

Thomas Theis

Inkl.  mit den  
Express-Versionen von

 Visual Studio

# Einstieg in Visual Basic 2013

Ideal für Programmierneinsteiger geeignet



- ▶ Schnell erste eigene Windows-Programme entwickeln
- ▶ Alle Grundlagen zu GUIs, Datenbank- und Internetanwendungen
- ▶ Inkl. Einführung in WPF und Windows Store Apps

4., aktualisierte Auflage

Galileo Computing



# Auf einen Blick

1	Einführung .....	17
2	Grundlagen .....	39
3	Fehlerbehandlung .....	107
4	Erweiterte Grundlagen .....	119
5	Objektorientierte Programmierung .....	185
6	Wichtige Klassen in .NET .....	223
7	Weitere Elemente eines Windows-Programms .....	297
8	Datenbankanwendungen mit ADO.NET .....	353
9	Internetanwendungen mit ASP.NET .....	407
10	Zeichnen mit GDI+ .....	433
11	Beispielprojekte .....	447
12	Windows Presentation Foundation .....	473
13	Windows Store-Apps für Windows 8.1 .....	499

# Inhalt

<b>1</b>	<b>Einführung</b>	17
<b>1.1</b>	<b>Aufbau dieses Buchs</b>	17
<b>1.2</b>	<b>Visual Studio 2013</b>	17
<b>1.3</b>	<b>Mein erstes Windows-Programm</b>	18
<b>1.4</b>	<b>Visual Basic-Entwicklungsumgebung</b>	19
1.4.1	Ein neues Projekt	19
1.4.2	Einfügen von Steuerelementen	21
1.4.3	Arbeiten mit dem Eigenschaften-Fenster	22
1.4.4	Speichern eines Projekts	25
1.4.5	Das Codefenster	25
1.4.6	Schreiben von Programmcode	26
1.4.7	Kommentare	28
1.4.8	Starten, Ausführen und Beenden des Programms	28
1.4.9	Ausführbares Programm	29
1.4.10	Projekt schließen, Projekt öffnen	29
1.4.11	Übung	30
1.4.12	Regeln für Zeilenumbrüche	30
<b>1.5</b>	<b>Arbeiten mit Steuerelementen</b>	31
1.5.1	Steuerelemente formatieren	31
1.5.2	Steuerelemente kopieren	33
1.5.3	Eigenschaften zur Laufzeit ändern	33
1.5.4	Vergabe und Verwendung von Namen	36
1.5.5	Verknüpfung von Texten, mehrzeilige Texte	36
1.5.6	Eigenschaft BackColor, Farben allgemein	37
<b>2</b>	<b>Grundlagen</b>	39
<b>2.1</b>	<b>Variablen und Datentypen</b>	39
2.1.1	Namen und Werte	39
2.1.2	Deklarationen	40

2.1.3	Datentypen .....	40
2.1.4	Gültigkeitsbereich .....	43
2.1.5	Konstanten .....	46
2.1.6	Enumerationen .....	47
<b>2.2</b>	<b>Operatoren</b> .....	<b>49</b>
2.2.1	Rechenoperatoren .....	49
2.2.2	Vergleichsoperatoren .....	51
2.2.3	Logische Operatoren .....	52
2.2.4	Verkettungsoperator .....	53
2.2.5	Zuweisungsoperatoren .....	54
2.2.6	Rangfolge der Operatoren .....	54
<b>2.3</b>	<b>Einfache Steuerelemente</b> .....	<b>56</b>
2.3.1	Panel .....	56
2.3.2	Zeitgeber .....	58
2.3.3	Textfelder .....	61
2.3.4	Zahlenauswahlfeld .....	65
<b>2.4</b>	<b>Verzweigungen</b> .....	<b>66</b>
2.4.1	Einzeiliges If...Then...Else .....	67
2.4.2	If...Then...Else-Block .....	69
2.4.3	Select Case .....	70
2.4.4	Funktion If .....	72
2.4.5	Funktion Choose .....	72
2.4.6	Übungen .....	73
<b>2.5</b>	<b>Verzweigungen und Steuerelemente</b> .....	<b>74</b>
2.5.1	Kontrollkästchen .....	74
2.5.2	Optionsschaltfläche .....	76
2.5.3	Mehrere Ereignisse in einer Prozedur behandeln .....	79
2.5.4	Mehrere Gruppen von Optionsschaltflächen .....	80
2.5.5	Prozedur ohne Ereignis, Modularisierung .....	83
<b>2.6</b>	<b>Schleifen</b> .....	<b>84</b>
2.6.1	For ... Next .....	85
2.6.2	Do ... Loop .....	87
2.6.3	With .....	91
2.6.4	Übungen .....	92
<b>2.7</b>	<b>Schleifen und Steuerelemente</b> .....	<b>94</b>
2.7.1	Listenfeld .....	94
2.7.2	Listenfeld füllen .....	94

