

<b>Vorwort</b>	<b>17</b>
<b>Einleitung</b>	<b>19</b>
<b>1 Die ersten Schritte mit Visual Basic 2005</b>	<b>23</b>
<b>2 Rundgang durch die Visual Basic 2005-IDE</b>	<b>53</b>
<b>3 Formulare und Steuerelemente</b>	<b>69</b>
<b>4 Die Befehle von Visual Basic 2005</b>	<b>85</b>
<b>5 Programmierung mit Klassen</b>	<b>123</b>
<b>6 Streifzug durch die große .NET-Klassenbibliothek</b>	<b>149</b>
<b>7 Einfache Windows-Anwendungen – Schritt für Schritt</b>	<b>169</b>
<b>8 Dateizugriffe und Streams</b>	<b>217</b>
<b>9 Benutzeroberflächen gestalten</b>	<b>235</b>
<b>10 Fehler abfangen und vermeiden</b>	<b>265</b>
<b>11 Multithreading mit Visual Basic</b>	<b>279</b>
<b>12 Internetprogrammierung</b>	<b>301</b>
<b>13 XML als universelles Datenformat</b>	<b>317</b>
<b>14 Datenbankprogrammierung mit Visual Basic 2005</b>	<b>335</b>
<b>15 Von der Idee zum fertigen Visual Basic-Programm</b>	<b>361</b>
<b>Anhang A: Das kleine .NET-Lexikon</b>	<b>385</b>
<b>Anhang B: Visual Basic-Befehlsreferenz</b>	<b>387</b>
<b>Anhang C: Lösungen zu den Übungsaufgaben</b>	<b>397</b>
<b>Stichwortverzeichnis</b>	<b>415</b>

<b>Vorwort</b>	17
<b>Einleitung</b>	19
<b>1 Die ersten Schritte mit Visual Basic 2005</b>	23
1.1 Eine Checkliste für den Start	23
1.1.1 Den SQL Server 2005 Express installieren	24
1.1.2 Die MSDN-Dokumentation installieren	25
1.1.3 Welche Windows-Version wird vorausgesetzt?	26
1.1.4 Welche Hardware wird vorausgesetzt?	26
1.1.5 Ein erster Probestart	26
1.2 Der Idealgewichtsrechner als erstes Beispielprojekt	27
1.2.1 Version 1 – der Idealgewichtsrechner berechnet ein Idealgewicht	28
1.2.2 Version 2 – der Idealgewichtsrechner überprüft die Eingabe	32
1.2.3 Version 3 – der Idealgewichtsrechner unterscheidet zwischen Männlein und Weiblein	33
1.2.4 Version 4 – der Idealgewichtsrechner zeigt Bildchen an	36
1.2.5 Version 5 – der Idealgewichtsrechner berechnet den BMI	38
1.2.6 Version 6 – der Idealgewichtsrechner merkt sich Einstellungen	46
1.2.7 Version 7 – der Idealgewichtsrechner lernt sprechen	48
1.3 Den Idealgewichtsrechner weitergeben	50
1.3.1 Die Rolle der Programmdatei	51
1.4 Zusammenfassung	52
1.5 Fragen	52

<b>2</b>	<b>Rundgang durch die Visual Basic 2005-IDE</b>	<b>53</b>
2.1	Der erste Start von Visual Basic 2005	54
2.1.1	Die Rolle der Projekte	54
2.1.2	Der Projektmappen-Explorer	54
2.1.3	Ein vorhandenes Projekt öffnen	56
2.1.4	Ein neues Projekt anlegen	56
2.1.5	Projekte starten, anhalten und beenden	57
2.1.6	Projekte speichern	57
2.1.7	Listings ausdrucken	58
2.1.8	Projekteigenschaften	59
2.1.9	Der Aufbau eines Visual Basic-Projekts	59
2.2	Die IDE als Nachhilfelehrer	61
2.3	Der Programmcode-Editor	63
2.4	Die IDE nach eigenem Geschmack anpassen	64
2.4.1	Schriftart und Schriftgröße einstellen	64
2.4.2	Farben des Hintergrunds und der Schrift einstellen	65
2.4.3	Zeilenumbruch und Zeilennummern einschalten	65
2.4.4	Menübefehle mit eigenen Tastenkombinationen belegen	65
2.4.5	Das Extras-Menü um neue Einträge erweitern	66
2.4.6	Funktionen, die es bei Visual Basic 2005 Express nicht gibt	66
2.5	Zusammenfassung	67
2.6	Fragen	67
<b>3</b>	<b>Formulare und Steuerelemente</b>	<b>69</b>
3.1	Anatomie eines Formulars	69
3.2	Steuerelemente und die Toolbox	70
3.2.1	Steuerelemente auf dem Formular anordnen	71
3.2.2	Die Rolle der Aufgabenliste	71
3.2.3	Die Rolle der Komponenten	72
3.2.4	Die Toolbox anpassen	73
3.3	Das Eigenschaftenfenster	74
3.4	Steuerelemente positionieren	74
3.4.1	Ein Blick in die Designer-Datei eines Formulars	75
3.5	Ereignisse und Ereignisprozeduren	76
3.6	Tipps zu Formularen und Steuerelementen	77
3.7	Die wichtigsten Steuerelemente (für den Anfang)	78
3.7.1	Das Label	78
3.7.2	Der Button	79
3.7.3	Die TextBox	79
3.7.4	Der Radiobutton	80
3.7.5	Die CheckBox	80
3.7.6	Die ListBox	81

