

## Excel für Microsoft 365

# Funktionen (Matrix) und dynamische Arrays





## Inhaltsverzeichnis

Einleitung.....	3
Integrierte Matrixfunktionen.....	3
Verwendung einer Matrixfunktion .....	5
Matrixkonstante.....	6
Zellbereich einer Matrix markieren.....	7
Beispiele für Matrixfunktionen.....	9
Dynamische Arrays.....	12
Neue Funktionen für die dynamische Arrays.....	13
Zusätzliche Informationen zu den dynamischen Arrays .....	21
Der Fehlerwert #ÜBERLAUF! .....	22
Einsatz von dynamischen Arrays umgehen.....	22

## Abbildungsverzeichnis




Abb. 1: <i>Eine Matrixberechnung bzw. Matrixfunktion in der Bearbeitungsleiste.....</i>	6
Abb. 2: <i>Dialogfeld <b>Gehe zu</b> .....</i>	8
Abb. 3: <i>Dialogfeld <b>Inhalte auswählen</b>.....</i>	8
Abb. 4: <i>Beispiel einer „normalen Funktion“ als Matrixfunktion .....</i>	9
Abb. 5: <i>Beispiel für eine Matrixfunktion mit zweidimensionalen Matrizen .....</i>	9
Abb. 6: <i>Beispiel für eine Matrixkonstante.....</i>	9
Abb. 7: <i>Wiederholung von Beispiel 3 mit anderer Anordnung der Matrixkonstanten .....</i>	10
Abb. 8: <i>Beispiele für Matrixkonstanten.....</i>	10
Abb. 9: <i>Beispiel für eine Trendberechnung .....</i>	10
Abb. 10: <i>Das letzte Wort in einem Satz extrahieren .....</i>	11
Abb. 11: <i>Die Ermittlung der Quadratwurzel als Matrixberechnung .....</i>	12
Abb. 12: <i>Ein dynamisches Array erkennen Sie an dem blauen Rahmen.....</i>	13
Abb. 13: <i>Ausschnitt aus der Tabelle <b>Autohändler.xlsx</b>.....</i>	14
Abb. 14: <i>Das dynamische Array mit den eindeutigen Verkäufernamen .....</i>	15
Abb. 15: <i>Die gefilterten Daten (Ausschnitt).....</i>	17
Abb. 16: <i>Die gefilterten Daten nur mit Verkäufer und Gewinn (Ausschnitt).....</i>	18
Abb. 17: <i>Die gefilterten Gewinne des Verkäufers Becker.....</i>	19
Abb. 18: <i>Beispiel für die Funktion <b>SEQUENZ</b>.....</i>	19
Abb. 19: <i>Beispiel für die Funktion <b>ZUFALLSMATRIX</b>.....</i>	19
Abb. 20: <i>Liste mit den unsortierten Messwerten.....</i>	20

Abb. 21: *Liste mit den sortierten Messwerten*..... 21

Abb. 22: *Ausgangssituation zur Berechnung einer Sparanlage* ..... 23

Abb. 23: *Ergebnis zur Berechnung einer Sparanlage*..... 24

## Einleitung

Eine Matrixformel kann mehrere Berechnungen durchführen und entweder ein einzelnes Ergebnis oder mehrere Ergebnisse liefern. Matrixformeln bearbeiten zwei oder mehr Wertsätze (Zellbereiche; keine einzelnen Tabellenzellen), die als *Matrixargumente* bezeichnet werden. Excel enthält spezielle Matrixfunktionen, aber es gibt auch Anwendungen, wo eine eigentlich „normale“ Formel als Matrixformel benutzt werden muss. Auch hierfür finden Sie in dem Skript ein Beispiel. Sie erstellen eine Matrixformel auf dieselbe Weise wie andere Formeln, aber mit dem Unterschied, dass Sie zur Bestätigung der Formeleingabe die Tastenkombination    verwenden. Neu hinzugekommen sind die dynamischen Arrays, die zwar den Matrixfunktionen ähneln, aber trotzdem anders gehandhabt werden. In diesem Skript wird in erster Linie die Version **Excel für Microsoft 365** behandelt. Bezüglich der Matrixfunktionen können Sie das Skript auch bei Verwendung der älteren Excel-Versionen einsetzen. Die dynamischen Arrays gibt es aber nur in *Excel für Microsoft 365* und *Excel 2021*.

## Integrierte Matrixfunktionen

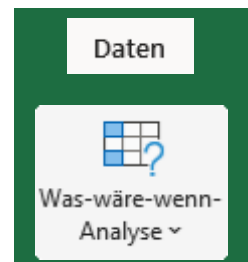
Zunächst eine kleine Übersicht von Matrixfunktionen, die bereits bei Excel integriert sind (die Übersicht ist nicht unbedingt vollständig; die in eckige Klammer eingeschlossenen Argumente sind optional; es gibt noch eine Reihe weiterer Funktionen, die zu den dynamischen Arrays gehören; siehe Kapitel **Neue Funktionen für dynamische Arrays**, Seite 13):

<i>Funktion</i>	<i>Beschreibung</i>
<b>BEREICH.VERSCHIEBEN</b>	Gibt einen Verweis auf einen Bereich zurück, der eine festgelegte Anzahl von Zeilen und Spalten von einer Tabellenzelle oder einem Zellbereich entfernt ist. Der zurückgegebene Verweis kann eine einzelne Tabellenzelle oder ein Zellbereich sein. Sie können die Anzahl der zurückzugebenden Zeilen und Spalten festlegen. <b>BEREICH.VERSCHIEBEN(Bezug;Zeilen;Spalten;[Höhe];[Breite])</b>
<b>BEREICHE</b>	Gibt die Anzahl der innerhalb eines Verweises aufgeführten Bereiche zurück. Ein Bereich (Teilbereich) kann sowohl aus mehreren zusammenhängenden Tabellenzellen (Zellbereich) als auch aus nur einer Tabellenzelle bestehen. <b>BEREICHE(Bezug)</b>
<b>HÄUFIGKEIT</b>	Berechnet, wie oft Werte innerhalb eines Wertebereichs auftreten, und gibt eine Häufigkeitsverteilung als einspaltige Matrix zurück. <b>HÄUFIGKEIT(Daten;Klassen)</b>
<b>INDEX</b>	Gibt den Wert eines Elements in einer Tabelle oder einer Matrix zurück. Das Element wird mit seiner Zeilen- und Spaltennummer angegeben. <b>INDEX(Matrix;[Zeile];[Spalte])</b> <b>INDEX(Bezug;Zeile;[Spalte];[Bereich])</b>
<b>KORREL</b>	Gibt den Korrelationskoeffizienten einer zweidimensionalen Zufallsgröße zurück, deren Werte in den Zellbereichen <b>Matrix1</b> und <b>Matrix2</b> stehen. Mit dem Korrelationskoeffizienten lässt sich feststellen, ob eine Beziehung zwischen zwei Eigenschaften vorliegt. <b>KORREL(Matrix1;Matrix2)</b>

<b>Funktion</b>	<b>Beschreibung</b>
<b>MDET</b>	Liefert die Determinante einer Matrix. <b>MDET(Matrix)</b>
<b>MINV</b>	Gibt die Inverse einer Matrix (die zu einer Matrix gehörende Kehrmatrix) zurück. <b>MINV(Matrix)</b>
<b>MMULT</b>	Gibt das Produkt zweier Matrizen zurück. Das Ergebnis ist eine Matrix, die dieselbe Anzahl an Zeilen wie <b>Matrix1</b> und dieselbe Anzahl an Spalten wie <b>Matrix2</b> aufweist. <b>MMULT(Array1;Array2)</b>
<b>MTRANS</b>	Mit dieser Funktion wird ein vertikaler Zellbereich als horizontaler Bereich zurückgegeben oder umgekehrt. Die Funktion muss als Matrixformel in einem Bereich eingegeben werden, der die gleiche Anzahl von Zeilen bzw. Spalten wie der Quellbereich hat. <b>MTRANS(Matrix)</b>
<b>PEARSON</b>	Gibt den pearsonschen Korrelationskoeffizienten r zurück. Dieser Koeffizient ist ein dimensionsloser Index mit dem Wertebereich zwischen -1,0 und 1,0 und ein Maß dafür, inwieweit zwischen zwei Datenmengen eine lineare Abhängigkeit besteht. <b>PEARSON(Matrix1;Matrix2)</b>
<b>RKP</b>	Liefert die Parameter eines exponentiellen Trends. In Regressionsanalysen berechnet diese Funktion eine Exponentialkurve, die möglichst gut an die von Ihnen bereitgestellten Daten angepasst ist, und liefert eine Matrix von Werten, die diese Kurve beschreibt. <b>RKP(Y_Werte;[X_Werte];[Konstante];[Stats])</b>
<b>SCHÄTZER</b>	Gibt den Schätzwert für einen linearen Trend zurück. Der Vorhersagewert ist ein y-Wert für einen gegebenen x-Wert. <b>SCHÄTZER(x;Y_Werte;X_Werte)</b>
<b>SPALTE</b>	Mit dieser Funktion wird die Spaltennummer des jeweiligen Zellbezug zurückgegeben. <b>SPALTE(Bezug)</b>
<b>SPALTEN</b>	Gibt die Anzahl der Spalten eines Bezugs zurück. <b>SPALTEN(Matrix)</b>
<b>SUMMEX2MY2</b>	Summiert für sich entsprechende Werte zweier Matrizen die aus Quadratzahlen gebildeten Differenzen. <b>SUMMEX2MY2(Matrix_x;Matrix_y)</b>
<b>SUMMEX2PY2</b>	Summiert für sich entsprechende Werte zweier Matrizen die aus Quadratzahlen gebildeten Summen. <b>SUMMEX2PY2(Matrix_x;Matrix_y)</b>
<b>SUMMEXMY2</b>	Gibt die Summe der quadrierten Differenzen einander entsprechender Werte in zwei Matrix zurück. <b>SUMMEXMY2(Matrix_x;Matrix_y)</b>

<b>Funktion</b>	<b>Beschreibung</b>
<b>SVERWEIS</b>	Durchsucht die erste Spalte einer Matrix und durchläuft die Zeile nach rechts, um den Wert einer Tabellenzelle zurückzugeben. <b>SVERWEIS(Suchkriterium;Matrix;Spaltenindex;[Bereich_Verweis])</b>
<b>TREND</b>	Gibt Werte zurück, die sich aus einem linearen Trend ergeben. <b>TREND(Y_Werte;X_Werte;Neue_X_Werte;[Konstante])</b>
<b>VERGLEICH</b>	Gibt die relative Position eines Elements in einer Matrix zurück, die einem bestimmten Wert in einer angegebenen Reihenfolge entspricht. <b>VERGLEICH(Suchkriterium;Suchmatrix;[Vergleichstyp])</b>
<b>VERWEIS</b>	Mit der Funktion wird ein Wert aus einem Bereich mit einer Zeile oder einer Spalte oder aus einer Matrix zurückgegeben. <b>VERWEIS(Suchkriterium;Suchvektor;Ergebnisvektor)</b> <b>VERWEIS(Suchkriterium;Matrix)</b>
<b>WVERWEIS</b>	Durchsucht die erste Zeile einer Matrix und durchläuft die Spalte nach unten, um den Wert einer Tabellenzelle zurückzugeben. <b>WVERWEIS(Suchkriterium;Matrix;Zeilenindex^;[Bereich_Verweis])</b>
<b>ZEILE</b>	Gibt die Zeilennummer eines Verweises zurück. <b>ZEILE([Bezug])</b>


Neben diesen Funktionen gibt es außerdem noch Befehle, die eine Matrixfunktion beinhalten. Dazu gehört beispielsweise die *Mehrfachoperation* (Register **Daten**, Gruppe *Datentools*, Symbol **Was-wäre-wenn-Analyse**, Befehl **Datentabelle**).