2.7.3	Wichtige Eigenschaften .....	95
2.7.4	Wechsel der Auswahl .....	97
2.7.5	Wichtige Methoden .....	99
2.7.6	Mehrfachauswahl .....	102
2.7.7	Kombinationsfelder .....	103
<b>3</b>	<b>Fehlerbehandlung</b> .....	<b>107</b>
<b>3.1</b>	<b>Entwicklung eines Programms</b> .....	<b>107</b>
<b>3.2</b>	<b>Fehlerarten</b> .....	<b>108</b>
<b>3.3</b>	<b>Syntaxfehler und IntelliSense</b> .....	<b>109</b>
<b>3.4</b>	<b>Laufzeitfehler und Exception Handling</b> .....	<b>111</b>
3.4.1	Programm mit Laufzeitfehlern .....	111
3.4.2	Einfaches Exception Handling .....	113
3.4.3	Erweitertes Exception Handling .....	115
<b>3.5</b>	<b>Logische Fehler und Debugging</b> .....	<b>116</b>
3.5.1	Haltepunkte und Einzelschrittverfahren .....	116
3.5.2	Überwachungsfenster .....	118
<b>4</b>	<b>Erweiterte Grundlagen</b> .....	<b>119</b>
<b>4.1</b>	<b>Steuerelemente aktivieren</b> .....	<b>119</b>
4.1.1	Ereignis Enter .....	119
4.1.2	Eigenschaften Enabled und Visible .....	122
<b>4.2</b>	<b>Bedienung per Tastatur</b> .....	<b>125</b>
4.2.1	Eigenschaften TabIndex und TabStop .....	125
4.2.2	Tastenkombination für Steuerelemente .....	126
<b>4.3</b>	<b>Ereignisgesteuerte Programmierung</b> .....	<b>127</b>
4.3.1	Eine Ereigniskette .....	128
4.3.2	Endlose Ereignisketten .....	129
4.3.3	Textfelder koppeln .....	130
<b>4.4</b>	<b>Mehrere Formulare</b> .....	<b>132</b>
4.4.1	Allgemeine Code-Module .....	135

