

**Wirtschaftsinformatik 1**  
**LE 05 – Schleifen**

Prof. Dr. Thomas Off  
<http://www.ThomasOff.de/lehre/beuth/wi1>

---

---

---

---

---

---

---

---

**Einordnung**



06 – Zusammenfassung

05 – Debugger und Testen

04 – Fortgeschrittene Konzepte

04.A Oberflächen (Teil 1) Elemente und Eigenschaften

04.B Oberflächen (Teil 2) Ereignisverarbeitung

04.C Zugriff auf Dateisystem und Anwendungen

03 – Grundkonzepte

03.A Wert Ausdruck Variable Konstante Datentyp

03.B Bedingte Ausführung/ Verzweigungen

03.C Schleifen

03.D Felder Mengen

03.E Prozedur Funktion Modul

02 – Grundlagen der Programmierung

01 – Grundbegriffe der Wirtschaftsinformatik

LE 05 - Schleifen 2

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 3

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 4

---

---

---

---

---

---

---

---

**Rückblick**



LE 05 - Schleifen 5

---

---

---

---

---

---

---

---

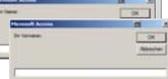
**Rückblick**



**Eingabe und Ausgabe**

- Eingabe mittels Dialog

```
' Generelle Syntax:  
' Let <Variable-vom-Typ-String> = InputBox("<Meldungstext>")  
' Beispiele  
Let strName = InputBox("Ihr Name.")  
Let strVorname = InputBox("Ihr Vorname.")  
Let bytAlter = Val(InputBox("Ihr Alter:"))
```



- Ausgabe im Debug-Bereich

```
' Beispiel  
Debug.Print "Hallo Welt!"
```



- Ausgabe im Meldungsfenster

```
' Generelle Syntax: MsgBox ("<Meldungstext>")  
' Beispiel:  
MsgBox ("Hallo Welt! Klicke auf OK.")
```



LE 05 - Schleifen 6

---

---

---

---

---

---

---

---

**Rückblick**

**Ausgangspunkt**

- Beschränkung bisheriger Programme auf linearen Programmablauf
- Notwendigkeit zu Verzweigungen

**Konzepte**

- Arten von Verzweigungen, z.B.

Bedingung ist	
Ja	Nein
Anweisung 4 / Anweisungsblock 4	Anw 5 / Anwb 5

Variable A			
Wert 1	Wert 2	Wert 3	Sonst
Anw4 / Anwb4	Anw5 / Anwb5	Anw6 / Anwb6	A7 / Ab7

- Formulierung von Bedingungen für Verzweigungen
  - als Ausdrücke, in der Regel mit Vergleichsoperator
  - mit Wahrheitswert als Ergebnis der Auswertung
  - Einsatz logischer Operationen

LE 05 - Schleifen 7

---

---

---

---

---

---

---

---

---

---

**Rückblick**

**Implementierung in VBA**

- Einfach Verzweigungen in Form von If-Then-Else-Elseif und End If
- Mehrfach Verzweigungen
  - Elseif-Erweiterung
  - Select-Case-Anweisung

```

' Beispiel
If intZahl > 4 Then
  Debug.Print "Größer 4"
End If

' Beispiel
If intZahl > 4 Then
  Debug.Print "Größer 4"
Else
  Debug.Print "Kleiner 5"
End If

' Beispiel
Select Case bytZahl
Case 2
  Debug.Print "Zwei"
Case 3
  Debug.Print "Drei"
Case Else
  Debug.Print "Etwas anderes"
End Select

' Beispiel
If intZahl > 4 Then
  Debug.Print "Größer 4"
Elseif intZahl = 4 Then
  Debug.Print "Gleich 4"
Else
  Debug.Print "Kleiner 4"
End If
    
```

LE 05 - Schleifen 8

---

---

---

---

---

---

---

---

---

---

**Rückblick**

**Programmierung als stufenweise Umsetzung**

- von Algorithmus und Datenstruktur
- in Quellcode einer Programmiersprache und
- der Überführung in ein ausführbares Programm im Maschinencode

Aufgabe/Problem

↓

Lösungsentwurf

↓

Im Vorfeld der Programmierung

Algorithmus

+ Datenstruktur

↓

Programm im Quellcode

↓

Ausführbares Programm in Maschinencode

↓

Aufgaben der Programmierung

...

Heute: Schleifen

LE 05 - Schleifen 9

---

---

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**



LE 05 - Schleifen 10

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 11

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 12

---

---

---

---

---

---

---

---

**Arten von Schleifen**

**Bisherige Programme**

- mit linearem Ablauf
- und mit Verzweigungen

**Was tun wenn eine Anweisung mehrfach ausgeführt werden soll?**

LE 05 - Schleifen 13

---

---

---

---

---

---

---

---

**Arten von Schleifen**

**Bisherige Programme**

- mit linearem Ablauf
- und mit Verzweigungen

**Was tun wenn eine Anweisung mehrfach ausgeführt werden soll?**

- solange es noch offene Rechnungen ohne Mahnung gibt, schicke eine Mahnung an den Rechnungsempfänger
- ..

LE 05 - Schleifen 14

---

---

---

---

---

---

---

---

**Arten von Schleifen**

**Bisherige Programme**

- mit linearem Ablauf
- und mit Verzweigungen

**Was tun wenn eine Anweisung mehrfach ausgeführt werden soll?**

- ...
- verbuche einen Artikel aus der Bestellung, wiederhole solange es noch unverbuchte Artikel der Bestellung gibt
- ...

