

Aufgaben aus der Praxis

1. Wie kann ich **SVERWEIS** auch nach links anwenden?

Oftmals müssen aus einer größeren Datentabelle Werte ausgelesen werden, die abhängig sind von einem Suchbegriff, der sich ebenfalls in der Tabelle befindet. Hierfür nutzt man die Tabellenfunktion **SVERWEIS**, um einen Datenbereich mittels eines Suchbegriffs in einer Spalte nach rechts zu durchsuchen. Dabei muss der Suchbegriff im Datenbereich jedoch links vom Ergebnis stehen. Mit dem folgenden Beispiel beschreiben wir zunächst die Standard-Funktionsweise und zeigen dann Alternativen, mit denen es möglich ist, die gleiche Aufgabe allerdings ohne Einschränkung der Funktion zu lösen.

Was aber können Sie machen, wenn der zu suchende Begriff eben nicht am linken Rand des Bereichs steht? Zur besseren Verständlichkeit sehen Sie sich einmal [Abbildung 1](#) an.

Wie Sie in der [Abbildung](#) sehen, wird in Zelle **E2** durch die Tabellenfunktion **SVERWEIS** über den Suchbegriff aus Zelle **D2** im Bereich **A5:B9** gesucht. Dabei steht der zu findende Wert in der ersten Spalte des zu durchsuchenden Bereichs. Über das letzte Argument der Funktion, hier die **2**, wird die Spalte im Bereich angegeben, in der das Ergebnis erwartet wird. Die Spaltenangabe **2** bedeutet somit die zweite Spalte im zu durchsuchenden Bereich **A5:B9**. In diesem Fall handelt es sich um die Spalte **B**.

Die Tabellenfunktion **SVERWEIS** funktioniert aber leider nicht, wenn eine Ausgangssituation wie in [Abbildung 2](#) vorliegt.

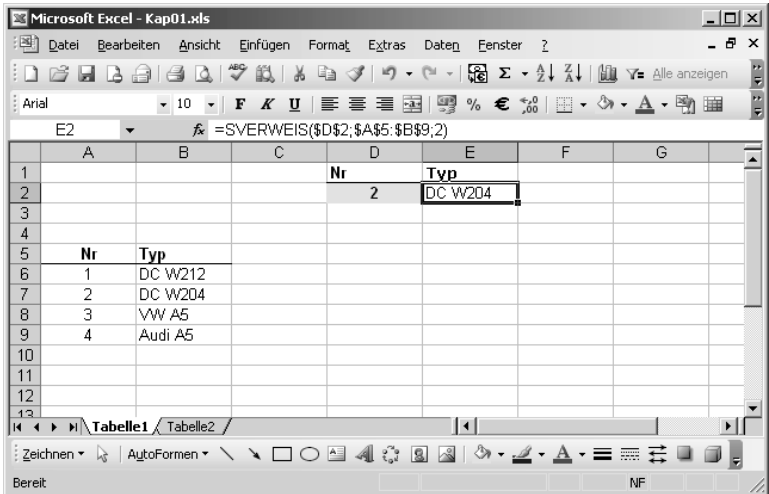


Abbildung 1: Der Standardfall

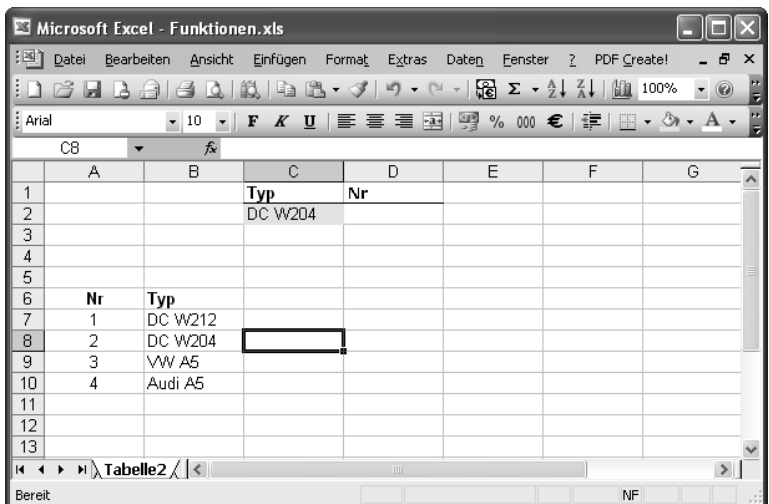


Abbildung 2: Der Sonderfall

In dieser Abbildung sehen Sie, wie nach einem anderen Suchbegriff gesucht werden soll. Die Suche soll nach dem Typ durchge-

führt werden. Dabei stehen die Typen jetzt aber in Spalte **B**. Es müsste also beispielsweise zum Typ **DC W204** die Nummer **2** ermittelt werden.

