

Inhalt

1	Einleitung.....	11
1.1	An wen richtet sich das Buch?.....	11
1.2	Ziel.....	11
1.3	1 = 2.....	11
1.4	Voraussetzungen	11
1.5	Was decken die einzelnen Kapitel ab?.....	12
1.6	Was nicht enthalten ist	14
1.7	Konventionen	15
1.8	Häufige Abkürzungen.....	15
1.9	Sourcecode und Ergänzungen	15
1.10	Errata.....	16
1.11	Link Sammlung.....	16
1.12	Literatur Sammlung	16
1.13	Microsoft C# Einstieg	16
1.14	Bildnachweis, Copyright und Warenzeichen.....	16
1.15	Glossar.....	16
1.16	Danksagung	16
2	Hello World, XXL !.....	17
2.1	Console Application	18
2.2	Windows Forms.....	23
2.3	Windows Phone mit Silverlight.....	27
2.4	Silverlight Webseite	30
2.5	ASP.NET Webseite	31
2.6	Fragen	31
2.7	Wo stehen wir?.....	31
3	.Net Framework.....	32
3.1	Was ist .NET?.....	32
3.2	Wie arbeitet man mit .NET und Visual Studio?	32
3.3	Überblick .NET	34
3.4	.NET Programmiersprachen	34
3.5	.NET Namespaces	35
3.6	Orientierungshilfe.....	36
3.7	Fragen	36
3.8	Kompetenz-Check.....	36
3.9	Wo stehen wir?.....	36
4	Visual Studio IDE.....	37
4.1	Start Page	37
4.2	Project Templates	38
4.3	Menü.....	38
4.4	Toolbar	39
4.5	Solution Explorer	40
4.6	Solution Navigator	41
4.7	Code Editor	42

4.7.1	Intellisense	42
4.7.2	Code Expansion	43
4.7.3	Code Snippets.....	43
4.7.4	Schriftgröße.....	44
4.7.5	Event Handler.....	44
4.7.6	XML Kommentar.....	44
4.7.7	Vollbild	45
4.7.8	Compiler Directives	45
4.8	Designer	47
4.9	Toolbox.....	47
4.10	Properties Window	48
4.11	Class View Window.....	48
4.12	Object Browser Window	49
4.13	Server Explorer Window	49
4.14	Help.....	49
4.15	Project Properties	50
4.16	Solution Properties	51
4.17	Referenzen	52
4.18	Debugger Fenster	52
4.19	Error List	53
4.20	Task List	53
4.21	Compilieren und Ausführen	54
4.22	Mehrere Bildschirme	54
4.23	Erweiterungen	54
4.24	Tastenkürzel	55
4.25	Refactoring.....	55
4.26	Multi-Targeting.....	55
4.27	Macros	55
4.28	Unit Tests	55
4.29	VSTS – Visual Studio Team System	56
4.30	Extension Manager	56
4.31	Grafische Entwurfswerkzeuge.....	56
4.32	Vom Code zum Diagramm.....	57
4.33	Vom Diagramm zum Code.....	58
4.34	Projektstruktur Praxis	58
4.35	Fragen	60
4.36	Kompetenz-Check.....	60
4.37	Wo stehen wir?.....	60
5	C# Grundlagen	61
5.1	Keywords und Syntax.....	61
5.2	Namensgebung.....	65
5.3	Variable	66
5.4	Value und Reference Types.....	66
5.4.1	Primitive Types	68
5.4.2	Value Types	68
5.4.3	Reference Types	71

