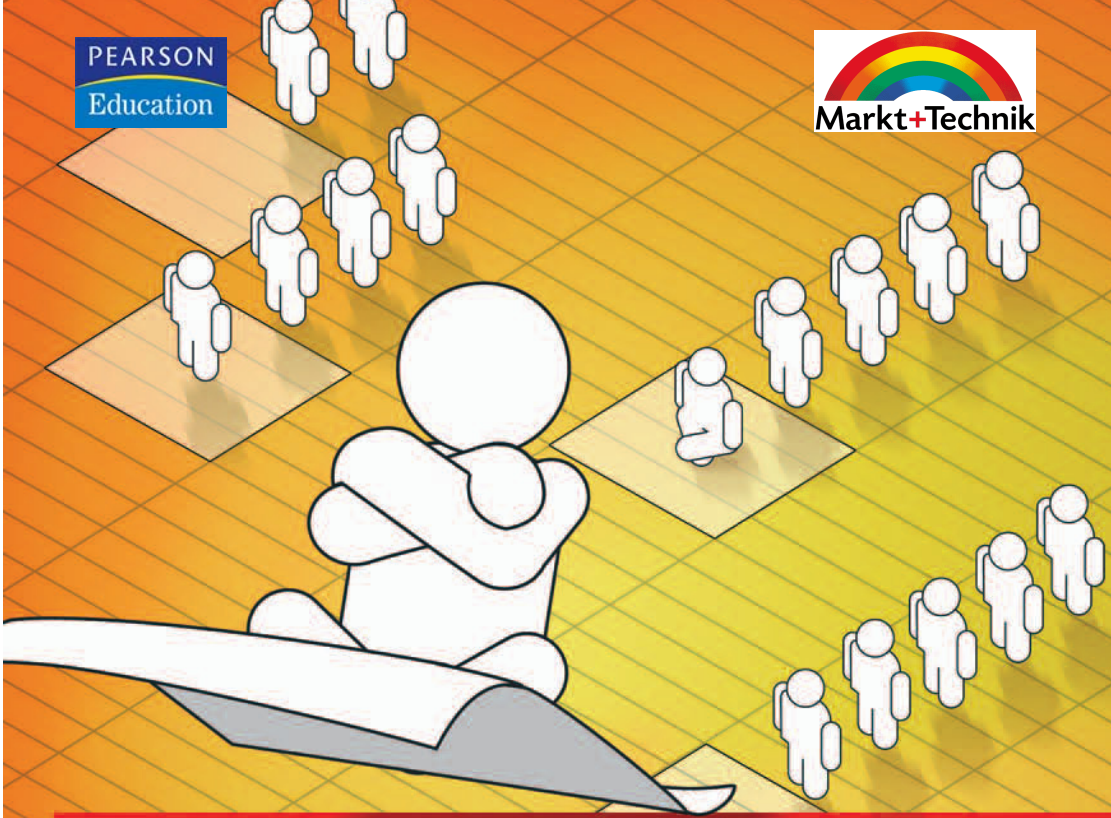


PEARSON
Education


Markt+Technik



Excel

Das Zauberbuch

raffinierte Zaubereien
für Excel-Kenner

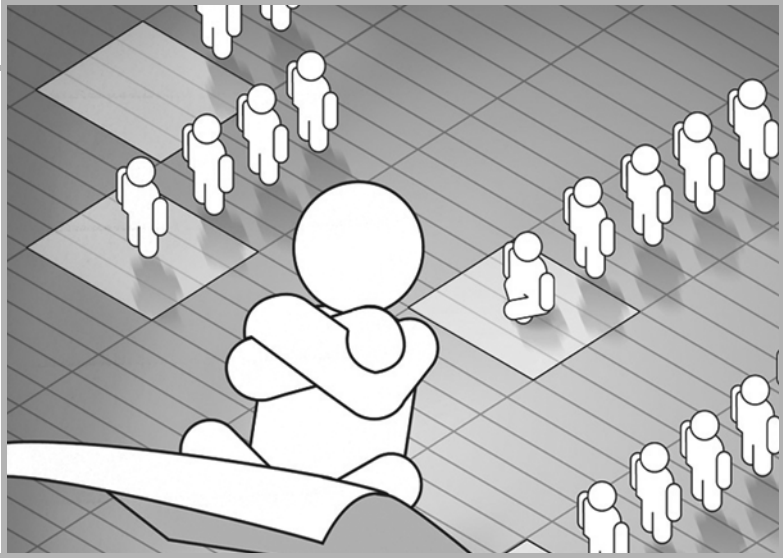
J. FLECK W. FRICKE B. GEORGI


Markt+Technik



KAPITEL 3

Anti-Add-in-Analyse- funktionen



Excel bietet standardmäßig 226 Funktionen an. Seit vielen Excel-Versionen blieb diese Zahl leider konstant. Der Funktionskatalog wird um so genannte Analysefunktionen ergänzt, die man optional installieren kann. Diese sind nützlich und rechnen richtig – falls sie denn rechnen. Aktiviert werden sie über den Menüpunkt *Extras>Add-Ins* (Abbildung 3.1).



