

# Inhalt

<b>1</b>	<b>C# und das .NET-Framework</b>	1
1.1	Ähnlichkeiten zwischen C# und Java	1
1.2	Unterschiede zwischen C# und Java	3
1.3	Das .NET-Framework	4
1.4	Übungsaufgaben	10
<b>2</b>	<b>Erste Schritte</b>	11
2.1	Hello World	11
2.2	Gliederung von Programmen	12
2.3	Symbole	13
2.4	Übungsaufgaben	16
<b>3</b>	<b>Typen</b>	17
3.1	Einfache Typen	18
3.2	Enumerationen	19
3.3	Arrays	20
3.4	Strings	23
3.5	Structs	24
3.6	Klassen	25
3.7	object	26
3.8	Boxing und Unboxing	27
3.9	Übungsaufgaben	28
<b>4</b>	<b>Ausdrücke</b>	31
4.1	Arithmetische Ausdrücke	31
4.2	Vergleichsausdrücke	32
4.3	Boolesche Ausdrücke	33
4.4	Bit-Ausdrücke	33
4.5	Shift-Ausdrücke	33
4.6	Überlaufprüfung	34
4.7	typeof	34
4.8	sizeof	35
4.9	Übungsaufgaben	35

<b>5</b>	<b>Deklarationen</b>	37
5.1	Deklarationen in Namensräumen	38
5.2	Deklarationen in Klassen, Structs und Interfaces	39
5.3	Deklarationen in Enumerationstypen	40
5.4	Deklarationen in Blöcken	40
5.5	Übungsaufgaben	42
<b>6</b>	<b>Anweisungen</b>	43
6.1	Leeranweisung	43
6.2	Zuweisung	43
6.3	Methodenaufruf	44
6.4	if-Anweisung	44
6.5	switch-Anweisung	45
6.6	while-Anweisung	46
6.7	do-while-Anweisung	46
6.8	for-Anweisung	46
6.9	foreach-Anweisung	47
6.10	break- und continue-Anweisungen	48
6.11	goto-Anweisung	48
6.12	return-Anweisung	49
6.13	Übungsaufgaben	50
<b>7</b>	<b>Ein-/Ausgabe</b>	51
7.1	Ausgabe auf den Bildschirm	51
7.2	Formatierte Ausgabe	51
7.3	Ausgabe auf eine Datei	54
7.4	Eingabe von der Tastatur	55
7.5	Eingabe von einer Datei	55
7.6	Lesen der Kommandozeilenparameter	56
7.7	Übungsaufgaben	57
<b>8</b>	<b>Klassen und Structs</b>	59
8.1	Sichtbarkeitsattribute	60
8.2	Felder	62
8.3	Methoden	63
8.4	Konstruktoren	70
8.5	Destruktoren	72
8.6	Properties	73
8.7	Indexer	76
8.8	Überladene Operatoren	78
8.9	Kurzform für Methoden	81
8.10	Geschachtelte Typen	82

8.11	Partielle Typen	83
8.12	Partielle Methoden	84
8.13	Statische Klassen	85
8.14	Unterschiede zu Java	85
8.15	Übungsaufgaben	86
<b>9</b>	<b>Vererbung</b>	<b>89</b>
9.1	Deklaration von Unterklassen	89
9.2	Kompatibilität zwischen Klassen	91
9.3	Überschreiben und Verdecken von Elementen	92
9.4	Dynamische Bindung	95
9.5	Konstruktoren in Ober- und Unterklasse	98
9.6	Abstrakte Klassen	99
9.7	Versiegelte Klassen	100
9.8	Die Klasse Object	101
9.9	Übungsaufgaben	103
<b>10</b>	<b>Interfaces</b>	<b>105</b>
10.1	Deklaration und Verwendung von Interfaces	105
10.2	Operationen auf Interfaces	107
10.3	Erweiterung von Interfaces	108
10.4	Namenskonflikte	109
10.5	Interface IDisposable	110
10.6	Übungsaufgaben	111
<b>11</b>	<b>Delegates und Events</b>	<b>113</b>
11.1	Einfache Delegates	113
11.2	Multicast-Delegates	114
11.3	Erzeugen von Delegate-Werten	114
11.4	Ereignisse (Events)	116
11.5	Anonyme Methoden	117
11.6	Übungsaufgaben	119
<b>12</b>	<b>Ausnahmen</b>	<b>121</b>
12.1	try-Anweisung	121
12.2	Ausnahmeklassen	123
12.3	Auslösen von Ausnahmen	124
12.4	Ausnahmen in aufgerufenen Methoden	126
12.5	Ausnahmen in Multicast-Delegates	126
12.6	Übungsaufgaben	127

