
Ricardo Hernández Garcia

1. Ausgabe, Juni 2016

ISBN 978-3-86249-566-5

VBA-Programmierung

**Integrierte Lösungen
mit Office 2016**

VBA2016



HERDT

In der Hilfe finden Sie unter dem Suchbegriff *Logische Operatoren* weitere logische Operatoren.



Verkettungsoperatoren

Mit dem Verkettungsoperator (&) werden Zeichenfolgen verbunden.

Operator	Operation	Beispiel (a = "Hallo", b = " Welt")
&	Zeichenfolgenverknüpfung	a & b → "Hallo Welt"

4.10 Programmablauf mit Kontrollstrukturen steuern

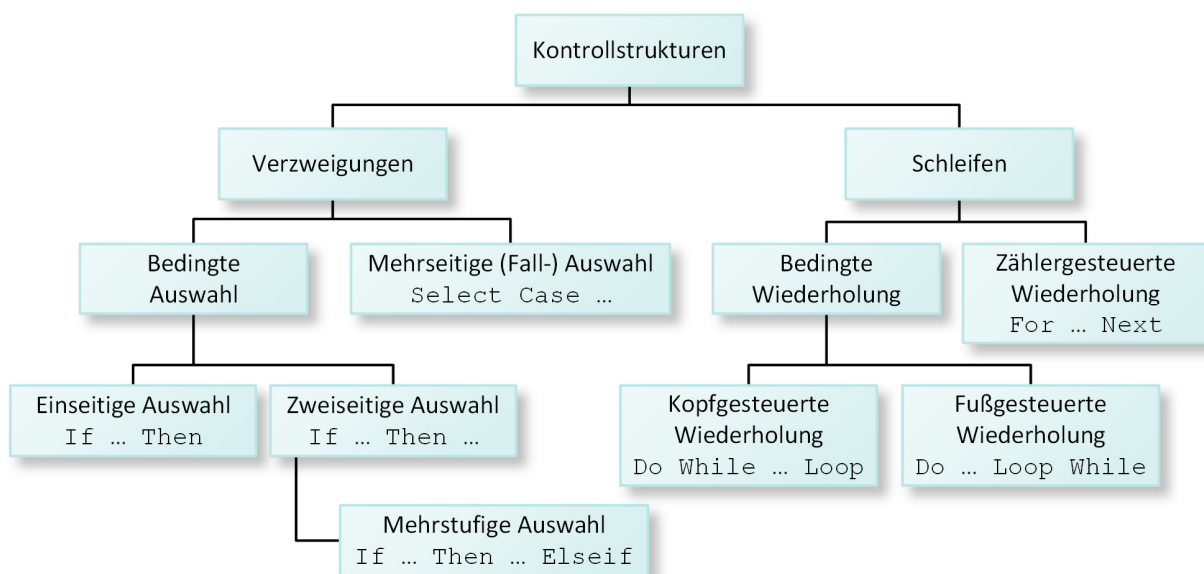
Plus **Beispieldatei:** Kap04.xlsm

Die Programme, die Sie bisher kennengelernt haben, bestanden aus einer Folge von Anweisungen, die genau einmal sequenziell (der Reihe nach) abgearbeitet wurden. Oft ist es jedoch erforderlich, dass Programmteile mehrmals oder gar nicht ausgeführt werden. Die VBA-Sprachelemente, mit denen der Programmablauf gesteuert werden kann, werden **Kontrollstrukturen** genannt. Die Entscheidung, nach welchen Kriterien der Ablauf gesteuert wird, wird in **Bedingungen** formuliert.

Es werden zwei Gruppen von Kontrollstrukturen unterschieden:

- ✓ **Verzweigungen:**
Es werden alternative Programmteile angeboten, in die – abhängig von einer Bedingung – beim Programmablauf verzweigt wird.
- ✓ **Schleifen (engl. Loops):**
Ein Programmteil kann – abhängig von einer Bedingung – mehrmals durchlaufen werden.