5.4.4	Nullable und DBNull.....	72
5.5	Enumerations	73
5.6	Arrays.....	74
5.7	Parameter Passing	74
5.8	Parameter Array	75
5.9	Scope.....	75
5.10	Konversionen.....	78
5.11	Boxing.....	78
5.12	Math Funktionen und Rundungen.....	79
5.13	Operatoren	79
5.13.1	Arithmetische Operatoren	79
5.13.2	Vergleichsoperatoren	80
5.13.3	Boolsche Operatoren	80
5.13.4	Bitwise Operators	81
5.13.5	Zuweisungsoperatoren	81
5.13.6	Rangfolge der Operatoren	81
5.13.7	Operator Overloading.....	81
5.14	Kontroll-Strukturen.....	82
5.14.1	If statement	82
5.14.2	Switch.....	82
5.14.3	For Schleife	83
5.14.4	Foreach Schleife.....	84
5.14.5	While Schleife	85
5.14.6	Do ... while Schleife.....	85
5.14.7	Schleifen-Abbruch.....	86
5.15	Operatoren	86
5.16	Zeichensatz.....	86
5.17	Ausgabeformatierung	86
5.18	Fragen	89
5.19	Kompetenz-Check.....	90
5.20	Wo stehen wir?.....	90
6	Objektorientiertes Programmieren	91
6.1	Beispiel Personen Klasse	92
6.2	Member Variables	93
6.3	Methoden ohne Rückgabewert.....	93
6.4	Methoden mit Rückgabewert.....	94
6.5	Methoden mit Parametern	94
6.6	Properties.....	95
6.7	Indexer	97
6.8	Events.....	98
6.8.1	Raising Events.....	98
6.8.2	Receiving Events	99
6.9	Konstruktoren.....	102
6.10	Termination und Cleanup, Destruktor	102
6.11	Finalize und Dispose.....	103
6.12	Fortgeschrittene Konzepte	104

6.12.1	Overloading Methods	104
6.12.2	Static Methods, Variables und Events	105
6.12.3	Static Variables	105
6.12.4	Static Methods	106
6.12.5	Static Events	106
6.13	Delegates.....	107
6.13.1	Deklaration eines Delegates.....	107
6.13.2	Benutzung eines Delegate Type	108
6.13.3	Implementierung einer Delegate Methode	108
6.13.4	Definition Delegates	110
6.14	Fragen	112
6.15	Kompetenz-Check.....	112
6.16	Wo stehen wir?.....	112
7	Fortgeschrittenes Objektorientiertes Programmieren	113
7.1	Inheritance	114
7.2	Beispiel Inheritance	115
7.2.1	Erzeugen einer Base Class.....	115
7.2.2	Erzeugen einer Subclass.....	116
7.2.3	Overloading Methods	119
7.2.4	Overriding Methods.....	122
7.2.5	Hiding.....	124
7.2.6	Mult-Level Inheritance.....	125
7.2.7	Zugriff auf Basisklasse, eigene Klasse, eigenes Objekt	127
7.2.8	Konstruktoren	130
7.2.9	Events and Inheritance	131
7.2.10	Abstract.....	131
7.2.11	Sealed.....	132
7.2.12	Creating an Abstract Base Class.....	132
7.3	Multiple Interfaces.....	132
7.3.1	Definition eines Interfaces	132
7.3.2	Benutzung eines Interfaces.....	133
7.3.3	Implementierung eines Interfaces	134
7.3.4	Bedeutung von Interfaces	135
7.4	Polymorphismus	135
7.5	Reflection	136
7.6	Late Binding.....	136
7.7	Bau eines Frameworks	136
7.8	Fragen	138
7.9	Kompetenz-Check.....	138
7.10	Wo stehen wir?.....	138
8	Exception Handling und Debugging.....	139
8.1	Exceptions	140
8.2	Beispiel 1	141
8.3	Beispiel 2	141
8.4	Try-Catch	142
8.5	Throw	143

8.6	User-defined Exceptions	143
8.7	Wie funktioniert Try-Catch?	143
8.8	Try-Catch und Multi-Threading	144
8.9	Application Events	144
8.10	Unhandled Exception Handler.....	145
8.11	Inner Exception Auswertung	145
8.12	Using Statement	145
8.13	Der My Namespace.....	146
8.14	Error Log und Trace.....	147
8.14.1	Wohin schreiben wir Meldungen?.....	147
8.14.2	Windows Error Reporting.....	147
8.14.3	Windows Event Log.....	148
8.14.4	Debug, Trace.....	149
8.15	Debugger	150
8.15.1	Wichtige Shortcuts.....	150
8.15.2	Aufruf des Debuggers aus Code	150
8.15.3	Remote Debugging	151
8.15.4	Assert Methoden	151
8.15.5	Compiler Constants.....	152
8.15.6	First und Last Chance Exception	152
8.15.7	IntelliTrace - Historical Debugging.....	153
8.16	Caller Info Attributes.....	155
8.17	Fragen	155
8.18	Kompetenz-Check.....	155
8.19	Wo stehen wir?.....	155
9	Generics.....	156
9.1	Ein Beispiel zu Beginn.....	156
9.2	Was sind Generics?.....	158
9.3	Namensgebung.....	159
9.4	Generic Class	159
9.5	Generic Method.....	159
9.6	Generic Interface	159
9.7	Generic Structure.....	160
9.8	Generic Delegate	160
9.9	Generic Constraints	161
9.10	Besonderheiten	161
9.11	Vergleich mit null.....	162
9.12	Nullable Types	162
9.13	Fragen	162
9.14	Kompetenz-Check.....	163
9.15	Wo stehen wir?.....	163
10	Collections	164
10.1	Was sind Collections?	164
10.2	Welche Collection soll ich verwenden?	164
10.3	Generic Collections	165
10.4	Collections in .NET 4.0.....	165