<b>4.5</b>	<b>Datenfelder</b> .....	137
4.5.1	Eindimensionale Datenfelder .....	138
4.5.2	Ein Feld durchsuchen .....	139
4.5.3	Weitere Feld-Operationen .....	141
4.5.4	Mehrdimensionale Datenfelder .....	143
4.5.5	Datenfelder initialisieren .....	147
4.5.6	Datenfelder sind dynamisch .....	148
4.5.7	Collections initialisieren .....	151
4.5.8	For-Each-Schleife .....	153
<b>4.6</b>	<b>Datenstruktur ArrayList</b> .....	154
4.6.1	ArrayList mit einfachen Typen .....	155
4.6.2	ArrayList mit Objekten .....	157
<b>4.7</b>	<b>Benutzerdefinierte Datentypen</b> .....	158
<b>4.8</b>	<b>Prozeduren und Funktionen</b> .....	161
4.8.1	Prozeduren .....	161
4.8.2	Übergabe per Referenz .....	164
4.8.3	Funktionen .....	166
4.8.4	Optionale Argumente .....	168
4.8.5	Beliebig viele Argumente .....	170
4.8.6	Datenfelder als Argumente .....	172
4.8.7	Rekursiver Aufruf .....	174
4.8.8	Übungen zu Prozeduren und Funktionen .....	176
<b>4.9</b>	<b>Konsolenanwendung</b> .....	177
4.9.1	Anwendung erzeugen .....	177
4.9.2	Ein- und Ausgabe von Text .....	178
4.9.3	Eingabe einer Zahl .....	179
4.9.4	Erfolgreiche Eingabe einer Zahl .....	180
4.9.5	Ausgabe formatieren .....	181
4.9.6	Aufruf von der Kommandozeile .....	183
<b>5</b>	<b>Objektorientierte Programmierung</b> .....	185
<b>5.1</b>	<b>Was ist Objektorientierung?</b> .....	185
<b>5.2</b>	<b>Klasse, Eigenschaft, Methode, Objekt</b> .....	186
5.2.1	Objektverweis und Instanz .....	189
<b>5.3</b>	<b>Eigenschaftsmethode</b> .....	190

<b>5.4</b>	<b>Konstruktor</b> .....	193
<b>5.5</b>	<b>Referenzen, Vergleiche und Typen</b> .....	196
5.5.1	Objekte vergleichen .....	198
5.5.2	Typ eines Objekts ermitteln .....	199
5.5.3	Typ eines Objekts durch Vergleich ermitteln .....	200
<b>5.6</b>	<b>Delegates</b> .....	201
<b>5.7</b>	<b>Statische Elemente</b> .....	203
<b>5.8</b>	<b>Vererbung</b> .....	207
<b>5.9</b>	<b>Konstruktoren bei Vererbung</b> .....	210
<b>5.10</b>	<b>Polymorphie</b> .....	213
<b>5.11</b>	<b>Schnittstellen</b> .....	217
5.11.1	Vorhandene Schnittstellen .....	220
<b>6</b>	<b>Wichtige Klassen in .NET</b> .....	223
<hr/>		
<b>6.1</b>	<b>Klasse String für Zeichenketten</b> .....	223
6.1.1	Eigenschaften der Klasse String .....	224
6.1.2	Trimmen .....	226
6.1.3	Splitten .....	227
6.1.4	Suchen .....	229
6.1.5	Einfügen .....	232
6.1.6	Löschen .....	234
6.1.7	Teilzeichenkette ermitteln .....	235
6.1.8	Zeichen ersetzen .....	237
6.1.9	Ausgabe formatieren .....	238
<b>6.2</b>	<b>Datum und Uhrzeit</b> .....	240
6.2.1	Eigenschaften von DateTime .....	240
6.2.2	Rechnen mit Datum und Uhrzeit .....	243
6.2.3	DateTimePicker .....	245
<b>6.3</b>	<b>Dateien und Verzeichnisse</b> .....	248
6.3.1	Lesen aus einer Textdatei .....	249
6.3.2	Schreiben in eine Textdatei .....	251
6.3.3	Sicheres Lesen aus einer Textdatei .....	252
6.3.4	Sicheres Schreiben in eine Textdatei .....	255
6.3.5	Datei mit wahlfreiem Zugriff .....	256