Variante 1 – über die Funktionen INDEX und VERGLEICH



Diese Variante finden Sie auf der CD-ROM im Verzeichnis *Beispieldateien* in der Mappe *Funktionen.xls* auf der *Tabelle2*.

Um diese Aufgabe zu lösen, verfahren Sie wie folgt:

1. Setzen Sie den Mauszeiger auf die Zelle **D2**.
2. Geben Sie die Formel
`=INDEX(A:A;VERGLEICH(C2;B:B;0))` ein.
3. Bestätigen Sie mit .

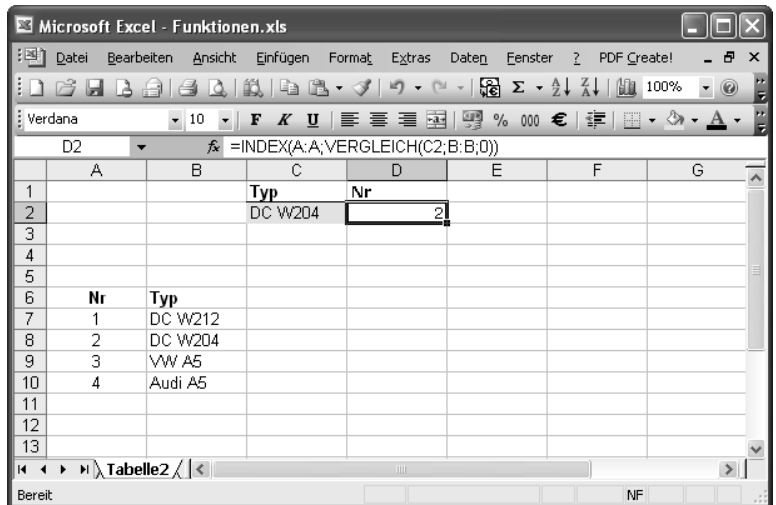


Abbildung 3: Die Suche funktioniert auch von rechts nach links

Möchten Sie die Nummer in Zelle C2 nicht per Hand eingeben, können Sie diese Zelle mit einer Gültigkeitsliste belegen. Dazu verfahren Sie wie folgt:

1. Setzen Sie den Mauszeiger in Zelle C2.
2. Wählen Sie aus dem Menü *Daten* den Befehl *Gültigkeit*.



Abbildung 4: Eine Gültigkeitsliste einstellen

3. Wählen Sie aus dem Kombinationsfeld *Zulassen* den Eintrag *Liste* aus.
4. Geben Sie im Feld *Quelle* den Zellenbezug **=B\$7:B\$10** an.
5. Wechseln Sie auf die Registerkarte *Eingabemeldung*.
6. Hier können Sie in den Feldern *Titel* und *Eingabemeldung* eine Nachricht definieren, die angezeigt werden soll, wenn die Zelle mit der Gültigkeitsliste aktiviert wird.
7. Wechseln Sie auf die Registerkarte *Fehlermeldung*.

8. In den Feldern *Titel* und *Fehlermeldung* können Sie eine Fehlermeldung festlegen, die angezeigt werden soll, wenn eine nicht in der Liste stehende Eingabe vorgenommen wird.
9. Indem Sie im Kombinationsfeld *Typ* das Symbol *Stopp* einstellen, werden tatsächlich keine anderen Eingaben als die von Ihnen vorher festgelegten zugelassen. Möchten Sie zwar eine Warnung als Reaktion auf eine Falscheingabe anzeigen, die Eingabe aber dennoch zulassen, dann wählen Sie das Symbol *Warnung* oder *Information*.
10. Bestätigen Sie Ihre Einstellung mit *OK*.