Abbildung 3.1: Dialog *Extras>Add-ins...*

3.1 Warum wir Add-in-Funktionen nicht mögen

Immer wieder sehen wir in Excel-Foren Postings von Add-in-geplagten Excel-Usern:

„Hallo an das Forum,

folgende Frage:

In einer Excel-Datei tauchen an (sehr vielen) bestimmten Stellen Fehlermeldungen auf (#Name?). Die Ursache des Fehlers kann ich nicht feststellen, die verwendete Funktion ist in der Version bekannt und unter „Einfügen>Funktion“ wird der korrekte Endwert angezeigt, obwohl die Zelle die Fehlermeldung auswirft.

Ich habe jetzt herausgefunden, dass die Fehlermeldung verschwindet, wenn ich in die jeweilige Zelle gehe, über „Einfügen>Funktion“ in die Bearbeitung der Formel gehe und diese dann über OK wieder schließe – obwohl ich nichts verändert habe. Da es sich aber um hunderte von Zellen handelt, ist es unmöglich, das auf diese Weise für alle Fehlermeldungen zu machen.

Hat jemand eine Idee?“

Wenn Sie alleine mit Ihrem Rechner ohne Internetzugang auf einer einsamen Insel hocken, sind diese Analysefunktionen ungefährlich. Kommunizieren Sie aber auch mit anderen Rechnern (Mail, Firmennetzwerk etc.), kommt es häufig vor, dass dort das Ergebnis Ihrer umfangreichen Kalkulation auf ein niederschmetterndes #NAME? reduziert wird. Excel will jetzt nicht Ihren Namen wissen, sondern teilt auf seine ureigene Art mit, dass es mit den verwendeten Analysefunktionen nichts anfangen kann. Das kann zwei Ursachen haben:

1. Die Analysefunktionen sind nicht aktiviert.
2. Das Add-in ist in einer anderen Sprache. Selbst wenn auf beiden Rechnern die gleichsprachige Excel-Version installiert ist, können Sie sich nicht sicher sein, dass die Add-in-Sprache auch identisch ist.

Was kann man dagegen tun?

Im ersten Fall müssen Sie im Normalfall nur die Add-in-Funktionen aktivieren und eine Neuberechnung durchführen.

Im zweiten Fall eigentlich nichts, außer Sie besorgen sich das anderssprachige Add-in.

Aber auch der erste Fall birgt eine böse Falle. Es ist nämlich wichtig, dass die Add-in-Funktionen aktiv sind, bevor Sie diese in eine Zelle eingeben. Wenn Sie dies nicht tun, Add-in-Funktionen in der Tabelle platzieren und #NAME? zunächst ignorieren, wird bei nachträglichem Aktivieren dieser unschöne Fehlerwert wacker die Stellung halten. Eine Neuberechnung mit **F9** bleibt erfolglos, selbst die erweiterte Berechnung mit **Strg + Alt + F9** geht ins Leere (die zweite Tastenkombination erzeugt eine erweiterte Berechnung der vollständigen Arbeitsmappe inklusive aller VBA-Funktionen – analog der CalculateFull-Methode in VBA). Selbst Beenden und Neuaufruf der Datei oder Beenden und Neustart von Excel bleiben wirkungslos. Die einzige Abhilfe ist das Editieren der entsprechenden Zelle und die Bestätigung mit **↵**.

Stellen Sie sich vor, Sie benutzen deutsche Add-ins und schreiben für einen Kunden eine Excel-Datei, von dem Sie wissen, dass er englische Add-ins besitzt. Jetzt sind Sie besonders pfiffig und übersetzen Ihre Add-in-Funktionen, die Sie in Ihren Formeln verwenden, direkt ins Englische. Dann erhalten Sie zwar #NAME?, aber das stört Sie ja nicht. Ziel ist, dass der Engländer nach Erhalt der Datei korrekte Berechnungen vorfindet. Sehr schöner Ansatz, aber mit dem Abschluss wird es leider nicht klappen. Er wird vergeblich **F9** drücken und kann die Formeln nur zum Leben erwecken, wenn er jede einzelne Zelle reanimiert (sprich: editiert).