Übersicht der Kontrollstrukturen in VBA



Übersicht der Verzweigungsarten in VBA

Verzweigungsart	Syntax	Beispiel
Einseitige Auswahl If-Then-Anweisung	If Ausdruck Then Anweisung1 Anweisung2 ... End If	If Zähler = 10 Then MsgBox "Berechnung beendet" End If
Zweiseitige Auswahl If-Then-Else-Anweisung	If Ausdruck Then Anweisungsblock1 Else Anweisungsblock2 End If	If GesamtPreis >= 100 Then Ausgabe = "Rabatt möglich!" Else Ausgabe = "Kein Rabatt möglich!" End If
Mehrstufige Auswahl If-Then-ElseIf-Anweisung	If Ausdruck1 Then Anweisungsblock1 ElseIf Ausdruck2 Then Anweisungsblock2 ElseIf Ausdruck3 Then Anweisungsblock3 Else Anweisungsblock End If	If GesamtPreis < 100 Then Ausgabe = "Kein Rabatt möglich!" ElseIf GesamtPreis < 200 Then Ausgabe = "5% Rabatt möglich!" ElseIf GesamtPreis < 250 Then Ausgabe = "10% Rabatt möglich!" Else Ausgabe = "15% Rabatt möglich!" End If
Fallauswahl Select-Case-Anweisung	Select Case Selektor Case Auswahlwert1 Anweisungsblock1 Case Auswahlwert2 Anweisungsblock2 ... Case Else Anweisungsblock End Select	Select Case Note Case 1 Text = "Note Sehr gut" Case 2 Text = "Note Gut" Case 3 To 4 Text = "Note Befriedigend" Case 5 To 6 Text = "Note Ungenügend" Case Else Text = "Ungültige Eingabe" End Select



Bei einer Fallauswahl können auch Operatoren wie `Is` zum Einsatz kommen, z. B. `Case Is < 20`.

Beispiel: *Kap04.xlsm*, Modul **Kontrollstrukturen**

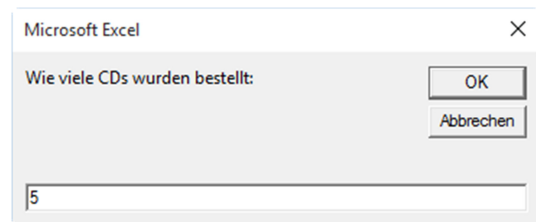
Ein CD-Händler gewährt je nach Bestellmenge einen unterschiedlich hohen Rabatt. In Abhängigkeit von einem festen Preis und einer einzugebenden Bestellmenge sollen der Rechnungsbetrag und der dafür mögliche Rabattwert angezeigt werden.

```

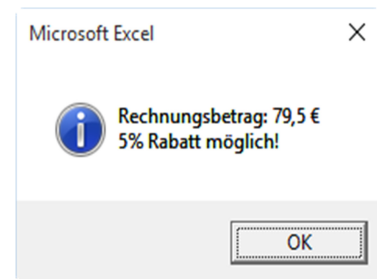
Sub Rabatt()
  Dim Anzahl As String, Ausgabe As String
  Dim EinzelPreis As Double, GesamtPreis As Double
  ① Anzahl = InputBox("Wie viele CDs wurden bestellt:")
  ② If Anzahl <> "" Then
    EinzelPreis = 15.90
    ③ GesamtPreis = Val(Anzahl) * EinzelPreis
    Ausgabe = "Rechnungsbetrag: " & GesamtPreis & " €" & vbCrLf
    ④ If GesamtPreis < 50 Then
      Ausgabe = Ausgabe & "Kein Rabatt möglich!"
    ⑤ ElseIf GesamtPreis < 100 Then
      Ausgabe = Ausgabe & "5% Rabatt möglich!"
    ⑥ ElseIf GesamtPreis < 150 Then
      Ausgabe = Ausgabe & "10% Rabatt möglich!"
    ⑦ Else
      Ausgabe = Ausgabe & "15% Rabatt möglich!"
    End If
    ⑧ MsgBox Ausgabe, vbOKOnly + vbInformation
  End If
End Sub