## Verwendung einer Matrixfunktion

Prinzipiell werden Matrixfunktionen genauso in Berechnungen verwendet wie „normale“ Funktionen. Trotzdem gibt es bei Matrixfunktionen auch Unterschiede. Hier drei, sehr bedeutsame Unterschiede:

1. Die meisten Matrixfunktionen bzw. Matrixformeln liefern mehrere Ergebniswerte. Dazu müssen auch mehrere Tabellenzellen ausgewählt, also markiert werden. Um wie viele Tabellenzellen es sich dabei genau handelt und ob dabei Tabellenzellen in einer Zeile oder Spalte (oder beides) markiert werden müssen, hängt von der jeweiligen Matrixfunktion bzw. Matrixformel und den Ausgangsdaten ab. Gegebenenfalls müssen Sie durch Ausprobieren herausfinden, welcher Zellbereich markiert werden muss.
2. Wie fast alle Funktionen, benötigen auch Matrixfunktionen Angaben in Form von Funktionsargumenten. Bei diesen Funktionsargumenten handelt es sich in den meisten Fällen um Matrizen, was in Excel identisch ist mit Zellbereichen. Auch hier kommt es darauf an, ob die Daten in Zeilen- oder Spaltenform (oder beides) angeordnet sind.


3. Wird eine Matrixfunktion direkt in die Tabellenzelle bzw. in den Zellbereich eingetragen, muss zur Eingabebestätigung die Tastenkombination  verwendet werden<sup>1</sup>.

Wenn die Eingabe der Matrixfunktion mit der Tastenkombination  abgeschlossen worden ist, wird die gesamte Berechnung (inkl. dem Gleichheitszeichen) in geschweifte Klammern { und } eingeschlossen (sichtbar in der Bearbeitungsleiste; siehe Abbildung 1).



● Die beiden geschweiften Klammern (hier rot eingefärbt) signalisieren, dass es sich um eine Matrixberechnung bzw. Matrixfunktion handelt.



Abb. 1: Eine Matrixberechnung bzw. Matrixfunktion in der Bearbeitungsleiste



**Anmerkung:** Wenn eine Matrixfunktion mehrere Ergebniswerte liefert, können Sie einen einzelnen Ergebniswert (oder einen Teil der Ergebniswerte) nicht bearbeiten oder löschen. Dies geht nur wenn der Zellbereich mit allen Ergebniswerten markiert wird. Wenn Sie trotzdem versuchen, eine einzelne Ergebniszelle zu bearbeiten oder zu löschen, können Sie den Versuch nur mit der Taste  abbrechen.

## Matrixkonstante

In einer gewöhnlichen Funktion können Sie entweder einen Bezug auf eine Tabellenzelle mit einem bestimmten Wert als Funktionsargument angeben, oder als konstanten Wert. Entsprechend können Sie in einer Matrixformel, in der Sie einen Bezug auf einen Zellbereich verwenden, die Werte dieses Zellbereichs auch als konstante Matrix eingeben. Die Matrix der konstanten Werte wird als *Matrixkonstante* bezeichnet. Im Allgemeinen werden Sie Matrixkonstanten anstelle von Bezügen verwenden, wenn Sie vermeiden wollen, jede Konstante in eine eigene Tabellenzelle in Ihrer Tabelle einzugeben.

So geben Sie eine Matrixkonstante ein:

1. Geben Sie die Werte direkt in die Funktion ein und schließen sie in geschweifte Klammern ein ({}).<sup>2</sup> Für die öffnende Klammer ( { ) verwenden Sie die Tastenkombination  und für die schließende Klammer ( } ) die Tastenkombination .
2. Trennen Sie Werte in verschiedenen **Spalten** durch **Punkte**.
3. Trennen Sie Werte in verschiedenen **Zeilen** durch **Semikola**.

<sup>1</sup> Liefert eine Matrixformel nur ein Ergebniswert (es wird also auch nur eine Tabellenzelle markiert), muss die Formel-eingabe nicht unbedingt mit der Tastenkombination  vorgenommen werden. In den meisten Fällen reicht die Eingabetaste () allein (es gibt aber auch Ausnahmefälle; siehe Abbildung 10, Seite 10).

<sup>2</sup> Bitte verwechseln Sie diese geschweiften Klammern nicht mit denen, die sich am Anfang bzw. Ende der Matrix-formeln befinden.



Die nachfolgende Tabelle zeigt einige Beispiele von Matrixkonstanten. Dabei wird die Excel-Schreibweise die der mathematischen Schreibweise gegenübergestellt.

Beispiele:

<i>Matrix-Schreibweise (Mathematik)</i>	<i>Matrix-Schreibweise (Excel)</i>
$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$	<b>{1.2.3;4.5.6}</b>
$(10 \ 20 \ 30 \ 40)$	<b>{10.20.30.40}</b>
$\begin{pmatrix} 32,6 \\ 54,9 \\ 12,7 \end{pmatrix}$	<b>{32,6;54,9;12,7}</b>
$\begin{pmatrix} 1 & 2 & 3 \\ a & b & c \end{pmatrix}$	<b>{1.2.3;"a"."b"."c"}</b>
$\begin{pmatrix} 56 & 34 \\ WAHR & FALSCH \end{pmatrix}$	<b>{56.34;WAHR.FALSCH}</b>

Matrixkonstanten können folgende Inhalte besitzen:


- Matrixkonstanten können Zahlen, Text, Wahrheitswerte oder Fehlerwerte enthalten.
- Zahlen in Matrixkonstanten können im Ganzzahlformat, Dezimalformat oder im wissenschaftlichen Zahlenformat (Exponentialschreibweise) angegeben werden.
- Text muss in doppelte Anführungszeichen (") eingeschlossen werden.
- In derselben Matrixkonstante können verschiedene Wertetypen gleichzeitig verwendet werden.
- Die Werte einer Matrixkonstante müssen Konstanten und dürfen keine Funktionen sein.
- Matrixkonstanten dürfen kein Dollarzeichen, Klammern oder Prozentzeichen enthalten.
- Matrixkonstanten dürfen keinen Zeilen oder Spalten ungleicher Länge enthalten.

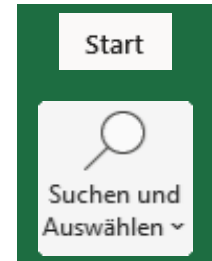
Excel erlaubt es nicht, in einer Matrixfunktion Zellbezüge oder Namen genauso wie konstante Werte aufzulisten. Beispielsweise können Sie nicht mit **{A1.B1.C1}** die Matrix angeben, die die Werte aus den Tabellenzellen **A1**, **B1** und **C1** enthält. Stattdessen müssen Sie den Bezug auf den Zellbereich **A1:C1** verwenden oder die Werte dieser Tabellenzellen direkt als Konstanten in die Matrix eingeben, z.B. **{10.20.30}**.

## Zellbereich einer Matrix markieren



Wenn Sie in einer Arbeitsmappe viele Matrixberechnungen durchgeführt haben, ist es häufig schwierig nachzuvollziehen, in welchen Zellbereichen sich die Matrixberechnungen befinden. Sie können natürlich jede einzelne Tabellenzelle auswählen und in der Bearbeitungsleiste nachschauen, ob sich in der Tabellenzelle eine Matrixformel befindet oder nicht (erkennbar an den geschweiften Klammern

vor und hinter der Formel; siehe Abbildung 1, Seite 6). Allerdings wissen Sie dann immer noch nicht, wie groß der zugehörige Zellbereich ist, in dem sich die Matrixformel befindet. Excel stellt aber zwei Möglichkeiten bereit, mit denen Sie den kompletten Zellbereich einer Matrixformel markieren lassen können. Voraussetzung ist allerdings, dass mindestens eine Tabellenzelle der Matrixformel ausgewählt ist:

- Wählen Sie im Register **Start** in der Gruppe **Bearbeiten** das Symbol **Suchen und Auswählen** und den Befehl **Gehe zu** (alternativ benutzen Sie die Funktionstaste ). Im Dialogfeld **Gehe zu** (siehe Abbildung 2) klicken Sie auf die Schaltfläche **Inhalte...**. Im Dialogfeld **Inhalte auswählen** wählen Sie die Option **Aktuelles Array** (siehe Abbildung 3) und bestätigen das Dialogfeld.



oder

- Benutzen Sie die Tastenkombination   .

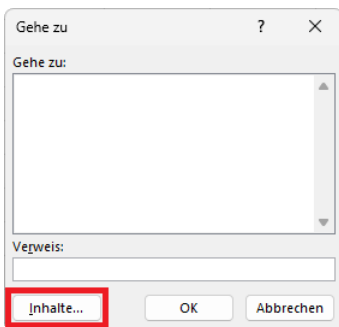


Abb. 2: Dialogfeld **Gehe zu**

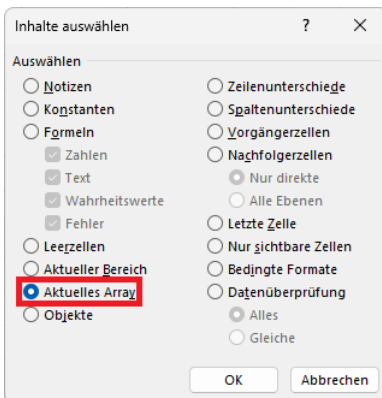


Abb. 3: Dialogfeld **Inhalte auswählen**

## Beispiele für Matrixfunktionen

Im ersten Beispiel<sup>3</sup> wird eine „normale“ Funktion als Matrixfunktion eingesetzt. Es handelt sich in diesem Beispiel um die Funktion **SIN**. Dabei soll der Sinus für die X-Werte von **-2,0** bis **+2,0** (in Schritten von **0,5**) ermittelt werden (siehe Abbildung 4).