10.4.1	Thread-Safe Collections	165
10.5	Programmierung.....	166
10.5.1	TypeCast.....	166
10.5.2	Mehrere Elemente einfügen	166
10.5.3	Löschen einzelner Elemente.....	167
10.5.4	Collections kopieren	167
10.5.5	Multithreaded Zugriff	167
10.5.6	Sortieren von Collections	168
10.6	Fragen	169
10.7	Kompetenz-Check.....	169
10.8	Wo stehen wir?.....	169
11	Lambda Expressions.....	170
11.1	Was sind Lambda Expressions?	170
11.2	Wie schreibt man Lambda Expressions?	171
11.3	Beispiel 1	171
11.4	Beispiel 2	172
11.5	Beispiel 3	173
11.6	Fragen	175
11.7	Kompetenz-Check.....	175
11.8	Wo stehen wir?.....	175
12	LINQ	176
12.1	Was ist LINQ?	176
12.2	LINQ Syntax	177
12.3	Rückgabewert.....	178
12.4	Beispiel 1	179
12.5	Beispiel 2	179
12.6	Deferred und Immediate Execution	180
12.7	Fragen	181
12.8	Kompetenz-Check.....	181
12.9	Wo stehen wir?.....	181
13	Streams	182
13.1	Was ist ein Stream?	182
13.2	Stream Members	183
13.3	Readers und Writers	184
13.4	File, FileInfo Klasse.....	184
13.5	Fragen	186
13.6	Kompetenz-Check.....	186
13.7	Wo stehen wir?.....	186
14	Files and Directories.....	188
14.1	Überblick	188
14.2	Laufwerke, Verzeichnisse, Dateien	188
14.3	Beispiel 1: FileSearch	188
14.4	Beispiel 2: Verzeichnis überwachen	189
14.5	Programmiertechnik.....	190
14.5.1	Using Statement	190
14.5.2	Path Klasse	191

14.5.3	Special Folder	191
14.5.4	File Security	191
14.5.5	Synchronisierung von Verzeichnissen.....	192
14.6	Fragen	193
14.7	Kompetenz-Check.....	193
14.8	Wo stehen wir?.....	193
15	XML	194
15.1	Beispiel.....	194
15.2	XML Regeln.....	196
15.3	Wann ist eine XML Datei korrekt?	197
15.4	Programmierung.....	197
15.4.1	Stream-basiert	198
15.4.2	DOM-basiert	202
15.4.3	XML Transformations.....	203
15.4.4	XPath	203
15.4.5	LINQ-to-XML	204
15.5	Wie kompatibel ist XML?	204
15.6	Fragen	205
15.7	Kompetenz-Check.....	205
15.8	Wo stehen wir?.....	205
16	Serialisierung	206
16.1	Beispiel.....	206
16.2	Wann ist ein Objekt serialisierbar?.....	208
16.3	ISerializable.....	209
16.4	Versionierung	209
16.5	Geheime Daten.....	209
16.6	Kompatibilität	209
16.7	Fragen	210
16.8	Kompetenz-Check.....	210
16.9	Wo stehen wir?.....	210
17	Tests	212
17.1	Unit Tests.....	213
17.2	Test Driven Development.....	216
17.3	Testdaten Generierung	216
17.4	Test Sprachversionen	217
17.5	Weitere Testwerkzeuge	217
17.6	Code Contracts	217
17.7	Fragen	218
17.8	Kompetenz-Check.....	218
17.9	Wo stehen wir?.....	218
18	Sonstiges	219
18.1	Timer	219
18.2	Zeitmessungen.....	219
18.3	Zufallsgenerator.....	220
18.4	GUID.....	220
19	Index.....	221