LE 05 - Schleifen 15

---

---

---

---

---

---

---

---

**Arten von Schleifen**

**Bisherige Programme**

- mit linearem Ablauf
- und mit Verzweigungen

**Was tun wenn eine Anweisung mehrfach ausgeführt werden soll?**

- ...
- Wiederhole x Mal eine <Anweisung 3>

Programm

Deklaration
Anweisung 1
Anweisung 2
Zähle von Start bis Ende
Anweisung 3

LE 05 - Schleifen 16

---

---

---

---

---

---

---

---

**Unterschiede im Überblick**

**Vorprüfende/Kopfgesteuerte**

- erst Bedingung prüfen
- dann ggf. Anweisung ausführen
- anschließend Wiederholung der Prüfung usw.

**Nachprüfende/Fußgesteuerte**

- erst Anweisung ausgeführt
- dann Bedingung prüfen
- anschließend Wiederholung der Anweisungsausführung usw.

**Zählerschleifen**

- Vorher bekannte Anzahl von Wiederholungen
- Anzahl gesteuert über Start und Ende
- Ausführung der Anweisung solange Anzahl Wiederholungen das Ende noch überschritten hat

Wiederhole solange Bedingung

Anweisung

Anweisung

Wiederhole solange Bedingung

Zähle von Start bis Ende

Anweisung

LE 05 - Schleifen 17

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 18

---

---

---

---

---

---

---

---

**Inhalt** 

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 19

---

---

---

---

---

---

---

---

**Inhalt** 

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 20

---

---

---

---

---

---

---

---

**Vorprüfende Schleife** 

**Schlüsselbegriffe**

- Variante 1
- Variante 2

LE 05 - Schleifen 21

---

---

---

---

---

---

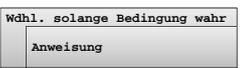
---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do While *<Bedingung>*  
*<Anweisung(en)>*  
Loop
- Variante 2  
• ...



LE 05 - Schleifen 22

---

---

---

---

---

---

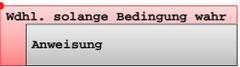
---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do While *<Bedingung>*  
*<Anweisung(en)>*  
Loop
- Variante 2  
• ...



LE 05 - Schleifen 23

---

---

---

---

---

---

---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do While *<Bedingung>*  
*<Anweisung(en)>*  
Loop
- Variante 2  
• ...



LE 05 - Schleifen 24

---

---

---

---

---

---

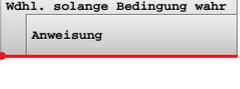
---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do **While** *<Bedingung>*  
*<Anweisung(en)>*  
**Loop**



LE 05 - Schleifen 25

---

---

---

---

---

---

---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do **While** *<Bedingung>*  
*<Anweisung(en)>*  
**Loop**
- Variante 2  
Do **Until** *<Bedingung>*  
*<Anweisung(en)>*  
**Loop**



LE 05 - Schleifen 26

---

---

---

---

---

---

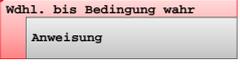
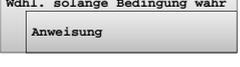
---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do **While** *<Bedingung>*  
*<Anweisung(en)>*  
**Loop**
- Variante 2  
**Do Until** *<Bedingung>*  
*<Anweisung(en)>*  
**Loop**



LE 05 - Schleifen 27

---

---

---

---

---

---

---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do **While** *<Bedingung>*  
*<Anweisung(en)>*  
Loop

Wdhl. solange Bedingung wahr
Anweisung

- Variante 2  
Do **Until** *<Bedingung>*  
*<Anweisung(en)>*  
Loop

Wdhl. bis Bedingung wahr
Anweisung

LE 05 - Schleifen 28

---

---

---

---

---

---

---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do **While** *<Bedingung>*  
*<Anweisung(en)>*  
Loop

Wdhl. solange Bedingung wahr
Anweisung

- Variante 2  
Do **Until** *<Bedingung>*  
*<Anweisung(en)>*  
Loop

Wdhl. bis Bedingung wahr
Anweisung

LE 05 - Schleifen 29

---

---

---

---

---

---

---

---

**Vorprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do **While** *<Bedingung>*  
*<Anweisung(en)>*  
Loop

Wdhl. solange Bedingung wahr
Anweisung

- Variante 2  
Do **Until** *<Bedingung>*  
*<Anweisung(en)>*  
Loop

Wdhl. bis Bedingung wahr
Anweisung

LE 05 - Schleifen 30

---

---

---

---

---

---

---

---

### Vorprüfende Schleife

**Generelle Syntax**

– Variante 1

```
Do While <Bedingung>
<Anweisung(en)>
Loop
```

– Variante 2

```
Do Until <Bedingung>
<Anweisung(en)>
Loop
```

**Beispiele**

```
Dim i As Integer
Let i = 0
Do While i < 5
  Let i = i + 1
  Debug.Print i
Loop
```

```
Dim j As Byte
Let j = 0
Do Until j > 4
  Let j = j + 1
  Debug.Print j
Loop
```

LE 05 - Schleifen
31

---

---

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.00

**Ziel**

– Vorprüfende Schleife mit Bedingungen verwenden

**Aufgabe**

– Schreiben Sie ein Programm mit einer vorprüfenden Schleife, die

- das die Zahlen von 1 bis 10 ausgibt
- schrittweise die Summe dieser Zahlen bildet, indem
  - auf die bisherige Summe die nächste Zahl addiert wird und
  - das Ergebnis als neue Summe verwendet wird
- nach der Schleife, die Gesamtsumme ausgibt



LE 05 - Schleifen
32

---

---

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.01

**Ziel**

– Vorprüfende Schleife mit Bedingungen verwenden

**Aufgabe**

– Schreiben Sie ein Programm mit einer vorprüfenden Schleife, die

- das Datum aller verbliebenden Tage bis zur Klausur am 19.06. ausgibt
- die Summe der Anzahl dieser Tage bildet und die Summe im Direktbereich ausgibt.



