

Klaus Fahnenstich, Rainer G. Haselier

Richtig einsteigen: Programmieren lernen mit Visual C# 2008

Microsoft[®]
Press

Klaus Fahnenstich, Rainer G. Haselier: Richtig einsteigen: Programmieren lernen mit Visual C# 2008
Microsoft Press Deutschland, Konrad-Zuse-Str. 1, 85716 Unterschleißheim
Copyright © 2008 by Microsoft Press Deutschland

Das in diesem Buch enthaltene Programmmaterial ist mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Autor, Übersetzer und der Verlag übernehmen folglich keine Verantwortung und werden keine daraus folgende oder sonstige Haftung übernehmen, die auf irgendeine Art aus der Benutzung dieses Programmmaterials oder Teilen davon entsteht. Die in diesem Buch erwähnten Software- und Hardwarebezeichnungen sind in den meisten Fällen auch eingetragene Marken und unterliegen als solche den gesetzlichen Bestimmungen. Der Verlag richtet sich im Wesentlichen nach den Schreibweisen der Hersteller.

Das Werk, einschließlich aller Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlags unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
10 09 08

ISBN 978-3-86645-207-7

© Microsoft Press Deutschland
(ein Unternehmensbereich der Microsoft Deutschland GmbH)
Konrad-Zuse-Str. 1, D-85716 Unterschleißheim
Alle Rechte vorbehalten

Umschlaggestaltung: Hommer Design GmbH, Haar (www.HommerDesign.com)
Layout, Satz und Korrektorat: rabbitsoft, Aachen
Gesamtherstellung: Kösel, Krugzell (www.KoeselBuch.de)

Inhaltsverzeichnis

Teil I Erste Schritte

1	Einleitung	11
1.1	Die Themenschwerpunkte dieses Buches	11
	Visual C# 2008 Express Edition	11
	C#	12
	.NET Framework	12
1.2	Wie Sie am besten mit diesem Buch arbeiten	15
	Aufbau des Buches und der einzelnen Kapitel	15
	Die DVD-ROM, Softlinks und die Website zu diesem Buch	16
	Die Symbole in diesem Buch	16
	Genug der Vorrede und hinein ins Vergnügen	16
2	Visual C# 2008 Express Edition installieren	17
2.1	Die Installation	17
2.2	Visual C# 2008 Express Edition registrieren	21
2.3	Änderungen an Ihrem PC durch die Installation	23
2.4	Die Beispieldateien herunterladen und installieren	25
2.5	Zusammenfassung	25

Teil II Grundlagen von C#

3	Die ersten C#-Programme erstellen	27
3.1	Die Oberfläche von Visual C# 2008 Express Edition	27
	Die Startseite	27
	Wichtige Komponenten der IDE	29
3.2	Das »klassische« erste Programm erstellen	29
3.3	»Hello World« unter der Lupe	33
	Die using-Direktiven	33
	Die namespace-Direktive	35
	Die Klasse Program	36
	Die Klassenmethode Main	36
3.4	Ein Programm mit Ein- und Ausgabe	37
	Den Programmcode zeilenweise ausführen lassen	39
3.5	Übungen zu diesem Kapitel	39
3.6	Zusammenfassung	40

4 Variablen, Datentypen und Operatoren	41
4.1 Informationen speichern	41
Deklaration von Variablen	43
Namen für Variablen	44
Groß- und Kleinschreibung für Variablennamen	45
4.2 Einen Mini-Rechner programmieren	46
Der Operator + hat verschiedene Bedeutungen	48
Zeichenketten in Zahlen umwandeln	49
Typumwandlung im Überblick	51
4.3 C#-Datentypen und das allgemeine Typsystem	52
Werttypen und Verweistypen	55
Die Bedeutung von null bei Verweistypen	55
4.4 Operatoren	56
Division von Integerzahlen, bei denen ein Divisionsrest bleibt	58
Der Modulo-Operator	59
Zusammengesetzte Zuweisungsoperatoren	60
Inkrement- und Dekrementoperator	60
4.5 Übungen zu diesem Kapitel	62
4.6 Zusammenfassung	63
5 Ausnahmen behandeln	65
5.1 Machen Sie Ihre Programme absturzsicher	65
Unbehandelte Ausnahmen	65
Einfacher try-catch-Block	68
Den Rechner absturzsicher machen, Teil 1	69
5.2 Mehrere catch-Blöcke verwenden	70
Das Programm absturzsicher machen, Teil 2	70
Weitere Informationen über die Ausnahme ermitteln	72
5.3 Informationen zu den Ausnahmen erhalten	73
5.4 finally: Sicherstellen, dass Code ausgeführt wird	74
5.5 throw: Ausnahmen auslösen und weiterreichen	76
5.6 Übungen zu diesem Kapitel	77
5.7 Zusammenfassung	78
6 Entscheidungen und Schleifen programmieren	79
6.1 Vergleichsoperatoren	79
6.2 Logische Operatoren	80
Die logischen Und-Operatoren & und &&	80
Die logischen Oder-Operatoren und 	81
Der logische Nicht-Operator !	81
6.3 Verzweigungen	82
Die if-Anweisung	82
Die if-else-Anweisung	84
if und else verschachteln	86
else-if-Ketten	87

