

### Datentypen

Boolean	True / False
Byte	0-255
Integer	ganze Zahlen
Long	grosse ganze Zahlen
Single	Kommazahlen
Double	doppelte Genauigkeit
String	Text
Variant	variabler Datentyp

### Variablendeklaration

```
Dim <var_name> as <data-type>
```

### Kommentare

' benutze den Apostroph

### Arithmetische Operatoren

+	Addition
.	Subtraktion
*	Multiplikation
/	Division
\	ganzzahlige Division
^	Potenz

### Fehlerbehandlung

On Error Resume	Fehler überspringen
Next	

### Prozeduren

```
[Public/Private] Sub <name>
    <statements>
End Sub
```

Prozeduren sind Unterprogramme

### Funktionen

```
[Public/Private] Function <name>
    (<arg>) As <result>
    <statements>
    return <result>
End Function
```

Funktionen benutzen eine Liste von Argumenten (<arg>) um einen Wert (<result>) zu berechnen

### Klassen

```
Public Class <name>
    Public <eigenschaftsname>
    Public Property <Name>() As
<datatype>
        Get
            <statements>
        End Get
        Set (ByVal value As
<datatype>)
            <statements>
        End Set
    End Property
End Class
```

### Namenskonvention

Label1	lblName	Ausgabe / Beschriftungsfeld
<input type="text"/>	tbName	Eingabe / Textfeld
Button1	btnName	Schaltfläche
<input type="radio"/>	rbName	Optionsfeld
<input type="checkbox"/>	cbName	Auswahlfeld
<input type="list"/>	cmbName	Kombinationsfeld
	pbName	Bildbehälter
GroupBox1	gbName	Behälter für Optionsfelder

### Beispielprogramm

```
Private Sub
    btn_Start_Click (ByVal ...)
        'Deklaration der Variablen
        Dim var1 As Single
        Dim var2 As Single
        'Eingabe der Variable
        var1=tb_var1.Text
        'Verarbeitung
        var2 = var1*2
        'Ausgabe der Variable 2
        lbl_ziel.Text = var2
    End Sub
```

### Mathematische Funktionen

Math.Round (<x>, <s>)	Runden
Math.Pow (<base>, <exp- >)	Potenz
Math.Sqrt (<var>)	Wurzel
<var1> Mod <var2>	Divisionsrest

s Anzahl Stellen

### Arrays deklarieren

```
Dim werte(0 to n) as <Datentyp>
Dim werte() as <Datentyp> =
{w1, w2, w3, w4}
```

### Arrays durchlaufen

```
For index = 0 to array.length-1
    Array(index) = ...
Next
```

### Sequenzen

```
<statements>
```

```
<statements>
```

```
<statements>
```



By Gregor Lüdi (Siniansung)  
[cheatography.com/siniansung/](https://cheatography.com/siniansung/)  
[www.ken.ch/%7elueg](http://www.ken.ch/%7elueg)

Published 3rd December, 2015.  
 Last updated 30th November, 2020.  
 Page 1 of 2.

Sponsored by **ApolloPad.com**  
 Everyone has a novel in them. Finish Yours!  
<https://apollopad.com>

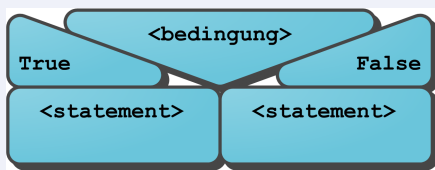
### Eingabe

```
' Eingabe über ein Objekt mit
Texteigenschaften
<variable> = <Objektnamen>.Text
```

### Ausgabe

```
' Ausgabe über ein Objekt mit
Texteigenschaft
<Objektnamen>.Text = <variable>
```

### Selektion



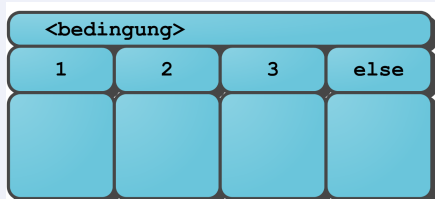
### Vergleichsoperatoren

=	gleich
<>	nicht gleich
<	kleiner als
>	größer als
<=	kleiner gleich
>=	größer gleich

### If ... Then

```
If <bedingung> Then
    <statements>
[ElseIf <bedingung> Then]
'optionale Bedingung
    <statements>
Else
    <statements>
End If
```

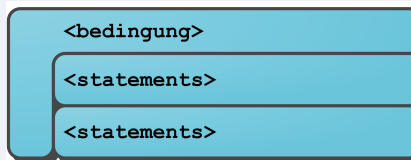
### SelectCase



### Select Case

```
Select Case <expression>
    Case 0 'einfacher Wert
        <statements>
    Case 1 To 5 'mehrere Werte
        <statements>
    Case Is <5 'Bereich von
Werten
        <statements>
    Case Else
        <statements>
End Select
```

### Iterationen



### For Next

```
For <zaehler> = <start> To
<end> Step <inc>
    <statements>
Next
```

### Do Loop (kopfgesteuert)

```
Do While <bedingung>
    <statements>
Loop
' oder
Do until <bedingung>
    <statements>
Loop
```

while: solange die Bedingung erfüllt ist  
until: bis die Bedingung erfüllt ist

### Iterationen



### Do Loop (fussgesteuert)

```
Do
    <statements>
Loop while <bedingung>
' oder
Do
    <statements>
Loop until <bedingung>
```

while: solange die Bedingung erfüllt ist  
until: bis die Bedingung erfüllt ist

### Graphiken einbinden

```
<Objektnamen>.Image = My.Resources.
<bildname>
```

Das Bild muss vorgängig in den Ressourcenspeicher geladen werden

### Audio abspielen

```
My.Computer.Audio.Play(<dateipfad>,
<option>)
<dateipfad> = My.Resources.<dateiname>
<option> = AudioPlayMode.Background
                AudioPlayMode.BackgroundLoop
                AudioPlayMode.WaitToComplete
```

