

Entwicklung einer benutzerdefinierten Funktion in MS-Excel

Eine Funktion (Excel Beispiele: SUMME, MAX etc.) ist eine eigenständige Programmeinheit, die aus jeder beliebigen Tabellenzelle aufgerufen wird, eine Berechnung durchführt und das Ergebnis an das aufrufende Programm (Ergebniszelle) zurückgibt.

Im Gegensatz zu den Standardfunktionen legt der Benutzer den Funktionsnamen, die Argumente und einen Rückgabewert selbst fest. Dabei können benutzerdefinierte Funktionen für den Einsatz in einem Tabellenblatt, für den Aufruf aus einem Makro und als Argument für eine andere Funktion entwickelt werden. Allgemeine Syntax einer benutzerdefinierten Funktion:

```

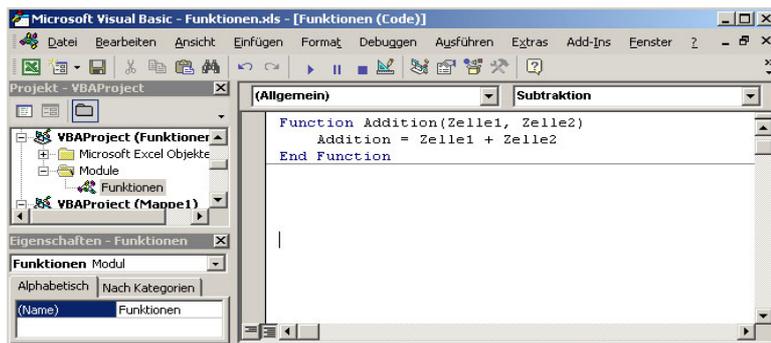
Function Funktionsname (Argument 1,..., Argument n)
    Anweisung 1
    ...
    Anweisung n
Funktionsname = Ausdruck
Ende Function
    
```

Vorgehensweise im VBA-Editor:

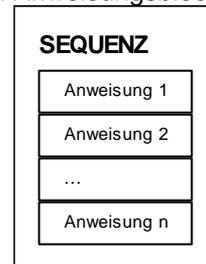
Legen Sie ein neues Modul an mit EINFÜGEN/MODUL. Sie können beobachten, dass sich das bisher leere Code – Fenster füllt und im Projektfenster ein Modul angelegt wird.

Geben Sie jetzt das Schlüsselwort *function* (entspricht dem Schlüsselwort *sub* beim Makro) gefolgt von dem Funktionsnamen ein. Achten Sie bei der Wahl des Funktionsnamens auf einen möglichst sprechenden Namen, und bestätigen Sie die Eingabe mit RETURN.

Grundstrukturen der Programmierung (1) - Folgestruktur



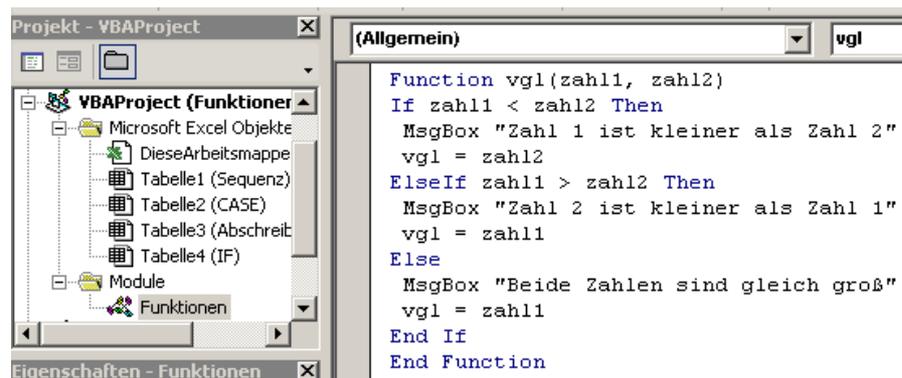
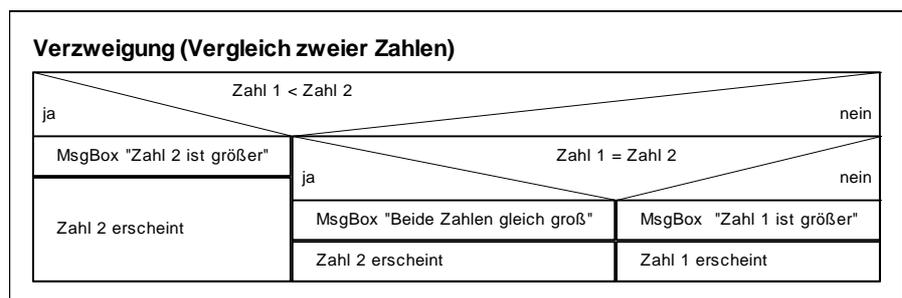
Die Struktur der Funktion „Addition“ ist eine so genannte **Folgestruktur** (Sequenz, lineare Programmstruktur) mit einem Anweisungsblock.



