

01 Gültigkeitsbereiche von Variablen (Dim, Static, Public)

Teil 1: Variablen, die nur in Prozeduren gültig sind

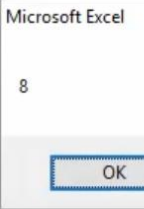
• Lokale Variablen:

```
Modul1
Sub Makro1()
    'Lokale Variablen definieren
    Dim i As Integer

    'Der Variable einen Wert zuweisen
    i = 8

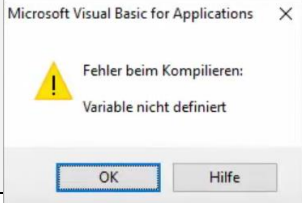
    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox i

End Sub
```



```
Modul1
Sub Makro2()
    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox i

End Sub
```



Lokale Variablen haben kein Gedächtnis:

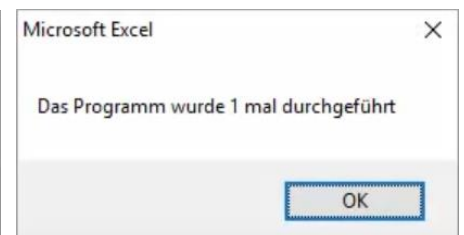
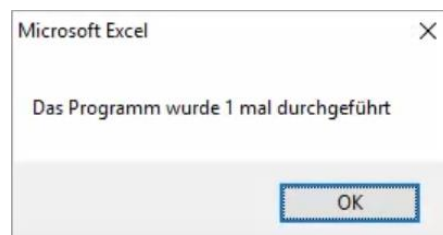
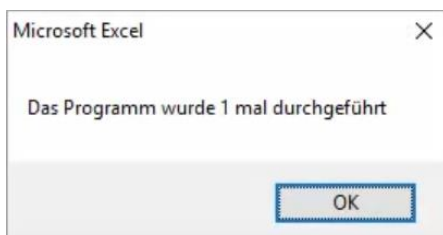
Das bedeutet, sobald die Prozedur mit *End Sub* beendet wird, wird der aktuelle Variablen-Wert wieder gelöscht. Egal wie oft wir die Prozedur „Beispiel1“ ausführen, es wird immer die gleiche Nachricht in der MsgBox angezeigt.

```
Modul1
Sub Beispiel1()
    'Lokale Variablen definieren
    Dim i As Integer

    'Den Wert der Variable immer um 1 erhöhen
    i = i + 1

    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox "Das Programm wurde " & i & " mal durchgeführt"

End Sub
```





01 Gültigkeitsbereiche von Variablen (Dim, Static, Public)

Statische Variablen haben ein Gedächtnis:

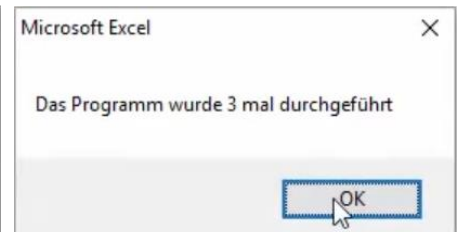
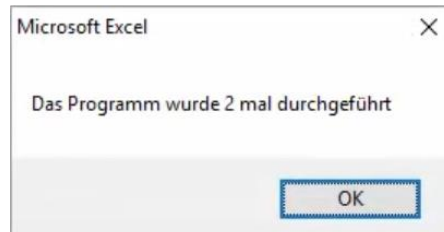
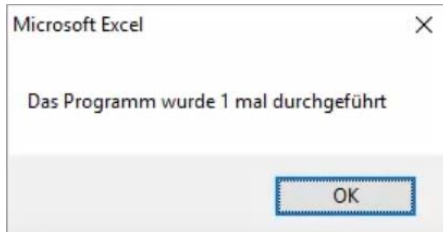
Das bedeutet, auch nachdem die Prozedur durch *End Sub* beendet wurde, bleibt der aktuelle Variablen-Wert gespeichert. Immer wenn wir die Prozedur „Beispiel1“ ausführen, wird die Zahl in der MsgBox um 1 erhöht.

```
Modul1
Sub Beispiel1()
    'Lokale Variablen definieren
    Static i As Integer

    'Den Wert der Variable immer um 1 erhöhen
    i = i + 1

    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox "Das Programm wurde " & i & " mal durchgeführt"

End Sub
```



