 424
7.3.5	Zusammenfassung 425
<b>7.4</b>	<b>XML-Dokumentation 425</b>
7.4.1	XML-Dokumente 426
7.4.2	Validierung von XML-Dokumenten 430
7.4.3	XML-Dateien in Visual Studio .NET erstellen 433
7.4.4	Generieren der XML-Dokumentation 436
7.4.5	Die Bearbeitung der XML-Dokumentation 439
<b>8</b>	<b>Schnittstellen 443</b>
<b>8.1</b>	<b>Einführung in die Schnittstellen 443</b>
8.1.1	Schnittstellendeklaration 443
8.1.2	Schnittstellenimplementierung 444
8.1.3	Abstrakte Klassen vs. Schnittstellen 452
8.1.4	Zusammenfassung 459

<b>9</b>	<b>Fehlerbehandlung und Debugging</b>	<b>461</b>
9.1	Die Behandlung von Laufzeitfehlern (Exceptions)	461
9.1.1	Laufzeitfehler erkennen	462
9.1.2	Die Behandlung von Exceptions	465
9.1.3	Die try...catch-Anweisung	465
9.1.4	Behandlung mehrerer Exceptions	468
9.1.5	Die finally-Anweisung	470
9.1.6	Das Weiterleiten von Ausnahmen	471
9.1.7	Die Hierarchie der Exceptions	478
9.1.8	Die Klassenhierarchie und die catch-Klauseln	479
9.1.9	Benutzerdefinierte Exceptions	481
9.2	Debugging	485
9.2.1	Einführung	485
9.2.2	Die Klasse »Debug«	486
9.2.3	Die Klasse »Trace«	490
9.2.4	Debuggen im Haltemodus	491
9.2.5	Das Befehlsfenster	495
9.2.6	Weitere Alternativen, um Variableninhalte zu prüfen	497
9.2.7	Das Fenster »Ausnahmen«	498
9.2.8	Das Fenster »Aufrufliste«	499
<b>10</b>	<b>Fundamentale Klassen des .NET-Frameworks</b>	<b>501</b>
10.1	Allgemeines	501
10.2	Die Klasse Object	502
10.2.1	Der Konstruktor	503
10.2.2	Die Methoden	503
10.2.3	Zusammenfassung	512
10.3	Die Klassen der Wertetypen	513
10.3.1	Die nativen Datentypen	514
10.3.2	Allgemeine Informationen	518
10.3.3	Ausgabeformatierung	519
10.3.4	Zusammenfassung	522
10.4	Die »String«-Klasse	522
10.4.1	Das Erzeugen eines Strings	523
10.4.2	Unveränderliche String-Objekte	524
10.4.3	Die Eigenschaften	525
10.4.4	Die Methoden der Klasse »String«	527
10.4.5	Zusammenfassung der Klasse String	540
10.5	Die Klasse »StringBuilder«	541
10.5.1	Die Konstruktoren der Klasse StringBuilder	542
10.5.2	Die Eigenschaften der Klasse StringBuilder	543
10.5.3	Die Methoden der Klasse StringBuilder	544
10.5.4	Allgemeine Anmerkungen	546
10.5.5	Zusammenfassung	546

<b>10.6</b>	<b>Die Klasse »Char«</b>	<b>547</b>
<b>10.7</b>	<b>Die Klasse »DateTime«</b>	<b>547</b>
10.7.1	Die Zeitspanne »Tick«	547
10.7.2	Die Konstruktoren der Klasse »DateTime«	548
10.7.3	Die Eigenschaften der Klasse »DateTime«	549
10.7.4	Die Methoden der Klasse DateTime	551
<b>10.8</b>	<b>Die Klasse »TimeSpan«</b>	<b>555</b>
10.8.1	Zusammenfassung	558
<b>10.9</b>	<b>Die Klasse »Array«</b>	<b>558</b>
10.9.1	Das Erzeugen eines Array-Objekts	559
10.9.2	Die Eigenschaften eines Array-Objekts	560
10.9.3	Die Methoden der Klasse Array	561
10.9.4	Zusammenfassung	571
<b>10.10</b>	<b>Objektauflistungen (Collections)</b>	<b>571</b>
10.10.1	Die fundamentalen Schnittstellen der Auflistungsklassen	572
10.10.2	Klassen, die die Schnittstelle IList implementieren	575
10.10.3	Klassen, die die Schnittstelle IDictionary implementieren	584
10.10.4	Die Klassen Queue und Stack	594
10.10.5	Objektauflistungen im Überblick	597
10.10.6	Zusammenfassung	598

