



Tutorial

Matrixformeln

Grundlegende Einführung

mit einfachen bis hin zu komplexen Beispielen

Stand: 01.05.2022

Download der Exceldatei mit allen Beispielen:

<https://hermann-baum.de/excel/tutorial/matrixformeln/>

Hermann Baum

mail@hermann-baum.de

Inhaltsverzeichnis

Begriffsklärung	2
Was ist eine Matrixformel (Arrayformel)?	3
Beispiel 1: Matrixformel, die mehrere Werte zurückgibt	4
1.1 Teile einer Matrix können nicht geändert werden	5
1.2 Der Zielbereich	5
1.3 Erweiterung des Zielbereichs	6
Beispiel 2: Matrixformel, die nur einen Wert zurückgibt	7
2.1 Formel auswerten - Array-Konstanten	7
2.2 Zielbereich zu groß	8
Matrixformeln, die sich auf mehrere Zellbereiche beziehen	9
Fall 1:	9
Fall 2:	10
Fall 3:	11
Fall 4:	12
Fall 5:	13
Fall 6:	14
Zusammenfassung:	14
Unterschied zwischen alten CSE-Formeln und neuen dynamischen Matrixformeln	15
Allgemeines zum Einsatz von Matrixformeln	16
Beispiel 3: Kopieren eines Bereiches	17
Beispiel 4: Gesamtbetrag berechnen	18
Beispiel 5: Summe der n größten Zahlen	20
5.1 Erster Schritt	20
5.2 Zweiter Schritt	21
5.3 Dritter Schritt	21
5.4 Letzter Schritt	21
Beispiel 6: Alphabetisches Sortieren	23
6.1 Sortieren ohne Matrixformel	24
6.2 Sortieren mit Matrixformel	25
Beispiel 7: SVERWEIS mit mehreren Suchkriterien	26
Beispiel 8: Ist mein Geburtsdatum eine Primzahl?	27
8.1 Primzahl-Test	27
8.2 Bestimmung aller Teiler der Zahl	28
8.2.1 Vorüberlegung mit praktischen Versuchen	29
8.2.2 Verallgemeinerung der bisherigen Matrixformel	30

Begriffsklärung

1. Matrixformel - Arrayformel

Im deutschsprachigen Raum sind das zwei unterschiedliche Bezeichnungen für ein und dasselbe, so wie die Begriffe 'Matrix' und 'Array' auch dasselbe bezeichnen. Im Englischen benutzt man im Allgemeinen den Begriff 'array formula'.

Nicht verwechseln darf man die Matrixformeln mit den Matrixfunktionen (engl. matrix functions). Damit bezeichnet man Excel-Funktionen, die spezielle mathematische Berechnungen im Zusammenhang mit Matrizen ausführen, z.B. die Multiplikation zweier Matrizen (MMULT) oder das Transponieren einer Matrix (MTRANS).

Hat man eine Liste von Zahlen (z.B. 12, 15, 20, 22, 25), so spricht man gerne von einem Array, und man kann sich diese Zahlenliste in Excel nebeneinander in einer Reihe oder untereinander in einer Spalte angeordnet vorstellen.

Hat man eine Anordnung von Zahlen, die sich über mehrere Reihen und mehrere Spalten erstreckt, so spricht man gerne von einem zweidimensionalen Array oder von einer zweidimensionalen Matrix. Das Zahlenfeld dehnt sich in zwei Dimensionen aus: nach rechts und nach unten.

Genauso gut kann man aber im Fall einer einzelnen Zeile oder einer einzelnen Spalte auch von einer eindimensionalen Matrix reden.

2. Alte Matrixformeln (CSE) - neue dynamische Matrixformeln

Seit dem Update vom September 2018 gibt es in Excel 365 die neuen sog. dynamischen Matrixformeln. Microsoft hat die bisherigen Matrixformeln als veraltet eingestuft. Auch Excel 2021 kennt die neuen dynamischen Matrixformeln.

Bis Excel 2019 gibt es nur die nicht-dynamischen, die auch als CSE-Formeln bezeichnet werden, weil man die Eingabe der Formel nicht mit Enter abschließt, sondern mit Control-Shift-Enter. Der Unterschied zwischen beiden wird später erklärt.

Hier in diesem Tutorial geht es um die CSE-Matrixformeln, auf welche die User von Office 2019 und älter angewiesen sind.

Was ist eine Matrixformel (Arrayformel)?

Eine normale Excel-Formel ermittelt ein Ergebnis oder einen Wert (z.B. Zahlenergebnis, Datum oder Text) und zeigt ihn in der Zelle an, in der die Formel steht. Man sagt, die Formel gibt einen einzelnen Wert zurück. Eine Matrixformel gibt eine ganze Reihe von Werten (ein Array von Werten oder eine Matrix von Werten) zurück. Allerdings kann eine Matrixformel auch einen einzelnen Wert zurückgeben. Und damit wären wir auch schon bei den zwei zu betrachtenden Fällen im Zusammenhang mit Matrixformeln:

1. Fall: Die Matrixformel gibt mehrere Werte zurück.
2. Fall: Die Matrixformel gibt nur einen Wert zurück.

Es wird Zeit für ein erstes Beispiel!

Beispiel 1: Matrixformel, die mehrere Werte zurückgibt

Wir öffnen ein leeres Excel-Blatt und tragen in den Bereich B3:D6 einige einfache Zahlen ein.

	A	B	C	D	E
1					
2					
3		2	4	7	
4		1	3	0	
5		5	5	5	
6		2	0	6	
7					

Damit haben wir eine 4×3-Matrix von Zahlenwerten (4 Reihen und 3 Spalten).

Würden wir nun in die Zelle F3 die simple Formel `=B3*10` eingeben, würde in F3 das Ergebnis 20 angezeigt werden. Wenn wir erreichen wollen, dass Excel mit einer einzigen Formel die gesamte Matrix B3:D6 mit 10 multipliziert und eine Matrix (ein Zahlenfeld) von Ergebnissen zurückgibt, dann können wir die Matrixformel `=B3:D6*10` verwenden. Wenn wir diese Formel in die Zelle F3 eingeben und die Taste ENTER drücken, bekommen wir allerdings den Error #WERT angezeigt.

Wir müssen bei der Eingabe von (CSE-)Matrixformeln zwei Regeln beachten:

Regel 1:

Wenn die Matrixformel mehrere Ergebnisse zurückgibt - in diesem Fall eine 4×3-Matrix von Zahlenwerten, müssen wir vor der Eingabe der Formel einen entsprechenden 4×3-Bereich auswählen (selektieren/markieren), so dass er grau hinterlegt ist (sog. Zielbereich).

Regel 2:

Die Eingabe der Formel darf nicht mit der Taste ENTER abgeschlossen werden, sondern mit der Tastenkombination CONTROL-SHIFT-ENTER bzw. STRG-UMSCHALT-EINGABE. Danach zeigt Excel diese Formel immer in geschweiften Klammern an, um deutlich zu machen, dass es sich hier um eine Matrixformel handelt - in unserem Beispiel also `{=B3:D6*10}`.

Die geschweiften Klammern dürfen nicht in die Bearbeitungszeile eingegeben werden! Das hätte zur Folge, dass Excel die vermeintliche Formel als simplen Text interpretiert.

Wir selektieren also den Bereich F3:H6, geben in der Bearbeitungszeile oben die Formel `=B3:D6*10` ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

F3		✕ ✓ fx		{=B3:D6*10}					
	A	B	C	D	E	F	G	H	I
1									
2									
3		2	4	7		20	40	70	
4		1	3	0		10	30	0	
5		5	5	5		50	50	50	
6		2	0	6		20	0	60	
7									

Excel berechnet ein 4×3-Zahlenfeld von Ergebnissen - die sog. Ergebnismatrix - und zeigt sie im Bereich F3:H6, dem Zielbereich, an.

1.1 Teile einer Matrix können nicht geändert werden

Klicken wir nun auf einzelne Zellen unserer Ergebnismatrix, so sehen wir in der Bearbeitungszeile oben immer dieselbe Matrixformel. Ein Versuch, den Inhalt einer einzelnen Zelle innerhalb der Ergebnismatrix zu löschen oder zu verändern, wird von Excel mit der Meldung "Teile einer Matrix können nicht geändert werden" zurückgewiesen.

Auch das Einfügen einer neuen Leerzeile ist nicht möglich, wenn diese Leerzeile mitten durch den Zielbereich einer Matrixformel verlaufen würde.

Um die Matrixformel zu ändern, klickt man auf eine beliebige Zelle innerhalb des Zielbereichs, ändert die Formel oben in der Bearbeitungszeile und schließt die Änderung mit STRG-UMSCHALT-EINGABE ab.

Um den Zellbereich F3:H6 anders zu nutzen, kann man ihn zuerst komplett auswählen und den Inhalt dann mit der ENTF-Taste löschen. Alternativ kann man auch auf eine beliebige Zelle innerhalb des Zielbereichs klicken, die Formel oben in der Bearbeitungszeile komplett löschen und die Änderung mit STRG-UMSCHALT-EINGABE abschließen.

1.2 Der Zielbereich

Der Zielbereich ist der Zellbereich, den man selektiert, bevor man die Matrixformel eingibt. In unserem Beispiel haben wir als Zielbereich den Bereich F3:H6 gewählt. Da die Matrixformel eine 4×3-Matrix zurückgibt, entspricht der Zielbereich auch genau der zurückgegebenen Ergebnismatrix.