	A	B	C	D	E	F	G	H	I	J
1	x	fx=sin(x)	fx=sin(x)							
2	-2,0	-0,9093	-0,9093		In die Tabellenzelle <b>B2</b> wird die Formel <b>=SIN(A2)</b> eingetragen und ganz normal mit der Eingabetaste (<Return>) bestätigt. Danach Tabellenzelle <b>B2</b> auswählen und die Ergebnisse für die Tabellenzellen <b>B3:B10</b> mit dem automatischen Ausfüllen bis zur Tabellenzelle <b>B10</b> kopieren.					
3	-1,5	-0,9975	-0,9975							
4	-1,0	-0,8415	-0,8415							
5	-0,5	-0,4794	-0,4794							
6	0,0	0,0000	0,0000							
7	0,5	0,4794	0,4794		Zunächst wird der Zellbereich <b>C2:C10</b> markiert und anschließend die Formel <b>=SIN(A2:A10)</b> eingetragen. Für die Berechnung der Ergebnisse muss nun noch die Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> gedrückt werden.					
8	1,0	0,8415	0,8415							
9	1,5	0,9975	0,9975							
10	2,0	0,9093	0,9093							

Abb. 4: Beispiel einer „normalen Funktion“ als Matrixfunktion

Das zweite Beispiel zeigt eine Matrixfunktion, die eine zweidimensionale Matrix als Ausgangsmatrix benötigt und die eine zweidimensionale Matrix als Ergebnis liefert. Für dieses Beispiel wurde die Funktion **MINV** (invertierte Matrix) genommen (siehe Abbildung 5).

	A	B	C	D	E	F	G	H	I
1		34	65	23			-0,041	0,163	0,020
2	Matrix A =	5	8	3		MINV(A) =	0,224	-1,898	0,024
3		77	66	22			-0,531	5,122	-0,098
4									
5							Zunächst wird der Zellbereich <b>G1:I3</b> markiert, dann die Formel <b>=MINV(B1:D3)</b> eingetragen und zum Abschluß mit der Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> bestätigt.		
6									
7									
8									

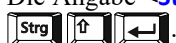
Abb. 5: Beispiel für eine Matrixfunktion mit zweidimensionalen Matrizen

Im dritten Beispiel werden die Elemente zweier Matrizen multipliziert (Funktion **MMULT**). Dabei wird die eine Matrix in einem Zellbereich angegeben, während die zweite Matrix in Form einer Matrixkonstante in die Formel eingetragen wird (siehe Abbildung 6).

	A	B	C	D	E	F	G	H	I	J	K
1		1	10	100			963				
2	Matrix A =	5	50	500			4815				
3		9	90	900			8667				
4											
5		$A * \begin{pmatrix} 3 \\ 6 \\ 9 \end{pmatrix} = \begin{pmatrix} 1 & 10 & 100 \\ 5 & 50 & 500 \\ 9 & 90 & 900 \end{pmatrix} * \begin{pmatrix} 3 \\ 6 \\ 9 \end{pmatrix}$					Zunächst wird der Zellbereich <b>G1:G3</b> markiert, dann die Formel <b>=MMULT(B1:D3;{3;6;9})</b> eingetragen und mit der Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> abgeschlossen.				
6											
7											
8											
9		Im grünen Bereich zunächst die mathematische Formel für die									
10		Matrixmultiplikation. Die Matrix A ist vorgegeben, während die									
11		zweite Matrix als Matrixkonstante in die Formel eingegeben									
12		wird.									

Abb. 6: Beispiel für eine Matrixkonstante

<sup>3</sup> Die Angabe **<Strg>+<Umschalten>+<Eingabetaste>** in den einzelnen Beispielen entspricht der Tastenkombination



Im vierten Beispiel wird noch mal Beispiel 3 wiederholt. Diesmal allerdings wird bei der Matrixkonstanten nicht das Semikolon als Trennzeichen zwischen den Werten **3**, **6** und **9** genommen, sondern der Punkt (siehe Abbildung 7).

	A	B	C	D	E	F	G	H	I	J
1		1	10	100			#WERT!			
2	Matrix A =	5	50	500			#WERT!			
3		9	90	900			#WERT!			
4										
5	$A * \begin{pmatrix} 3 \\ 6 \\ 9 \end{pmatrix} = \begin{pmatrix} 1 & 10 & 100 \\ 5 & 50 & 500 \\ 9 & 90 & 900 \end{pmatrix} * \begin{pmatrix} 3 \\ 6 \\ 9 \end{pmatrix}$				Zunächst wird der Zellbereich <b>G1:G3</b> markiert, dann die Formel <b>=MMULT(B1:D3;{3.6.9})</b> eingetragen und mit der Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> bestätigt. Das Ergebnis ist allerdings diesmal ein Fehlerwert, denn die Matrixkonstante wurde als 1x3-Matrix (Matrixelemente werden jeweils durch einen Punkt getrennt) und nicht als 3x1-Matrix (Matrixelemente werden jeweils durch ein Semikolon getrennt) angegeben. Dieses Beispiel zeigt wie wichtig es ist, den richtigen Zellbereich für die Matrixberechnung anzugeben.					
6										
7										
8										
9										
10										
11										
12										
13										

Abb. 7: Wiederholung von Beispiel 3 mit anderer Anordnung der Matrixkonstanten

Im fünften Beispiel wird nur eine Matrixkonstante verwendet. Als Beispiel wird eine transponierte Matrix mit der Funktion **MTRANS** erzeugt (siehe Abbildung 8).

	A	B	C	D	E	F	G	H	I	J
1	1	4	7		Den zunächst leeren Zellbereich <b>A1:C3</b> markieren, dann Formel <b>=MTRANS({1.2.3;4.5.6;7.8.9})</b> eingeben und mit der Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> bestätigen.					
2	2	5	8							
3	3	6	9							
4										
5	1	4			Den zunächst leeren Zellbereich <b>A5:B7</b> markieren, dann Formel <b>=MTRANS({1.2.3;4.5.6})</b> eingeben und mit der Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> bestätigen.					
6	2	5								
7	3	6								
8										
9	1	3			Den zunächst leeren Zellbereich <b>A9:B11</b> markieren, dann Formel <b>=MTRANS({1.2;3.4;5.6})</b> eingeben und mit der Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> bestätigen. Bei diesem Beispiel entsteht in den Tabellenzellen <b>A11</b> und <b>B11</b> jeweils ein Fehlerwert. Für diese Formel hätte eigentlich der Zellbereich <b>A9:C10</b> markiert werden müssen.					
10	2	4								
11	#NV	#NV								
12										
13										
14										

Abb. 8: Beispiele für Matrixkonstanten




Im sechsten Beispiel wird die Matrixfunktion **TREND** gezeigt, die eher einen Bezug zur Praxis hat als die Funktionen **MINV**, **MMULT** oder **MTRANS** aus der linearen Algebra. Ausgehend von einer Grundmenge an existierenden Werten berechnet die Funktion **TREND** eine Reihe von Werten, die, wenn Sie in einem XY-Diagramm als Linie dargestellt werden, eine steigende oder sinkende Gerade ergibt.

	A	B	C	D	E	F	G
1		Jan	Feb	Mrz	Apr	Mai	Jun
2	Verkaufszahlen	10.765	11.263	11.039	11.471	11.832	11.621
3	Trend	10.873	11.057	11.240	11.424	11.607	11.790
4							
5	Zellbereich <b>B3:G3</b> markieren, dann die Formel <b>=TREND(B2:G2)</b> eingeben und mit der Tastenkombination <b>&lt;Strg&gt;+&lt;Umschalten&gt;+&lt;Eingabetaste&gt;</b> bestätigen.						
6	Die Ergebnisse wurden mit dem Zahlenformat <b>#,##0</b> (ohne Dezimalstellen) formatiert.						
7							

Abb. 9: Beispiel für eine Trendberechnung

Beim siebten Beispiel handelt es sich auf den ersten Blick gar nicht um eine Matrixformel, allerdings muss sie als Matrixformel bestätigt werden, damit das gewünschte Ergebnis herauskommt. Dabei geht es in der Formel darum, aus einem längeren Satz, der aus mehreren Wörtern besteht, das letzte Wort zu extrahieren. Als Beispiel steht der Satz in der Tabellenzelle **A1**<sup>4</sup> und die Formel in der Tabellenzelle **B1**. Die Formel lautet:

```
=TEIL(A1;MAX((TEIL(A1;ZEILE(INDIREKT("1:"&LÄNGE(A1)))));1)=" ")*ZEILE(INDIREKT("1:"&LÄNGE(A1)))+1;LÄNGE(A1))
```

Die Formel sieht recht kompliziert aus und sie ist es auch. An dieser Stelle wird auf eine genaue Beschreibung der einzelnen Schritte der Formel verzichtet. Vertrauen Sie einfach darauf, dass die Formel genau das tut, was sie soll. Nach der Formeleingabe müssen Sie die Eingabe mit der Tastenkombination    abschließen, auch wenn es sich eigentlich gar nicht um eine Matrixberechnung handelt und die Formel auch nur in eine einzige Tabellenzelle eingetragen worden ist. Bestätigen Sie die Formeleingabe nur mit der Eingabetaste, erhalten Sie ein falsches Ergebnis. Abbildung 10 zeigt nochmal die Formel in Aktion.

	A	B	C	D	E	F
1	Das ist ein langer Satz.	Satz.				
2	Das ist ein ganz langer Satz ohne Punkt und Komma	Komma				
3						
4	Geben Sie in die Tabellenzelle B1 die Formel					
5						
6	=TEIL(A1;MAX((TEIL(A1;ZEILE(INDIREKT("1:"&LÄNGE(A1)))));1)=" ")*ZEILE(INDIREKT("1:"&LÄNGE(A1)))+1;LÄNGE(A1))					
7						
8	ein und bestätigen die Eingabe mit <Strg>+<Umschalten>+<Eingabetaste>.					

Abb. 10: Das letzte Wort in einem Satz extrahieren

**Anmerkung:** Sie werden sich vielleicht fragen, muss das so kompliziert sein, oder geht das nicht auch mit einer einfacheren Formel? Die Antwort ist: ja, es gibt eine einfachere Formel, die auch nicht als Matrixformel eingegeben werden muss. Die gezeigte Formel soll ja auch nur als Beispiel dafür dienen, dass es Situationen in Excel gibt, wo Sie zur Lösung eines Problems eine Matrixformel benötigen, auch wenn die Formel keine Matrixfunktionen enthält und sie auch nur einen Ergebniswert liefert. Sie können für dieses Beispiel auch die Blitzvorschau von Excel einsetzen (siehe Skript **Excel für Microsoft 365 – Blitzvorschau**). In diesem Fall benötigen Sie überhaupt keine Excel-Funktion. Übrigens: die einfachere Formel lautet (auch wenn sie immer noch recht lang ist und kompliziert aussieht):

```
=RECHTS(A1;LÄNGE(A1)-FINDEN("#";WECHSELN(A1;" ";"#";LÄNGE(A1)-LÄNGE(WECHSELN(A1;" "))))
```

Wie Sie am ersten Beispiel in Abbildung 10 sehen können, liefert die Formel (es spielt keine Rolle, ob es sich hierbei um die Matrixformel oder die in dieser Anmerkung gezeigten kürzeren Formel handelt) beim Ergebnis auch noch den Punkt hinter dem Wort. Die Formeln sind also nicht ganz perfekt.


<sup>4</sup> Der Satz dürfte sicherlich breiter sein als die Spalte, in der er steht. Aber es ist ja kein großes Problem, die Spaltenbreite an die Satzlänge anzupassen.