## 11 Multithreading 601

<b>11.1</b>	<b>Prozesse, Anwendungsdomänen und Threads</b>	<b>601</b>
11.1.1	Multitasking und virtueller Speicherraum	601
11.1.2	Multithreading	603
11.1.3	Thread-Zustände und Prioritäten	605
11.1.4	Einsatz von mehreren Threads	606
11.1.5	Anwendungsdomänen	607
<b>11.2</b>	<b>Die Entwicklung einer Multithread-Anwendung</b>	<b>609</b>
11.2.1	Die Klasse System.Threading.Thread	613
11.2.2	Threadpools	626
<b>11.3</b>	<b>Die Synchronisation von Threads</b>	<b>626</b>
11.3.1	Unsynchronisierte Threads	626
11.3.2	Der Monitor zur Synchronisation	628
<b>11.4</b>	<b>Asynchrone Aufrufe</b>	<b>638</b>
11.4.1	Eine kleine Einführung	638
11.4.2	Asynchroner Methodenaufruf	639
11.4.3	Das Ende des asynchronen Aufrufs	642
11.4.4	Asynchroner Aufruf mit Rückgabewerten	644
11.4.5	Eine Klasse mit asynchronen Methodenaufrufen	647

<b>12</b>	<b>Arbeiten mit Dateien und Streams</b>	<b>653</b>
12.1	Allgemeine Einführung	653
12.2	Dateien und Verzeichnisse	654
12.2.1	Die Klasse File	654
12.2.2	Das Behandeln von Ausnahmen bei E/A-Operationen	654
12.2.3	Methoden der Klasse File	655
12.2.4	Die Klasse »FileInfo«	664
12.2.5	Die Klassen »Directory« und »DirectoryInfo«	666
12.2.6	Die Klasse »Path«	671
12.3	Die XxxStream-Klassen	674
12.3.1	Die Klasse Stream	675
12.3.2	Die abgeleiteten Stream-Klassen	678
12.3.3	Die Klasse »FileStream«	679
12.4	Weitere Reader- und Writer-Klassen	687
12.4.1	Die Klassen TextReader und TextWriter	689
12.4.2	Die Klasse »StreamWriter«	690
12.4.3	Die Klasse »StreamReader«	694
12.4.4	Die Klassen »StringWriter« und »StringReader«	698
12.4.5	Die Klassen »BinaryReader« und »BinaryWriter«	698
12.4.6	Komplexe binäre Dateien	703
<b>13</b>	<b>Serialisierung und Attribute</b>	<b>711</b>
13.1	Einführung in die Serialisierung	711
13.1.1	Einfache Serialisierung	712
13.1.2	Serialisierung im XML-Format	722
13.1.3	Die Wahl des Serialisierungsformats	724
13.1.4	Benutzergesteuerte Serialisierung	724
13.1.5	Beispiel einer benutzergesteuerten Serialisierung	727
13.2	Attribute	730
13.2.1	Was ist ein Attribut?	730
13.2.2	Benutzerdefinierte Attribute	732
13.2.3	Benutzerdefinierte Serialisierung	739
<b>14</b>	<b>Einführung in die Programmierung grafischer Benutzeroberflächen</b>	<b>747</b>
14.1	Einführung	747
14.1.1	Die Entwicklungsumgebung	747
14.1.2	Eine erste Windows-Anwendung	748
14.2	Der elementare Code einer Windows-Anwendung	756
14.2.1	Die Methode »Application.Run«	758
14.2.2	Der letzte Schliff	759
14.2.3	Die Nachrichtenschleife	760
14.2.4	Eigenschaften der Form ändern	762
14.2.5	Steuerelemente in einem Formular	764