Die switch-Anweisung	89
Der bedingte Operator	91
6.4 Schleifen	91
Die for-Schleife	91
Der Programmfluss der for-Schleife	92
Die while-Schleife	93
Die do-while-Anweisung	95
6.5 Fortgeschrittene Schleifensteuerung	96
Die break-Anweisung	96
Die break-Anweisung in verschachtelten Schleifen	96
Die continue-Anweisung	98
Die goto-Anweisung	98
6.6 Übungen zu diesem Kapitel	99
6.7 Zusammenfassung	99
7 Arrays	101
7.1 Was ist ein Array?	101
Array-Variablen deklarieren	101
Array-Variablen instanzieren	102
Array-Variablen manuell initialisieren	103
7.2 Elemente eines Arrays ansprechen	103
Alle Elemente eines Arrays durchlaufen	105
Arrays mit foreach durchlaufen	105
Größe eines Arrays ändern	106
7.3 Mehrdimensionale Arrays	107
Mehrdimensionale Arrays durchlaufen	107
7.4 Nützliche Methoden für Arrays	108
Die GetLength-Methode	108
Daten zwischen Arrays kopieren	109
Arrays duplizieren	109
7.5 Die Klasse ArrayList	110
Neue Elemente in eine ArrayListe aufnehmen	111
Elemente aus einer ArrayListe entfernen	111
Das Beispielprogramm ListeFüllen	112
7.6 Übungen zu diesem Kapitel	113
7.7 Zusammenfassung	114

Teil III

Konzepte der objektorientierten Programmierung (OOP)

8 Methoden	115
8.1 Wozu werden Methoden benötigt?	115
8.2 Eigene Methoden erstellen	116
Methoden dokumentieren	117

Inhaltsverzeichnis

8.3	Die allgemeine Syntax von Methoden	119
	Zugriffsmodifizierer	119
	Rückgabewert	120
	Methodenname	121
	Parameter	121
	ref-Parameter	124
	out-Parameter	124
	params-Parameter	125
8.4	Refactoring Tools für Methoden	126
	Methodenstubs	126
	Methoden extrahieren	128
8.5	Gültigkeitsbereich von Variablen	129
	Methodenübergreifender Zugriff auf Variablen	132
8.6	Methoden überladen	132
8.7	Übungen zu diesem Kapitel	134
8.8	Zusammenfassung	134
9	Zeichenfolgen	137
9.1	Grundlagen und Besonderheiten	137
	Das Schlüsselwort string	137
	Escape-Sequenzen	138
	Das @-Symbol	138
	Zeichenfolgen vergleichen	139
	Zeichenfolgen verketteten	140
9.2	Zeichenfolgen bearbeiten	140
	Auf einzelne Zeichen zugreifen	141
	Zeichenfolgen durchsuchen	142
	Groß-/Kleinschreibung ignorieren	143
	Übung: Groß-/Kleinschreibung ignorieren	144
	Groß-/Kleinschreibung ändern	145
	Zeichenfolgen trimmen	146
	Zeichenfolgen einfügen und ersetzen	146
	Zeichenfolgen zerlegen	147
9.3	Das StringBuilder-Objekt	150
	StringBuilder-Objekte erzeugen	150
	StringBuilder-Objekte bearbeiten	151
	Übung: Performance-Test	152
9.4	Zeichenfolgen formatieren	154
	Formatierung mit ToString	157
9.5	Übungen zu diesem Kapitel	157
9.6	Zusammenfassung	158

10	Debuggen	159
10.1	Einleitung	159
10.2	Ausgeben von Meldungen	160
	Die Klasse Debug	160
	Primfaktoren zerlegen, 1. Fassung	161
	Assertions auslösen	166
10.3	Bedingte Kompilierung	167
10.4	Programme schrittweise ausführen	170
	Debugger-Crashkurs: Primfaktoren zerlegen, 2. Fassung	170
	Die Symbolleiste Debuggen	178
	Anzeigen von Daten mit der Schnellansicht	179
	Das Überwachungsfenster	180
10.5	Übungen zu diesem Kapitel	181
10.6	Zusammenfassung	182
11	Klassen erstellen	183
11.1	Einführung in Klassenkonzepte	183
	Was ist eine Klasse?	184
	Von der Klasse zum Objekt: der Konstruktor	186
	Die Klasse um eine Methode erweitern	190
	Zugriff auf die Daten mittels Eigenschaften und Accessoren	191
	Die Mutter aller Klassen in .NET: System.Object	193
	Methoden überschreiben	196
	Vererbung: Basisklasse festlegen	198
	Zugriffsmodifizierer: public, private & Co	201
11.2	Klasse zur Protokollierung von Ausnahmen	202
	Das Projekt erstellen	203
	Die Daten der Klasse – Enumerations	205
	Die Konstruktoren für die Klasse Logger erstellen	209
	Die Methoden zum Protokollieren erstellen	213
	Die neue Klasse testen	218
11.3	Klassenbibliothek in anderen Projekten verwenden	220
11.4	Statische Klassen erstellen	222
	Die Klasse und die erste Klassenmethode erstellen	222
	Die Klasse in einer Konsolenanwendung testen	224
	Der absolute Nullpunkt	225
	Die weiteren Konvertierungsmethoden einfügen	226
	Die Eigenschaften der Klasse erstellen	229
	Eine weitere Klasse im Namespace der Klassenbibliothek zur Verfügung stellen	230
11.5	Übungen zu diesem Kapitel	232
11.6	Zusammenfassung	233