## Dynamische Arrays

Die dynamischen Arrays<sup>5</sup> sind eine Neuerung von Excel für Microsoft 365 und auch nur dort bzw. in Excel 2021 verfügbar. In älteren Excel-Versionen können sie nicht eingesetzt werden. Sie vereinfachen den Umgang mit Matrixberechnungen und bieten außerdem noch andere Vorteile gegenüber den zuvor beschriebenen Matrixberechnungen. Zunächst ein kleines Einführungsbeispiel, um den Unterschied zwischen den bisherigen Matrixberechnungen und den dynamischen Arrays zu verdeutlichen. In Abbildung 11 sehen Sie in Spalte **A** die fünf Ziffern **1** bis **5**. In Spalte **B** sollen nun die Quadratwurzeln dieser 5 Ziffern mit der Excel-Funktion **WURZEL** in Form einer Matrixberechnung ermittelt werden:

1. Markieren Sie den Zellbereich **B1** bis **B5**.
2. Geben Sie folgende Formel ein:

**=WURZEL(A1:A5)**

3. Bestätigen Sie die Eingabe mit der Tastenkombination .


	A	B	C	D
1	1	1		
2	2	1,414214		
3	3	1,732051		
4	4	2		
5	5	2,236068		

Abb. 11: Die Ermittlung der Quadratwurzel als Matrixberechnung


Wie Sie am Inhalt der Bearbeitungsleiste erkennen können, ist die Formel in geschweifte Klammern eingeschlossen und damit als Matrixberechnung erkennbar. Soweit also nichts Besonderes. Jetzt wird das Ganze in der Spalte **C** wiederholt, aber diesmal als dynamisches Array:

1. Setzen Sie den Auswahlrahmen auf die Tabellenzelle **C1**. Sie wählen also nur eine Tabellenzelle und markieren keinen Zellbereich.
2. Geben Sie folgende Formel ein:

**=WURZEL(A1:A5)**

3. Bestätigen Sie die Eingabe mit der Taste .

Excel berechnet jetzt nicht nur in der Tabellenzelle **C1** das Ergebnis für den Wert aus der Tabellenzelle **A1**, sondern füllt die die nächsten vier Tabellenzellen (**C2** bis **C5**) mit den entsprechenden Ergebnissen aus. Im Gegensatz zur Matrixberechnung bietet die dynamischen Arrays folgende Vorteile:

- Sie markieren nur eine Tabellenzelle, keinen Zellbereich
- Sie verwenden zur Bestätigung nur die Taste  und keine Tastenkombination.

<sup>5</sup> Das englische Wort *array* hat sehr viele Bedeutungen im Deutschen (u.a. Feld, Bereich, Ordnung). Im Computerbereich steht das Wort *array* für Datenfeld, Datenbereich oder auch Matrix.

Außerdem wird die Formel nicht in geschweifte Klammern eingeschlossen, so wie das bei den Matrixberechnungen der Fall ist. Aber woran können Sie nun optisch erkennen, dass Excel ein dynamisches Array für die Berechnung eingesetzt hat? Das können Sie an drei Merkmalen erkennen:

- Als Funktionsargument wird weiterhin der Zellbereich angegeben (in diesem Beispiel **A1:A5**).
- Wenn Sie die Tabellenzelle **C1** auswählen, wird der Zellinhalt in der Bearbeitungsleiste in der Farbe Schwarz angezeigt. Wählen Sie stattdessen eine der Tabellenzellen **C2** bis **C5** aus, wird die Formel in der Bearbeitungsleiste in der Farbe Grau angezeigt. Der Zellbereich, wo die Formel in der Bearbeitungsleiste Grau angezeigt wird, wird als Überlaufbereich bezeichnet. Im konkreten Beispiel ist das der Zellbereich **C2** bis **C5**.
- Wenn Sie eine Tabellenzelle mit einem Ergebniswert auswählen, sehen Sie um den gesamten Zellbereich einen dünnen blauen Rahmen (siehe Abbildung 12; hier ist der blaue Rahmen stärker dargestellt als in Wirklichkeit).

	A	B	C	D
1	1	1	1	
2	2	1,41421356	1,18920712	
3	3	1,73205081	1,31607401	
4	4	2	1,41421356	
5	5	2,23606798	1,49534878	

Abb. 12: Ein dynamisches Array erkennen Sie an dem blauen Rahmen

### Neue Funktionen für die dynamische Arrays

Im Zusammenhang mit den dynamischen Arrays hat Microsoft auch einige neue Funktionen der Liste der verfügbaren Funktionen hinzugefügt:


Funktion	Beschreibung
<b>EINDEUTIG</b>	Gibt die eindeutigen Werte eines Bereichs oder Arrays zurück. <b>EINDEUTIG(Matrix;[Nach_Spalte];[genau_einmal])</b>
<b>FILTER</b>	Filtert einen Bereich oder eine Matrix. <b>FILTER(Matrix;Einschließen;[Wenn_leer])</b>
<b>SEQUENZ</b>	Gibt eine Sequenz von Zahlen zurück. <b>SEQUENZ(Zeilen;[Spalten];[Anfang];[Schritt])</b>
<b>SORTIEREN</b>	Sortiert einen Bereich oder Matrix. <b>SORTIEREN(Matrix;[Sortierindex];[Sortierreihenfolge];[nach_Spalte])</b>
<b>SORTIERENNACH</b>	Sortiert einen Bereich oder eine Matrix, basierend auf den Werten in einem entsprechenden Bereich oder einer entsprechenden Matrix. <b>SORTIERENNACH(Matrix;Nach_Matrix1[Sortierreihenfolge1];[Nach_Matrix2];[Sortierreihenfolge2];...)</b>
<b>ZUFALLSMATRIX</b>	Gibt eine Matrix von Zufallszahlen zurück. <b>ZUFALLSMATRIX([Zeilen];[Spalten];[min];[max];[ganze_Zahl])</b>

Anhand von ein paar Beispielen soll nun die eine oder andere Funktion näher gezeigt werden. Als Grundlage wird die Datei **Autohändler.xlsx** verwendet, die auch u.a. im Skript **Excel für Microsoft 365 – Pivot-Tabellen** verwendet wird. Abbildung 13 zeigt einen kleinen Ausschnitt der Liste, die insgesamt 3.000 Datensätze enthält.

	A	B	C	D	E	F
1	<b>Verkaufsdatum</b>	<b>Verkäufer</b>	<b>Filiale</b>	<b>Autotyp</b>	<b>Farben</b>	<b>Gewinn</b>
2	15.11.2018	Krause	Gießen	Kleinwagen	Rot	547,99 €
3	16.10.2019	Müller	Gießen	Kleinwagen	Silber	366,86 €
4	17.12.2019	Schmidt	Gießen	Kompaktklasse	Rot	213,75 €
5	25.10.2018	Westermann	Marburg	Cabrio	Rot	173,05 €
6	06.06.2018	Becker	Marburg	Cabrio	Schwarz	1.625,62 €
7	24.10.2019	Schulte	Marburg	Mittelklasse	Silber	1.431,99 €
8	19.12.2018	Becker	Marburg	Mittelklasse	Grün	1.219,22 €
9	19.11.2019	Müller	Gießen	Kompaktklasse	Silber	837,93 €
10	09.03.2018	Müller	Gießen	Kompaktklasse	Blau	811,59 €
11	13.03.2018	Schmidt	Gießen	Kompaktklasse	Schwarz	767,78 €
12	16.03.2018	Ziegler	Gießen	Mittelklasse	Anthrazit	765,14 €
13	22.05.2018	Schmidt	Gießen	Geländewagen	Grün	971,02 €
14	25.07.2019	Müller	Gießen	Mittelklasse	Blau	1.409,37 €
15	25.12.2019	Schmidt	Gießen	Geländewagen	Blau	893,79 €
16	07.01.2019	Westermann	Marburg	Cabrio	Grün	1.282,57 €
17	02.12.2019	Neumann	Marburg	Kleinwagen	Schwarz	731,25 €
18	08.05.2018	Westermann	Marburg	Kompaktklasse	Anthrazit	725,03 €
19	28.05.2019	Schulte	Marburg	Kompaktklasse	Schwarz	546,72 €
20	21.01.2019	Schmidt	Gießen	Mittelklasse	Rot	1.505,94 €

Abb. 13: Ausschnitt aus der Tabelle **Autohändler.xlsx**

Um die neuen Funktionen im Zusammenhang mit den dynamischen Arrays besser einsetzen zu können, ist es empfehlenswert, wenn Sie die die Liste zunächst in eine Tabelle umwandeln und der Tabelle dann auch noch einen Namen geben (siehe Skript **Excel für Microsoft 365 – Tabellen**). In diesem Beispiel bekommt die Tabelle den Namen **tblAutoverkauf**.

Im **ersten** Beispiel sollen die Namen der Verkäuferinnen bzw. Verkäufer aus der Tabelle extrahiert werden, um zu sehen, wie viele Verkäuferinnen bzw. Verkäufer es gibt und wie deren Namen lauten. Dazu wählen Sie zunächst eine leere Tabellenzelle, z.B. die Tabellenzelle **H2** (in der Tabellenzelle **H1** ist die Überschrift **Verkäufer** eingetragen<sup>6</sup>), geben folgende Formel ein und bestätigen die Eingabe mit der Taste :

**=EINDEUTIG(tblAutoverkauf[Verkäufer])**

Ab der Tabellenzelle **H2** füllt Excel nun die darunterliegenden Tabellenzellen mit den Namen der Verkäuferinnen bzw. Verkäufer. In Abbildung 14, Seite 15, sehen Sie das Ergebnis. Wenn Sie jetzt der Tabelle neue Datensätze hinzufügen, wobei es auch einen neuen Verkäufersnamen gibt, wird dieser neue Name in dem Zellbereich mit dem dynamischen Array automatisch hinzugefügt.

<sup>6</sup> Ob Sie in eine Tabellenzelle eine Überschrift eintragen, ist ganz Ihnen überlassen. Sie können die Überschrift auch weglassen und in diesem konkreten Beispiel das dynamische Array ab der Tabellenzelle **H1** beginnen.