<b>14.3</b>	<b>Ereignisse grafischer Windows-Komponenten</b>	<b>771</b>
14.3.1	Ereignisse einer grafischen Komponente	771
14.3.2	Ereignisse mit Argumentenliste	773
14.3.3	Die Entwicklungsumgebung und gemeinsame Ereignishandler	776
14.3.4	Ereignisbehandlung mit den OnXxx-Methoden	777
<b>14.4</b>	<b>Die Klasse »Control«</b>	<b>779</b>
<b>14.5</b>	<b>Die Klasse »Application«</b>	<b>783</b>
<b>15</b>	<b>Die Klasse »Form«</b>	<b>785</b>
<b>15.1</b>	<b>Formulare und Steuerelemente</b>	<b>785</b>
<b>15.2</b>	<b>Anwendungstypen</b>	<b>785</b>
<b>15.3</b>	<b>Die Basisklassen des Typs Form</b>	<b>786</b>
15.3.1	Die Klasse »ScrollableControl«	786
15.3.2	Die Klasse »ContainerControl«	787
<b>15.4</b>	<b>Eigenschaften der Klasse »Form«</b>	<b>788</b>
15.4.1	Eigenschaften des Designmodus	788
15.4.2	Eigenschaften zur Darstellung einer Form	789
15.4.3	Farbeinstellungen mit dem Typ »Color«	792
15.4.4	Schriftart mit dem Typ »Font«	793
15.4.5	Die Größe eines Formulars	799
15.4.6	Die Position eines Fensters	800
15.4.7	Die Arbeitsfläche des Fensters (der Clientbereich)	801
<b>15.5</b>	<b>Ereignisse eines Formulars</b>	<b>802</b>
15.5.1	Ereignisse beim Erzeugen eines Fenster	802
15.5.2	Ereignisreihenfolgen	803
15.5.3	Ereignisse beim Schließen eines Fensters	803
<b>15.6</b>	<b>Anwendungen mit mehreren Formularen</b>	<b>805</b>
15.6.1	Formulare laden, anzeigen, verstecken und schließen	805
15.6.2	Das Problem mit »Application.Run«	806
15.6.3	Formulare in einem eigenen Thread laufen lassen	809
<b>15.7</b>	<b>Modale Dialogfenster</b>	<b>810</b>
15.7.1	Beispiel eines Dialogfensters	811
15.7.2	Das aufrufende Formular	813
15.7.3	Die DialogResult-Eigenschaft eines Buttons	815
15.7.4	Die Weitergabe von Resultaten aus einem Dialog	816
15.7.5	Die »Übernehmen«-Schaltfläche	817
15.7.6	Der Feinschliff des modalen Dialogs	819
<b>15.8</b>	<b>Das Meldungsfenster</b>	<b>823</b>
15.8.1	Die Methode MessageBox.Show	824
15.8.2	Der Rückgabewert der Meldungsfenster	826
<b>15.9</b>	<b>Das Speichern von Formulardaten in der Registry</b>	<b>827</b>
15.9.1	Die Klassen »Registry« und »RegistKey«	827
15.9.2	Das Beispiel zur Speicherung in der Registrierung	831