Was passiert aber, wenn der Zielbereich kleiner ist als die zu erwartende Ergebnismatrix?

Um dies zu untersuchen, löschen wir den Bereich F3:H6 wieder. Wir selektieren jetzt nur den Bereich F3:G4 und geben die Matrixformel `=B3:D6*10` ein. Das Ergebnis sieht so aus:

	A	B	C	D	E	F	G	H
1								
2								
3		2	4	7		20	40	
4		1	3	0		10	30	
5		5	5	5				
6		2	0	6				
7								

Offensichtlich erscheint hier im Zielbereich nur der linke obere Teil der Ergebnismatrix. Der Rest wird nicht angezeigt. Man spricht hier von der "Schnittmenge". Mit "Schnittmenge" ist die Schnittmenge (also der gemeinsame oder überlappende Bereich) zwischen Zielbereich und der von Excel berechneten Ergebnismatrix gemeint.

So erklärt sich das anfangs häufige Phänomen, dass man vergisst, vor der Eingabe der Matrixformel den Zielbereich zu selektieren, und dann feststellt, dass in der einzigen Zelle, die nun als Zielbereich gilt, der linke obere Wert der Ergebnismatrix angezeigt wird.

Was passiert nun, wenn der Zielbereich zu groß gewählt wird?

Wir löschen erneut die bisherige Matrixformel und wählen als Zielbereich den Bereich F3:F8:

Beispiel 2: Matrixformel, die nur einen Wert zurückgibt

Für eine Matrixformel, die nur einen Wert zurückgibt, wird kein Zielbereich von mehreren Zellen vorselektiert, sondern die Matrixformel wird in eine einzelne Zelle eingegeben, und die Eingabe wird mit STRG-UMSCHALT-EINGABE abgeschlossen.

Wir nehmen an, wir möchten alle Zahlen des Bereiches F3:H6 mit 10 multiplizieren und die Summe der Ergebnisse bilden. Dazu geben wir in die Zelle F3 die folgende Matrixformel ein:

`=SUMME(B3:D6*10)`

und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

	A	B	C	D	E	F	G
1							
2							
3		2	4	7		400	
4		1	3	0			
5		5	5	5			
6		2	0	6			
7							

In der Zelle F3 erscheint das Ergebnis 400.

Was passiert hier?

Würden wir die Eingabe der Formel nur mit ENTER abschließen, würde in der Zelle F3 der Error #WERT angezeigt werden. Die Funktion SUMME erwartet als Parameter Zelladressen oder Konstanten und kann mit dem Rechenausdruck `B3:D6*10` nichts anfangen.

Wenn Excel aber durch die Tastenkombination STRG-UMSCHALT-EINGABE vermittelt wird, dass es sich um eine Matrixformel handelt, wird der Rechenausdruck `B3:D6*10` zu einer Ergebnismatrix ausgewertet, und zwar genau zu der, die wir beim Beispiel 1 im Zielbereich F3:H6 sehen. Die Funktion SUMME addiert dann alle Zahlen der Ergebnismatrix.

2.1 Formel auswerten - Array-Konstanten

Wir können uns diesen Vorgang anschauen, indem wir die Zelle F3 mit unserer Matrixformel auswählen, in der Menüleiste oben Formeln → Formel auswertung anklicken und im sich öffnenden Fenster auf 'Auswerten' klicken.

Matrixformeln, die sich auf mehrere Zellbereiche beziehen

Die bisher betrachteten Matrixformeln bezogen sich nur auf einen einzigen Zellbereich. Zum Beispiel bezog sich die Formel $\{=SUMME(B3:D6*10)\}$ im Beispiel 2 auf den Zellbereich B3:D6.

Was aber passiert, wenn in einer Matrixformel mehrere Zellbereiche vorkommen und sie unter Umständen verschiedene Ausdehnungen haben?

Dazu betrachten wir mehrere Fälle, in denen eine Matrixformel jeweils zwei verschiedene Zellbereiche zu verarbeiten hat. Damit das Ganze möglichst durchschaubar ist, nehmen wir eine einfache Addition dieser zwei Zellbereiche.

Fall 1:

Im einfachsten Fall haben beide Zellbereiche dieselbe Ausdehnung. Wir geben im Bereich B3:D6 die Zahlen 100, 200, 300, ... , 1200 ein und im Bereich F3:H6 die Zahlen 1, 2, 3, ... , 12. Wir selektieren den Zielbereich B9:D12, geben in die Bearbeitungszeile oben die Formel $=B3:D6+F3:H6$ ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

	A	B	C	D	E	F	G	H	I
1		Fall 1							
2									
3		100	200	300		1	2	3	
4		400	500	600		4	5	6	
5		700	800	900		7	8	9	
6		1000	1100	1200		10	11	12	
7									
8									
9		101	202	303					
10		404	505	606					
11		707	808	909					
12		1010	1111	1212					
13									

Wir sehen am Ergebnis, dass die Matrixformel eine Ergebnismatrix zurückgibt, in der B3+F3, C3+G3, usw. miteinander addiert werden. Die Ergebnismatrix hat also die Form:

$\{B3+F3.C3+G3.D3+H3; B4+F4.C4+G4.D4+H4; B5+F5.C5+G5.D5+H5; B6+F6.C6+G6.D6+H6\}$

Bei der Addition zweier 4×3 -Matrizen ist das Resultat wieder eine 4×3 -Matrix. Genau das haben wir auch erwartet.

Fall 2:

Als nächstes addieren wir eine 4×3 -Matrix und eine einzelne Zelle, also eine 1×1 -Matrix. Wir selektieren den Zielbereich B9:D12, geben in die Bearbeitungszeile oben die Formel `=B3:D6+F3` ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

	A	B	C	D	E	F	G	H	I
1		Fall 2							
2									
3		100	200	300		1	2	3	
4		400	500	600		4	5	6	
5		700	800	900		7	8	9	
6		1000	1100	1200		10	11	12	
7									
8									
9		101	201	301					
10		401	501	601					
11		701	801	901					
12		1001	1101	1201					
13									

Auch dieser zweite Fall überrascht uns wenig. Zu jeder Zahl im Bereich B3:D6 wird die Zahl 1 aus der Zelle F3 addiert. Die Ergebnismatrix hat die Form:

`{B3+F3.C3+F3.D3+F3; B4+F3.C4+F3.D4+F3; B5+F3.C5+F3.D5+F3; B6+F3.C6+F3.D6+F3}`

Man erhält dasselbe Ergebnis, wenn man zur Matrix im Bereich B3:D6 eine 4×3 -Matrix aus lauter Einsen addiert.

		100	200	300		1	1	1	
		400	500	600		1	1	1	
		700	800	900		1	1	1	
		1000	1100	1200		1	1	1	

Man kann sich deshalb auch vorstellen, dass Excel intern die 1×1 -Matrix auf die Größe 4×3 erweitert und mit Einsen auffüllt.

Fall 3:

Wir schauen nun, was passiert, wenn wir eine 4×3 -Matrix und eine 1×3 -Matrix addieren. Wir selektieren den Zielbereich B9:D12, geben in die Bearbeitungszeile oben die Formel $=B3:D6+F3:H3$ ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

	A	B	C	D	E	F	G	H	I
1		Fall 3							
2									
3		100	200	300		1	2	3	
4		400	500	600		4	5	6	
5		700	800	900		7	8	9	
6		1000	1100	1200		10	11	12	
7									
8									
9		101	202	303					
10		401	502	603					
11		701	802	903					
12		1001	1102	1203					
13									

In der ersten Spalte wird nun überall eine 1 addiert, in der zweiten Spalte eine 2 und in der dritten Spalte eine 3. Die Ergebnismatrix hat die Form:

$\{B3+F3.C3+G3.D3+H3; B4+F3.C4+G3.D4+H3; B5+F3.C5+G3.D5+H3; B6+F3.C6+G3.D6+H3\}$

Man erhält dasselbe Ergebnis, wenn man zur Matrix im Bereich B3:D6 eine 4×3 -Matrix mit 4 gleichen Zeilen $\{1.2.3\}$ addiert.

	100	200	300		1	2	3	
	400	500	600		1	2	3	
	700	800	900		1	2	3	
	1000	1100	1200		1	2	3	

Man kann sich deshalb auch vorstellen, dass Excel intern die 1×3 -Matrix auf die Größe 4×3 erweitert und die restlichen Zeilen mit den Werten der ersten Zeile auffüllt.

Fall 4:

Um Gewissheit zu haben, probieren wir es auch mit einer einzelnen Spalte aus. Wir addieren eine 4×3 -Matrix und eine 4×1 -Matrix. Wir selektieren den Zielbereich B9:D12, geben in die Bearbeitungszeile oben die Formel $=B3:D6+F3:F6$ ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

	A	B	C	D	E	F	G	H	I
1		Fall 4							
2									
3		100	200	300		1	2	3	
4		400	500	600		4	5	6	
5		700	800	900		7	8	9	
6		1000	1100	1200		10	11	12	
7									
8									
9		101	201	301					
10		404	504	604					
11		707	807	907					
12		1010	1110	1210					
13									

Wir sehen, dass in der ersten Zeile überall eine 1 addiert wird, in der zweiten Zeile jeweils eine 4, in der dritten Zeile eine 7 und in der vierten Zeile eine 10. Die Ergebnismatrix hat die Form:

$\{B3+F3.C3+F3.D3+F3; B4+F4.C4+F4.D4+F4; B5+F5.C5+F5.D5+F5; B6+F6.C6+F6.D6+F6\}$

Man erhält dasselbe Ergebnis, wenn man zur Matrix im Bereich B3:D6 eine 4×3 -Matrix mit 3 gleichen Spalten $\{1;4;7;10\}$ addiert.