	A	B	C	D	E	F	G	H
1	Verkaufsdatum	Verkäufer	Filiale	Autotyp	Farben	Gewinn		Verkäufer
2	15.11.2018	Krause	Gießen	Kleinwagen	Rot	547,99 €		Krause
3	16.10.2019	Müller	Gießen	Kleinwagen	Silber	366,86 €		Müller
4	17.12.2019	Schmidt	Gießen	Kompaktklasse	Rot	213,75 €		Schmidt
5	25.10.2018	Westermann	Marburg	Cabrio	Rot	173,05 €		Westermann
6	06.06.2018	Becker	Marburg	Cabrio	Schwarz	1.625,62 €		Becker
7	24.10.2019	Schulte	Marburg	Mitteklasse	Silber	1.431,99 €		Schulte
8	19.12.2018	Becker	Marburg	Mitteklasse	Grün	1.219,22 €		Ziegler
9	19.11.2019	Müller	Gießen	Kompaktklasse	Silber	837,93 €		Neumann
10	09.03.2018	Müller	Gießen	Kompaktklasse	Blau	811,59 €		
11	13.03.2018	Schmidt	Gießen	Kompaktklasse	Schwarz	767,78 €		
12	16.03.2018	Ziegler	Gießen	Mitteklasse	Anthrazit	765,14 €		
13	22.05.2018	Schmidt	Gießen	Geländewagen	Grün	971,02 €		
14	25.07.2019	Müller	Gießen	Mitteklasse	Blau	1.409,37 €		
15	25.12.2019	Schmidt	Gießen	Geländewagen	Blau	893,79 €		
16	07.01.2019	Westermann	Marburg	Cabrio	Grün	1.282,57 €		
17	02.12.2019	Neumann	Marburg	Kleinwagen	Schwarz	731,25 €		
18	08.05.2018	Westermann	Marburg	Kompaktklasse	Anthrazit	725,03 €		
19	28.05.2019	Schulte	Marburg	Kompaktklasse	Schwarz	546,72 €		
20	21.01.2019	Schmidt	Gießen	Mitteklasse	Rot	1.505,94 €		

Abb. 14: Das dynamische Array mit den eindeutigen Verkäufernamen

Im **zweiten** Beispiel wird das erste Beispiel dahingehend geändert, dass neben den Verkäufernamen auch noch die Filiale und der Autotyp angezeigt wird. Auch in diesem Beispiel geben Sie die Formel in die Tabellenzelle **H2** ein:

**=EINDEUTIG(tblAutoverkauf[Verkäufer]:[Autotyp])**

Bitte wundern Sie jetzt nicht, dass sowohl die Verkäufernamen als auch die Filialen und die Autotypen mehrmals als Ergebnis aufgelistet werden. Das liegt daran, dass es unterschiedliche Kombinationen aus Verkäufername, Filiale und Autotyp geben kann. Wenn Sie aber genau hinschauen, werden Sie feststellen, dass keine der Kombinationen doppelt oder mehrfach vorkommt.

Im **dritten** Beispiel sollen in der Spalte **H** (ab Tabellenzelle **H2**) die Verkäufernamen und in Spalte **I** (ab Tabellenzelle **I2**) die Autotypen eindeutig aufgelistet werden. In diesem Beispiel müssen Sie zwei getrennte Formeln eingeben. In die Tabellenzelle **H2** geben Sie die Formel aus dem ersten Beispiel ein:

**=EINDEUTIG(tblAutoverkauf[Verkäufer])**

In die Tabellenzelle **I2** geben Sie ein (und bestätigen die Eingabe):

**=EINDEUTIG(tblAutoverkauf[Autotyp])**

Wenn Sie sich die Ergebnisse der drei Beispiele genauer anschauen, werden Sie feststellen, dass die Ergebnisse unsortiert sind. Deshalb soll im **vierten** Beispiel das erste Beispiel noch mal hergenommen und die Ergebnisse dabei auch sortiert werden. Zur Lösung der Aufgabe können Sie die Funktion **SORTIEREN** einsetzen. Sie müssen allerdings auch die Funktion **EINDEUTIG** verwenden, da ansonsten jeder Verkäufernamen so oft wiederholt angezeigt wird, wie er in der Tabelle enthalten ist. Die Formel lautet:

**=SORTIEREN(EINDEUTIG(tblAutoverkauf[Verkäufer]))**

Im **fünften** Beispiel sollen nur die Datensätze angezeigt werden, bei den der Verkäufersname **Krause** ist. Auch für dieses Beispiel wird wieder die Tabellenzelle **H2** ausgewählt. Die Formel lautet:

```
=FILTER(tblAutoverkauf;tblAutoverkauf[Verkäufer]="Krause")
```

In diesem Beispiel werden jetzt nicht nur die Daten aus der Spalte *Verkäufer* im Ergebnis angezeigt, sondern auch von allen anderen Spalten. Die Ergebnismatrix besitzt also sechs Spalten.

**Anmerkung:** In der ersten Ergebnisspalte werden anstelle von Datumswerten nur ganze Zahlen angezeigt. Sie können die Werte in dieser Spalte aber markieren und ihnen die passende Zahlenformatierung zuweisen. Das gilt auch für die letzte Spalte, wo Sie zwar Zahlenwerte sehen, aber nicht als Währung formatiert. Weitere Informationen siehe Skript **Excel für Microsoft 365 – Tabellenzellen formatieren**.

Als Ergebnis bekommen Sie sehr viele Daten angezeigt, insgesamt 350 Datensätze mit dem Verkäufer **Krause**. Sie möchten die Filterung erweitern und neben dem Verkäufersnamen **Krause** auch noch (z.B.) nach der Wagenfarbe **Rot** filtern. Die Formel in der Tabellenzelle **H2** wird folgendermaßen ergänzt:

```
=FILTER(tblAutoverkauf;(tblAutoverkauf[Verkäufer]="Krause")*(tblAutoverkauf[Farben]="Rot"))
```

Das Sternchen (\*) in der Formel bedeutet jetzt nicht, dass eine Multiplikation durchgeführt wird, sondern hat in diesem Fall die Bedeutung der logischen Verknüpfung **UND**. Das heißt, nur die Datensätze werden angezeigt, bei denen der Verkäufersname **Krause** und die Wagenfarbe **Rot** ist. In diesem Beispiel reduziert sich die Anzahl der Datensätze von 350 auf 61.

Bei dem Beispiel mit der Funktion **FILTER** wurde die Filterung konkret auf den Verkäufersnamen **Krause** und der Wagenfarbe **Rot** durchgeführt. Sie können die Filterung aber auch allgemein formulieren. Dazu tragen Sie die gewünschten Daten in Tabellenzellen ein und verwenden die Namen dieser Tabellenzellen in der Formel. Als Beispiel wird in die Tabellenzelle **P2** der Name der Verkäuferin bzw. des Verkäufers eingetragen und in die Tabellenzelle **P3** die Farbe des Wagens. Die Formel in der Tabellenzelle **H2** wird nun folgendermaßen geändert:

```
=FILTER(tblAutoverkauf;(tblAutoverkauf[Verkäufer]=P2)*(tblAutoverkauf[Farben]=P3))
```


Jetzt tragen Sie in die beiden Tabellenzellen **P2** und **P3** einfach verschiedene Namen und Farben ein und erhalten in der Tabellenzelle **H2** und im Überlaufbereich die neuen Ergebnisse. Sie müssen allerdings aufpassen, dass Sie in die Tabellenzellen **P2** bzw. **P3** einen Verkäufersnamen bzw. eine Wagenfarbe eintragen, die es in der Tabelle auch tatsächlich gibt. Ansonsten bekommen Sie in der Tabellenzelle **H2** den Fehlerwert **#KALK!** angezeigt.

Sie können die Funktion **FILTER** auch einsetzen, wenn Sie nur bestimmte Spalten in der gefilterten Liste angezeigt bekommen möchten. Beispielsweise sollen in der gefilterten Liste nur die Namen der *Verkäufer* und die *Gewinne* angezeigt werden. Die gefilterten Daten werden rechts neben der Tabelle eingefügt, ähnlich wie beim Spezialfilter bei den Datenbanken (siehe Skript **Excel für Microsoft 365 – Datenbanken**, Kapitel **Der Spezialfilter**, Seite 22). Als Beispiel sollen die Gewinne für alle Verkäufer in Bezug auf den Autotyp (z.B. *Kleinwagen*) angezeigt werden. Zunächst wird der gesuchte Autotyp in eine Tabellenzelle eingetragen (in diesem Beispiel in die Tabellenzelle **I2**). In die Tabellenzellen **H4** und **I4** stehen die Namen **Verkäufer** und **Gewinn** (die Namen müssen nicht unbedingt mit den Namen in der Tabelle übereinstimmen). In die Tabellenzelle **H5** tragen Sie jetzt folgende

Formel ein:


```
JUSTUS-LIEBIG-  
UNIVERSITÄT  
GIESSEN
```

**=FILTER(tblAutoverkauf[[Verkaufsdatum]:[Gewinn]];tblAutoverkauf[Autotyp]=I2)**

Nachdem Sie die Eingabe der Formel mit der Eingabetaste () bestätigt haben, wird die Filterung durchgeführt. Allerdings bekommen Sie alle Spalten angezeigt, die auch die Ausgangstabelle enthält (siehe Abbildung 15; lassen Sie sich nicht an den fehlenden Formatierungen stören, dazu später mehr).




	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<b>Gewinne beim Autoverkauf</b>												
2								Autotyp	Kleinwagen				
3													
4	Verkaufsdatum	Verkäufer	Filiale	Autotyp	Farben	Gewinn		Verkäufer	Gewinn				
5	15.11.2018	Krause	Gießen	Kleinwagen	Rot	547,99 €		43419 Krause	Gießen	Kleinwagen	Rot		547,992705
6	16.10.2019	Müller	Gießen	Kleinwagen	Silber	366,86 €		43754 Müller	Gießen	Kleinwagen	Silber		366,863027
7	17.12.2019	Schmidt	Gießen	Kompaktklasse	Rot	213,75 €		43801 Neumann	Marburg	Kleinwagen	Schwarz		731,246234
8	25.10.2018	Westermann	Marburg	Cabrio	Rot	173,05 €		43735 Ziegler	Gießen	Kleinwagen	Schwarz		759,535551
9	06.06.2018	Becker	Marburg	Cabrio	Schwarz	1.625,62 €		43136 Schulte	Marburg	Kleinwagen	Grün		128,011365
10	24.10.2019	Schulte	Marburg	Mittekkklasse	Silber	1.431,99 €		43188 Krause	Gießen	Kleinwagen	Anthrazit		370,184002
11	19.12.2018	Becker	Marburg	Mittekkklasse	Grün	1.219,22 €		43313 Becker	Marburg	Kleinwagen	Anthrazit		550,19336
12	19.11.2019	Müller	Gießen	Kompaktklasse	Silber	837,93 €		43272 Müller	Gießen	Kleinwagen	Grün		199,363426
13	09.03.2018	Müller	Gießen	Kompaktklasse	Blau	811,59 €		43263 Ziegler	Gießen	Kleinwagen	Schwarz		457,426475
14	13.03.2018	Schmidt	Gießen	Kompaktklasse	Schwarz	767,78 €		43285 Westermann	Marburg	Kleinwagen	Grün		468,179288
15	16.03.2018	Ziegler	Gießen	Mittekkklasse	Anthrazit	765,14 €		43279 Becker	Marburg	Kleinwagen	Silber		417,273967
16	22.05.2018	Schmidt	Gießen	Geländewagen	Grün	971,02 €		43584 Neumann	Marburg	Kleinwagen	Anthrazit		473,928526
17	25.07.2019	Müller	Gießen	Mittekkklasse	Blau	1.409,37 €		43327 Müller	Gießen	Kleinwagen	Grün		182,41133
18	25.12.2019	Schmidt	Gießen	Geländewagen	Blau	893,79 €		43640 Westermann	Marburg	Kleinwagen	Rot		189,595895
19	07.01.2019	Westermann	Marburg	Cabrio	Grün	1.282,57 €		43405 Ziegler	Gießen	Kleinwagen	Grün		541,199893
20	02.12.2019	Neumann	Marburg	Kleinwagen	Schwarz	731,25 €		43703 Krause	Gießen	Kleinwagen	Blau		324,593279
21	08.05.2018	Westermann	Marburg	Kompaktklasse	Anthrazit	725,03 €		43496 Müller	Gießen	Kleinwagen	Blau		799,48378
22	28.05.2019	Schulte	Marburg	Kompaktklasse	Schwarz	546,72 €		43614 Schulte	Marburg	Kleinwagen	Blau		582,855991
23	21.01.2019	Schmidt	Gießen	Mittekkklasse	Rot	1.505,94 €		43532 Becker	Marburg	Kleinwagen	Silber		470,688586
24	27.09.2019	Ziegler	Gießen	Kleinwagen	Schwarz	759,54 €		43126 Schulte	Marburg	Kleinwagen	Rot		109,715672
25	06.04.2018	Schulte	Marburg	Cabrio	Blau	1.499,22 €		43775 Schmidt	Gießen	Kleinwagen	Schwarz		274,453818