<b>16</b>	<b>Elementare Steuerelemente 833</b>
16.1	<b>Gemeinsame Eigenschaften, Methoden und Ereignisse 833</b>
16.1.1	Größe und Position 833
16.1.2	Die Sichtbarkeit und der Aktivierungszustand 834
16.1.3	Die Eigenschaft »Text« 834
16.1.4	Die Farbeigenschaften 835
16.1.5	Grafische Komponenten 835
16.1.6	Die »Modifiers«-Eigenschaft 835
16.1.7	Die Eigenschaft »Tag« 836
16.1.8	Das Verankern von Steuerelementen 836
16.1.9	Die z-Reihenfolge 838
16.1.10	Die Tabulatorreihenfolge 839
16.1.11	Die Ereignisse beim Fokuswechsel 840
16.1.12	Maus- und Tastaturereignisse 841
16.2	<b>Schaltflächen (»Button«-Klasse) 842</b>
16.2.1	Die Eigenschaften einer Schaltfläche 843
16.2.2	Die Implementierung der Ereignishandler 848
16.3	<b>Die »CheckBox«-Klasse (Auswahlkästchen) 849</b>
16.3.1	Eigenschaften eines Kontrollkästchens 851
16.4	<b>Die Klasse »RadioButton« (Optionsschaltflächen) 853</b>
16.4.1	Die »GroupBox« als übergeordneter Container 856
16.5	<b>Texteingabefelder mit der Klasse »TextBox« 859</b>
16.5.1	Einzeilige Textboxen 859
16.5.2	Mehrzeilige Eingabefelder 867
16.6	<b>Das Steuerelement »RichTextBox« 870</b>
16.7	<b>Beschriftungen mit dem Steuerelement »Label« 872</b>
16.8	<b>Das »LinkLabel«-Steuerelement 873</b>
16.9	<b>Die Anzeige eines Quickinfotexts 875</b>
16.9.1	Methoden des ToolTip-Objekts 875
16.9.2	Eigenschaften eines ToolTip-Objekts 876
16.9.3	ToolTip-Texte im Windows Forms-Designer 877
16.10	<b>Das »ListBox«-Steuerelement 878</b>
16.10.1	Die Auflistung »ListBox.ObjectCollection« 879
16.10.2	Eigenschaften zur Darstellung einer Listbox 882
16.10.3	Der Zugriff auf Listenelemente 883
16.11	<b>Die Klasse »CheckedListBox« 890</b>
16.12	<b>Das Kombinationslistenfeld (ComboBox) 891</b>
<b>17</b>	<b>Standarddialoge 895</b>
17.1	<b>Einführung in die Standarddialoge 895</b>
17.2	<b>Die Klasse OpenFileDialog 896</b>
17.2.1	Die Beschriftung der Titelleiste 897
17.2.2	Festlegen eines Standardverzeichnisses 897

17.2.3	Die Klasse »Environment«	898
17.2.4	Einen Dateifilter setzen	899
17.2.5	Die ausgewählte Datei	901
17.2.6	Mehrfachauswahl von Dateien	901
17.2.7	Zusammenfassung der Eigenschaften und Methoden	902
17.2.8	Das Beispiel »Texteditor_1«	903
17.3	Die Klasse »SaveFileDialog«	905
17.3.1	»Speichern«-Implementierung am Beispiel Texteditor_2	907
17.4	Die Klasse »ColorDialog«	912
17.5	Die Klasse »FontDialog«	914
<b>18</b>	<b>Menü, Symbol- und Statusleiste</b>	<b>917</b>
18.1	Die Klasse »ImageList«	917
18.1.1	Instanzieren einer Bilderliste	917
18.1.2	Die Eigenschaften der Klasse »ImageList«	917
18.1.3	Das ImageList-Steuerelement im Windows Forms-Designer	919
18.2	Erstellen eines Menüs	920
18.2.1	Die Hierarchie der Menüklassen	921
18.2.2	Das Erzeugen eines Hauptmenüs	921
18.2.3	Menüelemente mit »MenuItem«	924
18.2.4	Die Eigenschaften von »MenuItem«	930
18.2.5	Kontextmenüs	937
18.2.6	Menüs mit dem Windows Forms-Designer	941
18.2.7	Kontextmenüs mit dem Windows Forms-Designer	942
18.2.8	Symbole in einem Menü	943
18.2.9	Beispiel einer Menüleiste mit Symbolen	945
18.3	Die Symbolleiste	950
18.3.1	Eine einfache Symbolleiste	950
18.3.2	Eine übliche Symbolleiste mit Schaltflächenbildern	951
18.3.3	Buttons mit dem Windows Forms-Designer hinzufügen	954
18.3.4	Die Darstellung einer Symbolleiste beeinflussen	954
18.3.5	Die Ereignisse der Symbolleiste	956
18.3.6	Weitere Gestaltungsmöglichkeiten	959
18.4	Die Statusleiste	963
18.4.1	Eine einfache Statusleiste	963
18.4.2	Eine »vollständige« Statusleiste	964
18.4.3	Das StatusBarPanel-Objekt	965
<b>19</b>	<b>Weitere Steuerelemente</b>	<b>969</b>
19.1	Bildlaufleisten	969
19.2	Schieberegler mit der Klasse »TrackBar«	971
19.3	Das ProgressBar-Steuerelement	974