01 Gültigkeitsbereiche von Variablen (Dim, Static, Public)

Teil 2: Variablen, die für ein Modul gültig sind


```
Modul1
'Variablen definieren, die für das gesamte Modul gilt
Dim b as Integer

Sub Makro1()

    'Der Variable einen Wert zuweisen
    b = 7

    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox b

End Sub
```




```
Modul1

Sub Makro2()

    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox b

End Sub
```



Diese Variablen haben ein Gedächtnis:

Das bedeutet, auch nachdem die Prozedur durch *End Sub* beendet wurde, bleibt der aktuelle Variablen-Wert gespeichert. Immer wenn wir die Prozedur „Beispiel2“ ausführen, wird die Zahl in der MsgBox um 1 erhöht.

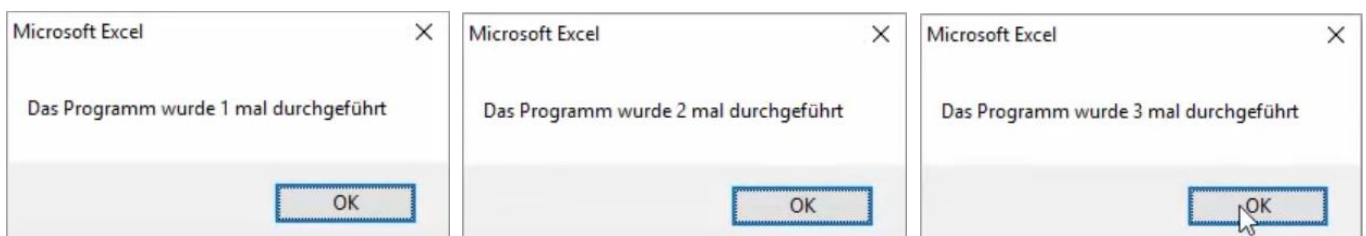
```
Modul1
'Variablen definieren, die für das gesamte Modul gilt
Dim b as Integer

Sub Beispiel2()

    'Den Wert der Variable immer um 1 erhöhen
    b = b + 1

    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox "Das Programm wurde " & b & " mal durchgeführt"

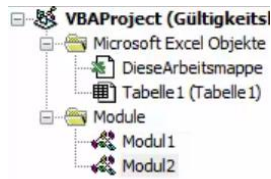
End Sub
```



01 Gültigkeitsbereiche von Variablen (Dim, Static, Public)

Teil 3: Variablen, die für alle Module gültig sind

Für dieses Beispiel erstellen wir ein zusätzliches Modul, sodass wir Modul1 und Modul2 haben.



- Öffentliche Variablen:

Modul1

```
'Variablen definieren, die für alle Module gilt
Public b as Integer

Sub Makro1 ()

    'Der Variable einen Wert zuweisen
    b = 4

    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox b

End Sub
```

The screenshot shows a 'Microsoft Excel' dialog box with a close button (X) in the top right corner. The dialog box contains the number '4' and an 'OK' button at the bottom.

Modul2

```
Sub Makro3 ()

    'Den Wert der Variable in einer MsgBox ausgeben
    MsgBox b

End Sub
```

The screenshot shows a 'Microsoft Excel' dialog box with a close button (X) in the top right corner. The dialog box contains the number '4' and an 'OK' button at the bottom.



01 Gültigkeitsbereiche von Variablen (Dim, Static, Public)

Zusammenfassung

1. Variablen, die nur in Prozeduren gültig sind	
<p>Lokale Variablen</p> <ul style="list-style-type: none">• werden im Bereich der Prozedur definiert• Beispiel: Dim i As Integer• Diese Variablen haben kein Gedächtnis	<p>Statische Variablen</p> <ul style="list-style-type: none">• werden im Bereich der Prozedur definiert• Beispiel: Static i as Integer• Diese Variablen haben ein Gedächtnis

2. Variablen, die für ein Modul gültig sind
<p>Lokale Variablen</p> <ul style="list-style-type: none">• werden im Bereich „Deklarationen“ definiert• Beispiel: Dim i As Integer• Diese Variablen haben ein Gedächtnis

3. Variablen, die für alle Modul gültig sind
<p>Lokale Variablen</p> <ul style="list-style-type: none">• werden im Bereich „Deklarationen“ definiert• Beispiel: Public i As Integer• Diese Variablen haben ein Gedächtnis