Abb. 15: Die gefilterten Daten (Ausschnitt)

Es sollen aber nur die Namen der Verkäufer und die zugehörigen Gewinne angezeigt werden. Zunächst löschen Sie das dynamische Array (einfach erste Tabellenzelle des dynamischen Arrays auswählen – in diesem Beispiel ist das die Tabellenzelle **H5** – und dann mit der Taste ) löschen). Belassen Sie den Auswahlrahmen auf der Tabellenzelle **H5** und geben folgende Formel ein:

**=FILTER(FILTER(tblAutoverkauf[[Verkaufsdatum]:[Gewinn]];tblAutoverkauf[Autotyp]=I2);{0.1.0.0.0.1})**

Nach Bestätigung der Formeleingabe sehen Sie jetzt nur noch die Verkäufernamen und die zugehörigen Gewinne (siehe Abbildung 16, Seite 18). Aber wieso werden jetzt nur noch die Daten dieser beiden Spalten angezeigt? Die Antwort auf diese Frage liegt darin begründet, dass eigentlich zwei Filterungen durchgeführt worden sind, also die Funktion **FILTER** ineinander verschachtelt wurden. Die innere Angabe entspricht der ersten Formel, bei der am Ende alle Spalten angezeigt worden sind. Bei der äußeren Funktion **FILTER** wurde noch eine Matrixkonstante (siehe Kapitel **Matrixkonstante**, Seite 6) hinzugefügt (**{0.1.0.0.0.1}**). Durch die Angabe **0** bzw. **1** legen Sie fest, welche Spalte bei der Filterung angezeigt werden soll und welche nicht. Mit der **0** wird festgelegt, dass die entsprechende Spalte nicht angezeigt wird, bei der **1** dagegen schon. In diesem Beispiel wird die 2. und 6. Spalte (also die *Verkäufer* und die *Gewinne*) angezeigt, die anderen vier Spalten aber nicht. Die Nullen und Einsen sind jeweils durch Punkte getrennt, da die Angaben *Verkaufsdatum*, *Verkäufer*, *Filiale*, *Autotyp*, *Farben* und *Gewinn* in einer Zeile nebeneinander stehen.

Jetzt fehlt nur noch die Formatierung (insbesondere das Zahlenformat in der Spalte *Gewinn*). Dazu wählen Sie zunächst die Tabellenzelle **I5**. Drücken Sie nun die Tastenkombination    und alle Tabellenzellen mit Zahlen in der Spalte **I** werden markiert. Jetzt können Sie das gewünschte Zahlenformat festlegen (siehe Skript **Excel für Microsoft 365 – Tabellenzellen formatieren**).

	A	B	C	D	E	F	G	H	I
1	<b>Gewinne beim Autoverkauf</b>								
2								Autotyp	Kleinwagen
3									
4	Verkaufsdatum	Verkäufer	Filiale	Autotyp	Farben	Gewinn		Verkäufer	Gewinn
5	15.11.2018	Krause	Gießen	Kleinwagen	Rot	547,99 €		Krause	547,9927054
6	16.10.2019	Müller	Gießen	Kleinwagen	Silber	366,86 €		Müller	366,8630272
7	17.12.2019	Schmidt	Gießen	Kompaktklasse	Rot	213,75 €		Neumann	731,2462339
8	25.10.2018	Westermann	Marburg	Cabrio	Rot	173,05 €		Ziegler	759,5355512
9	06.06.2018	Becker	Marburg	Cabrio	Schwarz	1.625,62 €		Schulte	128,0113646
10	24.10.2019	Schulte	Marburg	Mittekkilasse	Silber	1.431,99 €		Krause	370,1840025
11	19.12.2018	Becker	Marburg	Mittekkilasse	Grün	1.219,22 €		Becker	550,193358
12	19.11.2019	Müller	Gießen	Kompaktklasse	Silber	837,93 €		Müller	199,3634259
13	09.03.2018	Müller	Gießen	Kompaktklasse	Blau	811,59 €		Ziegler	457,4264746
14	13.03.2018	Schmidt	Gießen	Kompaktklasse	Schwarz	767,78 €		Westermann	468,1792881
15	16.03.2018	Ziegler	Gießen	Mittekkilasse	Anthrazit	765,14 €		Becker	417,273967
16	22.05.2018	Schmidt	Gießen	Geländewagen	Grün	971,02 €		Neumann	473,9285262
17	25.07.2019	Müller	Gießen	Mittekkilasse	Blau	1.409,37 €		Müller	182,41133
18	25.12.2019	Schmidt	Gießen	Geländewagen	Blau	893,79 €		Westermann	189,5958953
19	07.01.2019	Westermann	Marburg	Cabrio	Grün	1.282,57 €		Ziegler	541,1998931
20	02.12.2019	Neumann	Marburg	Kleinwagen	Schwarz	731,25 €		Krause	324,5932788
21	08.05.2018	Westermann	Marburg	Kompaktklasse	Anthrazit	725,03 €		Müller	799,483782
22	28.05.2019	Schulte	Marburg	Kompaktklasse	Schwarz	546,72 €		Schulte	582,8559911
23	21.01.2019	Schmidt	Gießen	Mittekkilasse	Rot	1.505,94 €		Becker	470,6885858
24	27.09.2019	Ziegler	Gießen	Kleinwagen	Schwarz	759,54 €		Schulte	109,7156724
25	06.04.2018	Schulte	Marburg	Cabrio	Blau	1.499,22 €		Schmidt	274,4538184

Abb. 16: Die gefilterten Daten nur mit Verkäufer und Gewinn (Ausschnitt)

Nun wird das Ganze noch ergänzt durch eine Sortierung nach dem Verkäufernamen. Die Formel, die Sie in die Tabellenzelle **H5** eingeben hat folgendes Aussehen (lassen Sie sich bitte nicht an dem Zeilenumbrech stören):

```
=SORTIEREN(FILTER(FILTER(tblAutoverkauf[[Verkaufsdatum]:[Gewinn]]);  
tblAutoverkauf[Autotyp]=I2);{0.1.0.0.1});1;1)
```

Die beiden letzten Angaben bedeuten, dass die gefilterte Liste nach der ersten Spalte (also nach dem Verkäufernamen) aufsteigend (das ist die zweite 1; bei absteigend müssten Sie -1 angeben) sortiert wird.

Im sechsten Beispiel wird erneut eine Filterung durchgeführt, diesmal aber sowohl nach den Zeilen als auch nach den Spalten. Es sollen z.B. nur die Gewinne des Verkäufers **Becker** aufgelistet werden. Für dieses Beispiel werden die dynamischen Array-Funktionen **INDEX**, **FILTER**, **SEQUENZ** und zusätzlich die Funktion **ZÄHLENWENN** benötigt. Mit der Funktion **FILTER** erstellen Sie eine Matrix, die nur die Daten eines bestimmten Kriteriums enthält, in diesem Beispiel der Name des Verkäufers (**Becker**). Mit der Funktion **INDEX** wird dann nicht nur eine einzelne Tabellenzelle aus der Liste ermittelt, sondern Sie können auch verschiedene Zeilen und Spalten aus der Liste extrahieren. Für die Ermittlung der Zeilen benötigen Sie eine Zahlenmatrix, die von 1 bis zur Anzahl der gefilterten Zeilen geht. Das machen Sie mit der Funktion **SEQUENZ**. Für die Spalten besteht die Zahlenmatrix aus den Nummern der entsprechenden Spalten. Sollen nur die Daten aus den Spalten **Verkäufer** (Spalte 2) und **Gewinn** (Spalte 6) ermittelt werden, geben Sie die konstante Matrix **{2.6}** an. Für die genaue Anzahl der gefilterten Zeilen wird noch die Funktion **ZÄHLENWENN** verwendet, die als erstes Argument für die Funktion **SEQUENZ** dient. Wenn jetzt nur die Gewinne des Verkäufers **Becker** ermittelt werden sollen, müssen Sie noch in eine Tabellenzelle das Filterkriterium (in diesem Beispiel der Name des Verkäufers) eintragen. Als Beispiel wird der Name **Becker** in die Tabellenzelle **H5** eingetragen. Dann wird z.B. die Formel in die Tabellenzelle **J5** eingetragen:

```
=INDEX(FILTER(tblAutoverkauf;tblAutoverkauf[Verkäufer]=H5;"");  
SEQUENZ(ZÄHLENWENN(tblAutoverkauf[Verkäufer];H5));{2.6})
```

Sie müssen dann noch die Spalte mit den Gewinnen als Währung formatieren. Das Ergebnis (Ausschnitt) sehen Sie in Abbildung 17, Seite 19.

	A	B	C	D	E	F	G	H	I	J	K
1	Gewinne beim Autoverkauf										
2											
3											
4	Verkaufsdatum	Verkäufer	Filiale	Autotyp	Farben	Gewinn		Verkäufer	Verkäufer	Gewinn	
5	15.11.2018	Krause	Gießen	Kleinwagen	Rot	547,99 €		Becker	Becker	1.625,62 €	
6	16.10.2019	Müller	Gießen	Kleinwagen	Silber	366,86 €		Becker	Becker	1.219,22 €	
7	17.12.2019	Schmidt	Gießen	Kompaktklasse	Rot	213,75 €		Becker	Becker	1.063,93 €	
8	25.10.2018	Westermann	Marburg	Cabrio	Rot	173,05 €		Becker	Becker	165,38 €	
9	06.06.2018	Becker	Marburg	Cabrio	Schwarz	1.625,62 €		Becker	Becker	970,73 €	
10	24.10.2019	Schulte	Marburg	Mittekkklasse	Silber	1.431,99 €		Becker	Becker	550,19 €	
11	19.12.2018	Becker	Marburg	Mittekkklasse	Grün	1.219,22 €		Becker	Becker	599,71 €	
12	19.11.2019	Müller	Gießen	Kompaktklasse	Silber	837,93 €		Becker	Becker	417,27 €	
13	09.03.2018	Müller	Gießen	Kompaktklasse	Blau	811,59 €		Becker	Becker	1.494,92 €	
14	13.03.2018	Schmidt	Gießen	Kompaktklasse	Schwarz	767,78 €		Becker	Becker	1.763,84 €	
15	16.03.2018	Ziegler	Gießen	Mittekkklasse	Anthrazit	765,14 €		Becker	Becker	183,35 €	
16	22.05.2018	Schmidt	Gießen	Geländewagen	Grün	971,02 €		Becker	Becker	789,13 €	
17	25.07.2019	Müller	Gießen	Mittekkklasse	Blau	1.409,37 €		Becker	Becker	470,69 €	
18	25.12.2019	Schmidt	Gießen	Geländewagen	Blau	893,79 €		Becker	Becker	592,02 €	
19	07.01.2019	Westermann	Marburg	Cabrio	Grün	1.282,57 €		Becker	Becker	1.790,49 €	
20	02.12.2019	Neumann	Marburg	Kleinwagen	Schwarz	731,25 €		Becker	Becker	220,12 €	
21	08.05.2018	Westermann	Marburg	Kompaktklasse	Anthrazit	725,03 €		Becker	Becker	1.784,48 €	

Abb. 17: Die gefilterten Gewinne des Verkäufers Becker

Sie können jetzt in die Tabellenzelle **H5** den Namen eines anderen Verkäufers eintragen und bekommen nur die Gewinne dieses Verkäufers ab Tabellenzelle **J5** angezeigt.