<b>19.4</b>	<b>Drehfeld-Steuerelemente (UpDown-Steuerelemente)</b>	<b>976</b>
19.4.1	Das NumericUpDown-Steuerelement	976
19.4.2	Das DomainUpDown-Steuerelement	978
<b>19.5</b>	<b>Das Timer-Steuerelement</b>	<b>979</b>
<b>19.6</b>	<b>Die Klasse »ErrorProvider«</b>	<b>981</b>
<b>19.7</b>	<b>Steuerelemente für die Datums- und Zeitangabe</b>	<b>983</b>
19.7.1	Das Steuerelement »MonthCalendar«	983
19.7.2	Das Steuerelement »DateTimePicker«	986
<b>19.8</b>	<b>Das Panel-Steuerelement</b>	<b>987</b>
<b>19.9</b>	<b>Das Steuerelement »NotifyIcon«</b>	<b>988</b>
<b>19.10</b>	<b>Die »PictureBox«</b>	<b>991</b>
<b>19.11</b>	<b>Das Registerkarten-Steuerelement</b>	<b>991</b>
19.11.1	Die Klasse »TabControl«	991
19.11.2	Objekte vom Typ »TabPage«	995
19.11.3	Beispielprogramm	996
<b>19.12</b>	<b>Das TreeView-Steuerelement</b>	<b>997</b>
19.12.1	Allgemeine Gestaltung	997
19.12.2	Bilder im TreeView-Steuerelement	1002
19.12.3	Beispiel mit Unterstützung der Entwicklungsumgebung	1003
19.12.4	Die Ereignisse des TreeView-Steuerelements	1004
19.12.5	Weitere Eigenschaften und Methoden des TreeView-Objekts	1006
19.12.6	Weitere Eigenschaften und Methoden des TreeNode-Objekts	1007
19.12.7	Beispiel zum Einlesen der Verzeichnisstruktur	1008
<b>19.13</b>	<b>Das ListView-Steuerelement</b>	<b>1012</b>
19.13.1	Die Klassen des ListView-Steuerelements	1012
19.13.2	Die Eigenschaften des ListView-Steuerelements	1013
19.13.3	Das Listenelement »ListViewItem«	1015
19.13.4	Das Element »ListViewSubItem«	1017
19.13.5	Der Typ »ColumnHeader«	1018
19.13.6	Die Ereignisse des ListView-Steuerelements	1018
19.13.7	Beispielanwendung	1020
<b>19.14</b>	<b>Das »Splitter«-Steuerelement</b>	<b>1025</b>
19.14.1	Die Eigenschaft »Control.Dock«	1025
19.14.2	Das Verhalten eines Splitters richtig festlegen	1028
19.14.3	Eigenschaften des Splitter-Steuerelements	1028
19.14.4	Ereignisse	1029
19.14.5	Ergänzung des Beispiels »SimpleExplorer«	1030
<b>20</b>	<b>Tastatur- und Mausereignisse</b>	<b>1031</b>
<b>20.1</b>	<b>Die Tastaturschnittstelle</b>	<b>1031</b>
20.1.1	Allgemeines	1031
20.1.2	Die Tastaturereignisse im Überblick	1031
20.1.3	»KeyDown« und »KeyUp« für den Tastendruck	1033

20.1.4	Die Aufzählung »Keys«	1034
20.1.5	Die Eigenschaften »KeyCode«, »KeyData« und »KeyModifiers«	1036
20.1.6	Eine ergänzende Alternative zu der Eigenschaft »Modifiers«	1038
20.1.7	Das Ereignis »KeyPress«	1039
20.1.8	Die »Handled«-Eigenschaft	1040
20.1.9	Die Tastaturereignisse der Form	1042
20.1.10	Senden von Tastatureingaben	1043
<b>20.2</b>	<b>Die Mausschnittstelle</b>	<b>1044</b>
20.2.1	Die Ereignisse »MouseDown«, »MouseMove« und »MouseUp«	1045
20.2.2	Das Ereignis »MouseWheel«	1048
20.2.3	Weitere Mausereignisse	1049
20.2.4	Die Ereignisse »Click« und »DoubleClick«	1049
20.2.5	Eigenschaften der Maus	1049