LE 05 - Schleifen

---

---

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

**Ziel**

- Vorprüfende Schleife verwenden

**Aufgabe**

- Teil 1: Vorprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Vorprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Vorprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt
- Teil 4: Vorprüfende Schleife, die von einem eingegebenen Datum die verbleibenden Tage bis Weihnachten zählt
- Teil 5: Vorprüfende Schleife, die die Summe der Zahlen 1, 2, 3, 4, ... usw. bildet, und solange läuft bis vorher eingegebene Zahl erreicht ist.



LE 05 - Schleifen 34

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

1 Programmcode

Deklaration Var
Initialisierung Var
Wiederhole solange, wie <Var ist kleiner 5>
Ausgabe Var
Var = Var + 1

2

```
Sub Bsp24()
    Dim i As Integer
    i = 1
    Do While i < 5
        Debug.Print i
        Let i = i + 1
    Loop
End Sub
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die für die Bedingung verwendet wird
  - Schleife, die von 1 bis 5 zählt und eine Ausgabeanweisung enthält
- **Nutzen von Do While Loop**
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen 35

---

---

---

---

---

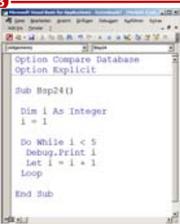
---

---

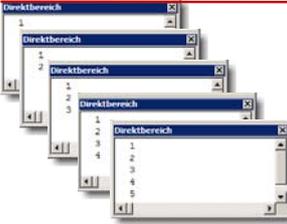
---

### Vorprüfende Schleife: Beispiel 05.02

3



4



- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen von 1 bis 5**

LE 05 - Schleifen 36

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

**Ziel**

- Vorprüfende Schleife verwenden

**Aufgabe**

- Teil 1: Vorprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Vorprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Vorprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt
- Teil 4: Vorprüfende Schleife, die von einem eingegebenen Datum die verbleibenden Tage bis Weihnachten zählt
- Teil 5: Vorprüfende Schleife, die die Summe der Zahlen 1, 2, 3, 4, ... usw. bildet, und solange läuft bis vorher eingegebene Zahl erreicht ist.



LE 05 - Schleifen

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

**1 Programmcode**

```

Deklaration Var
Initialisierung Var
Wiederhole solange, wie
<Var größer 0 ist>
Ausgabe Var
Var = Var - 1
        
```

**2**

```

Sub Bsp25()
Dim i As Integer
i = 5
Do While i > 0
Debug.Print i
i = i - 1
Loop
End Sub
        
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die zum Zählen verwendet wird
  - Schleife, die von 5 bis 1 rückwärts zählt und eine Ausgabeanweisung enthält
- **Nutzung Do While/Loop**
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen 38

---

---

---

---

---

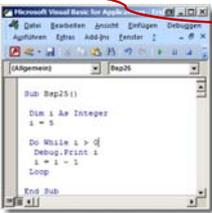
---

---

---

### Vorprüfende Schleife: Beispiel 05.02

**3**



**4 Direktbereich**



- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen von 5 bis 1**

LE 05 - Schleifen 39

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

→ Ziel

- Vorprüfende Schleife verwenden

**Aufgabe**

- Teil 1: Vorprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Vorprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Vorprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt
- Teil 4: Vorprüfende Schleife, die von einem eingegebenen Datum die verbleibenden Tage bis Weihnachten zählt
- Teil 5: Vorprüfende Schleife, die die Summe der Zahlen 1, 2, 3, 4, ... usw. bildet, und solange läuft bis vorher eingegebene Zahl erreicht ist.



LE 05 - Schleifen 40

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

1 Programmcode

```

Deklaration Var
Initialisierung Var
Wiederhole bis
<Var größer 10 ist>
Ausgabe Var
Var = Var + 2
        
```

2

```

Sub Bsp26()
Dim i As Integer
i = 1
Do Until i > 10
Debug.Print i
i = i + 2
Loop
End Sub
        
```

- Überführen in Programmcode
  - Deklaration einer Variablen, die zum Zählen verwendet wird
  - Schleife, die von 1 bis 10 in Zweierschritten zählt und eine Ausgabeanweisung enthält
- Nutzung Do Until/Loop
- Was passiert beim Ausführen des Programmcodes mit der Variable i?

LE 05 - Schleifen 41

---

---

---

---

---

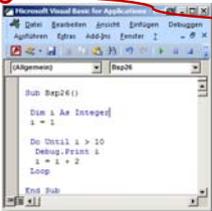
---

---

---

### Vorprüfende Schleife: Beispiel 05.02

3



4 Direktbereich



- Was ist im Direktbereich sichtbar?
- Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen 1, 3, 5, 7 und 9.

LE 05 - Schleifen 42

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

**Ziel**

- Vorprüfende Schleife verwenden

**Aufgabe**

- Teil 1: Vorprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Vorprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Vorprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt
- Teil 4: Vorprüfende Schleife, die von einem eingegebenen Datum die verbleibenden Tage bis Weihnachten zählt
- Teil 5: Vorprüfende Schleife, die die Summe der Zahlen 1, 2, 3, 4, ... usw. bildet, und solange läuft bis vorher eingegebene Zahl erreicht ist.