```

- ① Mit einem Eingabedialog wird die Anzahl der bestellten CDs abgefragt und als Zeichenfolge in der Variablen `Anzahl` gespeichert.
- ② Wird der Eingabedialog mit der Schaltfläche *Abbrechen* beendet oder wurde keine Anzahl eingegeben, enthält die Variable `Anzahl` eine leere Zeichenfolge. Die `If`-Anweisung überprüft dies und führt den folgenden Anweisungsblock nur aus, wenn die Variable nicht leer ist.
- ③ Mit der Funktion `Val` wird die Zeichenfolge `Anzahl` in einen numerischen Wert konvertiert.
- ④ Ist der Gesamtpreis unter 50 €, erfolgt ein Hinweis, dass kein Rabatt möglich ist.
- ⑤ Das Schlüsselwort `ElseIf` leitet eine neue, untergeordnete Bedingung ein, die überprüft wird, wenn der Gesamtpreis nicht unter 50 € liegt. Falls der Preis nun unter 100 € liegt, erfolgt ein entsprechender Hinweis.
- ⑥ Hier wird überprüft, ob der Gesamtpreis unter 150 € liegt. Wenn ja, wird ein Hinweis zur Ausgabe hinzugefügt.
- ⑦ Falls alle bisherigen Bedingungen falsch waren, wird die Anweisung nach `Else` ausgeführt. Dies ist für alle Gesamtpreise der Fall, die über 150 € liegen.
- ⑧ Das Meldungsfenster mit dem Ausgabebetext, der Schaltfläche *OK* und einem Informationssymbol (vgl. Abschnitt 6.7) wird angezeigt.



Eingeben der Anzahl bestellter CDs

Ausgabe der Prozedur *Rabatt*

Einfache Wertzuweisung mit einer Bedingung

Oft soll mit einer `If-Then-Else`-Anweisung kein großer Anweisungsblock ausgeführt werden, sondern nur eine Wertzuweisung an eine Variable in Abhängigkeit von einer Bedingung erfolgen. Für diesen Fall verfügt VBA über die vordefinierte Funktion `Iif`.

```
Variable = Iif(Ausdruck, Wert1, Wert2)
```

Ergibt die Auswertung des Ausdrucks den Wert `True`, wird der Variablen der Wert1 zugewiesen, sonst der Wert2.

Schleifen

Mit Schleifen (auch Wiederholungen genannt) wird eine bestimmte Anweisung bzw. ein bestimmter Anweisungsblock wiederholt ausgeführt. VBA unterscheidet dabei zählergesteuerte und bedingte Schleifen.

Übersicht der Schleifen in VBA

Schleife	Syntax	Beispiel
Zählergesteuerte Wiederholung For-Next-Schleife	<pre>For Zähler = Start To Ende [Step d] Anweisungsblock Next Zähler</pre> <p>Optional können Sie die Schrittweite durch Step und einen anschließenden Zahlenwert festlegen. Ohne diese Angabe wird die Schrittweite 1 verwendet.</p>	<pre>For Zähler = 1 To 20 Debug.Print "Nr. " & Zähler Next</pre>
Kopfgesteuerte bedingte Wiederholung Do-While/Until-Schleife	<pre>Do While/Until Ausdruck Anweisungsblock Loop</pre>	<pre>Do While Zahl2 <> 0 temp = Zahl2 Zahl2 = Zahl1 Mod Zahl2 Zahl1 = temp Loop</pre>
Fußgesteuerte bedingte Wiederholung Do-Loop While/Until-Schleife	<pre>Do Anweisungsblock Loop While/Until Bedingung</pre>	<pre>Do temp = Zahl2 Zahl2 = Zahl1 Mod Zahl2 Zahl1 = temp Loop Until Zahl2 = 0</pre>



Bei der Prüfung des Ausdrucks nach dem Schlüsselwort `While` wird eine Schleife so lange ausgeführt, bis die Auswertung den Wert `False` ergibt. Mit dem Schlüsselwort `Until` wird die Schleife dagegen durchlaufen, bis die Bedingung wahr ist.