<b>21</b>	<b>MDI-Anwendungen</b>	<b>1053</b>
21.1	Einführung	1053
21.2	Das Hauptfenster	1053
21.3	Die Subfenster	1054
21.4	<b>Zugriff auf die Subfenster</b>	<b>1055</b>
21.4.1	Zugriff auf alle Subfenster	1055
21.4.2	Zugriff auf das aktive Subfenster	1056
21.4.3	Zugriff auf Subfenster eines bestimmten Typs	1057
21.4.4	Das aktive Steuerelement eines Subfensters manipulieren	1058
21.5	<b>Das Menü in einer MDI-Anwendung</b>	<b>1058</b>
21.5.1	Subfenster anordnen	1058
21.5.2	Subfenster mit eigenen Menüs	1059
21.5.3	Die Liste der geöffneten untergeordneten Fenster	1060
21.6	<b>Beispiel einer MDI-Anwendung</b>	<b>1060</b>
<b>22</b>	<b>Die grafische Schnittstelle GDI+</b>	<b>1063</b>
22.1	<b>Die Namespaces der GDI+-Schnittstelle</b>	<b>1063</b>
22.2	<b>Das Graphics-Objekt</b>	<b>1064</b>
22.2.1	Das Paint-Ereignis	1064
22.2.2	Einfache Grafiken zeichnen	1066
22.2.3	Die Methode »CreateGraphics«	1069
22.2.4	Zerstören von grafischen Objekten (Dispose)	1070
22.2.5	Das Koordinatensystem	1071
22.2.6	Die Maßeinheit des Koordinatensystems	1071
22.2.7	Skalierung der Grafikausgabe	1072
22.2.8	Festlegen eines anderen Ursprungspunkts	1072
22.2.9	Beispielprogramm einer Transformation	1073
22.2.10	Die grafischen Methoden des »Graphics«-Objekts	1075
22.2.11	Eine Linie zeichnen	1076

22.2.12	Mehrere Linien zeichnen	1077
22.2.13	Rechtecke zeichnen	1077
22.2.14	Punkte zeichnen	1079
22.2.15	Polygone zeichnen	1080
22.2.16	Kreise und Ellipsen	1080
22.2.17	Kurvenzüge	1081
22.2.18	Bezierkurven	1082
<b>22.3</b>	<b>Elementare Klassen für grafische Operationen</b>	<b>1083</b>
22.3.1	Die Klasse »Brush«	1083
22.3.2	Die Klasse »Pen«	1096
22.3.3	Farbeinstellungen mit »Color«	1103
<b>22.4</b>	<b>Die Schriftdarstellung</b>	<b>1104</b>
22.4.1	Allgemeines	1104
22.4.2	Die Klassen »Font« und »FontFamily«	1104
22.4.3	Der Schriftstil mit »FontStyle«	1106
22.4.4	Die Größe einer Schrift	1107
22.4.5	Die grafische Ausgabe einer Zeichenfolge	1109
22.4.6	Die Abmessungen mit »MeasureString« ermitteln	1111
22.4.7	Die Klasse »StringFormat«	1111
<b>22.5</b>	<b>Bilddateien</b>	<b>1118</b>
22.5.1	Raster- und Vektorgrafiken	1118
22.5.2	Bilder und Grafiken der .NET-Klassenbibliothek	1119
22.5.3	Die Bitmap-Dateiformate	1120
22.5.4	Bilder mit der Klasse »Image«	1122
22.5.5	Bitmaps	1126

<b>23</b>	<b>Das Drucken (Printing)</b>	<b>1137</b>
23.1	Ein einführender Überblick	1137
<b>23.2</b>	<b>Der Druckauftrag »PrintDocument«</b>	<b>1137</b>
23.2.1	Die Ereignisse »BeginPrint« und »EndPrint«	1139
23.2.2	Das Ereignis »QueryPageSettings«	1139
23.2.3	Das Ereignis »PrintPage«	1139
23.2.4	Das Graphics-Objekt	1140
23.2.5	Die Seitenränder	1140
23.2.6	Das Beenden des Druckauftrags	1143
23.2.7	Programmbeispiel	1143
<b>23.3</b>	<b>Selteneinstellungen mit »PageSettings«</b>	<b>1145</b>
<b>23.4</b>	<b>Der Drucker als »PrinterSettings«-Objekt</b>	<b>1147</b>
23.4.1	Mehrere installierte Drucker	1147
23.4.2	Wichtige Eigenschaften des »PrinterSettings«-Objekts	1147
<b>23.5</b>	<b>Die Steuerelemente zum Drucken</b>	<b>1148</b>
23.5.1	Das »PrintDocument«-Steuerelement	1148
23.5.2	Das »PrintDialog«-Steuerelement	1149
23.5.3	Das »PageSetupDialog«-Steuerelement	1150