LE 05 - Schleifen

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

**1 Programmcode**

Deklaration Var
Initialisierung Var
Wiederhole bis <Var ist Weihnachten>
Ausgabe Var
Var = Var + 1 Tag

```
Sub Bsp27()
    Dim i As Integer, d As Date
    Let i = 0
    Let d = DateValue(InputBox("Datum: "))

    Do Until d >= #12/24/2011#
        Let i = i + 1
        Let d = d + 1
        Debug.Print d
    Loop

    Debug.Print "Tage: " & i
End Sub
```

- **Überführen in Programmcode**
  - Deklaration einer Bedingung, die das Datum prüft ("Ist Weihnachten?")
  - Schleife, die auf das eingegebene Datum 1 Tag addiert und eine Ausgabeanweisung enthält
- Nach Schleifenende Ausgabe der Tage bis Weihnachten
- **Nutzung Do Until/Loop**
- **Was passiert beim Ausführen des Programmcodes?**

LE 05 - Schleifen 44

---

---

---

---

---

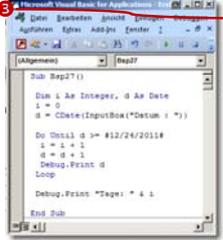
---

---

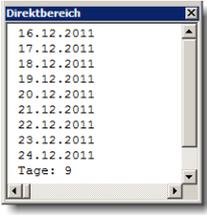
---

### Vorprüfende Schleife: Beispiel 05.02

**3**



**4 Direktbereich**



- Was ist im Direktbereich sichtbar?
- Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen von 5 bis 1

LE 05 - Schleifen 45

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

→ **Ziel**

- Vorprüfende Schleife verwenden

**Aufgabe**

- Teil 1: Vorprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Vorprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Vorprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt
- Teil 4: Vorprüfende Schleife, die von einem eingegebenen Datum die verbleibenden Tage bis Weihnachten zählt
- Teil 5: Vorprüfende Schleife, die die Summe der Zahlen 1, 2, 3, 4, ... usw. bildet, und solange läuft bis vorher eingegebene Zahl erreicht ist. **1**



LE 05 - Schleifen 46

---

---

---

---

---

---

---

---

### Vorprüfende Schleife: Beispiel 05.02

**1** Programmcode

Deklaration Var1, ...

Initialis. Var1, ...

Wiederhole bis  
<Var1 ...>

Ausgabe Var

Summe bilden

**2**

```
Sub Bsp28()
  Dim i As Integer, z As Integer
  Dim s As Integer
  Let i = 0
  Let s = 0
  Let z = Val(TextBox("Zahl: "))

  Do While s < z
    Let i = i + 1
    Let s = s + i
    Debug.Print i & ", " & s
  Loop
End Sub
```

- **Überführen in Programmcode**
  - Deklaration einer Bedingung ("Ist eingegebene Zahl erreicht?")
  - Schleife, die hochzählt, die Summe bildet und eine Ausgabeanweisung enthält
- **Nutzung Do While/Loop**
- **Was passiert beim Ausführen des Programmcodes?**

LE 05 - Schleifen 47

---

---

---

---

---

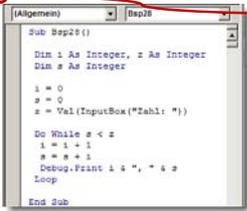
---

---

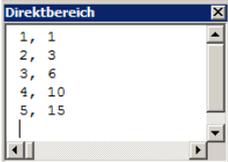
---

### Vorprüfende Schleife: Beispiel 05.02

**3**



**4** Fertig!



- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe des Zählers und der Summe**

LE 05 - Schleifen 48

---

---

---

---

---

---

---

---

**Vorprüfende Schleife: Beispiel 05.02**

**Ziel**

- Vorprüfende Schleife verwenden

**Aufgabe**

- Teil 1: Vorprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Vorprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Vorprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt
- Teil 4: Vorprüfende Schleife, die von einem eingegebenen Datum die verbleibenden Tage bis Weihnachten zählt
- Teil 5: Vorprüfende Schleife, die die Summe der Zahlen 1, 2, 3, 4, ... usw. bildet, und solange läuft bis vorher eingegebene Zahl erreicht ist.



LE 05 - Schleifen 49

---

---

---

---

---

---

---

---

**Inhalt**

Rückblick

Überblick

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

Abschluss und Ausblick



LE 05 - Schleifen 50

---

---

---

---

---

---

---

---

**Inhalt**

Rückblick

Überblick

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

Abschluss und Ausblick

LE 05 - Schleifen 51

---

---

---

---

---

---

---

---

**Nachprüfende Schleife** 

**Schlüsselbegriffe**

- Variante 1
- Variante 2

LE 05 - Schleifen 52

---

---

---

---

---

---

---

---

**Nachprüfende Schleife** 

**Schlüsselbegriffe**

- Variante 1

```
Do  
<Anweisung(en)>  
Loop While <Beding.>
```

Anweisung
Wdhl. solange Bedingung wahr

- Variante 2
- ...

LE 05 - Schleifen 53

---

---

---

---

---

---

---

---

**Nachprüfende Schleife** 

**Schlüsselbegriffe**

- Variante 1

```
Do  
<Anweisung(en)>  
Loop While <Beding.>
```

Anweisung
Wdhl. solange Bedingung wahr

- Variante 2
- ...

LE 05 - Schleifen 54

---

---

---

---

---

---

---

---

### Nachprüfende Schleife

**Schlüsselbegriffe**

- Variante 1  
Do  
<Anweisung(en)>  
Loop While <Beding.>



LE 05 - Schleifen 55

---

---

---

---

---

---

---

---

### Nachprüfende Schleife

**Schlüsselbegriffe**

- Variante 1  
Do  
<Anweisung(en)>  
Loop While <Beding.>



LE 05 - Schleifen 56

---

---

---

---

---

---

---

---

### Nachprüfende Schleife

**Schlüsselbegriffe**

- Variante 1  
Do  
<Anweisung(en)>  
Loop While <Beding.>
- Variante 2  
Do  
<Anweisung(en)>  
Loop Until <Beding.>