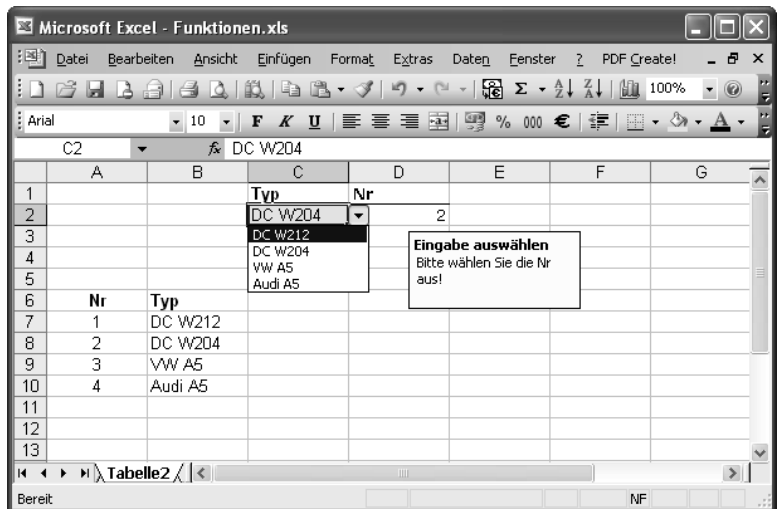


Abbildung 5: Der Typ muss nicht mehr per Hand eingegeben werden

Verwendete Funktionen in dieser Variante

INDEX

Mit Hilfe der Tabellenfunktion **INDEX** können Sie über einen Index einen bestimmten Wert aus einer Matrix finden.

=INDEX(Matrix;Zeile;Spalte)

gibt den Bezug auf eine oder mehrere Zellen zurück, die zur als Matrix angegebenen Matrix gehören. Das Argument **Matrix** ist dabei ein Zellenbereich oder eine Matrixkonstante. Besteht das Argument **Matrix** aus nur einer Zeile oder Spalte, ist das entsprechende Argument **Zeile** bzw. **Spalte** optional. Erstreckt sich eine **Matrix** über mehrere Zeilen und Spalten und ist nur eines der Argumente **Zeile** oder **Spalte** angegeben, liefert die Funktion eine Matrix, die der gesamten zugehörigen Zeile oder Spalte von **Matrix** entspricht.

VERGLEICH

Sucht Werte innerhalb eines Bezugs oder einer Matrix. Verwenden Sie **VERGLEICH** immer dann statt einer der **VERWEIS**-Funktionen, wenn Sie die Position eines Elements in einem Bereich und nicht das Element selbst benötigen.

=VERGLEICH(Suchkriterium;Suchmatrix;Vergleichstyp)

Das Argument **Suchkriterium** ist der Wert, aufgrund dessen der gewünschte Wert in einer Tabelle gesucht wird. **Suchkriterium** gibt den Wert an, gegen den Sie bestimmte Elemente der Suchmatrix abgleichen möchten. Das **Suchkriterium** kann ein Wert (eine Zahl, eine Zeichenfolge oder ein Wahrheitswert) oder ein Bezug auf eine Zelle sein, die eine Zahl, eine Zeichenfolge oder einen Wahrheitswert enthält.

Das Argument **Suchmatrix** ist ein zusammenhängender Zellbereich mit möglichen Vergleichskriterien. **Suchmatrix** muss eine Matrix oder ein Bezug auf eine Matrix sein.

Das Argument **Vergleichstyp** ist die Zahl **-1**, **0** oder **1**. **Vergleichstyp** gibt an, auf welche Weise Microsoft Excel die Werte in einer Suchmatrix mit den Suchkriterien vergleicht.

- ✓ **Vergleichstyp** gleich **1**: Die Funktion gibt den größten Wert zurück, der kleiner oder gleich dem Suchkriterium ist. Die Elemente der Suchmatrix müssen in aufsteigender Reihenfolge angeordnet sein.

- ✓ **Vergleichstyp** gleich 0: Die Funktion gibt den ersten Wert zurück, der gleich dem Suchkriterium ist. Die Elemente der Suchmatrix dürfen in beliebiger Reihenfolge angeordnet sein.
- ✓ **Vergleichstyp** gleich -1: Die Funktion gibt den kleinsten Wert zurück, der größer oder gleich dem Suchkriterium ist. Die Elemente der Suchmatrix müssen in absteigender Reihenfolge angeordnet sein.

Variante 2 – über ein Ereignismakro



Diese Variante finden Sie auf der CD-ROM im Verzeichnis *Beispieldateien* in der Mappe *Funktionen.xls* auf der *TabelleM*.