	100	200	300		1	1	1	
	400	500	600		4	4	4	
	700	800	900		7	7	7	
	1000	1100	1200		10	10	10	

Und wieder kann man sich vorstellen, dass Excel intern die 4×1 -Matrix auf die Größe 4×3 erweitert und die restlichen Spalten mit den Zahlen aus der ersten Spalte auffüllt.

Fall 5:

Noch ein weiterer Fall funktioniert problemlos: Wir addieren eine 1×3 -Matrix und eine 4×1 -Matrix. Wir selektieren den Zielbereich B9:D12, geben in die Bearbeitungszeile oben die Formel `=B3:D3+F3:F6` ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

	A	B	C	D	E	F	G	H	I
1		Fall 5							
2									
3		100	200	300		1	2	3	
4		400	500	600		4	5	6	
5		700	800	900		7	8	9	
6		1000	1100	1200		10	11	12	
7									
8									
9		101	201	301					
10		104	204	304					
11		107	207	307					
12		110	210	310					
13									

Wir sehen, dass die Ergebnismatrix wieder eine 4×3 -Matrix ist.

Das Ergebnis sieht nun so aus, als ob Excel beide Matrizen zu einer 4×3 -Matrix ergänzt hätte - so, wie es die folgende Abbildung zeigt:

	100	200	300		1	1	1		
	100	200	300		4	4	4		
	100	200	300		7	7	7		
	100	200	300		10	10	10		

Man kann sich also auch hier vorstellen, dass Excel intern beide Matrizen zu einer 4×3 -Matrix erweitert und durch Kopieren der Zeile bzw. der Spalte auffüllt.

Fall 6:

Abschließend betrachten wir einen Fall stellvertretend für die vielen anderen Fälle, die zu Fehlermeldungen führen. Wir versuchen die Addition einer 2×3 -Matrix und einer 4×1 -Matrix. Wir selektieren den Zielbereich B9:D12, geben in die Bearbeitungszeile oben die Formel `=B3:D4+F3:F6` ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

	A	B	C	D	E	F	G	H	I
1		Fall 6							
2									
3		100	200	300		1	2	3	
4		400	500	600		4	5	6	
5		700	800	900		7	8	9	
6		1000	1100	1200		10	11	12	
7									
8									
9		101	201	301					
10		404	504	604					
11		#NV	#NV	#NV					
12		#NV	#NV	#NV					
13									

Diesmal erhalten wir in den letzten beiden Reihen der Ergebnismatrix den Fehler #NV. Für Excel ist nicht mehr klar, mit welchen Inhalten die letzten beiden Zeilen der erweiterten Matrix aufgefüllt werden sollen.

	100	200	300		1	1	1	
	400	500	600		4	4	4	
	?	?	?		7	7	7	
	?	?	?		10	10	10	

Weitere Zeilen auffüllen kann Excel offenbar nur dann, wenn nur **eine** Zeile vorgegeben ist (siehe Fall 5). Sind bereits **mehrere** Zeilen vorgegeben, können keine weiteren automatisch aufgefüllt werden. Dasselbe gilt entsprechend für Spalten.

Zusammenfassung:

Sollen in einer Matrixformel zwei Zellbereiche miteinander verrechnet werden, die unterschiedliche Anzahl von Zeilen bzw. Spalten haben, kann man sich die Auswertung der Formel so vorstellen, dass Excel beide Zellbereiche intern auf die gleiche Größe erweitert. Zusätzlich geschaffene Zeilen (bzw. Spalten) können nur dann automatisch mit Werten aufgefüllt werden, wenn nur eine einzige Zeile (bzw. eine einzige Spalte) vorgegeben ist. Ansonsten tritt der Fehler #NV in der Zielmatrix auf.

Unterschied zwischen alten CSE-Formeln und neuen dynamischen Matrixformeln

Für die neuen dynamischen Matrixformeln gilt im Unterschied zu den alten CSE-Matrixformeln:

1. Es muss kein Zielbereich vorselektiert werden. Man gibt die Matrixformel in die Zelle F3 ein und Excel ermittelt den Zielbereich automatisch. Die Ergebniswerte fließen also von F3 nach rechts und nach unten in die entsprechenden Zellen. Gibt es in diesem Zielbereich nichtleere Zellen, erscheint eine entsprechende Warnmeldung.
2. Die Formeleingabe wird - wie bei anderen Formeln auch - nur mit der Taste ENTER abgeschlossen. Excel erkennt automatisch, dass es sich um eine Matrixformel handelt.
3. Bei einer Vergrößerung oder Verkleinerung des Bereichs, auf den sich die Matrixformel bezieht, passt sich der Zielbereich automatisch an. Löscht man beispielsweise eine Zeile im Bereich, auf den sich die Matrixformel bezieht, so wird automatisch im Zielbereich die entsprechende Zeile ebenfalls gelöscht. Deshalb bezeichnet man die neuen Matrixformeln als 'dynamisch'. Die Größe ihres Zielbereichs bzw. Ausdehnungsbereichs passt sich dynamisch an.

Allgemeines zum Einsatz von Matrixformeln

Um alle Zahlen im Bereich B3:D6 mit 10 zu multiplizieren, braucht man sicherlich keine Matrixformel. Man könnte auch in der Zelle F3 die Formel $=B3*10$ eintragen und in die restlichen Felder kopieren.

Doch hier ergibt sich bereits ein erster Aspekt zum Einsatz von Matrixformeln. Stellen wir uns vor, dass wir es nicht wie in unserem Beispiel mit einer 4×3 -Matrix zu tun haben, sondern mit einer 4000×3000 -Matrix, und alle Zahlen werden nicht mit zehn multipliziert, sondern einer komplexeren Rechenoperation unterzogen. Hier wäre es von Vorteil, diese Formel als Matrixformel einzugeben anstatt sie in 12 Millionen Zellen zu kopieren. Wird durch eine Unachtsamkeit eine dieser Einzelformeln verändert, ist die Fehlersuche sehr mühsam. Bei einer Matrixformel ist es nicht möglich, aus Versehen eine einzelne Formel in dem großen Zahlenfeld zu verändern. Diese Fehlerquelle ist sozusagen bei einer Matrixformel ausgeschlossen.

Die meisten Beispiele, die im Zusammenhang mit Matrixformeln (hier meine ich die alten CSE-Formeln) angeführt werden, lassen sich auch ohne Matrixformel lösen. Diese Einführungsbeispiele sind oftmals bewusst einfach gehalten, damit sie leichter nachvollzogen werden können.

Auf der anderen Seite liest man: 'Matrixformeln machen das Unmögliche möglich!'. Dies allerdings bezweifle ich. Was mit (CSE-)Matrixformeln möglich ist, lässt sich (auf Umwegen) auch mit konventionellen Formeln realisieren.

Die Berechnungen, die bei einer (CSE-)Matrixformel intern ablaufen, kann man meiner Meinung nach immer auch in Einzelschritte zerlegen, diese (z.B. in ausgeblendeten Bereichen) mit konventionellen Formeln berechnen und das Endergebnis da eintragen, wo es die Matrix-Formal auch tun würde. Mit anderen Worten: Was eine (CSE-)Matrixformel kann, ist auch durch konventionelle Formeln realisierbar.

Bei den neuen dynamischen Matrixformeln kann es durchaus sein, dass das dynamische Verhalten und ggf. noch andere Aspekte nicht durch konventionelle Formeln nachgebildet werden können.

Es entsteht natürlich ein erhöhter Platzbedarf, vielleicht auch eine geringere Performance, aber auf neuen leeren Excel-Blättern (evtl. ausgeblendeten) ist immer Platz vorhanden. Durch die zusätzlichen Formeln auf den ausgeblendeten Hilfsbereichen entsteht sicherlich auch ein höherer Wartungsaufwand. Im Gegenzug nimmt man bei platzsparenden, kompakten und teilweise genialen Matrixformeln in Kauf, dass sie vielleicht nur schwer verständlich sind und mehr Zeit erfordern, bis man sie analysiert hat.

Betrachten wir einige weitere Beispiele.

Beispiel 3: Kopieren eines Bereiches

Der einfachste Fall für eine Matrixformel, die mehrere Ergebnisse zurückgibt, ist das Kopieren eines ganzen Bereiches. In diesem Beispiel wird eine Kopie des Bereiches B2:D8 angelegt. In der Praxis kann es manchmal erwünscht sein, einen bestimmten Bereich (eine informative Tabelle) auch auf einem anderen Excel-Blatt sichtbar zu machen, ohne dass hier einzelne Werte geändert werden können. In unserem vereinfachten Beispiel erscheint der kopierte Bereich direkt unter dem Original.