3.7.7	Die ComboBox	82
3.7.8	Die PictureBox	82
3.7.9	Der Rauf- und Runterzähler (NumericUpDown)	83
3.8	Zusammenfassung	84
3.9	Fragen	84
<b>4</b>	<b>Die Befehle von Visual Basic 2005</b>	<b>85</b>
4.1	Schritt für Schritt oder wie wird ein Computerprogramm ausgeführt?	85
4.1.1	Computerprogramm = Befehle, Befehle und noch mehr Befehle	86
4.1.2	Fehler in einem Programm – die Bugs lauern überall	86
4.2	Variablen als das Gedächtnis des Computers	86
4.2.1	Variablen besitzen Werte	87
4.2.2	Variablen und ihr Anfangswert	88
4.3	Im Rechnen unschlagbar? Die Operatoren	89
4.3.1	Ein erstes Beispiel – ein kleiner Mehrwertsteuerrechner	90
4.4	Computerprogramme treffen Entscheidungen	92
4.4.1	Bedingungen verknüpfen – Mr. Boole lässt grüßen	93
4.4.2	Sind Sie volljährig oder bist Du es noch nicht? Der Computer findet es heraus	95
4.4.3	Der Mehrwertsteuerrechner wird erweitert	98
4.4.4	Mehrfachabfragen mit Select Case	99
4.4.5	Der Computer als Ratgeber und Freund	100
4.5	Wiederholungen erwünscht – die Programmschleifen	101
4.5.1	Die For Next-Schleife – die Anzahl der Durchläufe steht fest	102
4.5.2	Wiederholen bis eine Abbruchbedingung erfüllt ist	102
4.5.3	Die While End While-Schleife – wie Do Loop, nur ein wenig einfacher	103
4.5.4	Die For Each-Schleife – Spezialist für Auflistungen	103
4.5.5	Warten auf Godot – die Endlosschleife	103
4.6	Befehle zu Gruppen zusammenfassen – die Funktionen und Prozeduren	105
4.6.1	Prozeduren definieren	105
4.6.2	Variablendeklarationen innerhalb einer Prozedur – die lokalen Variablen	106
4.6.3	Prozeduren mit Argumenten	106
4.6.4	Funktionen definieren	107
4.6.5	Statische Variablen	108
4.7	Wie schnell ist der Computer?	109
4.7.1	Den Ausweis bitte – Prüfsummen bilden	111
4.8	Feldvariablen (Arrays)	113