Im **siebten** Beispiel soll mit Hilfe der Funktion **SEQUENZ** eine Zahlenreihe im Zellbereich **A1** bis **E5** (z.B. auf einem neuen leeren Arbeitsblatt) erstellt werden, wobei die Reihe mit dem Wert **5** beginnt und der Betrag für die schrittweise Erhöhung der Reihe **10** beträgt. Tragen Sie in die Tabellenzelle **A1** die folgende Formel ein und bestätigen die Eingabe (Ergebnis siehe Abbildung 18):

**=SEQUENZ(4;5;5;10)**

	A	B	C	D	E	F
1	5	15	25	35	45	
2	55	65	75	85	95	
3	105	115	125	135	145	
4	155	165	175	185	195	
5						

Abb. 18: Beispiel für die Funktion **SEQUENZ**

Im **achten** Beispiel schließlich wird ein Zellbereich (z.B. **A1** bis **D5**) mit Zufallszahlen erzeugt. Dabei sollen 20 ganze Zahlen im Bereich von **1** und **100** erzeugt werden. Nachdem Sie die Tabellenzelle **A1** ausgewählt haben, geben Sie folgende Formel ein und bestätigen die Eingabe:

**=ZUFALLSMATRIX(5;4;1;100;WAHR)**

Mit der Angabe **WAHR** als fünftes Funktionsargument legen Sie fest, dass die Funktion **ZUFALLSMATRIX** nur ganze Zahlen als Ergebnis anzeigt. Im anderen Fall werden Zahlen mit Dezimalstellen erzeugt. Ergebnis siehe Abbildung 19.

	A	B	C	D	E
1	45	48	43	29	
2	3	84	68	90	
3	2	72	83	69	
4	67	1	73	97	
5	63	89	86	14	
6					

Abb. 19: Beispiel für die Funktion **ZUFALLSMATRIX**

**Anmerkung:** Bei den hier gezeigten Beispielen handelt es sich um vergleichsweise einfache Beispiele. Wenn Sie weitere Beispiele und sich auch den genauen Syntaxaufbau der einzelnen Funktionen genauer anschauen möchten, sollten Sie sich die Hilfeseiten zu den Funktionen anzeigen lassen. Außerdem gibt es noch eine ganze Reihe an Webseiten im Internet, wo Sie weitere Informationen zum Thema *dynamische Arrays* erhalten. Sie müssen nur den entsprechenden Suchbegriff in eine Suchmaschine im Internet eingeben.

Im **neunten** Beispiel soll eine Messreihe sortiert werden, wobei die sortierten Daten an einem anderen Teil des Arbeitsblatts eingefügt werden sollen. Sie haben also am Ende zwei Zellbereiche: einer ohne Sortierung und einer mit Sortierung. Bei der Sortierung sollen nicht die Werte in einer Spalte auf- oder absteigend sortiert werden (so wie das sonst üblich ist), sondern in den Zeilen. Dazu ein konkretes Beispiel. Wie bereits erwähnt, soll eine eingegebene Messreihe sortiert werden. In der Spalte **A** stehen Datumswerte, wann die Messungen durchgeführt worden sind. In den Spalten **B** bis **H** stehen die erfassten Messwerte. Dabei wurden an jedem der 10 Termine insgesamt 7 Messungen durchgeführt. Nun sollen die Messungen für jedes Datum (also jede Zeile) aufsteigend (von links nach rechts) sortiert werden. Dabei soll zum Vergleich die unsortierte Liste erhalten bleiben, damit sie mit der sortierten Liste verglichen werden kann. Abbildung 20 zeigt die Ausgangssituation (die Datumsangaben stehen in diesem Beispiel im Zellbereich **A4:A13** und die Messwerte im Zellbereich **B4:H13**).

	A	B	C	D	E	F	G	H
1	<b>Messwerte für eine wissenschaftliche Studie</b>							
2								
3	<b>Datum</b>	<b>Messwert 1</b>	<b>Messwert 2</b>	<b>Messwert 3</b>	<b>Messwert 4</b>	<b>Messwert 5</b>	<b>Messwert 6</b>	<b>Messwert 7</b>
4	05.06.2023	22,06	65,91	38,55	59,37	28,58	31,01	66,78
5	12.06.2023	92,74	11,99	17,47	91,55	89,19	4,69	70,81
6	19.06.2023	20,66	6,64	17,54	28,39	23,06	42,36	81,08
7	26.06.2023	83,55	14,69	21,05	27,30	26,07	36,11	19,69
8	03.07.2023	7,50	73,59	32,47	24,28	98,55	19,81	26,28
9	10.07.2023	81,89	11,04	43,48	79,64	79,72	72,11	34,44
10	17.07.2023	35,53	17,76	40,96	28,41	50,60	26,87	36,18
11	24.07.2023	70,87	67,49	93,21	92,37	40,31	24,66	25,02
12	31.07.2023	52,39	76,87	1,31	23,75	20,64	46,05	10,96
13	07.08.2023	82,87	32,78	9,57	68,24	23,74	16,24	14,36

Abb. 20: Liste mit den unsortierten Messwerten

Nun werden zunächst die Überschriftzeile und die Datumsangaben in einen anderen Zellbereich kopiert (in diesem Beispiel werden die Datumsangaben in den Zellbereich **A17:A26** kopiert und die Überschriftzeile in den Zellbereich **A16:H16**). Als nächstes wählen Sie die Tabellenzelle **B17**, tragen folgende Formel ein und bestätigen die Eingabe:

**=SORTIEREN(B4:H4;;1;WAHR)**

Beim ersten Funktionsargument handelt es sich um den Zellbereich, dessen Inhalt sortiert werden soll. Das ist in diesem Beispiel die erste durchgeführte Messreihe. Das zweite Funktionsargument ist der Sortierindex, mit dem Sie die Zeile bzw. Spalte angeben, nach der sortiert werden soll. Diese Angabe ist bei diesem Beispiel nicht nötig, daher kann dieses Funktionsargument weggelassen werden. Mit dem dritten Funktionsargument geben Sie an, ob die Sortierung aufsteigend erfolgen soll (**1**) oder absteigend (**-1**). In diesem Beispiel ist die Sortierung aufsteigend. Und mit dem vierten Funktionsargument geben Sie die Sortierrichtung an. Die Angabe **WAHR** steht für eine Spaltensortierung, die Angabe **FALSCH** für eine Zeilensortierung. In diesem Beispiel werden die Werte in den Spalten sortiert. Üblicherweise erfolgt eine Sortierung nach Zeilen. Sobald Sie die Eingabe bestätigt haben,

sehen Sie die sortierte Messreihe für das erste Datum. Markieren Sie die Tabellenzelle **B17** und kopieren die Formel mit dem Verfahren *Automatisches Ausfüllen* (siehe Skript **Excel für Microsoft 365 – Automatisches Ausfüllen**) bis zur Tabellenzelle **B26**. Das Ergebnis sehen Sie in Abbildung 21.

	A	B	C	D	E	F	G	H
16	Datum	Messwert 1	Messwert 2	Messwert 3	Messwert 4	Messwert 5	Messwert 6	Messwert 7
17	05.06.2023	22,06	28,58	31,01	38,55	59,37	65,91	66,78
18	12.06.2023	4,69	11,99	17,47	70,81	89,19	91,55	92,74
19	19.06.2023	6,64	17,54	20,66	23,06	28,39	42,36	81,08
20	26.06.2023	14,69	19,69	21,05	26,07	27,30	36,11	83,55
21	03.07.2023	7,50	19,81	24,28	26,28	32,47	73,59	98,55
22	10.07.2023	11,04	34,44	43,48	72,11	79,64	79,72	81,89
23	17.07.2023	17,76	26,87	28,41	35,53	36,18	40,96	50,60
24	24.07.2023	24,66	25,02	40,31	67,49	70,87	92,37	93,21
25	31.07.2023	1,31	10,96	20,64	23,75	46,05	52,39	76,87
26	07.08.2023	9,57	14,36	16,24	23,74	32,78	68,24	82,87

Abb. 21: Liste mit den sortierten Messwerten

## Zusätzliche Informationen zu den dynamischen Arrays

Beim Einsatz von dynamischen Arrays müssen Sie ein paar Dinge beachten, da Excel ansonsten Fehlerwerte anstelle der eigentlichen Ergebnisse anzeigt. Hier eine entsprechende Auflistung:

- Wenn sich die Bezüge bei den Funktionsargumenten auf Tabellenzellen einer anderen Arbeitsmappe beziehen, muss diese andere Arbeitsmappe in Excel geöffnet sein, da sonst als Ergebnis der Fehlerwert **#BEZUG!** angezeigt wird.
- Sie können dynamische Arrays nicht in Tabellen einsetzen. Das bedeutet, wenn Sie eine Liste in eine Tabelle umgewandelt haben (siehe Skript **Excel für Microsoft 365 – Tabellen**) dürfen Sie keine dynamischen Arrays innerhalb dieser Tabelle einsetzen. Ansonsten erhalten Sie den Fehlerwert **#ÜBERLAUF!**. Das gilt auch für die „normale“ Matrixberechnung.
- Wenn Sie den kompletten Zellbereich, der durch Verwendung des dynamischen Arrays erzeugt worden ist, verschieben wollen, reicht es aus, die Tabellenzelle auszuwählen, in die Sie die Formel eingegeben haben. Wenn Sie dann diese Tabellenzelle mit der Maus verschieben, wird der komplette Zellbereich mit den Ergebniswerten automatisch mitverschoben.
- Wie bereits erwähnt, handelt es sich bei den dynamischen Arrays um eine Neuerung von Excel für Microsoft 365. Außer in dieser Version, gibt es die dynamischen Arrays auch in Excel 2021. Wenn Sie eine Arbeitsmappe, in der dynamische Arrays enthalten sind, mit einer älteren Excel-Version öffnen (z.B. in Excel 2019), wandelt Excel die dynamischen Arrays in „normale“ Matrixberechnungen um. Werden in den Formeln die neuen Funktionen verwendet, wandelt Excel diese in spezielle integrierte Funktionen mit englischen Namen um. So wird z.B. aus der Funktion **SEQUENZ** die Funktion **\_xlfn.SEQUENCE**. Das passiert auch mit den anderen neuen Funktionen. Allerdings wird nicht immer auch das gewünschte Ergebnis angezeigt, sondern in manchen Fällen ein Excel-Fehlerwert. Gegebenenfalls müssen Sie ausprobieren, bei welcher älteren Excel-Version eine automatische Umwandlung durchgeführt wird und bei welcher Version nicht.