	A	B	C	D	E
1					
2		Alter	männlich	weiblich	
3		0 - 12 Jahre	7	7	
4		13 - 14 Jahre	6	4	
5		15 - 16 Jahre	4	2	
6		17 - 18 Jahre	2	3	
7		19 - 21 Jahre	5	1	
8		über 21 Jahre	2	2	
9					
10					
11		Kopie der Tabelle:	{=\$B\$2:\$D\$8}		
12					
13		Alter	männlich	weiblich	
14		0 - 12 Jahre	7	7	
15		13 - 14 Jahre	6	4	
16		15 - 16 Jahre	4	2	
17		17 - 18 Jahre	2	3	
18		19 - 21 Jahre	5	1	
19		über 21 Jahre	2	2	
20					

Zum Anlegen dieser Kopie markiert man einen Bereich gleicher Größe - hier der Bereich B13:D19 - und trägt in die Bearbeitungszeile die Formel `=B2:D8` ein, die mit STRG-UMSCHALT-EINGABE abgeschlossen wird. Die Formel erscheint danach in der Bearbeitungszeile in geschweiften Klammern: `{=B2:D8}`

Im kopierten Bereich ist es nicht möglich, einzelne Zellen zu verändern. Man kann nur den Bereich als Ganzes löschen oder die Formel für den gesamten Bereich verändern.

Beispiel 4: Gesamtbetrag berechnen

Das folgende Beispiel wird gerne im Zusammenhang mit einführenden Beispielen für Matrixformeln angeführt, obwohl es nicht wirklich praxisrelevant ist. Es soll auch hier nicht fehlen, da es auf einfache Weise illustriert, wie eine Matrixformel arbeitet.

Beispiele, in denen der Einsatz von Matrixformeln wirklich angebracht ist, führen sehr schnell zu komplexeren Formeln, die nicht mehr auf den ersten Blick durchschaubar sind, wie die letzten Beispiele dieses Tutorials zeigen.

	A	B	C	D	E
1					
2					
3		Menge	Einzelpreis		
4		1	10,00 €		
5		2	25,00 €		
6		5	8,00 €		
7		1	1.000,00 €		
8		2	50,00 €		
9			Gesamt:	1.200,00 €	
10					

Es geht um eine simple Rechnung, in der die Mengenangaben mit den Einzelpreisen multipliziert werden und alle Ergebnisse zu einem Gesamtpreis addiert werden.

Die Funktion SUMME erwartet als Parameter einzelne Werte oder ganze Bereiche.

Mit einem Rechenausdruck **Bereich * Bereich** - in unserem Beispiel **\$B\$4:\$B\$8 * \$C\$4:\$C\$8** - kann sie nichts anfangen und quittiert die Formel mit dem Error #WERT, wenn man vergisst, die Formeleingabe mit STRG-UMSCHALT-EINGABE abzuschließen.

Wird sie jedoch mit Hilfe der Tastenkombination STRG-UMSCHALT-EINGABE als Matrixformel eingegeben, läuft die Formelauswertung anders. Wie man durch Aufrufen des Menüpunktes 'Formelauswertung' sehen kann (im ► [Beispiel 2.1](#) näher erläutert), wird der Rechenausdruck in der Klammer zuerst zu der Matrix (dem Array) {10;50;40;1000;100} ausgewertet und danach die Funktion SUMME auf diese Matrix angewendet.

Dieses Beispiel ist deshalb nicht sehr praxisrelevant, weil in der Praxis sicherlich die Zwischenergebnisse in der Spalte D aufgeführt wären und man in der Zelle D9 dann eine einfache Summenformel einsetzt.

Interessanter wird die Matrixformel dann, wenn mehrere Mengenspalten vorkommen:

	A	B	C	D	E
11					
12					
13		Menge Januar	Menge Februar	Einzelpreis	
14		1	3	10,00 €	
15		2	2	25,00 €	
16		5	6	8,00 €	
17		1	2	1.000,00 €	
18		2	4	50,00 €	
19				Gesamt:	3.528,00 €
20					

Hier wird der Rechenausdruck **\$B\$14:\$C\$18 * \$D\$14:\$D\$18** so ausgewertet, dass sowohl alle Mengenangaben in der Spalte B als auch alle Mengenangaben in der Spalte C mit den Einzelpreisen in Spalte D

multipliziert werden (vergleiche oben ► [Fall 4](#)). Zwei Spalten mal eine Spalte ergibt eine zweispaltige Matrix, in unserem Beispiel die Matrix

$\{10.30; 50.50; 40.48; 1000.2000; 100.200\}$.

Die Zwischenergebnisse dieser zweispaltigen Matrix addiert die Funktion SUMME dann zu einem Gesamtbetrag in Zelle E19.

Beispiel 5: Summe der n größten Zahlen

Von nun an wird's komplexer und die Matrixformeln zeigen ihre wirkliche Stärke.

Um die Summe der 10 größten Zahlen ($n = 10$) aus einer Zahlentabelle zu berechnen, könnte man auch ohne Einsatz einer Matrixformel die zehn größten Zahlen irgendwo auf dem Arbeitsblatt mit Hilfe der Funktion KGRÖSSTE darstellen und darunter eine einfache Summenformel setzen.

Der Einsatz einer Matrixformel wird jedoch erst so richtig motiviert, wenn die Zahl für n nicht fest vorgegeben ist, sondern vom Benutzer des Arbeitsblattes in eine Zelle eingegeben wird. Dieser Fall soll hier beleuchtet werden.

So sieht das Ganze aus:

	A	B	C	D	E	F
1						
2						
3		10	20	10		
4		22	400	25		
5		5	600	8		
6		90	1	1000		
7		2	10	50		
8						
9				Summe der	5	
10				größten Zahlen:		
11				2140		
12						

In der Zelle D11 steht die Formel:

```
{=SUMME (KGRÖSSTE ($B$3:$D$7; ZEILE (BEREICH.VERSCHIEBEN ($B$1; ; ; $E$9))))}
```

Die Formel berechnet die Summe der n größten Zahlen aus dem Zahlenfeld im Bereich B3:D7, wobei der Wert für die Zahl n aus der Zelle E9 ausgelesen wird.

Wir nähern uns dieser Formel am besten schrittweise an.

5.1 Erster Schritt

Tragen wir versuchsweise in die Zelle G3 die Formel `=KGRÖSSTE(B3:D7; 1)` ein, so bekommen wir den Wert 1000 angezeigt. Dies ist die größte Zahl des Zahlenfeldes.

Um die zweit- und drittgrößte Zahl zu erhalten, könnten wir beispielsweise darunter in der Zelle G4 die Formel `=KGRÖSSTE(B3:D7; 2)` und in Zelle G5 die Formel `=KGRÖSSTE(B3:D7; 3)` eintragen.

Damit man für die fünf größten Zahlen nicht fünf Formeln eintippen muss, greift man zu einem kleinen Trick: In der ersten Formel (ganz oben) ersetzt man die 1 am Ende der Formel durch `ZEILE(A1)`. Der Ausdruck `ZEILE(A1)` erzeugt auch die Zahl 1. Nun kann man jedoch die Formel nach unten kopieren, denn in der zweiten Zeile wird `ZEILE(A1)` dann zu `ZEILE(A2)`, in der dritten zu `ZEILE(A3)` usw.

An Stelle von A1 kann man auch B1 oder X1 schreiben. Die Spalte spielt für die Funktion ZEILE keine Rolle.

Wir probieren es aus: Wir tragen in die Zelle H3, die Formel `=KGRÖSSTE(B3:D7; ZEILE(A1))` ein. Wenn wir die Formel nach unten bis zur Zelle H7 kopieren, bekommen wir die fünf größten Zahlen angezeigt und können problemlos die Summe bilden.

ABER: Das Ganze ist nicht flexibel. Es ist auf die Summe von fünf Zahlen festgelegt.

5.2 Zweiter Schritt

Wir gehen nun zu einer Matrixformel über. Anstatt eine konventionelle Formel mehrmals nach unten zu kopieren, markieren wir den Bereich I3:I7, geben in die Bearbeitungszeile oben die Formel `=KGRÖSSTE(B3:D7; {1;2;3;4;5})` ein und schließen die Eingabe mit STRG-UMSCHALT-EINGABE ab.

Da der zweite Parameter der Funktion KGRÖSSTE nun nicht aus einem einzelnen Wert besteht, sondern aus einer Matrix von fünf Zahlen, und die Formel durch die Tastenkombination STRG-UMSCHALT-EINGABE als Matrixformel deklariert wurde, gibt die Formel auch eine Matrix von fünf Ergebnissen zurück und verteilt sie auf den markierten Bereich. Wir sehen jetzt im Bereich I3:I7 die fünf größten Zahlen: 1000; 600; 400; 90; 50.

5.3 Dritter Schritt

Der nächste Schritt ist, die Matrix {1;2;3;4;5} von der Funktion ZEILE erzeugen zu lassen. Dies erreichen wir mit dem Ausdruck `Zeile(A1:A5)`.

Wir probieren es aus:

Wir markieren den Bereich J3:J7, geben die Formel `=KGRÖSSTE(B3:D7; ZEILE(A1:A5))` ein und schließen mit der Tastenkombination STRG-UMSCHALT-EINGABE ab. Wir sehen wieder die fünf größten Zahlen.

5.4 Letzter Schritt

Der letzte Schritt besteht nun darin, dass wir an Stelle des festen Bezugs A1:A5 für die Funktion ZEILE einen variablen Bezug "A1:An" basteln.

Adressen für einzelne Zellen oder auch für ganze Zellbereiche wie z.B. D4:F25 nennt man auch Bezüge. Sie sind das, worauf sich Funktionen wie SUMME oder ZEILE beziehen, das heißt, sie haben diese Bezüge als Parameter. In der Mehrzahl der Fälle geben die Funktionen Werte zurück (Zahlenwerte, Datumswerte oder Texte). Es gibt aber auch Funktionen, die Bezüge zurückgeben wie zum Beispiel INDIREKT oder BEREICH.VERSCHIEBEN.