4.8.1	Arrays definieren	113
4.8.2	Einen Wert in einem Array ansprechen	114
4.8.3	Die Größe eines Arrays dynamisch ändern	114
4.8.4	Arrays Element für Element durchlaufen	115
4.9	Etwas fortgeschrittenere Themen der Visual Basic-Programmierung	116
4.9.1	Die Rolle der Datentypen	116
4.9.2	Übergaben von Argumenten als Referenz	118
4.9.3	Zeichenkettenoperationen mit der String-Klasse	119
4.10	Zusammenfassung	120
4.11	Fragen und Antworten	120
<b>5</b>	<b>Programmierung mit Klassen</b>	123
5.1	Was ist ein Objekt?	123
5.1.1	Auf zur objektorientierten Programmierung	124
5.2	Eine Klasse wird definiert	124
5.2.1	Eine Klasse besitzt Mitglieder	125
5.2.2	Felder definieren	125
5.3	Eigenschaften definieren	125
5.4	Aus der Klasse wird ein Objekt	126
5.4.1	Ich instanziiere, also bin ich – das Schlüsselwort Me	127
5.4.2	Die Rolle des Konstruktors	127
5.5	Methoden definieren und überladen – was steckt hinter dem Gültigkeitsbereich?	128
5.5.1	Methoden überladen	129
5.6	Klassen über freigegebene Mitglieder ansprechen	130
5.7	Klassen ableiten	132
5.8	Mitglieder überschreiben	134
5.9	Abstrakte Basisklassen – ab jetzt wird es ein wenig anspruchsvoller	136
5.10	Schnittstellen	137
5.10.1	Schnittstellen bei der Variablendeklaration	139
5.10.2	Schnittstellen führen zu (etwas) mehr Flexibilität	139
5.10.3	Ein praktisches Beispiel für die Implementierung einer Schnittstelle	140
5.10.4	Polymorphie – damit auch das einmal geklärt wird	141
5.11	Events	143
5.12	Alles besitzt einen Typ	145
5.12.1	Den Typ eines Objekts feststellen	145
5.12.2	Alles über einen Typen – das Type-Objekt gibt Auskunft	146
5.13	Zusammenfassung	147
5.14	Fragen	147

<b>6</b>	<b>Streifzug durch die große .NET-Klassenbibliothek</b>	<b>149</b>
6.1	Ein erster Überblick	150
6.1.1	Die Environment-Klasse als Musterknabe	150
6.1.2	Instanzenmitglieder bei Basisklassen benutzen	151
6.1.3	Ohne Hilfe geht es (fast) nicht	152
6.2	Die Rolle der Namespaces	153
6.2.1	Der Imports-Befehl	153
6.2.2	Das Global-Schlüsselwort von Visual Basic	154
6.3	Die Klassenbibliothek enthält nicht nur Klassen	155
6.4	Die wichtigsten Namespaces im Schnellüberblick	156
6.4.1	Die Namespace-Hierarchie mit dem Objektbrowser erforschen	157
6.4.2	Die Suche nach einem Namen	157
6.4.3	Klassen für den Dateizugriff	157
6.4.4	Klassen für Formulare und Steuerelemente	158
6.4.5	Klassen für den Datenbankzugriff	158
6.4.6	Klassen für den Umgang mit XML	159
6.4.7	Klassen für die Internetprogrammierung	159
6.4.8	Ausnahmeklassen	161
6.4.9	Mathematikklassen	161
6.5	Auflistungen	162
6.5.1	Die ArrayList-Klasse – eine wirklich universelle Ablage	163
6.5.2	Die Hashtable – Zugriff nur über einen Schlüssel	164
6.5.3	Die List-Klasse – nur ein Typ erlaubt	164
6.5.4	Die Dictionary-Klasse – mit Typ und Schlüssel	165
6.6	Der My-Namespace	166
6.6.1	Mehr über den My-Namespace	167
6.7	Zusammenfassung	168
6.8	Fragen	168
<b>7</b>	<b>Einfache Windows-Anwendungen – Schritt für Schritt</b>	<b>169</b>
7.1	Tastatureingaben	169
7.1.1	Was ist genau passiert? Die Aufgabe des e-Arguments	170
7.1.2	Der Unterschied zwischen KeyDown, KeyUp und KeyPress	170
7.1.3	Die Rolle des Eingabefokus	172
7.1.4	Den Eingabefokus per Programm setzen	173
7.1.5	Wie das Formular Tastaturereignisse erhält	173
7.1.6	Automatische Zeilenschaltung – kurzer Ausflug zu SendKeys	173
7.2	Mausaktionen	174
7.2.1	Den Mauszeiger »verschönern«	176
7.3	Ein Zinsrechner – Schritt für Schritt	176
7.4	Ein Bahnfahrt-Entscheidungshelfer – Schritt für Schritt	182