Übrigens haben die Excel-Standardfunktionen mit unterschiedlichen Sprachen überhaupt kein Problem. Sie werden – außer bei Textargumenten wie z.B. =ZELLE("Dateiname") oder =INDIREKT("Z1S1";0) – automatisch übersetzt.

Die beste Lösung für alle Probleme: keine Analysefunktionen verwenden!

3.2 Wie Sie Add-in-Funktionen vermeiden können

Für viele Add-ins gibt es, zum Teil sehr einfache, Pendanten in den Standardfunktionen. Diese sind nachfolgend aufgelistet. Komplexere Lösungen zur Umgehung von Add-in-Funktionen finden Sie auch noch in anderen Kapiteln dieses Buches.

| Add-in-Funktion | Standardformel-Ersatz |
|--|----------------------------------|
| Jahre mit Nachkommastellen zwischen zwei Datumsangaben | |
| =BRTEILJAHRE(A1;B1;2) | =(B1-A1)/360 |
| Wenn Zahlen in zwei Zellen identisch sind, wird 1 ausgegeben – ansonsten 0 | |
| =DELTA(A1;B1) | =(A1=B1)*1 |
| Gibt 1 aus, wenn die erste Zahl größer als die zweite ist | |
| =GGANZZAHL(A1;B1) | =(A1>B1)*1 |
| Prüft, ob eine Zahl gerade ist | |
| =ISTGERADE(A1) | =REST(KÜRZEN(A1);2)=0 |
| Prüft, ob eine Zahl ungerade ist | |
| =ISTUNGERADE(A1) | =REST(KÜRZEN(A1);2)=1 |
| Gibt das Monatsende eines Datums (A1) nach x Monaten (B1) aus | |
| =MONATSENDE(A1;B1) | =DATUM(JAHR(A1);MONAT(A1)+1+B1;) |
| Gibt den ganzzahligen Teil einer Division aus | |
| =QUOTIENT(A1;B1) | =KÜRZEN(A1/B1) |
| Gibt eine auf das gewünschte Vielfache gerundete Zahl aus | |
| =VRUNDEN(A1;B1) | =RUNDEN(A1/B1;0)*B1 |

| Add-in-Funktion | Standardformel-Ersatz |
|--|--|
| Liefert die Fakultät einer Zahl mit Schrittlänge 2 | |
| =ZWEIFAKULTÄT(A1) | {=PRODUKT(A1-ZEILE(INDIREKT("1:"&AUFRUNDEN(A1/2;0)))*2+2)} |
| Liefert eine ganze Zufallszahl im festgelegten Bereich | |
| =ZUFALLSBEREICH(vonZahl;bisZahl) | =KÜRZEN(ZUFALLSZAHL()*bisZAHL)+vonZahl |
| Gibt die Wurzel aus einer mit Pi multiplizierten Zahl zurück | |
| =WURZELPI(A1) | =(A1*Pi())^0,5 |

Tabelle 3.1: Beispiele zu Add-In-Funktionen

Wie Sie sehen, verbergen sich hinter den genannten Add-ins großteils Banalitäten. Manch eine Analysefunktion kann auch aufgrund ihrer Bezeichnung zu missverständlichen Ergebnissen führen. Beispielsweise liefert die Funktion ISTGERADE für die Zahl 4,63 im Ergebnis WAHR, was erst nach einem gründlichen Lesen der Excel-Hilfe verständlich wird, da die Funktion keine Nachkommastellen berücksichtigt. Andere Analysefunktionen haben nur dann eine Berechtigung, solange Sie einerseits die bereits genannten Fehlerquellen ausschließen können oder aber einfach nur auf die Schnelle eine Berechnung durchführen möchten. Als Beispiel sei die Funktion NETTOARBEITSTAGE genannt, deren Nachbau mit den Standardfunktionen zwar möglich, aber sehr kompliziert ist.

=NETTOARBEITSTAGE(A1;A2;C1:C20)

kann ersetzt werden durch die Arrayformel (mit  +  +  abschließen):

{=SUMME((WOCHENTAG(ZEILE(INDIREKT(A1&"."&A2));2)<6)*1)-
SUMME(WENN((C1:C20>=A1)*(C1:C20<=A2)*(WOCHENTAG(C1:C20;2)<6);1;0)))}

A1 enthält das Startdatum, A2 das Enddatum und C1:C20 auszuschließende Feiertage.