LE 05 - Schleifen 57

---

---

---

---

---

---

---

---

**Nachprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do  
<Anweisung(en)>  
Loop While <Beding.>

- Variante 2  
Do  
<Anweisung(en)>  
Loop Until <Beding.>

LE 05 - Schleifen 58

---

---

---

---

---

---

---

---

**Nachprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do  
<Anweisung(en)>  
Loop While <Beding.>

- Variante 2  
Do  
<Anweisung(en)>  
Loop Until <Beding.>

LE 05 - Schleifen 59

---

---

---

---

---

---

---

---

**Nachprüfende Schleife**

**Schlüsselbegriffe**

- Variante 1  
Do  
<Anweisung(en)>  
Loop While <Beding.>

- Variante 2  
Do  
<Anweisung(en)>  
Loop Until <Beding.>

LE 05 - Schleifen 60

---

---

---

---

---

---

---

---

### Nachprüfende Schleife

**Schlüsselbegriffe**

– Variante 1

```
Do
<Anweisung(en)>
Loop While <Beding.>
```

Anweisung

WdhL. solange Bedingung wahr

– Variante 2

```
Do
<Anweisung(en)>
Loop Until <Beding.>
```

Anweisung

WdhL. bis Bedingung wahr

LE 05 - Schleifen 61

---

---

---

---

---

---

---

---

### Nachprüfende Schleife

**Generelle Syntax**

– Variante 1

```
Do
<Anweisung(en)>
Loop While <Beding.>
```

– Variante 2

```
Do
<Anweisung(en)>
Loop Until <Beding.>
```

**Beispiele**

```
Dim i As Integer
Let i = 0
Do
Let i = i + 1
Debug.Print i
Loop While i < 5
```

```
Dim j As Byte
Let j = 0
Do
Let j = j + 1
Debug.Print j
Loop Until j > 4
```

LE 05 - Schleifen 62

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.03

**Ziel**

- Nachprüfende Schleife mit Bedingungen verwenden

**Aufgabe: Schreiben Sie ein Programm mit einer nachprüfenden Schleife**

– Teil 1

- In der Schleife sollen alle Vielfachen von 3 ausgegeben werden
- Schleife soll enden, wenn die Vielfachen 100 überschreiten

– Teil 2

- bilden Sie die Zwischensumme der Vielfachen von 3
- Geben Sie das aktuelle Vielfache von 3 im Direktbereich aus
- Geben Sie die Zwischensumme im Direktbereich aus
- Die Schleife soll enden, sobald die Summe größer als 75 ist.



LE 05 - Schleifen 63

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.04

**Ziel**

- Weitere Möglichkeiten der nachprüfenden Schleife kennenlernen

**Aufgabe**

- Teil 1: Nachprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Nachprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Nachprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen 64

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.04

**1 Programmcode**

```

Deklaration Var
Initialisierung Var
Ausgabe Var
Var = Var + 1
Wiederhole solange
<Var ist kleiner 5>
        
```

**2 Sub Bsp()**

```

Dim i As Integer
Let i = 1

Do
  Debug.Print i
  Let i = i + 1
Loop While i < 5

End Sub
        
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die für die Bedingung verwendet wird
  - Schleife, die von 1 bis 5 zählt und eine Ausgabeanweisung enthält
- **Nutzen von Do/Loop While**
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen 65

---

---

---

---

---

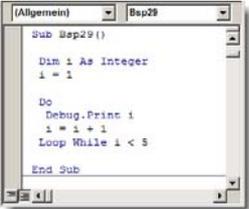
---

---

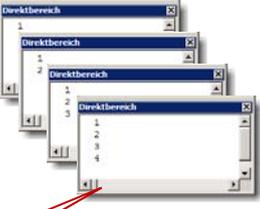
---

### Nachprüfende Schleife: Beispiel 05.04

**3**



**4**



- **Was ist im Direktbereich sichtbar?**

Huch!  
Wo ist die 5?

Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen von 1 bis 4

LE 05 - Schleifen 66

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.04

**5 Ziel**

- Weitere Möglichkeiten der nachprüfenden Schleife kennenlernen

**Aufgabe**

- Teil 1: Nachprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Nachprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Nachprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen 67

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.04

Programmcode

**1**

```

Deklaration Var
Initialisierung Var
Ausgabe Var
Var = Var - 1
Wiederhole solange
<Var größer 0>
            
```

Sub Bsp30()

**2**

```

Sub Bsp30()
    Dim i As Integer
    i = 5

    Do
        Debug.Print i
        i = i - 1
    Loop While i > 0

End Sub
            
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die zum Zählen verwendet wird
  - Schleife, die von 5 bis 1 rückwärts zählt und eine Ausgabeanweisung enthält
- **Nutzung Do While/Loop**
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen 68

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.04

**3**

```

Sub Bsp30()
    Dim i As Integer
    i = 5

    Do
        Debug.Print i
        i = i - 1
    Loop While i > 0
            
```

**4**

Direktbereich

```

5
4
3
2
1
|
|
|
            
```

- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen von 5 bis 1**

LE 05 - Schleifen 69

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.04

**Ziel**

- Weitere Möglichkeiten der nachprüfenden Schleife kennenlernen

**Aufgabe**

- Teil 1: Nachprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Nachprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Nachprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen 70

---

---

---

---

---

---

---

---

### Nachprüfende Schleife: Beispiel 05.04

**1 Programmcode**

```

Deklaration Var
Initialisierung Var
Ausgabe Var
Var = Var + 2
Wiederhole bis
<Var größer 10>
        
```