## Der Fehlerwert #ÜBERLAUF!

Da der Umgang mit den dynamischen Arrays neu und auch nicht ganz so einfach ist, kann es schnell passieren, dass Sie anstelle des erwarteten Ergebnisses den Fehlerwert **#ÜBERLAUF!** angezeigt bekommen. Für die Anzeige dieses Fehlerwerts kann es unterschiedliche Gründe geben, die in der nachfolgenden Tabelle beschrieben werden.

<b>Grund</b>	<b>Beschreibung</b>
<b>Überlaufbereich ist nicht leer</b>	In eine oder mehrere Tabellenzellen des Überlaufbereichs wurde nachträglich manuell ein neuer Wert eingetragen. Der Überlaufbereich wird von Excel bei der Eingabe der Formel automatisch erzeugt. Sie dürfen keine Daten manuell in den Überlaufbereich eintragen.
<b>Unbestimmte Größe</b>	Die Größe des Überlaufbereichs konnte von Excel nicht ermittelt werden. Beispielsweise liefert die Formel <b>=SEQUENZ(ZUFALLSBEREICH(1;100))</b> den Fehlerwert <b>#ÜBERLAUF!</b> .
<b>Erstreckt sich über den Rand des Arbeitsblatts hinaus</b>	Beim Ergebnis des dynamischen Arrays ragt der Überlaufbereich über den Rand des Arbeitsblatts hinaus.
<b>Tabellenformel</b>	Sowohl für Matrixberechnungen als auch für dynamische Arrays gilt, dass sie nicht in Tabellen (siehe Skript <b>Excel für Microsoft 365 – Tabellen</b> ) eingesetzt werden dürfen. Sie können nur außerhalb von Tabellen verwendet werden oder die Tabelle muss in einen normalen Bereich konvertiert werden.
<b>Nicht genügend Arbeitsspeicher</b>	Es kann passieren, dass der Überlaufbereich so groß wird, dass die Informationen nicht mehr in den Arbeitsspeicher des PCs passen. Versuchen Sie die Formel so zu verändern, dass ein kleinerer Überlaufbereich erzeugt wird.
<b>Überlaufen in verbundene Tabellenzellen</b>	In dem Zellbereich, der für den Überlaufbereich benötigt wird, befinden sich verbundene Tabellenzellen. Heben Sie die Zusammenführung der betreffenden Tabellenzellen auf.
<b>Nicht erkannt/Fallback</b>	Der Fehlerwert wird angezeigt, wenn nicht alle erforderlichen Argumente bzw. Daten für die Berechnung des dynamischen Arrays angegeben worden sind.

## Einsatz von dynamischen Arrays umgehen

Auch wenn der Einsatz von dynamischen Arrays viele Vorteile mit sich bringt, so gibt es dennoch Situationen, bei denen die dynamischen Arrays keine große Hilfe sind und es in diesen Fällen besser wäre, auf sie zu verzichten. Wie Sie in solchen Fällen die dynamischen Arrays umgehen, soll an einem konkreten Beispiel gezeigt werden.



Für einen vorgegebenen Zinssatz, eine vorgegebene Laufzeit und einem vorgegebenen Betrag, der monatlich auf ein Konto eingezahlt wird, soll Excel für verschiedene einmalige Anlagebeträge den jeweiligen Endbetrag ermitteln, der am Ende der Laufzeit sich auf dem Bankkonto befindet. Abbildung 22 zeigt die Ausgangssituation.

	A	B	C	D	E	F	G
1	Anlagebetrag bis 5.000,00 €				Anlagebetrag ab 6.000,00 €		
2							
3	Zinssatz:	2,00%			Zinssatz:	3,00%	
4	Laufzeit:	96 Monate			Laufzeit:	96 Monate	
5	Monatliche Einzahlung:	-50,00 €			Monatliche Einzahlung:	-50,00 €	
6							
7	Anlagebeträge		Auszahlung				
8	-1.000,00 €						
9	-2.000,00 €						
10	-3.000,00 €						
11	-4.000,00 €						
12	-5.000,00 €						
13	-6.000,00 €						
14	-7.000,00 €						
15	-8.000,00 €						
16	-9.000,00 €						
17	-10.000,00 €						


Abb. 22: Ausgangssituation zur Berechnung einer Sparanlage

Dabei gelten für Anlagebeträge bis maximal 5.000,00 € ein anderer Zinssatz als für Anlagebeträge über 5.000,00 €. Zunächst bekommen einige Tabellenzellen und ein Zellbereich benutzerdefinierte Namen zugewiesen, die bei der eigentlichen Berechnung verwendet werden:

Tabellenzelle/Zellbereich	Name
<b>B3</b>	Zinssatz1
<b>F3</b>	Zinssatz2
<b>B4</b>	Laufzeit
<b>B5</b>	Monatliche_Einzahlung
<b>A8:A17</b>	Anlagebetrag

In der Spalte **C**, genauer gesagt ab Tabellenzelle **C8**, soll nun die Berechnung des Endbetrags erfolgen, der nach Ablauf der Laufzeit sich auf dem Bankkonto befindet. Dafür wird die finanzmathematische Funktion **ZW** (Zukünftiger Wert) verwendet. Die Formel, die Sie in die Tabellenzelle **C8** eintragen lautet in diesem Beispiel:

**=ZW(Zinssatz1/12;Laufzeit;Monatliche\_Einzahlung;Anlagebetrag)**

Beachten Sie bitte, dass der Zinssatz durch 12 dividiert werden muss, da es sich bei dem Zinssatz um den jährlichen Zinssatz handelt, die Funktion **ZW** aber den Endbetrag auf der Basis von Monaten berechnet (u.a. wegen den monatlichen Einzahlungen). Wenn Sie die Eingabe mit der Taste  bestätigen, wird nicht nur in der Tabellenzelle **C8** der Endbetrag für den Anlagebetrag **-1.000,00 €** berechnet, sondern die Formel wird automatisch bis zur Tabellenzelle **C17** erweitert. Dank der dynamischen Arrays. Das scheint zunächst durchaus sinnvoll und vermutlich auch wünschenswert zu sein. Allerdings beträgt der Zinssatz für Anlagebeträge ab mehr als **5.000,00 €** nicht mehr nur **2,00%**, sondern **3,00 %**. Diese Änderung wird aber bei der Übertragung der Ergebnisse in den Bereich **C9** bis **C17** (durch das dynamische Array) nicht berücksichtigt. Vielleicht denken Sie jetzt, dass Sie nur den

Zellbereich **C13** bis **C17** markieren und löschen müssen, damit Sie dann die neue Formel in die Tabellenzelle **C13** eingeben und bestätigen können. Aber wenn Sie das versuchen, werden alle andere Ergebnisse auch gelöscht (obwohl Sie diese Tabellenzellen nicht markiert haben) und in der Tabellenzelle **C8** erscheint der Fehlerwert **#ÜBERLAUF!**.

Sie sehen also, bei diesem Beispiel ist der Einsatz der dynamischen Arrays nicht erwünscht. Es muss also eine Lösung ohne dynamische Arrays geben. Leider gibt es keine Möglichkeit, die Verwendung der dynamischen Arrays zu deaktivieren. Damit aber Excel die dynamischen Arrays nicht einsetzt, müssen Sie einen kleinen Trick anwenden. Geben Sie in die Tabellenzelle **C8** folgende Formel ein und bestätigen die Eingabe:

**=ZW(Zinssatz1/12;Laufzeit;Monatliche\_Einzahlung; @Anlagebetrag)**

Durch die zusätzliche Angabe des **@**-Zeichens vor dem benutzerdefinierten Namen **Anlagebetrag** teilen Sie Excel mit, dass von dem Zellbereich mit dem Namen **Anlagebetrag** nur der Wert aus der aktuellen Zeile für die Berechnung verwendet werden soll, also der Wert aus der Tabellenzelle **A8**. Sie erhalten also nur ein Ergebnis. Jetzt können Sie die Tabellenzelle **C8** erneut auswählen und die Formel mit dem Verfahren *Automatisches Ausfüllen* (siehe Skript **Excel für Microsoft 365 – Automatisches Ausfüllen**) nach unten kopieren, bis zur Tabellenzelle **C13** einschließlich. Dann wählen Sie die Tabellenzelle **C13** und ändern die enthaltene Formel dahingehend ab, dass Sie den benutzerdefinierten Namen **Zinssatz1** ersetzen durch den Namen **Zinssatz2**. Nach Bestätigung der Änderung wählen Sie die Tabellenzelle **C13** erneut aus und kopieren die Formel mit dem Verfahren *Automatisches Ausfüllen* bis zur Tabellenzelle **C17**. Das komplette Ergebnis sehen Sie in Abbildung 23.

	A	B	C	D	E	F	G	H	I	J
1	Anlagebetrag bis 5.000,00 €				Anlagebetrag ab 6.000,00 €					
2										
3	Zinssatz:	2,00%			Zinssatz:	3,00%				
4	Laufzeit:	96 Monate			Laufzeit:	96 Monate				
5	Monatliche Einzahlung:	-50,00 €			Monatliche Einzahlung:	-50,00 €				
6										
7	Anlagebeträge		Auszahlung							
8	-1.000,00 €		6.373,99 €		Formel in C8: =ZW(Zinssatz1/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
9	-2.000,00 €		7.547,35 €		Formel in C9: =ZW(Zinssatz1/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
10	-3.000,00 €		8.720,70 €		Formel in C10: =ZW(Zinssatz1/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
11	-4.000,00 €		9.894,06 €		Formel in C11: =ZW(Zinssatz1/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
12	-5.000,00 €		11.067,41 €		Formel in C12: =ZW(Zinssatz1/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
13	-6.000,00 €		13.042,58 €		Formel in C13: =ZW(Zinssatz2/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
14	-7.000,00 €		14.313,45 €		Formel in C14: =ZW(Zinssatz2/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
15	-8.000,00 €		15.584,32 €		Formel in C15: =ZW(Zinssatz2/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
16	-9.000,00 €		16.855,19 €		Formel in C16: =ZW(Zinssatz2/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					
17	-10.000,00 €		18.126,05 €		Formel in C17: =ZW(Zinssatz2/12;Laufzeit;Monatliche_Einzahlung;@Anlagebetrag)					

Abb. 23: Ergebnis zur Berechnung einer Sparanlage

**Anmerkung:** Bei den Werten für die monatliche Einzahlung und die Anlagebeträge handelt es sich um negative Zahlenwerte, damit als Ergebnis ein positiver Wert angezeigt wird. Bildlich gesprochen geben Sie Geld an die Bank, wo es auf das Konto eingezahlt wird, und erhalten am Ende den ersparten Geldbetrag von der Bank zurück.

Bei der Verwendung des **@**-Zeichens handelt es sich um einen strukturierten Verweis, wie er in Tabellen zur Anwendung kommt (siehe Skript **Excel für Microsoft 365 – Tabellen**, Kapitel **Strukturierte Verweise**, Seite 15).