<b>13 Namensräume und Assemblies</b>	129
13.1 Namensräume	129
13.2 Assemblies	132
13.2.1 Assemblies und Module	132
13.2.2 Versionierung von Assemblies	133
13.2.3 Assemblies versus Namensräume	136
13.3 Übungsaufgaben	137
<b>14 Generische Bausteine</b>	139
14.1 Generische Typen	140
14.2 Constraints	141
14.3 Vererbung bei generischen Typen	142
14.4 Generische Methoden	144
14.5 Generische Delegates	145
14.6 Nullwerte	147
14.7 Ko- und Kontravarianz bei generischen Typen	147
14.8 Was geschieht hinter den Kulissen?	151
14.9 Unterschiede zu Java	151
14.10 Übungsaufgaben	152
<b>15 Threads</b>	155
15.1 Die Klasse Thread	155
15.2 Zustände eines Threads	158
15.3 Abbrechen eines Threads	159
15.4 Thread-Synchronisation	160
15.5 Übungsaufgaben	165
<b>16 Iteratoren</b>	167
16.1 Allgemeine Iteratoren	167
16.2 Spezifische Iteratoren	169
16.3 Übungsaufgaben	172
<b>17 Attribute</b>	173
17.1 Schreibweise von Attributen	173
17.2 Parameter von Attributen	174
17.3 Attribute für spezifische Programmelemente	175
17.4 Attribut Serializable	176
17.5 Attribut Conditional	178
17.6 Attribut DllImport	179
17.7 Deklaration eigener Attribute	180
17.8 Übungsaufgaben	181

<b>18 Dokumentationskommentare</b>	183
18.1 XML-Elemente	183
18.2 Erzeugte XML-Datei	185
18.3 Übungsaufgaben	186
<b>19 Auszug aus der .NET-Klassenbibliothek</b>	187
19.1 Hilfsklassen	188
19.2 Collections	191
19.3 Ein-/Ausgabe	200
19.4 Reflection	206
19.5 Übungsaufgaben	210
<b>20 LINQ</b>	213
20.1 Motivation	213
20.2 Lambda-Ausdrücke	214
20.3 Erweiterungsmethoden	218
20.4 Objektinitialisierer	220
20.5 Anonyme Typen	222
20.6 Query-Ausdrücke	224
20.7 LINQ und XML	231
20.8 Übungsaufgaben	233
<b>21 Asynchrone Methoden und Parallelität</b>	235
21.1 Asynchronität	235
21.2 Tasks	236
21.3 Asynchrone Methoden	238
21.4 Explizite Parallelität	244
21.5 Übungsaufgaben	246
<b>22 Tupel</b>	247
22.1 Tupel-Typen	247
22.2 Zuweisungen	248
22.3 Anwendungen	249
22.4 Zerlegung von Tupeln	251
22.5 Dekonstruktoren für Klassen und Structs	252
22.6 Übungsaufgaben	253
<b>23 Pattern Matching</b>	255
<b>24 Interoperabilität mit COM</b>	259
24.1 COM-Objekte von .NET aus ansprechen	260
24.2 .NET-Assemblies von COM aus ansprechen	263
24.3 Übungsaufgaben	265

<b>25 Dynamisch getypte Variablen</b>	267
25.1 Typ dynamic	267
25.2 Operationen auf dynamic-Variablen	269
<b>26 Diverses</b>	271
26.1 Null-fähige Werttypen	271
26.2 Bedingter Zugriff über Referenzen	274
26.3 using static	275
26.4 Geschachtelte Methoden	276
26.5 Rückgabe von Funktionswerten by reference	277
<b>27 Fallstudien</b>	279
27.1 Anwendungen mit grafischer Benutzeroberfläche	279
27.2 Ein Web-Service für Börsenkurse	288
27.3 Dynamische Webseiten mit ASP.NET	293
27.4 Übungsaufgaben	299
<b>A Anhang</b>	301
A.1 Compileroptionen	301
A.2 Werkzeuge unter .NET	304
A.2.1 ildasm	304
A.2.2 Globaler Assembly-Cache	305
A.3 Grammatik von C#	308
A.4 Unicode und ASCII	315
<b>Literatur</b>	317
<b>Index</b>	319