6.3.6	Die Klassen File und Directory .....	263
6.3.7	Das aktuelle Verzeichnis .....	264
6.3.8	Eine Liste der Dateien .....	265
6.3.9	Eine Liste der Dateien und Verzeichnisse .....	266
6.3.10	Informationen über Dateien und Verzeichnisse .....	267
6.3.11	Bewegen in der Verzeichnishierarchie .....	269
<b>6.4</b>	<b>XML-Dateien</b> .....	<b>270</b>
6.4.1	Aufbau von XML-Dateien .....	271
6.4.2	Schreiben in eine XML-Datei .....	272
6.4.3	Lesen aus einer XML-Datei .....	274
6.4.4	Schreiben von Objekten .....	275
6.4.5	Lesen von Objekten .....	277
<b>6.5</b>	<b>Rechnen mit der Klasse Math</b> .....	<b>279</b>
<b>6.6</b>	<b>Zugriff auf MS Office</b> .....	<b>285</b>
6.6.1	MS Word-Datei erstellen .....	286
6.6.2	MS Excel-Datei erstellen .....	289
<b>6.7</b>	<b>Formular drucken</b> .....	<b>291</b>
6.7.1	Druck und Seitenvorschau .....	292
6.7.2	Druckeinstellungen .....	293

## **7 Weitere Elemente eines Windows-Programms** 297

---

<b>7.1</b>	<b>Hauptmenü</b> .....	<b>297</b>
7.1.1	Erstellung des Hauptmenüs .....	297
7.1.2	Code des Hauptmenüs .....	299
7.1.3	Klasse Font .....	302
7.1.4	Schriftart .....	302
7.1.5	Schriftgröße .....	304
7.1.6	Schriftstil .....	305
<b>7.2</b>	<b>Kontextmenü</b> .....	<b>306</b>
7.2.1	Erstellung des Kontextmenüs .....	306
7.2.2	Code des Kontextmenüs .....	307
<b>7.3</b>	<b>Symbolleiste</b> .....	<b>309</b>
7.3.1	Erstellung der Symbolleiste .....	309
7.3.2	Code der Symbolleiste .....	310



<b>7.4</b>	<b>Statusleiste</b> .....	313
7.4.1	Erstellung der Statusleiste .....	313
7.4.2	Code der Statusleiste .....	314
<b>7.5</b>	<b>Eingabe-Dialogfeld</b> .....	316
<b>7.6</b>	<b>Ausgabe-Dialogfeld</b> .....	319
<b>7.7</b>	<b>Standarddialogfelder</b> .....	325
7.7.1	Datei öffnen .....	325
7.7.2	Datei speichern unter .....	328
7.7.3	Verzeichnis auswählen .....	329
7.7.4	Farbe auswählen .....	331
7.7.5	Schrifteigenschaften auswählen .....	332
<b>7.8</b>	<b>Steuerelement ListView</b> .....	333
<b>7.9</b>	<b>Steuerelement Chart</b> .....	337
<b>7.10</b>	<b>Steuerelement DataGridView</b> .....	341
<b>7.11</b>	<b>Lokalisierung</b> .....	346
<b>8</b>	<b>Datenbankanwendungen mit ADO.NET</b> .....	353
<hr/>		
<b>8.1</b>	<b>Was sind relationale Datenbanken?</b> .....	353
8.1.1	Beispiel »Lager« .....	354
8.1.2	Indizes .....	357
8.1.3	Relationen .....	358
8.1.4	Übungen .....	362
<b>8.2</b>	<b>Anlegen einer Datenbank in Microsoft Access</b> .....	364
8.2.1	Aufbau von Access .....	364
8.2.2	Datenbankentwurf in Access 2013 .....	365
8.2.3	Übungen .....	370
<b>8.3</b>	<b>Datenbankzugriff mit Visual Basic</b> .....	370
8.3.1	Beispiel-Datenbank .....	371
8.3.2	Ablauf eines Zugriffs .....	371
8.3.3	Verbindung .....	372
8.3.4	SQL-Befehl .....	372
8.3.5	OleDb .....	372
8.3.6	Auswahlabfrage .....	373
8.3.7	Aktionsabfrage .....	376