Eine der Funktionen, die keinen Wert zurückgibt, sondern einen Bezug, ist die Funktion BEREICH.VERSCHIEBEN (engl. OFFSET). Sie hat die Syntax:

BEREICH.VERSCHIEBEN (Bezug; Zeilen; Spalten; [Höhe]; [Breite])

Die Parameter bedeuten:

- **Bezug:**
Startwert für den Bezug (einzelne Zelle oder Bereich)
- **Zeilen:**
Versatz nach unten oder (bei negativen Werten) nach oben
- **Spalten:**
Versatz nach rechts oder (bei negativen Werten) nach links
- **Höhe (optional):**
Anzahl der Zeilen des zurückgegebenen Bezugs
- **Breite (optional):**
Anzahl der Spalten des zurückgegebenen Bezugs

Fehlen die Parameter Höhe und Breite, werden die Höhe und Breite des ersten Parameters übernommen.

So würde beispielsweise die Formel `=BEREICH.VERSCHIEBEN (D2; 1; 3; 4; 2)` den Bezug G3:H6 zurückgeben.

Ausgehend von D2 haben wir einen Versatz um eine Zeile nach unten (D3) und 3 Spalten nach rechts (G3). Die linke obere Zelle des zurückzugebenden Bezugs ist also G3. Da die Höhe 4 Zeilen und die Breite 2 Spalten sein soll, ergibt sich der Bezug G3:H6.

Die Formel `=BEREICH.VERSCHIEBEN (A1; ; ; 5)` würde den Bezug A1:A5 zurückgeben. Die Parameter 'Zeilen' und 'Spalten' sind nicht angegeben. Sie nehmen daher den Default-Wert 0 an (kein Versatz nach unten und kein Versatz nach rechts). Der Parameter 'Höhe' ist mit 5 angegeben und der optionale Parameter 'Breite' fehlt; dieser hat damit den Wert 1, weil als Startbezug nur eine einzelne Zelle angegeben ist (`A1`), welche die Breite 1 besitzt. Wir bekommen ausgehend vom Startwert `A1` einen Bereich der Höhe 5 (d.h. 5 Zeilen) und der Breite 1 (d.h. 1 Spalte) - also A1:A5.

Und damit hätten wir's geschafft, denn wir brauchen an Stelle der 5 für den Parameter 'Höhe' nur noch die Zelladresse `E9` einzutragen. Dann wird die Zahl für den Parameter 'Höhe' aus dieser Zelle entnommen.

Steht in der Zelle E9 z.B. die Zahl 8, erzeugt der Ausdruck `BEREICH.VERSCHIEBEN (A1; ; ; E9)` den Bezug A1:A8, und der Ausdruck `ZEILE (BEREICH.VERSCHIEBEN (A1; ; ; E9))` wird aufgelöst zu `ZEILE(A1:A8)`, was wiederum die Matrix `{1;2;3;4;5;6;7;8}` produziert.

Der Ausdruck `KGRÖSSTE (B3:D7; ZEILE (BEREICH.VERSCHIEBEN (A1; ; ; E9)))` wird dann aufgelöst zu `KGRÖSSTE (B3:D7; {1;2;3;4;5;6;7;8})` und dieser wiederum zu `{1000; 600; 400; 90; 50; 25; 22; 20}`.

Die Funktion SUMME bildet dann die Summe dieser acht Zahlen.

Beispiel 6: Alphabetisches Sortieren

Das Thema 'Sortieren' wurde bereits ausführlich in [► diesem Tutorial](#) behandelt. Hier soll nur als kleine Ergänzung eine Möglichkeit aufgezeigt werden, wie man eine Liste von Namen mit Hilfe einer einzigen Matrixformel sortieren kann. - Im Office 365 und ab Office 2021 steht die neue Funktion SORTIEREN zur Verfügung, die alle diese Bemühungen überflüssig macht.

Seit Ende Januar 2022 gibt es die Funktion SORTIEREN auch für ältere Excel-Versionen (Excel 2007 - Excel 2019)! Sie kann kostenfrei heruntergeladen und verwendet werden:

► <https://hermann-baum.de/excel/hbSort/de/>.

Damit wäre das nachfolgende Problem mit der einfachen Formel `{=Sortieren (B4:B20)}` gelöst.

Wer jedoch als Lerneffekt den Aufbau einer Matrixformel zum Sortieren studieren möchte - oder wer ein Open Source Produkt benutzt, auf dem Microsoft VBA nicht läuft, dem wird empfohlen, den Rest dieses Abschnitts doch noch zu lesen.

Voraussetzung für die hier vorgestellte Matrixformel ist, dass kein Name doppelt vorkommt (diese Einschränkung entfällt bei Verwendung der Funktion SORTIEREN). Sie eignet sich daher insbesondere für sog. IDs, also Kombinationen aus Buchstaben, Zeichen und Ziffern, die irgendetwas, z.B. einen Verkaufsartikel, eindeutig identifizieren. Die zu sortierende Liste darf Lücken aufweisen.

	A	B	C	D	E	F	G
1		Sortieren					
2				Einzelformeln		Eine Formel	
3		Mitarbeiter					
4		Waldecker		Andersson		Andersson	
5		Baumann		Baumann		Baumann	
6		Stolzinger		Brecht		Brecht	
7		Kaufmann		Glatzer		Glatzer	
8		Neumeier		Hirzer		Hirzer	
9		Brecht		Kaufmann		Kaufmann	
10		Glatzer		Meininger		Meininger	
11		Hirzer		Mombach		Mombach	
12		Meininger		Neumeier		Neumeier	
13				Stolzinger		Stolzinger	
14				Waldecker		Waldecker	
15		Andersson					
16		Mombach					
17							
18							
19							
20							
21							

In der Zelle D4 steht die Matrixformel

```
{=WENNFEHLER (INDEX ($B$4:$B$20; VERGLEICH (ZEILE ($A1); ZÄHLENWENN ($B$4:$B$20; "<="&$B$4:$B$20); 0)); "" )}.
```

Sie wurde nach unten kopiert bis zur Zelle D20.

Um die Formel zu verstehen, schauen wir uns den Sortiervorgang zuerst noch einmal ohne Matrixformeln an.

6.1 Sortieren ohne Matrixformel

In der folgenden Abbildung wurden die Spalten D bis I ausgeblendet.

	A	B	C	J	K	L
1		Sortieren				
2						
3		Mitarbeiter				
4		Waldecker		11	Andersson	
5		Baumann		2	Baumann	
6		Stolzinger		10	Brecht	
7		Kaufmann		6	Glatzer	
8		Neumeier		9	Hirzer	
9		Brecht		3	Kaufmann	
10		Glatzer		4	Meiningen	
11		Hirzer		5	Mombach	
12		Meiningen		7	Neumeier	
13				0	Stolzinger	
14				0	Waldecker	
15		Andersson		1		
16		Mombach		8		
17				0		
18				0		
19				0		
20				0		
21						

Die Spalten D bis I wurden in dieser Abbildung ausgeblendet.

In die Zelle J4 tragen wir die Formel `=ZÄHLENWENN(B4:B20;"<="&$B4)` ein und kopieren sie nach unten bis zur Zelle J20. Diese Formel zählt für jeden Namen in der Liste, wie viele Namen kleiner oder gleich sind. Damit haben wir für jeden Namen eine Zahl, welche die Position des Namens in der sortierten Liste angibt.

Zum Beispiel errechnet die Formel für den Namen 'Waldecker' den Wert 11, denn alle 11 Namen sind kleiner oder gleich 'Waldecker'. Für den Namen 'Baumann' wird der Wert 2 berechnet, denn die beiden Namen 'Andersson' und 'Baumann' sind kleiner oder gleich 'Baumann' usw.

In die Zelle K4 tragen wir die Formel `=WENNFEHLER(INDEX(B4:B20;VERGLEICH(ZEILE($A1);J$4:J$20;0));"")` ein und kopieren sie ebenfalls nach unten bis zur Zelle K20.

Schauen wir uns diese Formel am besten von innen nach außen an. Der Ausdruck `ZEILE($A1)` gibt die Zahl 1 zurück. Beim Kopieren nach unten wird daraus `ZEILE($A2)`, `ZEILE($A3)` usw. Das heißt, es werden einfach die Zahlen 1, 2, 3, ... erzeugt.

Die Funktion `VERGLEICH` gibt eine relative Position in einer Liste zurück. Der Ausdruck `VERGLEICH(ZEILE($A1);J$4:J$20;0)` ist in der ersten Zeile gleichbedeutend mit dem Ausdruck `VERGLEICH(1;J$4:J$20;0)`. Es wird also die Position der '1' in der Liste J\$4:J\$20 gesucht. Die '1' wird an der Position 12 gefunden. Man beachte, dass der alphabetisch erste Name ('Andersson') an der 12. Position in Spalte B steht.

In der zweiten Zeile findet der Ausdruck `VERGLEICH(2;J$4:J$20;0)` die Zahl '2' an der Position 2. In der dritten Zeile findet der Ausdruck `VERGLEICH(3;J$4:J$20;0)` die Zahl '3' an der Position 6. Das bedeutet, dass der dritte Name der alphabetischen Reihenfolge ('Brecht') an der 6. Position steht.