**2**

```

Sub Bsp31()
Dim i As Integer
i = 1

Do
Debug.Print i
i = i + 2
Loop Until i > 10

End Sub
        
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die zum Zählen verwendet wird
  - Schleife, die von 1 bis 10 in Zweierschritten zählt und eine Ausgabeanweisung enthält
- **Nutzung Do Until/Loop**
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen 71

---

---

---

---

---

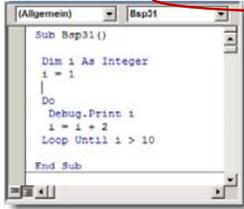
---

---

---

### Nachprüfende Schleife: Beispiel 05.04

**3**



**Fertig!**

**4**



- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen 1, 3, 5, 7 und 9.**

LE 05 - Schleifen 72

---

---

---

---

---

---

---

---

**Nachprüfende Schleife: Beispiel 05.04** 

**Ziel**

- Weitere Möglichkeiten der nachprüfenden Schleife kennenlernen

**Aufgabe**

- Teil 1: Nachprüfende Schleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Nachprüfende Schleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Nachprüfende Schleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen 73

---

---

---

---

---

---

---

---

**Hinweise vor-/nachprüfende Schleife** 

**Do While vs. Do Until bzw. Loop While vs. Loop Until**

- While bedeutet, dass solange die Bedingung wahr ist die Schleife ausgeführt wird
- Until bedeutet, dass bis die Bedingung wahr wird, die Schleife ausgeführt wird
- Äquivalenz zwischen While und Until durch Einsatz der logischen Negation

<pre>Dim i As Byte Let i = 0  Debug.Print "While:" Do While i &lt; 5   Debug.Print i   Let i = i + 1 Loop</pre>	<pre>Dim i As Byte Let i = 0  Debug.Print "Until Not:" Do Until Not (i &lt; 5)   Debug.Print i   Let i = i + 1 Loop</pre>
---	---

Alternative Negation:  
 i >= 5

LE 05 - Schleifen 74

---

---

---

---

---

---

---

---

**Inhalt** 

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**



LE 05 - Schleifen 75

---

---

---

---

---

---

---

---

**Inhalt**

Rückblick

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**



LE 05 - Schleifen 76

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form
- Erweiterungen

Zähle von Start bis Ende
Anweisung



LE 05 - Schleifen 77

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form

```
For  
  <Var> = <Beginn>  
  To <Ende>  
  <Anweisung(en)>  
Next
```

Zähle von Start bis Ende
Anweisung

- Erweiterungen
- ...



LE 05 - Schleifen 78

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form

```
For  
  <Var> = <Beginn>  
  To <Ende>  
  <Anweisung(en)>  
Next
```

– Erweiterungen

- ...

LE 05 - Schleifen 79

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form

```
For  
  <Var> = <Beginn>  
  To <Ende>  
  <Anweisung(en)>  
Next
```

– Erweiterungen

- ...

LE 05 - Schleifen 80

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form

```
For  
  <Var> = <Beginn>  
  To <Ende>  
  <Anweisung(en)>  
Next
```

– Erweiterungen

- ...

LE 05 - Schleifen 81

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form

```
For  
  <Var> = <Beginn>  
  To <Ende>  
  <Anweisung(en)>  
Next
```

– Erweiterungen

- ...

LE 05 - Schleifen 82

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form

```
For  
  <Var> = <Beginn>  
  To <Ende>  
  <Anweisung(en)>  
Next
```

– Erweiterungen

- ...

LE 05 - Schleifen 83

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form

```
For  
  <Var> = <Beginn>  
  To <Ende>  
  <Anweisung(en)>  
Next
```

– Erweiterungen

- ...

LE 05 - Schleifen 84

---

---

---

---

---

---

---

---

**Zählerschleife**

**Schlüsselbegriffe**

- Grundlegende Form
 

Zähle von Start bis Ende
Anweisung

```

For
  <Var> = <Beginn>
  To <Ende>
  <Anweisung(en)>
Next
      
```
- Erweiterungen
  - Schrittweite/-richtung beim Zählen **Step**

Zähle von Start bis Ende in Schrittweite
Anweisung

```

For <Var> = <Wert oder Ausdruck> To <Ausdruck> Step <Schrittweite>
  <Anweisung(en)>
Next
      
```

LE 05 - Schleifen 85

---

---

---

---

---

---

---

---

**Zählerschleife**

**Grundlegende Form**

- Generelle Syntax
 

```

For <Variable> = <Wert oder Ausdruck> To <Ausdruck>
  <Anweisung(en)>
Next
      
```
- Beispiele
 

<pre> Dim i As Integer For i = 1 To 5   Debug.Print i Next           </pre>	<pre> 1 2 3 4 5           </pre>
---	----------------------------------

LE 05 - Schleifen 86

---

---

---

---

---

---

---

---

**Zählerschleife**

**Grundlegende Form**

- Erweiterungen
 

```

For <Var> = <Wert/Ausdr> To <Ausdr> Step <Schritt>
  <Anweisung(en)>
Next
      
```
- Beispiele
 

<pre> Dim i As Integer For i = 2 To 8 Step 2   Debug.Print i Next           </pre>	<pre> 2 4 6 8           </pre>
<pre> Dim i As Integer For i = 9 To 1 Step -3   Debug.Print i Next           </pre>	<pre> 9 6 3           </pre>

LE 05 - Schleifen 87

---

---

---

---

---

---

---

---

**Zählerschleife: Beispiel 05.05**

**Aufgabe**

- Teil 1: Zählerschleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Zählerschleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Zählerschleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen

---

---

---

---

---

---

---

---

**Zählerschleife: Beispiel 05.05**

Programmcode

Deklaration Var  
 Zähle  
 Von 1  
 Bis 5  
 und bearbeite  
 Ausgabe Var

```
Sub Bsp21()  

    Dim i As Integer  

    For i = 1 To 5  

        Debug.Print i  

    Next  

End Sub
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die zum Zählen verwendet wird
  - Schleife, die von 1 bis 5 zählt und eine Ausgabeanweisung enthält
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen

89

---

---

---

---

---

---

---

---

**Zählerschleife: Beispiel 05.05**

- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen von 1 bis 5**

LE 05 - Schleifen

90

---

---

---

---

---

---

---

---

### Zählerschleife: Beispiel 05.05

**Aufgabe**

- Teil 1: Zählerschleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Zählerschleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Zählerschleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen

---

---

---

---

---

---

---

---

### Zählerschleife: Beispiel 05.05

Programmcode

**Deklaration Var**

Zähle  
Von 5 Bis 1  
rückwärts  
und bearbeite

**Ausgabe Var**

```
Sub Bsp22()
    Dim i As Integer
    For i = 5 To 1 Step -1
        Debug.Print i
    Next
End Sub
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die zum Zählen verwendet wird
  - Schleife, die von 5 bis 1 rückwärts zählt und eine Ausgabeanweisung enthält
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen 92

---

---

---

---

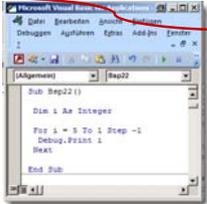
---

---

---

---

### Zählerschleife: Beispiel 05.05



Direktbereich

```
5
4
3
2
1
```

- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen von 5 bis 1**

LE 05 - Schleifen 93

---

---

---

---

---

---

---

---

### Zählerschleife: Beispiel 05.05

**Aufgabe**

- Teil 1: Zählerschleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Zählerschleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Zählerschleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen

---

---

---

---

---

---

---

---

### Zählerschleife: Beispiel 05.05

Programmcode

Deklaration Var

Zähle  
Von 1 Bis 10  
in Zweierschritten  
und bearbeite

Ausgabe Var

```
Sub Bsp23()
    Dim i As Integer
    For i = 1 To 10 Step 2
        Debug.Print i
    Next
End Sub
```

- **Überführen in Programmcode**
  - Deklaration einer Variablen, die zum Zählen verwendet wird
  - Schleife, die von 1 bis 10 in Zweierschritten zählt und eine Ausgabeanweisung enthält
- **Was passiert beim Ausführen des Programmcodes mit der Variable i?**

LE 05 - Schleifen 95

---

---

---

---

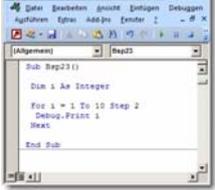
---

---

---

---

### Zählerschleife: Beispiel 05.05



Direktbereich

```
1
3
5
7
9
```

- **Was ist im Direktbereich sichtbar?**
- **Ausführen des Programmcodes liefert im Direktbereich schrittweise die Ausgabe der Zahlen 1, 3, 5, 7 und 9.**

LE 05 - Schleifen 96

---

---

---

---

---

---

---

---

**Zählerschleife: Beispiel 05.05**



**Aufgabe**

- Teil 1: Zählerschleife, die von 1 bis 5 die Zahlen im Direktbereich ausgibt
- Teil 2: Zählerschleife, die von 5 bis 1 die Zahlen im Direktbereich ausgibt
- Teil 3: Zählerschleife, die von 1 bis 10 die ungeraden Zahlen im Direktbereich ausgibt



LE 05 - Schleifen

---

---

---

---

---

---

---

---

**Zählerschleife: Beispiel 05.06**



**Ziel**

- Zählerschleife praktisch anwenden

**Aufgabe**

- Schreiben Sie ein Programm mit einer Zählerschleife
- Es soll die Summe aller Zahlen im Bereich von 0 bis 25 bilden
- Die jeweilige Zwischensumme und die zuletzt addierte Zahl sollen im Direktbereich ausgegeben werden
- Die Schleife soll bei 25 enden.
- Das Endergebnis ist hinter der Schleife im Direktbereich auszugeben.
- Berechnen Sie das Ergebnis zusätzlich mit der Gaußschen Summenformel und geben Sie es zum Vergleich aus:
  - $1 + 2 + \dots + n = n * (n+1)/2$



LE 05 - Schleifen

98

---

---

---

---

---

---

---

---

**Zählerschleife Hinweise**



**Schleifenvariable**

- für Zählerschleife ist eine Variable notwendig
- Variable muss vorher deklariert werden
- Initialisierung der Variable erfolgt im Schleifenkopf durch Zuweisung des Startwertes

LE 05 - Schleifen

99

---

---

---

---

---

---

---

---

### Zählerschleife Hinweise



Welchen Wert hat die Schleifenvariable, nachdem die Schleife vollständig durchlaufen wurde?

– Beispiel

```
Dim i As Integer
For i = 2 To 8 Step 2
    Debug.Print "Innen: " & i
Next
Debug.Print "Dahinter: " & i
```

```
Innen: 2
Innen: 4
Innen: 6
Innen: 8
Dahinter: 10
```

– Antwort: Den Endwert der Schleife plus einen weiteren Schritt entsprechend der Schrittweite

LE 05 - Schleifen

100

---

---

---

---

---

---

---

---

### Inhalt



Rückblick

Überblick

- Arten von Schleifen
- Unterschiede im Überblick

Implementierung von Schleifen mit VBA

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

Abschluss und Ausblick



LE 05 - Schleifen

101

---

---

---

---

---

---

---

---

### Inhalt



Rückblick

Überblick

- Arten von Schleifen
- Unterschiede im Überblick

Implementierung von Schleifen mit VBA

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

Abschluss und Ausblick

LE 05 - Schleifen

102

---

---

---

---

---

---

---

---

**Weitere Möglichkeiten**



**Risiko von Endlosschleifen**

- Anweisungen innerhalb der Schleife müssen bei Vor- und nachprüfenden Schleifen Einfluss auf die zu prüfende Bedingung nehmen
- Zählerschleifen sollten die Zählervariable in den Anweisungen innerhalb der Schleife nur lesen, bei ändernden Zugriffen besteht die Gefahr der Endlosschleife
- Endlosschleifen in VBA abbrechen mit Tastenkombination
  - STRG + Pause
  - STRG + ALT + Pause (Laborrechner in B045L)
- Rechner ohne Pause-Taste (z.B. MacBook, Samsung Ativ)
  - Zusatzprogramm SharpKey installieren (<http://www.randyrants.com/sharpkeys/>) und damit eine Taste zur Pause-Taste (E0\_46) machen

LE 05 - Schleifen

103

---

---

---

---

---

---

---

---

**Weitere Möglichkeiten**



**Schachteln von Schleifen**

- Äußere Schleife wird durchlaufen
- Innerhalb der Äußeren Schleife wird die innere Schleife durchlaufen
- mit allen Arten von Schleifen möglich
- Beispiel: Geschachtelte Zählerschleifen