<b>8.4</b>	<b>SQL-Befehle</b> .....	378
8.4.1	Auswahl mit select .....	378
8.4.2	Ändern mit update .....	383
8.4.3	Löschen mit delete .....	383
8.4.4	Einfügen mit insert .....	384
8.4.5	Typische Fehler in SQL .....	384
<b>8.5</b>	<b>Ein Verwaltungsprogramm</b> .....	385
8.5.1	Initialisierung .....	386
8.5.2	Alle Datensätze sehen .....	387
8.5.3	Datensatz einfügen .....	389
8.5.4	Datensatz ändern .....	391
8.5.5	Datensatz löschen .....	394
8.5.6	Datensatz suchen .....	396
<b>8.6</b>	<b>Abfragen über mehrere Tabellen</b> .....	398
<b>8.7</b>	<b>Verbindung zu MySQL</b> .....	403
8.7.1	.NET-Treiber .....	403
<b>9</b>	<b>Internetanwendungen mit ASP.NET</b> .....	407
<b>9.1</b>	<b>Grundlagen von Internetanwendungen</b> .....	407
9.1.1	Statische Internetanwendungen .....	407
9.1.2	Dynamische Internetanwendungen .....	408
9.1.3	Vorteile von ASP.NET .....	409
<b>9.2</b>	<b>Ein lokaler Webserver</b> .....	409
9.2.1	Eine erste Internetanwendung .....	410
<b>9.3</b>	<b>Eine erste ASP.NET-Anwendung</b> .....	412
9.3.1	Fehlerhafte Programmierung .....	414
<b>9.4</b>	<b>Formatierung von Internetseiten</b> .....	415
<b>9.5</b>	<b>Senden und Auswerten von Formulardaten</b> .....	416
<b>9.6</b>	<b>Weitere Formularelemente</b> .....	419
<b>9.7</b>	<b>Ein Kalenderelement</b> .....	422
<b>9.8</b>	<b>ASP.NET und ADO.NET</b> .....	424
<b>9.9</b>	<b>Datenbank im Internet ändern</b> .....	427

<b>10</b>	<b>Zeichnen mit GDI+</b>	433
<b>10.1</b>	<b>Grundlagen von GDI+</b>	433
<b>10.2</b>	<b>Linie, Rechteck, Polygon und Ellipse zeichnen</b>	433
10.2.1	Grundeinstellungen	434
10.2.2	Linie	435
10.2.3	Rechteck	435
10.2.4	Polygon	436
10.2.5	Ellipse	437
10.2.6	Dicke und Farbe ändern, Zeichnung löschen	437
<b>10.3</b>	<b>Text schreiben</b>	438
<b>10.4</b>	<b>Bilder darstellen</b>	441
<b>10.5</b>	<b>Dauerhaft zeichnen</b>	443
<b>10.6</b>	<b>Zeichnen einer Funktion</b>	444
<b>11</b>	<b>Beispielprojekte</b>	447
<b>11.1</b>	<b>Spielprogramm Tetris</b>	447
11.1.1	Spielablauf	447
11.1.2	Programmbeschreibung	448
11.1.3	Steuerelemente	449
11.1.4	Initialisierung des Programms	450
11.1.5	Erzeugen eines neuen Panels	452
11.1.6	Der Zeitgeber	454
11.1.7	Panel löschen	455
11.1.8	Panel seitlich bewegen	459
11.1.9	Panel nach unten bewegen	459
11.1.10	Pause	460
<b>11.2</b>	<b>Lernprogramm Vokabeln</b>	461
11.2.1	Benutzung des Programms	461
11.2.2	Erweiterung des Programms	463
11.2.3	Initialisierung des Programms	463
11.2.4	Ein Test beginnt	464
11.2.5	Zwei Hilfsprozeduren	467
11.2.6	Die Antwort prüfen	468
11.2.7	Das Benutzermenü	469