Beispiel: *Kap04.xlsm*, Modul **Kontrollstrukturen**

Eine Prozedur soll die Anzahl der Tage von heute bis zu einem einzugebenden Datum berechnen. Der Eingabedialog soll so oft angezeigt werden, bis der Anwender die Schaltfläche *Abbrechen* wählt.

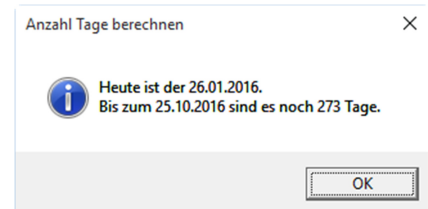
```

Sub AnzahlTage ()
  Dim Heute As Date, Datum As String, Ausgabe As String, Auswahl As Integer
  ① Heute = Date
  ② Do
  ③   Datum = InputBox("Berechne die Anzahl der Tage bis zum:", _
                     "Anzahl Tage berechnen")
  ④   If Datum <> "" Then
  ⑤     Datum = DateValue(Datum)
     Ausgabe = "Heute ist der " & Heute & "." & vbCrLf & _
               "Bis zum " & Datum & " sind es noch " & _
  ⑥     DateDiff("d", Heute, Datum) & " Tage."
     Auswahl = MsgBox(Ausgabe, vbOKOnly + vbInformation, _
                     "Anzahl Tage berechnen")

     End If
  ⑦   Loop While Datum <> ""
End Sub

```

- ① Die Funktion `Date` liefert das aktuelle Systemdatum.
- ② Mit dem Schlüsselwort `Do` beginnt die Schleifenstruktur. Die Schleife wird mindestens einmal durchlaufen, denn die Bedingung wird erst am Ende geprüft.
- ③ Die Funktion `InputBox` erzeugt ein Eingabefenster, das den Anwender zur Eingabe eines Datums auffordert.
- ④ Wenn der Anwender die Schaltfläche *Abbrechen* betätigt, enthält `Datum` eine leere Zeichenfolge. In diesem Fall soll keine Berechnung erfolgen.
- ⑤ Mit der Funktion `DateValue` wird das eingegebene Datum in ein gültiges Format umgewandelt.
- ⑥ Die Funktion `DateDiff` berechnet die Differenz zwischen zwei Datumsangaben. Da das Ergebnis in Tagen ausgegeben werden soll, wird als Argument der Wert "d" übergeben.
- ⑦ Nach dem Schlüsselwort `Loop` beginnt die Prüfung des Ausdrucks, der die Abbruchbedingung enthält. In diesem Fall wird die Schleife so lange ausgeführt, bis der Anwender die Schaltfläche *Abbrechen* betätigt und die Variable `Datum` eine leere Zeichenfolge enthält.



Ausgabe der Prozedur `AnzahlTage`

Schleifenabbruch mit `Exit`

Sie haben die Möglichkeit, Schleifen in Abhängigkeit von einer Bedingung vorzeitig zu verlassen, um so z. B. auf einen Abbruch durch den Anwender oder eine unzulässige Eingabe zu reagieren.

Mit der Anweisung `Exit` können sowohl zählergesteuerte als auch bedingte Wiederholungen abgebrochen werden:

- ✓ Eine zählergesteuerte Wiederholung wird mit `Exit For` abgebrochen.
- ✓ Eine bedingte Wiederholung kann mit `Exit Do` verlassen werden.

Mit der Anweisung `Exit Sub` können Sie die Ausführung einer Prozedur abbrechen.



