

Funktionen für Anweisungen

Inhalt

- `$IncDate([Datum],[+Tage],[+Monate])`
- `$Calc(<Formel>,<Formatanweisung>,<Nachkommastellen>,<Dezimal Separator>,<Tausender Separator>)`
- `$Mid(string, start, anzahl)`
- `$IncWhour()`
- Eingangs-Parameter und Ausgabe
- Besonderheiten bei der Eingabe der Stunden innerhalb der Formel
- Stunden anteilig berechnen
- IDs der Arbeitszeit des jeweiligen Teams nutzen
- Funktionen "\$Min" und "\$Max": Minimalen und maximalen Wert ermitteln
- Zweck der Funktionen "Min" und "Max"

L

`$IncDate([Datum],[+Tage],[+Monate])`

Dadurch können zu einem Datum sowohl Tage als auch Monate addiert werden.



Bitte beachten Sie:

"Tage" darf auch Null und negativ sein (negative Werte werden in Abzug gebracht)

"Monate" darf auch Null und negativ sein (negative Werte werden in Abzug gebracht)

Wird nur `IncDate([Datum])` angegeben, so wird genau ein Tag zum Datum addiert.

`$Calc(<Formel>,<Formatanweisung>,<Nachkommastellen>,<Dezimal Separator>,<Tausender Separator>)`

Mit der Funktion `$Calc()` können Sie die verschiedensten Berechnungen durchführen.

Mit dieser Funktion können in Feldzuweisungen Zahlenwerte berechnet werden, wobei die vier Grundrechenarten (addieren +, subtrahieren -, multiplizieren *, dividieren /) verwendet werden können. (siehe auch Beispiel "Berechnen des Einzelgewichtes einer Vorgangsposition" im [Kapitel: Regeln mit der Auswertungsposition "Vor dem Speichern \(nach einer Neuanlage, Änderung oder beim Import\)"](#))

Es ergeben sich folgenden mögliche Werte:

- `<Formel>`: String Ausdruck von Werten die berechnet werden +-* / (Punkt-vor-Strich)
- `<Formatanweisung>`: ohne Angabe: 'Betrag' ansonsten 'Menge', 'Gewicht' oder 'Format'
- `<Nachkommastellen>`: Anzahl, wie viele Nachkommastellen verwendet werden, maximal: 9
- `<Dezimal Separator>`: nur wirksam bei Formatanweisung: 'Format'. Hier wird der Dezimal Separator angegeben (nur ein Zeichen).
- `<Tausender Separator>`: nur wirksam bei Formatanweisung: 'Format'. Hier wird der Tausender Separator angegeben (nur ein Zeichen).

Wird der Parameter Format ohne den Parameter `<Tausender Separator>` verwendet, erfolgt die Ausgabe ohne Tausender Trennzeichen.

Beispiele:

- `$Calc(«Vsd.Gew»*3,"",0)`
 - In diesem Beispiel wird das im Versanddatensatz angegebene Gewicht mit der Zahl 3 multipliziert. Das Ergebnis wird kaufmännisch gerundet und ohne Nachkommastellen dargestellt.
- `$Calc(«Variable»+1.55,Format,4,.) = 1234.1234`
 - Freie Formatierung ohne Tausender Separator.
- `$Calc(«Variable»+1.55,Format,5,.,",") = 1,234.12345`
 - Freie Formatierung mit Tausender Separator.



Bitte beachten Sie:

Im letzten Beispiel ist der letzte Parameter ein , (Komma) womit es eigentlich einen weiteren Parameter darstellen würde. Um dennoch ein , (Komma) als Parameter übergeben zu können, wurde das , (Komma) durch zwei " separiert = " , "

M

\$Mid(string, start, anzahl)

In "string" werden ab "start" "anzahl" Zeichen zurückgegeben.

Beispiel:

`$Mid(testen, 2, 3)`

Man erhält: est

In "Testen" werden ab der 2. Position 3 Zeichen ausgegeben.

L

Wie kann man in einer Artikelbezeichnung Text ersetzen?

Funktionsaufbau: `$Replace(Text,Suchtext,Ersetzungstext)`

In den Regeln kann mit dieser Funktion ein Wert in einem Feld ersetzt werden.