Grundstrukturen der Programmierung (2): Auswahlstrukturen (Selektion)

Im Lösungsalgorithmus fast jeden Problems kommt man an einen Punkt, an dem der weitere Fortgang von der Erfüllung oder Nichterfüllung einer Bedingung abhängig ist. Möglichkeiten sind:

- a) **Einseitige Verzweigung** (Sonst-Zweig bleibt leer)
- b) **Zweiseitige Verzweigung** (Klassische Wenn-Dann-Sonst-Beziehung)
- c) **Verschachtelte Verzweigung** (weitere Wenn-Anweisungen im Sonst-Fall)



Wie das Beispiel des Zahlenvergleichs (Struktogramm und Code-Editor) benötigt die Funktion „vgl“ die beiden Argumente **zahl1** und **zahl2**. Die normale *If* (Wenn) - Anweisung wird bei dieser Dreifachauswahl durch die *Elseif*-Anweisung ergänzt. Eine *If*-Anweisung muss durch eine *End If* – Anweisung beendet werden. Diese hat die Syntax:

```

If Bedingung 1 Then
    [Anweisungsblock 1]
[Elseif Bedingung 2 Then
    [Anweisungsblock 2]]
[Elseif Bedingung 3 Then
    [Anweisungsblock 3]
    usw.
    [Else
    [Anweisungsblock n]]
End If

```

Die im Beispiel verwendete Message-Box (MsgBox) zeigt eine Meldung in einem Dialogfeld an und wartet darauf, dass der Benutzer auf eine Schaltfläche klickt. Es wird dann einen Wert zurückgegeben, der anzeigt, auf welche Schaltfläche der Benutzer geklickt hat (in unserem Beispiel aber nur die Schaltfläche **OK**)

Grundstrukturen der Programmierung (3): Mehrfachauswahl (Case)

Mehrfach geschachtelte *If*-Anweisungen werden schnell unübersichtlich. Eine Alternative zur Blockform der *If*-Anweisung bildet die *Select Case* -Anweisung. Ihr Vorteil gegenüber der *If*- Anweisung liegt darin, dass sie in einfacherer und übersichtlicherer Form die Definition von Anweisungsgruppen erlaubt. Intern wird sie vom System in einen effizienteren Code umgesetzt und beschleunigt dadurch die Programmausführung. Ihre Syntax lautet:

```

Select Case Test
    [Case Werteliste 1
        [Anweisungen1]]
    [Case Werteliste 2
        [Anweisungen2]]
    [Case Else
        [Anweisungen n]]
End Select

```

So werden im nebenstehenden Beispiel 5 Fälle der Gehalts-Nettoberechnung untersucht, die in Abhängigkeit vom Bruttolohn einen vorgegebenen Steuersatz ermitteln. Umgesetzt wird diese Mehrfachverzweigung durch die *CASE*-Anweisung.

Gehalt?				
Fall 1	Fall 2	Fall 3	Fall 4	sonst
< 410,00 €	< 800,00 €	<1.200,00 €	<2.500,00 €	25% Steuern
keine Steuern	10 % Steuern	15 % Steuern	20% Steuern	

Im VBA-Code wird diese *Case*-Anweisung durch ein *Select Case* begonnen und mit *End Select* beendet. Werden feste Werte abgefragt, ist jeder Fall mit der Anweisung *Case Is* einzuleiten.

Steuern

Bruttogehalt	Nettogehalt	Steuersatz
300,00 €	300,00 €	0%
650,00 €	585,00 €	10%
900,00 €	765,00 €	15%
1.500,00 €	1.200,00 €	20%
2.500,00 €	1.875,00 €	25%

```

Function netto(gehalt)
Select Case gehalt
Case Is < 410
netto = gehalt
Case Is < 800
netto = gehalt * 0.9
Case Is < 1200
netto = gehalt * 0.85
Case Is < 2500
netto = gehalt * 0.8
Case Else
netto = gehalt * 0.75
End Select
End Function

```

Aufgabenstellung:

Erstellen Sie Funktionen zur Verkaufspreisberechnung nach folgendem Schema:

1. VPreis1: Faktor 2,5 vom Einkaufspreis
2. VPreis2: Liegt der Einkaufspreis unter 1.000,00 € wird der Faktor 1,5 vom EP genommen, sonst Faktor 2,5
3. VPreis3: Liegt der EP unter 1.000,00€ und ein Rabatt von über 5% des Lieferanten vor, so wird der VP mit dem Faktor 3 bewertet. Falls dies nicht der Fall ist wird Faktor 2 gewählt.
4. VPreis4: Preis unter 500: Faktor 2, Preis unter 1000: Faktor 2,5, Preis unter 1500: Faktor 3, Preis unter 2000: Faktor 3,5 ansonsten Faktor 4.