```
Dim i As Byte, j As Byte
For i = 1 To 3
  For j = 1 To 3
    Debug.Print i & "*" & j & "=" & i * j
  Next
  Debug.Print
Next
```



LE 05 - Schleifen

104

---

---

---

---

---

---

---

---

**Weitere Möglichkeiten**



**Vorzeitiges Verlassen von Schleifen**

- Schleifen können verlassen werden, auch wenn Ende-Bedingung noch nicht erfüllt ist
- vor- und nachprüfende Schleifen mit Schlüsselwort: **Exit Do**
- Zählerschleifen mit Schlüsselwort: **Exit For**

**Springen zum nächsten Schleifendurchlauf**

- ist der aktuelle Schleifendurchlauf noch nicht beendet, soll aber keine weitere Anweisung innerhalb der Schleife verarbeitet werden, dann
- kann mit **Continue Do** bzw. **Continue For** zum jeweils nächsten Schleifendurchlauf "gesprungen" werden
- aber nicht in VBA ☹️

**Alte Form der Schleife: While-Wend-Schleife wird nicht mehr benutzt**

Nicht in VBA  
Nicht mehr üblich

LE 05 - Schleifen

105

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**



LE 05 - Schleifen 106

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 107

---

---

---

---

---

---

---

---

**Inhalt**

**Rückblick**

**Überblick**

- Arten von Schleifen
- Unterschiede im Überblick

**Implementierung von Schleifen mit VBA**

- Vorprüfende Schleifen
- Nachprüfende Schleifen
- Zählerschleifen
- Weitere Möglichkeiten

**Abschluss und Ausblick**

LE 05 - Schleifen 108

---

---

---

---

---

---

---

---

**Abschluss**

**Vorprüfende/Kopfprüfende Schleife**

– Einsatzzweck

- erst Bedingung prüfen
- dann ggf. Anweisung ausführen
- anschließend Wiederholung der Prüfung usw.

WdhL. solange Bedingung

---

Anweisung

– Generelle Syntax

```
Do While <Bedingung>
  <Anweisung(en)>
Loop
```

```
Do Until <Bedingung>
  <Anweisung(en)>
Loop
```

– Beispiel

```
Dim i As Integer
Let i = 0
Do While i < 5
  Let i = i + 1
  Debug.Print i
Loop
```

```
Dim j As Byte
Let j = 0
Do Until j > 4
  Let j = j + 1
  Debug.Print j
Loop
```

LE 05 - Schleifen 109

---

---

---

---

---

---

---

---

**Abschluss**

**Nachprüfende/Fußprüfende Schleife**

– Einsatzzweck

- erst Anweisung ausgeführt
- dann Bedingung prüfen
- anschließend Wiederholung der Anweisungsausführung usw.

Anweisung

---

WdhL. solange Bedingung wahr

– Generelle Syntax

```
Do
  <Anweisung(en)>
Loop While <Beding.>
```

```
Do
  <Anweisung(en)>
Loop Until <Beding.>
```

– Beispiel

```
Dim i As Integer
Let i = 0
Do
  Let i = i + 1
  Debug.Print i
Loop While i < 5
```

```
Dim j As Byte
Let j = 0
Do
  Let j = j + 1
  Debug.Print j
Loop Until j > 4
```

LE 05 - Schleifen 110

---

---

---

---

---

---

---

---

**Abschluss**

**Zählerschleifen**

– Einsatz

- Vorher bekannte Anzahl von Wiederholungen
- Anzahl gesteuert über Start und Ende
- Ausführung der Anweisung solange Anzahl Wiederholungen das Ende noch überschritten hat

– Generelle Syntax

```
For <Var> = <Wert/Ausdr> To <Ausdr> Step <Schrittw>
  <Anweisung(en)>
Next
```

– Beispiel

```
Dim i As Integer
For i = 1 To 10 Step 2
  Debug.Print i
Next
```

LE 05 - Schleifen 111

---

---

---

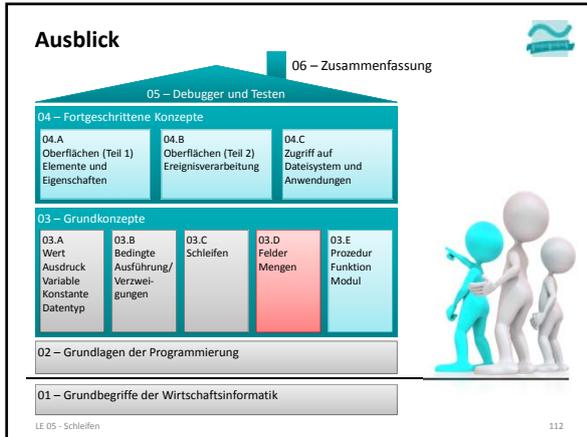
---

---

---

---

---



---

---

---

---

---

---

---

---

 BEUTH HOCHSCHULE FÜR TECHNIK BERLIN  
University of Applied Sciences

**Wirtschaftsinformatik 1**  
**LE 05 – Schleifen**

Prof. Dr. Thomas Off  
<http://www.ThomasOff.de/lehre/beuth/wi1>

---

---

---

---

---

---

---

---