Soll gänzlich auf den Einsatz von Formeln verzichtet werden, dann können Sie diese Aufgabe auch über ein Ereignismakro lösen. Dabei verfahren Sie wie folgt:

1. Klicken Sie mit der rechten Maustaste auf die Registerlasche der Tabelle und wählen Sie aus dem Kontextmenü den Befehl *Code anzeigen*.
2. In der Entwicklungsumgebung wählen Sie im Codefenster oben aus dem ersten Kombinationsfeld den Befehl **Worksheet**.
3. Wählen Sie aus dem rechts daneben stehenden Kombinationsfeld das Ereignis **Change**.
4. Kompletieren Sie den noch leeren Ereignisrahmen wie folgt:

```
Private Sub Worksheet_Change(ByVal Target As Range)
    Dim Treffer As Range
```

```
    If Target.Address <> "$C$1" Then Exit Sub
```

```
    Set Treffer = Columns(2).Find(what:=Target.Value, _
        lookat:=xlWhole)
```

```
If Treffer Is Nothing Then
```

```
Else
```

```
Target.Offset(0, 1).Value = _  
Cells(Treffer.Row, 1).Value
```

```
End If
```

```
End Sub
```



Das Ereignis **Change** wird dann automatisch ausgeführt, wenn Sie auf der Tabelle eine Eingabe in einer beliebigen Zelle vornehmen. Um dieses Feature ein wenig einzuschränken, legen Sie fest, dass das Ereignis nur dann weitergeführt werden soll, wenn die Eingabe in Zelle **C1** durchgeführt wurde. Dazu fragen Sie die Adresse der Zelle über die Eigenschaft **Address** ab, in der gerade eine Eingabe vorgenommen wurde. Im Argument **Target** kann dabei genau diese Zelle identifiziert werden.

Führen Sie danach eine Suche über die Methode **Find** durch. Dabei geben Sie zunächst an, wo Sie genau in der Tabelle suchen möchten. Da der zu suchende Begriff in Spalte **B** steht, geben Sie hier **Columns(2)** an. Den eigentlichen Suchbegriff geben Sie im Argument **what** an. Im Argument **lookat** können Sie die Art und Weise festlegen, wie Excel die Suche durchführen soll. Über die Konstante **xlWhole** legen Sie fest, dass Excel nur nach dem exakten Suchbegriff im Suchbereich suchen soll. Ist die Suche erfolgreich verlaufen, dann ist die Variable **Treffer** belegt, sonst bleibt diese leer (**Nothing**). Im ersten Fall übergeben Sie dann das Ergebnis der Suche an die Zelle gleich neben der Zelle, in der Sie den Suchbegriff eingegeben haben. Dazu setzen Sie die Eigenschaft **Offset** ein. Über die Anweisung **Target.Offset(0, 1).Value** sagen Sie, dass Sie eine Zelle weiter rechts das Ergebnis speichern möchten.

2. Wie kann ich WVERWEIS auch nach oben ausführen?

Standardmäßig können Sie mit Hilfe der Tabellenfunktion **WVERWEIS** einen Datenbereich mittels eines Suchbegriffs nach unten durchsuchen. Dabei muss der Suchbegriff jedoch zu Beginn des oberen Datenbereichs stehen. Was aber können Sie machen, wenn der zu suchende Begriff eben nicht am oberen Rand des Bereichs steht? Zur besseren Verständlichkeit sehen Sie sich einmal die folgende Abbildung an.

	A	B	C	D	E	F	G
1	Kd.Nr	4711					
2	Kundenname						
3	Umsatz	250					
4							
5							
6							
7	Kundenname	Balmer KG	König	Leo Ludwig	Max M. KG	Fashion T.	KLM GmbH
8	Kd.Nr	4708	4709	4710	4711	4712	4713
9	Umsatz	100	150	200	250	300	350
10							
11							
12							
13							

Abbildung 6: Den Umsatz zu ermitteln ist kein Problem

Aber wenn Sie mit der gleichen Funktion den Kundennamen ermitteln möchten, dann geht dies über die Funktion **WVERWEIS** nicht, da die Kundennummer leider nicht im oberen Bereich des zu durchsuchenden Bereichs steht, sondern in der Mitte. Trotzdem müssen Sie nicht verzweifeln. Es gibt ein paar Varianten, um die Aufgabe dennoch zu lösen.

Variante 1 – über die Funktionen INDEX und VERGLEICH



Diese Variante finden Sie auf der CD-ROM im Verzeichnis *Beispieldateien* in der Mappe *Funktionen.xls* auf der *Tabelle47*.