7.5	Ein Aufgabenplaner – Schritt für Schritt	190
7.5.1	Eine wichtige Erweiterung – Abspeichern und Laden von Aufgaben	197
7.6	Ein Gedächtnistrainer – Schritt für Schritt	200
7.7	Zusammenfassung	215
7.8	Fragen	215
<b>8</b>	<b>Dateizugriffe und Streams</b>	217
8.1	Im Mittelpunkt steht die File-Klasse	217
8.2	Abspeichern und Einlesen von Texten	218
8.2.1	Einlesen von Texten aus Textdateien	218
8.2.2	Feststellen, ob eine Datei existiert	218
8.2.3	Dateien löschen	219
8.2.4	Die Größe einer Datei feststellen	219
8.2.5	Berechtigungen bei Dateien abfragen und setzen	219
8.3	Abspeichern und Laden von Zahlen	219
8.3.1	Speichern von Zahlen	220
8.3.2	Das Einlesen von Zahlen	221
8.4	Im Mittelpunkt steht der Stream	221
8.4.1	Streams im Arbeitsspeicher anlegen	222
8.4.2	Ein Visual Basic-Horoskopberater	223
8.4.3	Eine Frage der Zeichensätze	226
8.5	Mit Dateien, Verzeichnissen und Laufwerken umgehen	227
8.5.1	Dateien und Verzeichnisse über eine Dialogbox auswählen	227
8.5.2	Die Directory-Klasse	229
8.5.3	Rekursive Dateisuche	229
8.5.4	Backups mit Datumsvergleich	230
8.5.5	Den Laufwerksbuchstaben des CD-/DVD-Laufwerks ermitteln	232
8.6	Die sehr nützliche Path-Klasse	232
8.7	Zusammenfassung	233
8.8	Fragen	233
<b>9</b>	<b>Benutzeroberflächen gestalten</b>	235
9.1	Auf die richtigen »Zutaten« kommt es an	235
9.1.1	Die Menüleiste – das MenuStrip-Steuerelement	236
9.1.2	Die Symbolleiste – das ToolStrip-Steuerelement	238
9.1.3	Die Statusleiste	239
9.1.4	Kontextmenüs mit dem ContextMenuStrip-Steuerelement	240
9.1.5	Die passenden Bitmaps finden	240
9.1.6	Verschiebbare Flächen mit dem SplitContainer-Steuerelement	241
9.2	Der Bildbetrachter stellt sich vor	241

9.2.1	Die Umsetzung Schritt für Schritt	242
9.2.2	Erweiterung 1 – eine Dateiliste	254
9.2.3	Erweiterung 2 – ein Begrüßungsbildschirm	254
9.2.4	Die Anwendung weitergeben	255
9.2.5	Weitere Erweiterungen	255
9.3	Weitere wichtige Steuerelemente	256
9.3.1	Die ganze Welt im eigenen Programm – das Webbrowser-Steuerelement	256
9.3.2	Texte endlich einmal bunt – die RTFTextBox	258
9.3.3	Listen mit Details und Spalten – die ListView	259
9.3.4	Es lebe die Hierarchie – das TreeView-Control	261
9.4	Zusammenfassung	263
9.5	Fragen	264
<b>10</b>	<b>Fehler abfangen und vermeiden</b>	265
10.1	Was sind eigentlich Programmfehler?	266
10.2	Über das Wesen einer Ausnahme	267
10.2.1	Ausnahmen basieren auf regulären Klassen	268
10.3	Das Ausnahmefenster	269
10.4	Ausnahmen abfangen	270
10.4.1	Variablen in einem Try-Block	270
10.4.2	Mehrere Ausnahmen abfangen	271
10.4.3	Damit immer etwas passiert – der Finally-Befehl	272
10.4.4	Ausnahmen im Debugger abfangen	272
10.5	Der integrierte Debugger stellt sich vor	273
10.5.1	Ein erster Überblick über den Debugger	273
10.5.2	Den Debugger über die Tastatur steuern	273
10.5.3	Das Direktfenster	274
10.5.4	Ein Programm im Einzelschritt-Modus ausführen	274
10.5.5	Festlegen der nächsten Anweisung	275
10.5.6	Haltepunkte setzen	275
10.5.7	Bedingte Haltepunkte einrichten	276
10.5.8	Ausdrücke überwachen	276
10.6	Tipps, um Fehler zu vermeiden	277
10.7	Zusammenfassung	277
10.8	Fragen	278
<b>11</b>	<b>Multithreading mit Visual Basic</b>	279
11.1	Was ist eigentlich Multithreading?	279
11.2	Ein »Hallo, Welt«-Beispiel für Multithreading	280
11.2.1	Ein wenig Theorie	281
11.3	Multithreading in Windows-Anwendungen – die BackgroundWorker-Komponente im Einsatz	284