Zu guter Letzt gibt es aber auch Analysefunktionen, die dermaßen leistungsstark sind, dass sie mit Bordmitteln kaum oder gar nicht nachzustellen sind. Beispielhaft sei hier die Funktion GGT genannt, die den größten gemeinsamen Teiler aus bis zu 30 Zahlen ermittelt. Das mit Standardfunktionen nachzubauen ist möglich – aber beschränkt auf 10.922 für die kleinste Zahl und ziemlich kompliziert:

{=WENN(UND(REST(B1:E1;A1)=0);A1;MAX(WENN(MMULT(REST(A1:E1;ZEILE(INDIREKT("1:"&GANZZAHL(\$A\$1/2)))));{1;1;1;1;1})=0;ZEILE(INDIREKT("1:"&GANZZAHL(\$A\$1/2))))))}

Da macht =GGT(A1:E1) doch 'nen richtig schlanken Fuß und ist außerdem leistungsstärker.

Aber: Diese Analysefunktionen haben trotzdem ein Rad ab!

Gibt man eine „normale“ Funktion ohne jeglichen Bezug in eine Zelle ein, erhält man beispielsweise bei =SUMME: :#NAME?

Bei =NETTOARBEITSTAGE oder =QUOTIENT oder =WURZELPI erhält man z.B. tatsächlich „Ergebnisse“: -1.335.885.768 und -643.039.222 und -1.707.081.719.

Das eben gelobte GGT ergibt bei Eingabe von =GGT -839.712.728 (sehr logisch!).

DIE SPINNEN, DIE RÖMER!

Unschön ist außerdem, dass die Add-ins überhaupt keine Kontrolle über eine valide Eingabesyntax haben. Wenn man bei Standardfunktionen zu viele oder zu wenige Argumente eingibt, erscheint eine Fehlermeldung (*Abbildung 3.2*).

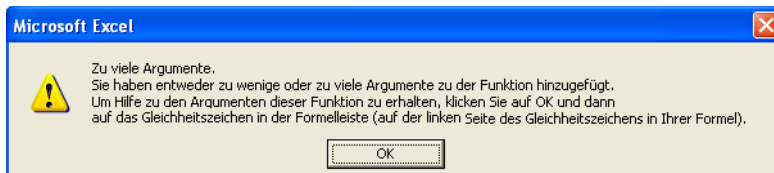


Abbildung 3.2: Fehlermeldung aufgrund der Argumentenanzahl in Funktionen

Enthalten die Argumente verkehrte Datentypen, ist der resultierende Fehlerwert eigentlich immer sinnvoll.

Nicht so bei den Add-ins:

=QUOTIENT() =#NV

=QUOTIENT(1;2;3) = #WERT!

Aber man soll ja versuchen, in allem etwas Positives zu sehen. Falls Sie einmal nicht sicher sind, ob eine Funktion ein Add-in oder Standard ist, schreiben Sie den Namen (mit Gleichheitszeichen) in eine Zelle. Wenn Sie eine abstruse Zahl sehen, ist es eine Add-in-Funktion.

=WENN(ISTZAHL(DEZINBIN);"Hallo, ich bin ein Add-in";"")

3.3 Funktionsdolmetscher

Wenn Sie übrigens häufig sowohl deutsche als auch englische Funktionsnamen benötigen, gibt es zwei Tricks, wie Sie sich einen ständig verfügbaren Funktionsdolmetscher einrichten können.

Wenn Sie den englischen Namen einer Funktion nicht kennen, können Sie ja die deutsche Funktion eingeben und dabei den Makrorekorder laufen lassen. In dem aufgezeichneten Makro finden Sie dann die englische Bezeichnung. Dies gilt nur für die Standardfunktionen, bei den Add-in-Funktionen klappt das nicht, dort wird die deutschsprachige Bezeichnung aufgezeichnet.