1. Setzen Sie den Mauszeiger in Zelle **B2**.
2. Erfassen Sie die Formel


$$=INDEX(7:7;VERGLEICH(B1;8:8;0)).$$
3. Drücken Sie .
4. Setzen Sie den Mauszeiger in Zelle **B1**.
5. Wählen Sie aus dem Menü *Daten* den Befehl *Gültigkeit*.



Abbildung 7: Eine Gültigkeitsregel aufstellen

6. Wählen Sie aus dem Kombinationsfeld *Zulassen* den Eintrag *Liste* aus.
7. Geben Sie im Feld *Quelle* den Zellenbezug $=\$B\$8:\$G\8 an.

8. Wechseln Sie auf die Registerkarte *Eingabemeldung*.
9. Hier können Sie in den Feldern *Titel* und *Eingabemeldung* eine Nachricht definieren, die angezeigt werden soll, wenn die Zelle mit der Gültigkeitsliste aktiviert wird.
10. Wechseln Sie auf die Registerkarte *Fehlermeldung*.
11. In den Feldern *Titel* und *Fehlermeldung* können Sie eine Fehlermeldung festlegen, die angezeigt werden soll, wenn eine nicht in der Liste stehende Eingabe vorgenommen wird.
12. Indem Sie im Kombinationsfeld *Typ* das Symbol *Stopp* einstellen, werden tatsächlich keine anderen Eingaben als die von Ihnen vorher festgelegten zugelassen. Möchten Sie zwar eine Warnung als Reaktion auf eine Falscheingabe anzeigen, die Eingabe aber dennoch zulassen, dann wählen Sie das Symbol *Warnung* oder *Information*.
13. Bestätigen Sie Ihre Einstellung mit *OK*.

Verwendete Funktionen in dieser Variante

INDEX

Mit Hilfe der Tabellenfunktion **INDEX** können Sie über einen Index einen bestimmten Wert aus einer Matrix finden.

=INDEX(Matrix;Zeile;Spalte)

gibt den Bezug auf eine oder mehrere Zellen zurück, die zur als **Matrix** angegebenen Matrix gehören. Das Argument **Matrix** ist dabei ein Zellenbereich oder eine Matrixkonstante. Besteht das Argument **Matrix** aus nur einer Zeile oder Spalte, ist das entsprechende Argument **Zeile** bzw. **Spalte** optional. Erstreckt sich **Matrix** über mehrere Zeilen und Spalten und ist nur eines der Argumente **Zeile** oder **Spalte** angegeben, liefert die Funktion eine Matrix, die der gesamten zugehörigen Zeile oder Spalte von **Matrix** entspricht.

VERGLEICH

Sucht Werte innerhalb eines Bezugs oder einer Matrix. Verwenden Sie **VERGLEICH** immer dann statt einer der **VERWEIS**-Funktionen, wenn Sie die Position eines Elements in einem Bereich und nicht das Element selbst benötigen.

=**VERGLEICH**(Suchkriterium;Suchmatrix;Vergleichstyp)

Das Argument **Suchkriterium** ist der Wert, aufgrund dessen der gewünschte Wert in einer Tabelle gesucht wird. **Suchkriterium** gibt den Wert an, gegen den Sie bestimmte Elemente der Suchmatrix abgleichen möchten. Das **Suchkriterium** kann ein Wert (eine Zahl, eine Zeichenfolge oder ein Wahrheitswert) oder ein Bezug auf eine Zelle sein, die eine Zahl, eine Zeichenfolge oder einen Wahrheitswert enthält.

Das Argument **Suchmatrix** ist ein zusammenhängender Zellbereich mit möglichen Vergleichskriterien. **Suchmatrix** muss eine Matrix oder ein Bezug auf eine Matrix sein.

Variante 2 – über ein Ereignismakro



Diese Variante finden Sie auf der CD-ROM im Verzeichnis *Beispieldateien* in der Mappe *Funktionen.xls* auf der *Tabelle47M*.

Soll gänzlich auf den Einsatz von Formeln verzichtet werden, dann können Sie diese Aufgabe auch über ein Ereignismakro lösen. Dabei verfahren Sie wie folgt:

1. Klicken Sie mit der rechten Maustaste auf die Registerlasche der Tabelle und wählen Sie aus dem Kontextmenü den Befehl *Code anzeigen*.
2. In der Entwicklungsumgebung wählen Sie im Codefenster oben aus dem ersten Kombinationsfeld den Befehl **Worksheet**.
3. Wählen Sie aus dem rechts daneben stehenden Kombinationsfeld das Ereignis **Change**.