Die Funktion INDEX verwandelt nun die gefundenen Positionszahlen in die entsprechenden Namen um. Zum Beispiel heißt der entsprechende Ausdruck in der dritten Zeile INDEX (\$B\$4:\$B\$20; 6). Die INDEX-Funktion liefert uns also in der dritten Zeile den Namen an der 6. Position, nämlich 'Brecht'.

Die innerste Funktion ZEILE(. . .) produziert ab der 12. Zeile die Zahlen 12, 13, 14, usw., die in der Formel einen Fehler erzeugen. Dieser Fehler wird durch die äußere Funktion WENNFEHLER abgefangen. Diese Funktion gibt im Fall eines Fehlers einen leeren String zurück.

6.2 Sortieren mit Matrixformel

Für die Sortierung ohne Matrixformel benötigt man eine Hilfsspalte, in der die Positionierung der einzelnen Namen ermittelt wird. In den Sortierformeln in der Spalte K greift man dann auf diese Zahlen zu, indem man den Bezug J\$4:J\$20 als Parameter der Funktion VERGLEICH einsetzt.

Die Idee ist nun, an Stelle des Bezugs J\$4:J\$20 denjenigen Funktionsaufruf einzusetzen, der diese Liste von Positionierungszahlen erzeugt und als Matrix von 17 Zahlen zurückgibt. Dazu nehmen wir die Formel in der J-Spalte, nämlich =ZÄHLENWENN (\$B\$4:\$B\$20; "<="&\$B4) und ersetzen den Ausdruck "<="&\$B4) durch "<="&\$B\$4:\$B\$20).

Als kleines Zwischenexperiment tragen wir einmal diese neue Formel

=ZÄHLENWENN (\$B\$4:\$B\$20; "<="&\$B\$4:\$B\$20)

in die Zelle H4 ein. Ob wir nun die Formeleingabe mit EINGABE abschließen oder mit STRG-UMSCHALT-EINGABE, wir sehen immer nur die erste Zahl, die 11. Wenn wir aber den Bereich H4:H20 vorher markieren und dann die Formeleingabe mit STRG-UMSCHALT-EINGABE abschließen, sehen wir die ganze Zahlenliste der 17 Positionierungszahlen.

Ersetzen wir in der konventionellen Sortierformel aus Spalte K nun den Bezug J\$4:J\$20 durch den Funktionsaufruf ZÄHLENWENN (\$B\$4:\$B\$20; "<="&\$B\$4:\$B\$20) und schließen die Bearbeitung mit STRG-UMSCHALT-EINGABE ab, haben wir die Matrixformel in Zelle D4 erzeugt. Sie gibt den ersten Namen der sortierten Liste zurück.

Die Matrixformel in Zelle D4 wurde nach unten kopiert. Wir haben deshalb in der D-Spalte 17 einzelne, unterschiedliche Matrixformeln. Sie unterscheiden sich nur dadurch, dass in der Mitte der Formel aus ZEILE(A1) in der nächsten Zeile ZEILE(A2) usw. wird.

Um nur eine einzige Matrixformel für die ganze Spalte zu erhalten, markieren wir z.B. den Bereich F4:F20 und geben dieselbe Matrixformel ein, nur an Stelle von ZEILE(A1) schreiben wir ZEILE (\$A\$1:\$A\$17). Die Eingabe schließen wir mit STRG-UMSCHALT-EINGABE ab. Nun hat der ganze Bereich F4:F20 dieselbe Matrixformel. Einzelne Zellen dieses Bereichs können nicht mehr verändert oder gelöscht werden. Diese Lösung wäre also etwas sicherer gegen eine versehentliche Veränderung einzelner Zellen.

Die Formel in Zelle F4 lautet also:

{=WENNFEHLER (INDEX (\$B\$4:\$B\$20; VERGLEICH (ZEILE (\$A\$1:\$A\$17); ZÄHLENWENN (\$B\$4:\$B\$20; "<="&\$B\$4:\$B\$20); 0)); ""}}.

Dies ist die endgültige Formel.

Beispiel 7: SVVERWEIS mit mehreren Suchkriterien

In den Zellen G3 und G4 kann man zwei Suchkriterien auswählen. Die Formel in Zelle G5 sucht dann wie ein SVVERWEIS die passende Einwohnerzahl aus der Tabelle. Die Formel lautet:

```
{=SVVERWEIS(G3&G4; WAHL({1.2}); $B$4:$B$9&$C$4:$C$9; $D$4:$D$9); 2; 0)}
```

G5								
A	B	C	D	E	F	G	H	
1	SVVERWEIS mit mehreren Suchkriterien							
2								
3	Alter	Geschlecht	Einwohner		Alter:	5 - 9		
4	0 - 4	männlich	528		Geschlecht:	weiblich		
5	0 - 4	weiblich	522		Einwohner:	833		
6	5 - 9	männlich	798					
7	5 - 9	weiblich	833					
8	10 - 14	männlich	1466					
9	10 - 14	weiblich	1522					
10								

Wenn man in der obigen Tabelle mit Hilfe der Funktion SVVERWEIS nach der Altersgruppe "5 - 9" suchen will, um die Einwohnerzahl für diese Altersgruppe zurückzugeben, wendet man die Formel `=SVVERWEIS(G3; B4:D9; 3; 0)` an.

Die Funktion SVVERWEIS findet die erste Übereinstimmung in Zeile 6 und gibt die Einwohnerzahl 798 zurück. Es ist aber auch möglich, nach der Altersklasse "5 - 9" und dem Geschlecht "weiblich" zu suchen.

Dazu nimmt man als Suchkriterium die String-Verkettung beider Suchkriterien, in unserem Fall also G3&G4.

Mit Hilfe der Funktion WAHL erstellt man eine Tabelle, deren erste Spalte (die Suchspalte für die Funktion SVVERWEIS) aus den String-Verkettungen der beiden Spalten B und C besteht und deren zweite Spalte aus der Spalte D mit den Einwohnerzahlen besteht.

Der Ausdruck `WAHL({1.2}; B4:B9&C4:C9; D4:D9)` gibt genau diese Tabelle zurück.

Auf diese Weise kann man auch mehr als zwei Suchkriterien mit "&" verknüpfen.

Beispiel 8: Ist mein Geburtsdatum eine Primzahl?

Bei diesem Beispiel kommt die Stärke der Matrixformeln so richtig zur Geltung. Seine praktische Bedeutung für die Business-Welt ist zwar gleich null, aber es vermittelt außerordentlich viele Aspekte des Einsatzes von Matrixformeln und ist deshalb sehr lehrreich.

Im ersten Schritt wird gezeigt, wie man bei einer vorgegebenen Zahl mit Hilfe einer einzigen (Matrix-)Formel testet, ob es sich um eine Primzahl handelt.

Im zweiten Schritt stelle ich eine Möglichkeit vor, sämtliche Teiler der zu prüfenden Zahl mit Hilfe einer Matrixformel berechnen zu lassen. Allerdings wird die Formel in diesem Fall schon ziemlich komplex.

8.1 Primzahl-Test

Zuerst schauen wir uns an, wie bei einer Zahl geprüft werden kann, ob sie eine Primzahl ist.

	A	B	C	D	E
1	Primzahl-Test				
2					
3			23082001	ist Primzahl?	WAHR
4					

In die Zelle C3 gibt man die zu prüfende Zahl ein, z.B. sein achtstelliges Geburtsdatum ohne die Punkte. Die Formel in Zelle E3 prüft dann, ob es sich um eine Primzahl handelt. In der Abbildung sehen wir, dass der 23.08.2001 (ohne die Punkte) eine Primzahl ist.

Die (Matrix-)Formel in Zelle E3 lautet:

```
{=SUMME(1*(REST($C$3;ZEILE(BEREICH.VERSCHIEBEN($A$1;;;GANZZAHL(WURZEL($C$3))))=0))=1}
```

Das hier verwendete Prüfverfahren ist mathematisch sehr einfach gehalten und wenig optimiert. Es geht ja in diesem Tutorial nicht um Primzahl-Testverfahren, sondern um Matrixformeln.

Die Zahl wird durch alle Zahlen {1; 2; 3; 4; 5; ...} dividiert, die kleiner sind als die Wurzel der Zahl. Wenn der Rest dieser Division gleich null ist, ist ein Teiler gefunden worden. Die äußere Funktion SUMME in der Formel ermittelt die Anzahl der gefundenen Teiler. Diese Anzahl ist bei einer Primzahl gleich 1, da nur einmal - nämlich bei der Division durch 1 - der Rest gleich null ist.

Erklärung der Formel:

Der Ausdruck `ZEILE(BEREICH.VERSCHIEBEN(A1;;;GANZZAHL(WURZEL(C3)))` erzeugt die Zahlenmatrix {1; 2; 3; 4; 5; ...; n}, wobei die letzte Zahl n gleich `GANZZAHL(WURZEL(C3))` ist, also die größte Ganzzahl, die noch kleiner ist als die Wurzel der zu prüfenden Zahl. Die Erzeugung dieser Zahlenmatrix wird im [Abschnitt 5.4](#) ausführlich erklärt.

Die Wurzel aus 23082001 zum Beispiel ist gleich 4804,373. In diesem Fall wird die Zahlenmatrix {1; 2; 3; 4; 5; ...; 4804} erzeugt. Unsere Gesamtformel sieht nach diesem Berechnungsschritt so aus:
`{=SUMME(1*(REST(C3;{1;2;3;4;5;...;4804})=0))=1}`.