<b>12</b>	<b>Windows Presentation Foundation</b>	473
12.1	Layout .....	474
12.2	Steuerelemente .....	477
12.3	Frame-Anwendung .....	480
12.4	Zweidimensionale Grafik .....	483
12.5	Dreidimensionale Grafik .....	486
12.6	Animation .....	490
12.7	WPF und Windows Forms .....	493
12.7.1	Windows Forms in WPF .....	493
12.7.2	WPF in Windows Forms .....	495
<b>13</b>	<b>Windows Store-Apps für Windows 8.1</b>	499
13.1	Projektvorlagen für Windows Store-Apps .....	499
13.2	Projektvorlage Blank .....	501
13.3	Steuerelemente .....	502
13.4	Seitenvorlagen für Windows Store-Apps .....	506
13.5	Eine Reihe von Seiten .....	507
13.6	Eine geteilte Seite .....	510
13.7	Seitenvorlage Standardseite .....	512
13.8	Projektvorlage Grid .....	515
13.9	Projektvorlage Split .....	517
13.10	Prüfen einer App .....	518
<b>A</b>	<b>Installation und technische Hinweise</b>	521
A.1	Inhalt des Datenträgers zu diesem Buch .....	521
A.2	Installation der Express-Versionen von Visual Studio 2013 .....	521
A.3	Arbeiten mit einer Formularvorlage .....	522

A.4	Arbeiten mit einer Projektvorlage .....	523
A.5	Weitergabe eigener Windows-Programme .....	524
A.6	Konfigurationsdaten .....	525
A.7	Datenbankzugriff unter Vista 64-Bit-Version .....	527
<b>B</b>	<b>Lösungen der Übungsaufgaben</b> .....	<b>529</b>
B.1	Lösung der Übungsaufgabe aus Kapitel 1 .....	529
B.2	Lösungen der Übungsaufgaben aus Kapitel 2 .....	529
B.3	Lösungen der Übungsaufgaben aus Kapitel 4 .....	544
B.4	Lösungen der Übungsaufgaben aus Kapitel 8 .....	548
Index	.....	551

# Kapitel 2

## Grundlagen

*In diesem Kapitel erlernen Sie auf anschauliche Weise die Sprachgrundlagen von Visual Basic in Verbindung mit den gängigen Steuerelementen von Windows-Programmen.*

In den folgenden Abschnitten lernen Sie wichtige Elemente der Programmierung, wie Variablen, Operatoren, Verzweigungen und Schleifen, gemeinsam mit wohlbekanntem und häufig verwendeten Steuerelementen kennen.

### 2.1 Variablen und Datentypen

Variablen dienen zur vorübergehenden Speicherung von Daten, die sich zur Laufzeit eines Programms ändern können. Eine Variable besitzt einen eindeutigen Namen, unter dem sie angesprochen werden kann.

#### 2.1.1 Namen und Werte

Für die Namen von Variablen gelten in Visual Basic die folgenden Regeln:

**Namensregeln**

- ▶ Sie beginnen mit einem Buchstaben.
- ▶ Sie können nur aus Buchstaben, Zahlen und einigen wenigen Sonderzeichen (z. B. dem Unterstrich `_`) bestehen.
- ▶ Innerhalb eines Gültigkeitsbereichs darf es keine zwei Variablen mit dem gleichen Namen geben (siehe Abschnitt 2.1.4).

Variablen erhalten ihre Werte durch Zuweisung per Gleichheitszeichen. Falls eine Variable als Erstes auf der rechten Seite des Gleichheitszeichens genutzt wird, sollte ihr vorher ein Wert zugewiesen werden. Dadurch werden Programme eindeutiger, lesbarer und fehlerfreier.

### 2.1.2 Deklarationen

Neben dem Namen besitzt jede Variable einen Datentyp, der die Art der Information bestimmt, die gespeichert werden kann. Sie als Entwickler wählen den Datentyp danach aus, ob Sie Text, Zahlen ohne Nachkommastellen, Zahlen mit Nachkommastellen oder z. B. logische Werte speichern möchten.

#### Auswahl des Datentyps

Außerdem müssen Sie sich noch Gedanken über die Größe des Bereichs machen, den die Zahl oder der Text annehmen könnte, und über die gewünschte Genauigkeit bei Zahlen. Im folgenden Abschnitt 2.1.3 finden Sie eine Liste der Datentypen.