4. Komplettieren Sie den noch leeren Ereignisrahmen wie folgt:

```
Private Sub Worksheet_Change(ByVal Target As Range)
    Dim Treffer As Range

    If Target.Address <> "$B$1" Then Exit Sub

    Set Treffer = Rows(8).Find(what:=Target.Value, _
        lookat:=xlWhole)

    If Treffer Is Nothing Then

    Else

        Range("B2").Value = _
            Cells(7, Treffer.Column).Value

    End If

End Sub
```



Das Ereignis **Change** wird dann automatisch ausgeführt, wenn Sie auf der Tabelle eine Eingabe in einer beliebigen Zelle vornehmen. Um dieses Feature ein wenig einzuschränken, legen Sie fest, dass das Ereignis nur dann weitergeführt werden soll, wenn die Eingabe in Zelle **B1** durchgeführt wurde. Dazu fragen Sie die Adresse der Zelle über die Eigenschaft **Address** ab, in der gerade eine Eingabe vorgenommen wurde. Im Argument **Target** kann dabei genau diese Zelle identifiziert werden.

Führen Sie danach eine Suche über die Methode **Find** durch. Dabei geben Sie zunächst an, wo Sie genau in der Tabelle suchen möchten. Da der zu suchende Begriff in Zeile **8** steht, geben Sie hier **Rows(8)** an. Den eigentlichen Suchbegriff geben Sie im Argument **what** an. Im Argument **lookat** können Sie die Art und Weise festlegen, wie Excel die Suche durchführen soll. Über die Konstante **xlWhole** legen Sie fest, dass Excel nur nach dem ex-

akten Suchbegriff im Suchbereich suchen soll. Ist die Suche erfolgreich verlaufen, dann ist die Variable **Treffer** belegt, sonst bleibt diese leer (**Nothing**). Im ersten Fall übergeben Sie dann das Ergebnis der Suche an die Zelle **B2**.

	A	B	C	D	E	F	G
1	Kd.Nr	4709					
2	Kundename	König					
3	Umsatz	150					
4							
5							
6							
7	Kundename	Balmer KG	König	Leo Ludwig	Max M. KG	Fashion T.	KLM GmbH
8	Kd.Nr	4708	4709	4710	4711	4712	4713
9	Umsatz	100	150	200	250	300	350
10							
11							
12							
13							

Abbildung 8: Bei der Ereignisvariante werden keine Formeln geschrieben

3. Wie kann ich aus einer Wochennummer den dazugehörigen Montag ermitteln?

In der folgenden Aufgabenstellung soll ausgehend von einer Wochennummer der dazugehörige Montag gefunden werden. Sehen Sie sich zunächst einmal die folgende Abbildung an.

The screenshot shows a Microsoft Excel window titled 'Microsoft Excel - Kap01.xls'. The spreadsheet has the following data:

	A	B	C	D	E
1	Wochennummer	dazugehöriger Montag			
2	19				
3	20				
4	21				
5	33				
6	45				
7	52				
8					
9					
10					

Abbildung 9: Zur Wochennummer den dazugehörigen Montag ermitteln

Variante 1 – über eine Formelschachtelung



Diese Variante finden Sie auf der CD-ROM im Verzeichnis *Beispieldateien* in der Mappe *Funktionen.xls* auf der *Tabelle3*.

Um diese Aufgabe zu lösen, gehen Sie wie folgt vor:

1. Markieren Sie den Zellenbereich **B2:B7**.
2. Erfassen Sie die Formel

$$=A2*7+DATWERT("1.1." \& \text{JAHR}(\text{HEUTE}()))-\text{WOCHENTAG}("1.1." \& \text{JAHR}(\text{HEUTE}()))+2.$$
3. Schließen Sie die Formel über die Tastenkombination **Strg** + **↵** ab.
4. Kontrollieren Sie diese Formel, indem Sie den Bereich **C2:C7** markieren, die Formel **=B2** eingeben und die Eingabe über die Tastenkombination **Strg** + **↵** abschließen.
5. Drücken Sie die Tastenkombination **Strg** + **F1**, um den Dialog *Zellen formatieren* aufzurufen.

6. Im Register *Zahlen* gehen Sie auf die Kategorie *Benutzerdefiniert*.
7. Geben Sie im Feld *Typ* das Format **TTTT** an.
8. Bestätigen Sie mit *OK*.