Beispiel:

In der Artikelbezeichnung soll der Wert "Auskünfte" gegen "Rückfragen oder eine Beratung" ersetzt werden.

In der Regel für Artikel ist die Auswertungsposition "Suchen und Ersetzen" auszuwählen. Es sollte noch eine Bedingung hinterlegt werden, dass es nur die Artikel betrifft, welche in der Artikelbezeichnung 1 den Wert "Auskünfte" haben. Diese Formel könnte z.B. lauten:

`Pos('Auskünfte',ToText(«Art.Bez1»)) >0`

Anschließend ist eine Anweisung zu erstellen, welche das Feld "Bezeichnung1" enthält.

In der Art der Feldzuweisung wird "Feldwert über Formel füllen" eingestellt.

Als Vorgabewert wird folgende Formel hinterlegt:

`$Replace(«Art.Bez1»,Auskünfte,Rückfragen oder eine Beratung)`

Beachten Sie: Die Formatierung im Artikel-Text kann nicht beibehalten werden.



Beachten Sie:

Die Replace-Funktion unterscheidet nicht zwischen Groß- und Kleinschreibung beim Ersetzen.

Beispiel: Bei einer Replace-Funktion: `$Replace(«Art.Bez1»,Ä,ae)`, die den Großbuchstaben "Ä" durch "ae" ersetzen soll, wird ebenfalls auch der Kleinbuchstabe "ä" durch ein "ae" ersetzt.

M

Wie kann in einer Feldzuweisung mit einem Datum gerechnet werden?

`$IncDate(Datum,Wert)`

Es werden zum Datum die bei "Wert" angegebenen Tage addiert.

Beispiel:

Mit `$IncDate(«Vsd.Dat»,2)` werden auf das Versanddatum 2 Tage addiert.

Wie kann in einem Vorgang ein frühester und spätester Liefertermin angegeben werden?

In einer Vorgangsposition wird bei einem Artikel ein Lieferdatum hinterlegt. Da sich das Lieferdatum in einem Zeitraum von plus/minus einem Monat bewegt, wird hierfür ein Selektionsfeld für den frühesten Termin und ein Selektionsfeld für den spätesten Termin angelegt. Über eine Regel mit Feldzuweisung für Vorgangspositionen wird dieses Datum mittels folgenden Formeln ermittelt:

Formel für den frühesten Termin: `$IncDate(«VPo.LiefDat»,-1)`

Formel für den spätesten Termin: `$IncDate(«VPo.LiefDat», , 1)`

Wie kann man die Informationen aus den Selektionen des Ansprechpartners in die Positionen des Vorgangs übergeben?

Möchte man mit der Regel-Erweiterung "Feldänderung während der Vorgangserfassung" (NE-1081) in ein Positionsselektionsfeld (Selektion 1) den Inhalt eines Selektionsfeldes (auch Selektion 1) aus dem Ansprechpartner der Rechnungsanschrift des Vorgangs eintragen lassen, so kann dies beispielsweise mit folgender Formel realisiert werden:

`$DBInfo(Ansprechpartner,IX;AdrNrAnsNrNa;3,«Vog.AdrNr»,«Vog.AnsNr»,«Vog.Re.Asp.Ans»,Sel1)`

Beachten Sie:

- Es wird ein abweichender Index übergeben.
- Der Name des Ansprechpartners muss eindeutig sein.



Soll eine `$DBInfo`-Formel in einer `$IF`-Abfrage mit einem String verglichen werden, muss die `$DBInfo`-Formel zwischen Hochkomma stehen.

Beispiel:

Der Inhalt eines Vertreter-Selektionsfeldes soll abgefragt und mit einem Text verglichen werden:

`$if('$DBInfo(Vetreter,«Vog.VtrNr»,Sel1)' = 'bis 50% Rabatt' ,«VPo.ProvSz»,')`



Wie lassen sich Reaktionszeiten in Abhängigkeit zu den Arbeitszeiten einer Firma berechnen?