23.5.4	Keine Software ohne Bugs	1151
23.5.5	Das »PrintPreviewDialog«-Steuerelement	1152
23.5.6	Das »PrintPreviewControl«-Steuerelement	1153
<b>23.6</b>	<b>Drucken von Grafiken</b>	<b>1154</b>
23.6.1	Das Problem der verschiedenen Maßeinheiten	1154
23.6.2	Festlegung der Einheiten und Skalierung	1155
23.6.3	Beispielprogramm zum Drucken einer Grafik	1156
<b>23.7</b>	<b>Ausdruck von mehrseitigem Text</b>	<b>1158</b>

## **24 Zwischenablage/Drag & Drop** 1167

<b>24.1</b>	<b>Die Zwischenablage</b>	<b>1167</b>
24.1.1	Das Speichern von Daten in der Zwischenablage	1167
24.1.2	Das Abrufen von Daten aus der Zwischenablage	1167
24.1.3	Beispielprogramm	1168
<b>24.2</b>	<b>Drag &amp; Drop-Operationen</b>	<b>1171</b>
24.2.1	Allgemeines	1171
24.2.2	Die Ereignisse einer Drag & Drop-Aktion	1172
24.2.3	Das Einleiten einer Drag & Drop-Operation	1174

## **25 Assemblies und Verteilung** 1177

<b>25.1</b>	<b>Die COM-Technologie</b>	<b>1177</b>
25.1.1	Allgemeines	1177
25.1.2	Überblick über die COM-Technologie	1179
25.1.3	Die Registrierung von COM-Komponenten	1180
25.1.4	Die Problematik mit COM	1182
<b>25.2</b>	<b>Das Konzept der Assemblies</b>	<b>1184</b>
<b>25.3</b>	<b>Der Inhalt einer Assembly</b>	<b>1186</b>
25.3.1	Die Struktur einer Assembly	1186
25.3.2	Manifest und Metadaten	1188
<b>25.4</b>	<b>Einzeldatei-Assemblies</b>	<b>1193</b>
<b>25.5</b>	<b>Mehrdateien-Assemblies</b>	<b>1195</b>
25.5.1	Manifest und MSIL-Code in einer Datei	1196
25.5.2	Zusammenfassung	1201
25.5.3	Das Manifest in einer separaten Datei	1201
<b>25.6</b>	<b>Das Verteilen von Assemblies</b>	<b>1202</b>
25.6.1	Gemeinsame genutzte Assemblies	1203
25.6.2	Das Erstellen einer globalen Assembly	1205
25.6.3	Ein Installationsprogramm erstellen	1208
25.6.4	Eine gemeinsam benutzte Assembly in einer Anwendung	1211
25.6.5	Versionierung	1212

- 25.7 Globale Assemblies im praktischen Einsatz 1213**
  - 25.7.1 Die Entwicklung einer globalen Assembly 1213
  - 25.7.2 Der Endanwender von MeiersClientApp 1214
  - 25.7.3 Ein zweiter .NET-Anwendungsentwickler 1215
  - 25.7.4 Die überarbeitete Version einer globalen Assembly 1215
  - 25.7.5 Zwei gleichnamige, versionsverschiedene Komponenten auf einem Rechner 1216
  - 25.7.6 Konfiguration einer Anwendung 1217
  - 25.7.7 Die Attribute »oldVersion« und »newVersion« 1220
  - 25.7.8 Die Entwicklung von Konfigurationsdateien 1220
  - 25.7.9 Die Beispiele auf der CD 1220
- 25.8 Installation einer Windows-Anwendung 1221**
  - 25.8.1 Der Setup-Assistent 1221
  - 25.8.2 Installation des .NET-Frameworks 1223
  - 25.8.3 Die Installationsdateien 1223

**Index 1225**