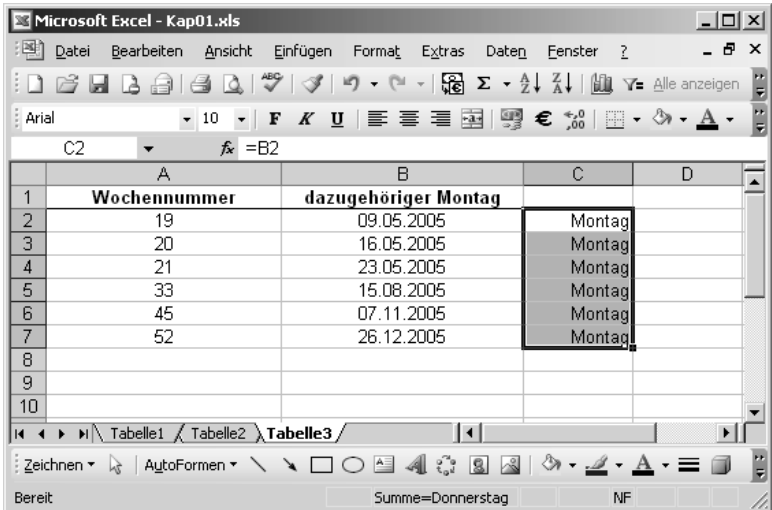


Abbildung 10: Der jeweilige Montag wurde ausgegeben

Verwendete Funktionen in dieser Aufgabe

DATWERT

Mit Hilfe der Tabellenfunktion **DATWERT** können Sie nicht erkannte Datumsangaben, die beispielsweise nach einem Import von Fremddaten in Excel eingefügt wurden, in ein lesbares Excel-Datum umwandeln. Die Funktion **DATWERT** wandelt ein als Text vorliegendes Datum in eine fortlaufende Zahl um.

=DATWERT(Datumstext)

Im Argument **Datumstext** wird der Text bzw. der Zellenbezug, der ein Datum enthalten soll, angegeben. Entspricht **Datumstext** einem Datum, das außerhalb dieses Bereichs liegt, gibt

DATWERT den Fehlerwert **#WERT!** zurück. Ist in **Datumstext** kein Jahr angegeben, verwendet **DATWERT** die Jahresangabe, die von der in Ihrem Computer eingebauten Uhr zur Verfügung gestellt wird.

WOCHENTAG

Mit Hilfe der Tabellenfunktion **WOCHENTAG** können Sie anhand eines Datums bestimmen, ob es sich um einen Arbeitstag oder um einen Tag am Wochenende handelt. Der Tag wird standardmäßig als ganze Zahl ausgegeben, die einen Wert von 1 (Sonntag) bis 7 (Samstag) annehmen kann.

=WOCHENTAG(Datum;Typ)

Im Argument **Datum** wird das gewünschte Datum, zu dem Sie den Wochentag ermitteln möchten, eingegeben oder über einen Zellenbezug hergestellt.

Beim Argument **Typ** geben Sie eine Zahl (1, 2 oder 3) an, die den Typ des Rückgabewertes bestimmt. Mögliche Zahlenwerte dabei sind:

1 oder nicht angegeben – Zahl 1 (Sonntag) bis 7 (Samstag). Verhält sich wie frühere Microsoft Excel-Versionen.

2 – Zahl 1 (Montag) bis 7 (Sonntag).

3 – Zahl 0 (Montag) bis 6 (Sonntag).

JAHR

Mit Hilfe der Tabellenfunktion **JAHR** können Sie eine fortlaufende Zahl in eine Jahreszahl umwandeln oder auch aus einem bereits erfassten Datum das Jahr extrahieren. Das Jahr wird als ganze Zahl zurückgegeben, die einen Wert von 1900 bis 9999 annehmen kann.

=JAHR(Zahl)

Im Argument **Zahl** geben Sie das Datum (bzw. die Zahl) an, dessen Jahr Sie ermitteln möchten.

HEUTE

Mit Hilfe der Tabellenfunktion **HEUTE** können Sie das aktuelle Tagesdatum in einer Zelle einfügen. Dabei wird diese Funktion bei jedem Öffnen der Arbeitsmappe aktualisiert. Excel orientiert sich bei dieser Funktion am aktuellen Tag, der in der Systemsteuerung von Windows eingestellt ist.

Variante 2 – über eine benutzerdefinierte Funktion



Diese Variante finden Sie auf der CD-ROM im Verzeichnis *Beispieldateien* in der Mappe *Funktionen.xls* auf der *Tabelle3M*.