Die Funktion REST berechnet den Rest einer Ganzzahldivision.

So würde beispielsweise die Formel `=REST(27; 8)` den Wert 3 zurückgeben.

In unserem Fall gibt die Funktion `REST(C3;{1;2;3;4;5;...;4804})` die Matrix aller Reste zurück, nämlich {0;1;1;1;...;3585}. In der Formel steht hinter der REST-Funktion `=0`, d.h. alle Zahlen in der Matrix der

Restwerte werden auf Gleichheit mit null geprüft, und es entsteht eine Matrix aus lauter WAHR- und FALSCH-Werten.

Durch die Multiplikation mit 1 werden die WAHR- und FALSCH-Werte für die SUMME-Funktion zu Einsen und Nullen konvertiert. Die SUMME-Funktion gibt also die Anzahl der Einsen und damit die Anzahl der Teiler zurück. Wenn $SUMME(. . .)=1$ ist, liegt eine Primzahl vor.

8.2 Bestimmung aller Teiler der Zahl

Die Teiler einer Zahl erhält man mit einer Matrixformel, die auf den ersten Blick aussieht wie eine Horror-Formel, und ich habe Verständnis für jeden, der an dieser Stelle nicht mehr weiterliest.

Die Formel lautet:

```
{=WENN(ZEILE()-5>SUMME(1*(REST($C$3; ZEILE(BEREICH.VERSCHIEBEN($A$1; ; ; GANZZAHL(WURZEL($C$3))))))=0); ""; KGRÖSSTE((REST($C$3; ZEILE(BEREICH.VERSCHIEBEN($A$1; ; ; GANZZAHL(WURZEL($C$3))))=0)* (ZEILE(BEREICH.VERSCHIEBEN($A$1; ; ; GANZZAHL(WURZEL($C$3))))); SUMMENPRODUKT(1*(REST($C$3; ZEILE(BEREICH.VERSCHIEBEN($A$1; ; ; GANZZAHL(WURZEL($C$3))))=0))-ZEILE(BEREICH.VERSCHIEBEN($A$1; ; ; GANZZAHL(WURZEL($C$3))))+1))}
```

	A	B	C	D	E	F
1		Primzahl-Test				
2						
3			31122000	ist Primzahl?	FALSCH	
4						
5			Alle Teiler der Zahl		Anzahl der Teiler	
6			1	31122000	480	
7			2	15561000		
8			3	10374000		
9			4	7780500		
10			5	6224400		
11			6	5187000		
12			7	4446000		
13			8	3890250		
14			9	3458000		
15			10	3112200		
16			12	2593500		
17			13	2394000		
18			14	2223000		
19			15	2074800		
20			16	1945125		
21			18	1729000		
22			19	1638000		
23			20	1556100		
24			21	1482000		
25			24	1296750		

Wer jedoch trotzdem weiterliest, wird feststellen, dass der äußere Umfang der Formel täuscht und die Formel gar nicht so komplex ist wie sie aussieht.

Sie wird sich, wenn man die äußere WENN-Funktion zum Abfangen von Fehlercodes erst mal weglässt, auf eine einfache KGRÖSSTE-Funktion reduzieren.

Sie hat also die Struktur $=WENN(. . . ; "" ; KGRÖSSTE(. . . ; . . .))$.

Die KGRÖSSTE-Funktion liefert dann die Teiler nach Größe sortiert.

Wie kommt man nun auf eine solche Formel?

8.2.1 Vorüberlegung mit praktischen Versuchen

Als Erstes überlegen wir, wie man die Teilersuche überhaupt (ohne Matrixformeln) anstellen könnte. Wir nehmen dazu als überschaubares Beispiel die Zahl 20.

Wir könnten die Zahl 20 der Reihe nach durch alle Zahlen von 1 bis 20 dividieren und schauen, bei welchen Divisoren der Rest null ist. Diese Divisoren sind dann die Teiler der Zahl 20.

Da die Teiler paarweise auftreten (bei 20 also 1·20, 2·10, 4·5), genügt es, diejenigen Divisoren zu prüfen, die kleiner oder gleich der Wurzel aus 20 sind, in unserem Fall die Zahlen von 1 bis 4. Die damit gepaarten Teiler kann man dann durch einfache Divisionen ZAHL/TEILER berechnen.

Damit das Ganze keine graue Theorie bleibt, lösen wir das kleine Beispiel mit der Zahl 20 auf dem Excel-Blatt.

In die Zelle C3 tragen wir die Zahl 20 ein.

In die Zellen G6:G9 tragen wir die Zahlen von 1 bis 4 ein. Das sind unsere Divisoren, bei denen geprüft werden soll, ob es sich um Teiler von 20 handelt.

G	H	I	J	K
Für diese kleine Beispieltabelle muss in C3 eine 20 eingetragen werden				
Divisor	=REST(\$C\$3;\$G6)	=REST(\$C\$3;\$G6)=0	=\$G6*(REST(...)=0)	Teiler
1	0	WAHR	1	1
2	0	WAHR	2	2
3	2	FALSCH	0	4
4	0	WAHR	4	

Da uns die REST-Funktion von Excel =REST(Zahl; Divisor) direkt den Rest der Division liefert, tragen wir in die Zelle H6 die Formel =REST(\$C\$3; \$G6) ein und kopieren sie nach unten bis zur Zelle H9. Wir erhalten die Reste {0;0;2;0}.

Achtung: Bei \$G6) steht vor der 6 kein Dollarzeichen!

In der nächsten Spalte wandeln wir die Reste in WAHR/FALSCH-Werte um. WAHR bedeutet dabei, dass hier ein Teiler vorliegt. In die Zelle I6 tragen wir die Formel =REST(\$C\$3; \$G6)=0 ein und kopieren sie nach unten bis zur Zelle I9.

In die Zelle J6 tragen wir die Formel =\$G6*(REST(\$C\$3; \$G6)=0) ein und kopieren sie nach unten bis zur Zelle J9. In der J-Spalte haben wir damit alle Divisoren, die Teiler von 20 sind. An den Plätzen der anderen wird eine Null erzeugt.

Nun kommt das finale Werk der KGRÖSSTE-Funktion: sie listet die Teiler der Größe nach auf - ohne die Nullen dazwischen. In die Zelle K6 tragen wir die Formel =KGRÖSSTE (\$J\$6:\$J\$9; 3) ein, in die Zelle K7 die Formel =KGRÖSSTE (\$J\$6:\$J\$9; 2) und in die Zelle K8 die Formel =KGRÖSSTE (\$J\$6:\$J\$9; 1).

Vermutlich hat dein Inneres bereits mit Kopfschütteln auf diese letzten drei Formeln reagiert, denn sie sind ganz auf die Zahl 20 zugeschnitten und versagen bereits, wenn wir die Zahl 25 in C3 eingeben. Der nächste Schritt muss also sein, diese Speziallösung zu verallgemeinern.

Als Erstes werden wir dazu den Übergang zu Matrixformeln betrachten.

Statt wie in der K-Spalte drei Einzelformeln einzugeben, markieren wir die drei Zellen L6:L9 und tragen die Formel `=KGRÖSSTE (J6:J9; {3;2;1})` ein. Diesmal schließen wir die Eingabe mit STRG-UMSCHALT-EINGABE ab.

Das Verhalten der Matrixformel in der L-Spalte ist genauso wie das der Formeln in der K-Spalte: Durch die Matrix $\{3;2;1\}$ wird in der ersten Zeile der drittgrößte Teiler ermittelt, in der zweiten Zeile der zweitgrößte und in der dritten Zeile der größte.

Unser Ziel ist es, zu einer Matrixformel zu gelangen, die ohne zwischengespeicherte Zahlenreihen auskommt.

Das ist ja gerade der Haupteffekt beim Einsatz von Matrixformeln, dass umfangreiche Zwischenergebnisse nicht auf dem Excel-Blatt gespeichert werden, sondern bei der Formelbewertung intern berechnet und gleich weiter verwertet werden.

Die Formel in der L-Spalte greift noch auf die Ergebnisse in der J-Spalte zurück. Deshalb schauen wir, wie diese Ergebnisse berechnet wurden und setzen den entsprechenden Rechenausdruck in unsere Matrixformel ein. In unserer Matrixformel setzen wir also an Stelle des Bezugs `J6:J9` den Rechenausdruck `$G6*(REST ($C$3; $G6)=0)` ein. - Doch Stopp!

Dieser Rechenausdruck berechnet nur den ersten Wert. Wir brauchen einen auf Matrixformeln zugeschnittenen Rechenausdruck, der eine Matrix von allen vier Werten in der J-Spalte zurückgibt! An zwei Stellen ersetzen wir deshalb in diesem Ausdruck das `$G6` durch `$G6:$G9`. Unsere neue Matrixformel heißt dann:

`=KGRÖSSTE ($G6:$G9*(REST (C3; $G6:$G9)=0); {3;2;1})`

Um die Formel zu testen, markieren wir die Zellen M6:M8, geben die Formel in die Bearbeitungszeile ein und schließen die Eingabe wieder mit STRG-UMSCHALT-EINGABE ab.

Wenn wir jetzt an beiden Stellen in der Formel den Bezug `$G6:$G9` noch durch die Matrix $\{1;2;3;4\}$ ersetzen, haben wir eine Matrixformel, die ganz ohne zwischengespeicherte Ergebnisse auskommt. Ihr einziger Bezug ist die Zelle C3, wo die zu prüfende Zahl steht.