4.11 Schnellübersichten

Was bedeutet ... ?	
Variable	Symbolischer Bezeichner für einen Speicherbereich im Computer, in dem während des Programmablaufs Werte gespeichert werden können. Diese Werte lassen sich jederzeit auslesen und verändern.
Konstante	Symbolischer Bezeichner für einen Speicherbereich im Computer, dem einmalig ein Wert zugewiesen wird. Der Wert kann nach der Definition nicht mehr geändert werden.
Datentyp	Legt fest, welche Daten eine Variable oder Konstante speichern kann (Wertebereich), und bestimmt die Speicherplatzgröße
Anweisung	Arbeitsschritt, den ein Programm ausführen soll und der nach bestimmten Regeln (Syntax) geschrieben wird
Deklaration	Vereinbarung von Variablen oder Konstanten im Programm und Zuweisen eines Datentyps und/oder eines Anfangswerts
Array	Datenfeld, das mehrere Elemente eines Datentyps enthält, auf die über einen Index zugegriffen wird
Ausdruck	Ein Ausdruck ist allgemein eine Kombination aus Werten, Variablen, Konstanten und Operatoren.

Sie möchten ...	
eine Sub-Prozedur deklarieren	Sub ... End Sub
eine Function-Prozedur deklarieren	Function ... End Function
Variablen deklarieren	Dim Variablenname [As Datentyp]
Konstanten deklarieren	Const Konstantenname [As Datentyp] = Wert
ein Array (Datenfeld) deklarieren	Dim Arrayname (Dimension1, Dimension2, ...) As Datentyp
auf einen Array-Wert zugreifen	Variable = Arrayname (Index1, Index2, ...)
mehrere Zeichenfolgen verknüpfen	String1 & String2 & ... oder String1 + String2 ...
Vergleichsoperatoren einsetzen	=, <, >, <>, <=, >=, Like , Is
logische Operatoren einsetzen	Not , And , Or , ...
eine Auswahl erstellen	If...Then... [ElseIf] ... [Else] ...End If
eine Fallauswahl verwenden	Select Case...Case...Case...Case Else...End Select
eine zählergesteuerte Wiederholung erstellen	For..To... [Step] ...Next
eine bedingte Wiederholung einsetzen	Do While/Until...Loop Do ... Loop While/Until

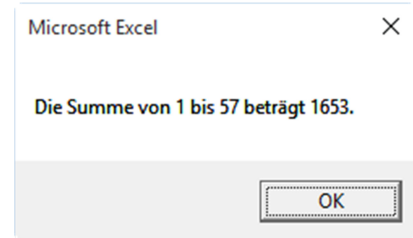
4.12 Übung

Übung 1: Zählergesteuerte Wiederholung

Übungsdatei: *Kontrollstrukturen.xlsm*

Ergebnisdatei: *Kontrollstrukturen-E.xlsm*

1. Öffnen Sie die Datei *Kontrollstrukturen.xlsm* und rufen Sie die VBA-Entwicklungsumgebung auf.
2. Erstellen Sie im Modul Übung eine Prozedur `AddZahlen`, die alle Zahlen von 1 bis zu einem Endwert addiert.
3. Benutzen Sie eine Konstante, um den Endwert festzulegen.
4. Addieren Sie alle Zahlen mit einer zählergesteuerten Wiederholung (For-Next-Schleife) von 1 bis zum Endwert.
5. Geben Sie das Ergebnis und den Endwert in einem Meldungsfenster aus.