Inhaltsverzeichnis

12	Strukturen und Schnittstellen	235
12.1	Strukturen	235
	Die Verwandtschaft von Strukturen und Klassen	236
	Deklaration einer Struktur	238
	Verschachtelte Strukturen	241
	Direkten Zugriff auf die Felder von Strukturen verhindern	242
	Die Unterschiede von Strukturen und Klassen im Überblick	243
12.2	Schnittstellen	244
	Was ist eine Schnittstelle?	244
	Implementieren einer Schnittstelle	246
	Eigene Schnittstellen erstellen	252
	Unterschiede zwischen Schnittstellen und Klassen	258
12.3	Übungen zu diesem Kapitel	258
12.4	Zusammenfassung	259

Teil IV

Windows Forms-Anwendungen

13	Windows Forms-Anwendungen erstellen	261
13.1	Projekt für Windows Forms-Anwendungen erstellen	261
13.2	Steuerelemente einfügen	266
13.3	Auf Ereignisse reagieren	269
	Ereignishandler für Steuerelemente erstellen	271
	Ereignishandler für das Form erstellen	274
	RGB-Farbwert in Hexadezimaldarstellung umwandeln	275
13.4	Anwendungseinstellungen verwenden	277
	Anwendungseinstellungen definieren	277
	Anwendungseinstellungen beim Programmstart einlesen	279
	Anwendungseinstellungen beim Programmende speichern	281
13.5	Übungen zu diesem Kapitel	282
13.6	Zusammenfassung	283
14	Steuerelemente erstellen	285
14.1	Benutzerdefinierte Steuerelemente	285
	Verschiedene Arten benutzerdefinierter Steuerelemente	285
14.2	Zusammengesetztes Steuerelement erstellen	287
	Das Projekt und die Benutzeroberfläche erstellen	288
	Prüfen, ob Eingaben in den TextBox-Steuerelementen vorhanden sind	291
	Die Eigenschaft ClickCount und deren Logik implementieren	293
	Das Ereignis programmieren	295
	Das Testprogramm erstellen	299
	Eigene Steuerelemente dauerhaft in die Toolbox einfügen	300

14.3	Abgeleitetes Steuerelement erstellen	301
	Die Projektmappe erstellen und konfigurieren	302
	Die Funktionalität des Steuerelements FilterTextBox implementieren	303
	Das Testprogramm erstellen	308
14.4	Übungen zu diesem Kapitel	309
14.5	Zusammenfassung	311
15	Menüs, Symbolleisten und Dialogfelder erstellen und verwenden	313
15.1	MyNotepad im Überblick	313
15.2	Die Benutzeroberfläche erstellen	314
	Das Menü einfügen	317
15.3	Dateien öffnen und speichern	319
	Konzeptionelle Überlegungen zum Datei-Menü	321
	Datei-Operationen, die zweite Runde	322
	MyNotepad »sauber« beenden	327
	Exkurs: Die Klasse MessageBox	328
15.4	Das Bearbeiten-Menü fertigstellen	330
15.5	Ein Info-Dialogfeld einfügen	332
15.6	Eine Standardsymbolleiste ergänzen	335
15.7	Das Suchen-Dialogfeld erstellen	337
	Die Optik des Dialogfeldes erstellen	337
	Das Dialogfeld aufrufen	340
	Datenaustausch zwischen NotepadForm und SearchForm	341
	Den Code für die Suche schreiben	342
15.8	Zusätzliche Features für den Editor	344
	Schriftart einstellen – Standarddialogfeld nur mit Code verwenden	344
	Der Zeilenumbruchmodus	345
15.9	Übungen zu diesem Kapitel	346
15.10	Zusammenfassung	348
16	Datenbankzugriff mit ADO.NET	349
16.1	Die Beispieldatenbank	349
16.2	Datenbanken in ein Projekt einfügen	351
	Datenquellen	353
	Daten auf einem Formular anzeigen	354
	Speicherort der MDB-Datei	358
	Arbeiten mit Unterformularen	359
	ComboBox mit Daten aus der Datenquelle füllen	362
16.3	Installationsprogramm erstellen	369
16.4	Zusammenfassung	371
	Stichwortverzeichnis	373