Zum Testen der Formel markieren wir den Bereich N6:N9 und tragen die Formel `=KGRÖSSTE ({1;2;3;4} * (REST (C3; {1;2;3;4})=0); {3;2;1})` **<F0>** ein. Abschluss mit STRG-UMSCHALT-EINGABE!

So weit die Vorüberlegung. - Wir wissen jetzt, wie die Struktur der Matrixformel aussehen muss, nur darf sie eben nicht auf den Fall 20 spezialisiert sein.

8.2.2 Verallgemeinerung der bisherigen Matrixformel

Wenn wir die letzte Matrixformel betrachten, sehen wir, dass wir an zwei Stellen eine Zahlenfolge von 1 bis n erzeugen müssen, wobei n die diejenige ganze Zahl ist, die gerade noch kleiner als die Wurzel der zu prüfenden Zahl ist. Die Berechnung von n erledigt der Rechenausdruck `GANZZAHL (WURZEL (C3))`.

Die Erzeugung einer Zahlenreihe variabler Länge geschieht wieder mit `ZEILE (BEREICH.VERSCHIEBEN (A1; ; ; n))`. Die Erzeugung dieser Zahlenreihe wird im [Abschnitt 5.4](#) ausführlich erklärt. Für n setzen wir den Rechenausdruck `GANZZAHL (WURZEL (C3))` ein und erhalten: `ZEILE (BEREICH.VERSCHIEBEN (A1; ; ; GANZZAHL (WURZEL (C3))))` **<F1>**

Dieser Rechenausdruck erzeugt also eine Zahlenmatrix der Gestalt $\{1;2;3;\dots;n\}$ bis zu einer letzten Zahl n, die von der zu prüfenden Zahl abhängt. Für die Zahl 20 in unserer Vorüberlegung war $n = 4$.

Außer dieser Zahlenreihe (Zahlenmatrix) müssen wir nur noch eine weitere erzeugen: Am Ende der Formel müssen wir die Matrix $\{3;2;1\}$ ersetzen durch eine Zahlenfolge, die mit der Anzahl der Teiler unterhalb des

Wurzelwärts beginnt und dann abwärts läuft. Bei der Zahl 20 war die Anzahl der Teiler unterhalb der Wurzel aus 20 gleich 3.

Diese Anzahl 3 würden wir erhalten, wenn wir bei unserem kleinen Beispiel im vorherigen Abschnitt 8.2.1 die Anzahl der Nullen in der H-Spalte zählen würden. Diese Zählung geschieht mit der Matrixformel $=\text{SUMME}(1*(\text{REST}(\$C\$3; \{1;2;3;4\})=0))$.

N	O	P
	$=\text{SUMME}(1*(\text{REST}(\$C\$3; \{1;2;3;4\})=0))$	
Teiler verbessert	Anzahl der Teiler	
1	3	
2		
4		

Zum Verständnis dieser Formel kann man sich die Formelauswertung von Excel anzeigen lassen. Wir tragen die Formel als Matrixformel in die Zelle O6 ein. Die schrittweise Formelauswertung ergibt:

```
SUMME(1*(REST(20; {1;2;3;4})=0))
SUMME(1*({0;0;2;0}=0))
SUMME(1*({WAHR;WAHR;FALSCH;WAHR}))
SUMME({1;1;0;1})
3
```

Wenn wir den Ausdruck $\{1;2;3;4\}$ wieder wie oben ersetzen durch den verallgemeinerten Ausdruck $\text{ZEILE}(\text{BEREICH.VERSCHIEBEN}(\$A\$1; ; ; \text{GANZZAHL}(\text{WURZEL}(\$C\$3))))$, erhalten wir als Rechenausdruck für die Ermittlung der gesuchten Teileranzahl:

```
SUMME(1*(REST(\$C\$3; ZEILE(BEREICH.VERSCHIEBEN(\$A\$1; ; ; GANZZAHL(WURZEL(\$C\$3))))))=0)) <F2>
```

Um die absteigende Zahlenfolge $\{3;2;1\}$ am Ende der Formel $\langle F0 \rangle$ zu erhalten, bilden wir einen Rechenausdruck der Form $\text{Teileranzahl}-\{1;2;3;\dots;n\}+1$.

Wenn die Teileranzahl zum Beispiel gleich 15 ist, wird der Rechenausdruck dann zu $\{15;14;13;12;11;\dots\}$ ausgewertet.

Für Teileranzahl setzen wir den soeben entwickelten Rechenausdruck $\langle F2 \rangle$ ein und für $\{1;2;3;\dots;n\}$ den Rechenausdruck $\langle F1 \rangle$. Wir erhalten den Rechenausdruck für die absteigende Zahlenfolge:

```
SUMME(1*(REST(\$C\$3; ZEILE(BEREICH.VERSCHIEBEN(\$A\$1; ; ; GANZZAHL(WURZEL(\$C\$3))))))=0))-ZEILE(BEREICH.VERSCHIEBEN(\$A\$1; ; ; GANZZAHL(WURZEL(\$C\$3))))+1 <F3>
```

Jetzt sind wir soweit, dass wir in der Formel $\langle F0 \rangle$ aus Abschnitt 8.2.1 die drei Zahlenreihen (rot) durch die verallgemeinerten Rechenausdrücke $\langle F1 \rangle$ und $\langle F2 \rangle$ ersetzen können.

Die Formel

```
=KGRÖSSTE({1;2;3;4}*(REST(\$C\$3; {1;2;3;4})=0); {3;2;1})
```

wird zu:

```
=KGRÖSSTE(<F1>*(REST(\$C\$3; <F1>)=0); <F3>)
```

Den unübersichtlichen vollständigen Text dieser Formel ersparen wir uns an dieser Stelle.

Das könnte nun die Endformel sein. Aber auf Grund der Tatsache, dass die Liste der Divisoren <F1> (im Beispiel {1;2;3;4}) länger ist als die Liste der Teiler, geht die absteigende Liste <F3> (im Beispiel {3;2;1}) ab einem bestimmten Punkt auf null und in die Minuszahlen. Die Funktion KGRÖSSTE gibt dann den Fehlercode #NV (nicht validierbar) aus.

Dieser Fehlercode könnte noch mit WENNNV abgefangen werden. Die Funktion WENNNV hat jedoch keine Wirkung, wenn ein #NV-Fehler ausgegeben wird, weil der Zielbereich der Matrixformel zu groß gewählt wurde. Damit aber auch bei Zahlen mit vielen Teilern (z.B. 31122000 mit 480 Teilern) alle Teiler ausgegeben werden, müssen wir den Zielbereich groß genug wählen, obwohl bei der nächsten Zahl 31122001 zum Beispiel nur noch 4 Teiler auftreten.

Um in diesem Fall nicht Hunderte von #NV-Fehlercodes auf dem Excel-Blatt zu haben, setzen wir vor die bereits sehr unübersichtliche Gesamtformel noch eine WENN-Funktion:

```
=WENN (ZEILE()-5>AnzahlTeiler; ""; ...)
```

Da die erste Zeile des Zielbereichs unserer Matrixformel die Zeile 6 ist, muss von ZEILE() die Zahl 5 subtrahiert werden, damit die Zählung bei 1 beginnt. Verschiebt man den Zielbereich nach oben oder nach unten, muss die Zahl 5 entsprechend angepasst werden.

Bei der Zahl 31122000 zum Beispiel steht der letzte gefundene Teiler (der 240.) in Zeile 245; in der Zeile 246 wird der Fehlercode #NV erzeugt. Da ZEILE() - 5, also 246 - 5, größer als 240 (Anzahl der Teiler) ist, sorgt die WENN-Funktion dafür, dass ein leerer String ("") in die Zelle geschrieben wird.

Unsere finale Gesamtformel sieht also so aus:

```
=WENN (ZEILE()-5 > <F2>; ""; KGRÖSSTE (<F1> * (REST ($C$3; <F1>) = 0); <F3>))
```

Oder - weil es so prächtig aussieht - nochmal im vollständigen Text:

```
{=WENN (ZEILE()-5 > SUMME (1*(REST ($C$3; ZEILE (BEREICH.VERSCHIEBEN ($A$1; ; ; GANZZAHL (WURZEL ($C$3))))=0)); ""; KGRÖSSTE ((REST ($C$3; ZEILE (BEREICH.VERSCHIEBEN ($A$1; ; ; GANZZAHL (WURZEL ($C$3))))=0) * (ZEILE (BEREICH.VERSCHIEBEN ($A$1; ; ; GANZZAHL (WURZEL ($C$3))))); SUMMENPRODUKT (1*(REST ($C$3; ZEILE (BEREICH.VERSCHIEBEN ($A$1; ; ; GANZZAHL (WURZEL ($C$3))))=0))-ZEILE (BEREICH.VERSCHIEBEN ($A$1; ; ; GANZZAHL (WURZEL ($C$3))))+1))}
```

In die Zelle D6 tragen wir die konventionelle Formel =WENNFehler (\$C\$3/C6; "") ein und kopieren sie nach unten bis zur Zelle D305. Im Bereich C6:D305 können maximal 600 Teiler angezeigt werden. Da im Eingabefeld als größte Zahl 31129999 zugelassen ist, reicht dies in jedem Fall aus. In diesem Zahlenbereich können maximal 480 Teiler auftreten.