Variablen sollten in Visual Basic immer mit einem Datentyp deklariert werden. Das beugt Fehlern vor, die aufgrund einer falschen Verwendung der Variablen entstehen könnten.

### 2.1.3 Datentypen

Die folgende Liste enthält die wichtigsten von Visual Basic unterstützten Datentypen mit ihrem jeweiligen Wertebereich.

- ▶ Datentyp **Boolean**, Werte `True` oder `False` (*wahr* oder *falsch*)
- ▶ Datentyp **Byte**, ganze Zahlen von 0 bis 255
- ▶ Datentyp **Char**, einzelne Zeichen
- ▶ Datentyp **Date**, Datumsangaben vom 1. Januar des Jahres 1 bis zum 31. Dezember 9999
- Double** ▶ Datentyp **Double**, Gleitkommazahl mit doppelter Genauigkeit, Werte von  $-1,79769313486231570 \times 10^{308}$  bis  $4,94065645841246544 \times 10^{308}$  im negativen Bereich und von  $4,94065645841246544 \times 10^{308}$  bis  $1,79769313486231570 \times 10^{308}$  im positiven Bereich
- Integer** ▶ Datentyp **Integer**, ganze Zahlen von  $-2.147.483.648$  bis  $2.147.483.647$
- ▶ Datentyp **Long**, ganze Zahlen von  $-9.223.372.036.854.775.808$  bis  $9.223.372.036.854.775.807$
- ▶ Datentyp **Object**, beliebige Werte
- ▶ Datentyp **Short**, ganze Zahlen von  $-32768$  bis  $32767$
- ▶ Datentyp **Single**, Gleitkommazahl mit einfacher Genauigkeit; Werte von  $3,4028235 \times 10^{38}$  bis  $-1,401298 \times 10^{-45}$  im negati-

ven Bereich und  $1,401298 \cdot 10^{\text{hoch } -45}$  bis  $3,4028235 \cdot 10^{\text{hoch } 38}$  im positiven Bereich

- ▶ Datentyp **String**, Zeichenkette mit variabler Länge **String**
- ▶ benutzerdefinierte Struktur, jedes Element hat seinen eigenen Datentyp und damit seinen eigenen Wertebereich

Im folgenden Beispiel werden Variablen der wichtigsten Typen deklariert, mit Werten versehen und in einem Label angezeigt (Projekt *Datentypen*).

```
Public Class Form1
    Private Sub cmdAnzeige_Click(...) Handles ...
        Dim Bo As Boolean
        Dim By As Byte
        Dim Ch As Char
        Dim Dt As Date
        Dim Db As Double
        Dim It As Integer
        Dim Lg As Long
        Dim Sh As Short
        Dim Sg As Single
        Dim St As String

        Bo = True
        By = 200
        Ch = "a"
        Dt = "18.02.2014"
        Db = 1 / 7
        It = 2000000000
        Lg = 3000000000
        Sh = 30000
        Sg = 1 / 7
        St = "Zeichenkette"

        lblAnzeige.Text =
            "Boolean: " & Bo & vbCrLf &
            "Byte: " & By & vbCrLf &
            "Char: " & Ch & vbCrLf &
            "Double: " & Db & vbCrLf &
            "Date: " & Dt & vbCrLf &
            "Integer: " & It & vbCrLf &
```



```
        "Long: " & Lg & vbCrLf &  
        "Short: " & Sh & vbCrLf &  
        "Single: " & Sg & vbCrLf &  
        "String: " & St  
    End Sub  
End Class
```

### Listing 2.1 Projekt »Datentypen«

Das Programm hat nach Betätigung des Buttons die Ausgabe wie in Abbildung 2.1 dargestellt.