In der Datei *vbaliste.xls* werden auch englische und deutsche Funktionsnamen gegenübergestellt. Vielleicht wollen Sie diese Datei aber nicht jedes Mal suchen bzw. öffnen. Es gibt aber einen recht originellen Speicherplatz, an dem Sie sich die Funktionen der *vbaliste.xls* im Hintergrund bereitstellen können, und zwar als benutzerdefinierte Liste. Ihnen ist sicherlich bekannt, dass eine Zelle mit dem Inhalt „Montag“ nach unten gezogen in der nächsten Zelle „Dienstag“ ergibt? Das liegt daran, dass die Wochentage (und Monate) in *Benutzerdefinierten Listen* gespeichert sind. Diese finden Sie unter *Extras>Optionen>Benutzerdefinierte Listen* – in früheren Excel-Versionen hieß die Funktion *AutoAusfüllen* (Abbildung 3.3):

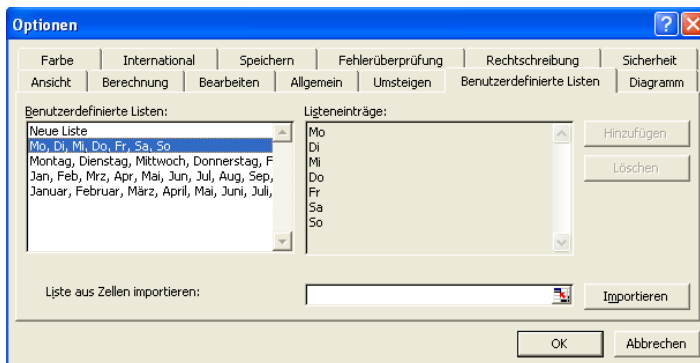


Abbildung 3.3: Dialog Optionen – Benutzerdefinierte Listen

Die Funktionen der Datei *vbaliste.xls* können Sie nun auch als Listen hinterlegen. Für jede Funktion wird eine Liste importiert, die jeweils aus zwei Einträgen besteht – der deutschen Bezeichnung und der englischen Bezeichnung. Und so geht's:

Geben Sie in das Feld *Liste aus Zellen importieren* den Bereich ein, in dem die deutschen und englischen Funktionsnamen stehen. In der Originaldatei *vbaliste.xls* müsste das A4:B229 sein. Dann drücken Sie auf den *Importieren*-Button. Jetzt kommt das wichtigste, Sie müssen die Listen aus Zeilen importieren (*Abbildung 3.4*):



Abbildung 3.4: Dialog Benutzerdefinierte Listen

Falls Sie Spalten anklicken, wird eine Liste für die deutschen Funktionen und eine für die englischen Funktionen erstellt. Das ist vielleicht auch ganz nützlich, aber nicht für das, was wir vorhaben. Jedenfalls dann, wenn Sie *Zeilen* ausgewählt haben, müsste das Ergebnis wie in *Abbildung 3.5* aussehen:

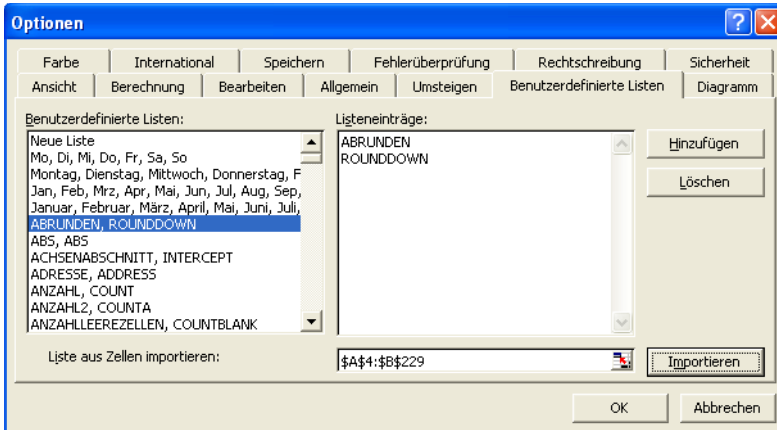
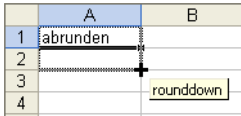


Abbildung 3.5: Dialog Optionen – Benutzerdefinierte Listen mit Excel-Funktionen

Wenn Sie jetzt in eine Zelle „Abrunden“ schreiben (ohne Gleichheitszeichen) und die Zelle eine Zelle nach unten ziehen, erhalten Sie die englische Bezeichnung (*Abbildung 3.6*):



| | A | B |
|---|----------|-----------|
| 1 | abrunden | |
| 2 | | |
| 3 | | rounddown |
| 4 | | |

Abbildung 3.6: Beispiel für Funktionsdolmetscher per Benutzerliste

Das funktioniert auch in die andere Richtung sowie nach links und rechts. Die benutzerdefinierten Listen stehen immer der Anwendung auf dem aktuellen Rechner zur Verfügung, sind also nicht an eine Arbeitsmappe gebunden. Logischerweise können Sie so auch andere Begriffe verwalten, die Sie immer schnell griffbereit haben wollen.