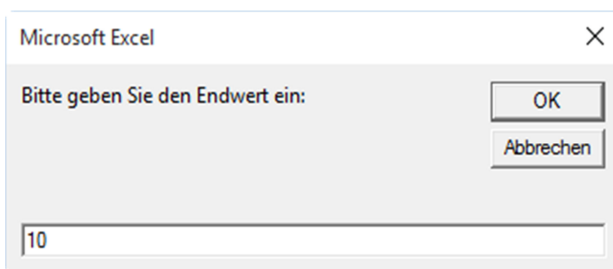
Ausgabe der Prozedur `AddZahlen`

Übung 2: Geschachtelte Schleifen

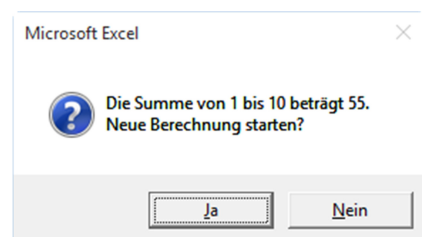
Übungsdatei: *Kontrollstrukturen.xlsm*

Ergebnisdatei: *Kontrollstrukturen-E.xlsm*

1. Erstellen Sie im Modul Übung eine neue Prozedur `AddZahlenWiederholt`, indem Sie die Prozedur aus der vorherigen Übung kopieren.
2. Der Endwert soll diesmal nicht mit einer Konstanten festgelegt, sondern durch einen Eingabedialog vom Anwender abgefragt werden. Verwenden Sie dazu die VBA-Funktion `InputBox` und speichern Sie die Rückgabe in einer Variablen.
3. Der Anwender soll auf Nachfrage eine neue Berechnung starten können. Erweitern Sie dazu das Meldungsfenster um eine Abfrage, ein Fragezeichen (Konstante `vbQuestion`) und die Schaltflächen Ja und Nein (Konstante `vbYesNo`).
4. Die Berechnung soll mindestens einmal durchgeführt werden und danach so oft, bis der Anwender die Schaltfläche Nein betätigt. Benutzen Sie dazu eine `Do-Loop-Until`-Schleife. Prüfen Sie dabei den Rückgabewert von `MsgBox` auf den Wert `vbNo`.



Endwert abfragen



Ausgabe und Wiederholungsabfrage

Impressum

Matchcode: VBA2016

Autor: Ricardo Hernández Garcia

Redaktion: Andrea Weikert

Produziert im HERDT-Digitaldruck

1. Ausgabe, Juni 2016

HERDT-Verlag für Bildungsmedien GmbH

Am Kümmerling 21-25

55294 Bodenheim

Internet: www.herd.com

E-Mail: info@herd.com

© HERDT-Verlag für Bildungsmedien GmbH, Bodenheim

Alle Rechte vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren) ohne schriftliche Genehmigung des Verlags reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Dieses Buch wurde mit großer Sorgfalt erstellt und geprüft. Trotzdem können Fehler nicht vollkommen ausgeschlossen werden. Verlag, Herausgeber und Autoren können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Wenn nicht explizit an anderer Stelle des Werkes aufgeführt, liegen die Copyrights an allen Screenshots beim HERDT-Verlag. Sollte es trotz intensiver Recherche nicht gelungen sein, alle weiteren Rechteinhaber der verwendeten Quellen und Abbildungen zu finden, bitten wir um kurze Nachricht an die Redaktion.

Die in diesem Buch und in den abgebildeten bzw. zum Download angebotenen Dateien genannten Personen und Organisationen, Adress- und Telekommunikationsangaben, Bankverbindungen etc. sind frei erfunden. Eventuelle Übereinstimmungen oder Ähnlichkeiten sind unbeabsichtigt und rein zufällig.

Die Bildungsmedien des HERDT-Verlags enthalten Verweise auf Webseiten Dritter. Diese Webseiten unterliegen der Haftung der jeweiligen Betreiber, wir haben keinerlei Einfluss auf die Gestaltung und die Inhalte dieser Webseiten. Bei der Bucherstellung haben wir die fremden Inhalte daraufhin überprüft, ob etwaige Rechtsverstöße bestehen. Zu diesem Zeitpunkt waren keine Rechtsverstöße ersichtlich. Wir werden bei Kenntnis von Rechtsverstößen jedoch umgehend die entsprechenden Internetadressen aus dem Buch entfernen.

Die in den Bildungsmedien des HERDT-Verlags vorhandenen Internetadressen, Screenshots, Bezeichnungen bzw. Beschreibungen und Funktionen waren zum Zeitpunkt der Erstellung der jeweiligen Produkte aktuell und gültig. Sollten Sie die Webseiten nicht mehr unter den angegebenen Adressen finden, sind diese eventuell inzwischen komplett aus dem Internet genommen worden oder unter einer neuen Adresse zu finden. Sollten im vorliegenden Produkt vorhandene Screenshots, Bezeichnungen bzw. Beschreibungen und Funktionen nicht mehr der beschriebenen Software entsprechen, hat der Hersteller der jeweiligen Software nach Drucklegung Änderungen vorgenommen oder vorhandene Funktionen geändert oder entfernt.