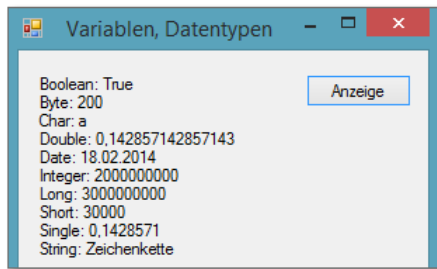


Abbildung 2.1 Wichtige Datentypen

Zur Erläuterung:

- ▶ Variablen werden mit `Dim ... As ...` deklariert.
- Wertebereich** ▶ Bei den Zahlen-Datentypen führt eine Über- oder Unterschreitung des Wertebereichs zu einer Fehlermeldung.
- Genauigkeit** ▶ Die Datentypen `Single` und `Double` für Zahlen mit Nachkommastellen unterscheiden sich in ihrer Genauigkeit.
- ▶ Werte für Zeichen, Zeichenketten und Datumsvariablen müssen in doppelten Anführungszeichen angegeben werden.

Mehrere Variablen des gleichen Typs können, durch Kommata getrennt, in einer Zeile deklariert werden (z. B. `Dim x, y As Integer`).

### Übung

#### Übung ÜDatentypen

Schreiben Sie ein Programm, in dem Ihr Nachname, Vorname, Ihre Adresse, Ihr Geburtsdatum und Ihr Alter jeweils in Variablen eines geeigneten Datentyps gespeichert und anschließend wie in Abbildung 2.2 ausgegeben werden.

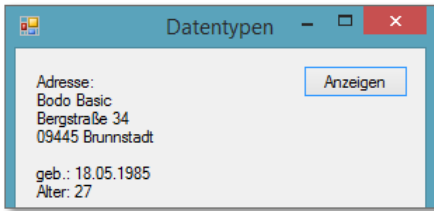


Abbildung 2.2 Übung ÜDatentypen

### 2.1.4 Gültigkeitsbereich

Variablen, die innerhalb einer Prozedur vereinbart wurden, haben ihre Gültigkeit nur in der Prozedur. Außerhalb der Prozedur sind sowohl Name als auch Wert unbekannt. Solche Variablen bezeichnet man auch als lokale Variablen. Sobald die Prozedur abgearbeitet wurde, steht der Wert auch nicht mehr zur Verfügung. Beim nächsten Aufruf der gleichen Prozedur werden diese Variablen neu deklariert und erhalten neue Werte.

Lokal

Anders verhält es sich mit statischen Variablen. Diese behalten ihren Wert, solange das Programm läuft. Ein wiederholter Aufruf der gleichen Prozedur kann auf den letzten gespeicherten Wert einer Variablen zugreifen. Eine statische Variable vereinbaren Sie z. B. wie folgt: `Static Sx As Integer`.

Statisch

Variablen, die außerhalb von Prozeduren vereinbart werden, sind innerhalb der gesamten Klasse gültig, hier also innerhalb der Klasse des Formulars. Ihr Wert kann in jeder Prozedur gesetzt oder abgerufen werden und bleibt erhalten, solange das Formular im laufenden Programm existiert. Sie können sie auch mit dem Schlüsselwort `Private` deklarieren: `Private Mx As Integer`. Weitere Einzelheiten zu klassenweit gültigen Variablen finden Sie in Kapitel 5.

Klassenweit gültig

Variablen, die mit dem Schlüsselwort `Public` vereinbart werden, sind öffentlich. Damit sind sie auch außerhalb der jeweiligen Klasse, also z. B. auch in anderen Formularen, gültig. Mehr dazu in Abschnitt 4.4.

Public

Gibt es in einem Programmabschnitt mehrere Variablen mit dem gleichen Namen, gelten folgende Regeln:

- ▶ Lokale Variablen mit gleichem Namen in der gleichen Prozedur sind nicht zulässig.
- ▶ Eine klassenweit gültige Variable wird innerhalb einer Prozedur von einer lokalen Variablen mit dem gleichen Namen ausgeblendet.

Ausblenden