Möchten Sie für diese Aufgabe eine eigene, etwas handlichere Funktion schreiben, dann gehen Sie wie folgt vor:

1. Drücken Sie die Tastenkombination **[Alt] + [F11]**, um in die Entwicklungsumgebung zu gelangen.
2. Wählen Sie aus dem Menü *Einfügen* den Befehl *Modul*.
3. Erfassen Sie folgende benutzerdefinierte Funktion.

Function MoWoTag(WoNr As Integer)

```
MoWoTag = WoNr * 7 + DateValue("1.1." & Year(Date)) -  
- Weekday("1.1." & Year(Date)) + 2
```

End Function

4. Wechseln Sie zurück auf Ihre Tabelle.
5. Markieren Sie den Bereich **B2:B7**.
6. Rufen Sie die gerade erstellte benutzerdefinierte Funktion auf, indem Sie die Formel **=moWoTag(A2)** erfassen.
7. Schließen Sie die Eingabe über die Tastenkombination **[Strg] + [↵]** ab.

	A	B	C	D	E
1	Wochennummer	dazugehöriger Montag			
2	19	09.05.2005	Montag		
3	20	16.05.2005	Montag		
4	21	23.05.2005	Montag		
5	33	15.08.2005	Montag		
6	45	07.11.2005	Montag		
7	52	26.12.2005	Montag		
8					
9					
10					

Abbildung 11: Die etwas kürzere Lösung über eine benutzerdefinierte Funktion



Die benutzerdefinierte Funktion wurde im Prinzip 1:1 aus der ersten Variante übernommen.

Verwendete VBA-Funktionen in dieser Aufgabe

DATEVALUE

Diese Funktion wandelt eine Zeichenfolge in einen gültigen Datumswert um.

DateValue(Datum)

Das erforderliche Argument **Datum** ist normalerweise ein Zeichenfolgenausdruck, der ein Datum aus dem Bereich vom 1. Januar 100 bis zum 31. Dezember 9999 repräsentiert. Alternativ können Sie für **Datum** aber auch einen beliebigen Ausdruck angeben, der ein Datum, eine Zeit oder beides aus diesem Bereich darstellen kann.

YEAR

Über diese Funktion können Sie aus einem Datum den Jahresanteil abfragen.

Year(Datum)

Das erforderliche Argument **Datum** ist ein beliebiger Wert vom Typ **Variant**, ein numerischer Ausdruck, ein Zeichenfolgenausdruck oder eine beliebige Kombination, die ein Datum darstellen kann.

WEEKDAY

Diese Funktion gibt den Wochentag aus einem Datumswert zurück (1 bis 7).

Weekday(date, [firstdayofweek])

Die Syntax für die Weekday-Funktion besteht aus folgenden benannten Argumenten:

Teil	Beschreibung
date	Erforderlich. Wert vom Typ Variant , numerischer Ausdruck, Zeichenfolgenausdruck oder eine beliebige Kombination, die ein Datum darstellen kann. Wenn date den Wert 0 enthält, wird 0 zurückgegeben.
firstdayofweek	Optional. Eine Konstante, die den ersten Wochentag angibt. Wird die Konstante nicht angegeben, so wird vbSunday angenommen.

Tabelle 1: Die Argumente der Funktion Weekday

Das Argument **firstdayofweek** hat folgende Einstellungen:

Konstante	Wert	Beschreibung
vbUseSystem	0	NLS-API-Einstellung wird verwendet.
vbSunday	1	Sonntag (Voreinstellung)
vbMonday	2	Montag
vbTuesday	3	Dienstag

Konstante	Wert	Beschreibung
vbWednesday	4	Mittwoch
vbThursday	5	Donnerstag
vbFriday	6	Freitag
vbSaturday	7	Samstag

Tabelle 2: Die Argumente von firstdayofweek

4. Wie kann ich aus einer Zahleneingabe eine Uhrzeit machen?

Bei der folgenden Aufgabenstellung liegen in einer Tabelle Uhrzeiten im Zahlenformat vor. So repräsentiert beispielsweise die Zahl 925 die Zeitangabe 9:25 Uhr. Sehen Sie sich zunächst einmal die folgende Abbildung an.



The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D
1	Uhrzeit als Zahl	umgewandelte Uhrzeit		
2		925		1.Methode
3		1355		
4		1559		
5		2105		
6		2315		
7		2354		
8				
9				
10				
11				

Abbildung 12: Die Uhrzeiten sollen umgewandelt werden