11.4	Multithreading für etwas Fortgeschrittenere	288
11.5	Asynchrone Aufrufe	294
11.5.1	Asynchrones Lesen und Schreiben von Dateien	296
11.6	Der Timer als Alternative	296
11.7	Zusammenfassung	299
11.8	Fragen	300
<b>12</b>	<b>Internetprogrammierung</b>	<b>301</b>
12.1	Das Internet in 10 Minuten	301
12.1.1	Auf die Protokolle kommt es an	303
12.1.2	Die Rolle der Ports	304
12.1.3	Die Klassen im Namespace System.Net	305
12.1.4	Netzwerkverbindung und IP-Adresse feststellen	305
12.1.5	Andere Computer im Netzwerk anpingen	307
12.2	Dateien per HTTP herunterladen	308
12.3	Dateien per FTP hochladen	309
12.4	E-Mails versenden mit SMTP	313
12.4.1	Die Mehrfachauswahl aus einer ListBox	314
12.4.2	E-Mails abholen	315
12.5	Zusammenfassung	315
12.6	Fragen	315
<b>13</b>	<b>XML als universelles Datenformat</b>	<b>317</b>
13.1	XML in 10 Minuten	317
13.2	XML-Dokumente anfertigen	319
13.3	XML-Dokumente lesen und schreiben	322
13.3.1	Ein XML-Dokument Knoten für Knoten lesen	322
13.3.2	XML-Dokumente aktualisieren und schreiben	323
13.4	XML-Daten in Steuerelementen anzeigen	324
13.5	XML-Dokumente mit XPath durchsuchen	326
13.6	Eine kleine XML-Anwendung	328
13.7	Zusammenfassung	332
13.8	Fragen	333
<b>14</b>	<b>Datenbankprogrammierung mit Visual Basic 2005</b>	<b>335</b>
14.1	Datenbanken in 10 Minuten	335
14.1.1	Was steckt hinter einer Datenbank?	336
14.1.2	Was ist eigentlich ein SQL Server?	337
14.1.3	Welches Datenbanksystem ist das Beste für Visual Basic?	337
14.1.4	Wie werden SQL Server-Datenbanken erstellt und administriert?	338
14.1.5	Die Beispieldatenbank Northwind	339
14.1.6	Datenbanken in Visual Basic 2005 Express anzeigen	341

14.2	Datenbankdaten in eine Windows-Anwendung integrieren	341
14.2.1	Visual Basic 2005 lässt ein wenig die Muskeln spielen – eine Abfrage hinzufügen	349
14.2.2	Funktioniert das Ganze auch mit anderen Datenbanken?	351
14.2.3	Ein Blick hinter die Kulissen	351
14.3	Die wunderbare Genialität von SQL	352
14.4	SQL-Abfragen mit Visual Basic ausführen	354
14.4.1	Aktualisierungen durchführen	357
14.4.2	Funktioniert das Ganze auch mit anderen Datenbanken?	357
14.4.3	Das DataGridView unabhängig von einer Datenbank ansprechen	358
14.5	Zusammenfassung	359
14.6	Fragen	359
<b>15</b>	<b>Von der Idee zum fertigen Visual Basic-Programm</b>	<b>361</b>
15.1	Eine Programmidee entsteht	361
15.2	Vor dem ersten Befehl steht die Planung	362
15.2.1	Den Funktionsumfang festlegen	362
15.2.2	Wie soll das Programm heißen?	364
15.3	Die Benutzeroberfläche wird entworfen	365
15.3.1	Benutzerspezifische Einstellungen berücksichtigen	366
15.4	Die Programmierung beginnt	367
15.4.1	Oberfläche und Code müssen sauber getrennt werden	368
15.5	Die Codierung beginnt	368
15.6	Das Programm wird debuggt	376
15.7	Kleinigkeiten, die noch fehlen	377
15.7.1	Einen Begrüßungsbildschirm hinzufügen	377
15.7.2	Das Formular erhält ein eigenes Icon	377
15.7.3	Die Versionsnummer und andere Programminformationen werden gesetzt	378
15.8	Das Programm wird ausgeliefert	378
15.8.1	Eine Installerdatei bei Visual Basic 2005 Express anfertigen	379
15.8.2	Vom Debug- auf den Release-Modus umschalten	379
15.8.3	Die Rolle der .NET-Laufzeit	380
15.8.4	Das Programm zum Download anbieten	380
15.9	Zusammenfassung und Ausblick	382
<b>Anhang A: Das kleine .NET-Lexikon</b>		<b>385</b>
<b>Anhang B: Visual Basic-Befehlsreferenz</b>		<b>387</b>
<b>Anhang C: Lösungen zu den Übungsaufgaben</b>		<b>397</b>
<b>Stichwortverzeichnis</b>		<b>415</b>