`$IncWhour()`

Über die Funktion: **\$incWhour** lässt sich die Zeit bestimmen, bis wann eine Reaktion in Abhängigkeit von Arbeitszeit und Feiertagen zu erwarten ist. So kann z. B. anhand des Erstelldatums eines Vorgang- oder Projekt-Datensatzes eine Reaktionszeit errechnet werden. Neben der Berechnung der Reaktionszeit anhand der Arbeitszeit ([Öffnungs- und Arbeitszeiten](#)), werden auch die [Feiertage](#) berücksichtigt. Sollten Ihre Teams verschiedene Arbeitszeiten besitzen, können die Arbeitszeiten des Teams, anhand der ID dieser Datensätze ausgewählt werden.



Beachten Sie:

- Der Wert wird **immer in der Einheit: "Stunden"** interpretiert
- Wir empfehlen die Werte der Reaktionszeit in **"Anführungszeichen"** zu setzen, damit die Werte innerhalb des Strings korrekt / eindeutig interpretiert werden können. Dies ist insbesondere dann relevant, wenn **mehrere IDs** für die Arbeitszeiten bestehen

Eingangs-Parameter und Ausgabe

1. Startdatum mit der Zeitangabe, von welcher aus gerechnet wird
2. Stunden, die berechnet werden sollen
3. Optional kann auch die **ID der zu beachtenden Öffnungs- und Arbeitszeiten-Tabelle** angegeben werden - so kann z. B. nach Abteilungen unterschieden werden: [Öffnungs- und Arbeitszeiten](#).

Eingangs-Parameter:

```
IncWhour([Ausgangs Datum mit Uhrzeit im Format TT.MM.JJJJ SS:MM:SS], [Anzahl der Stunden], [Optional: ID der Öffnungszeit])
```

Ausgabe:

```
Datum & Uhrzeit im Format TT.MM.JJJJ SS:MM:SS
```

Besonderheiten bei der Eingabe der Stunden innerhalb der Formel

Die Anzahl der Stunden, welche auf einen Zeitpunkt addiert werden sollen, lassen sich wie folgt in der Formel erfassen:

- Verwendung des mittels \$IncWHour mitgegebenen Wertes z. B. mit oder ohne **Komma-Trennzeichen**: "1,0" bzw. "1" = **1 Stunde**

```
$IncWHour(«Prj.ErstDat», "1,0")
```

```
$IncWHour(«Prj.ErstDat», "1")
```



Beachten Sie:

Es ist wichtig, dass bei Eingabe des Wertes das **Komma-Trennzeichen** gesetzt wird und **nicht der Punkt!** In der Software wird der Punkt als Tausender-Separator interpretiert. Nutzen Sie deshalb bitte das Komma-Trennzeichen.

Stunden anteilig berechnen

Das Addieren von Stunden stellt die einfachste Methode dar, eine Reaktionszeit zu berechnen. Weiterhin lassen sich die Stunden auch anteilig übergeben.

Beispiele:

- \$IncWHour(«Prj.ErstDat», "0,5") für 0,5 Stunden (**30 Minuten**)
- \$IncWHour(«Prj.ErstDat», "0,4") steht für 0,4 Stunden (60 x 0,4 = **24 Minuten**)

IDs der Arbeitszeit des jeweiligen Teams nutzen

Besitzen Ihre Teams unterschiedliche Arbeitszeiten, muss auch die korrekte ID des Öffnungszeiten-Datensatzes mitgegeben werden

Beispiele:

- \$IncWHour(«Prj.ErstDat», "0,5", "1") für 0,5 Stunden (30 Minuten) der Arbeitszeiten mit ID=1
- \$IncWHour(«Prj.ErstDat», "0,5", "2") für 0,5 Stunden (30 Minuten) der Arbeitszeiten mit ID=2

XXL

Funktionen "\$Min" und "\$Max": Minimalen und maximalen Wert ermitteln

Die Funktionen "Min" und "Max" stehen in **Formeln für Feldzuweisungen** zur Verfügung. Bei Min & Max dürfen nur 2 Werte angegeben werden. Die zwei Übergabewerte sind der Formel nach folgendem Aufbau mitzugeben:

- **\$Min(Übergabewert1,Übergabewert2)**
- **\$Max(Übergabewert1,Übergabewert2)**



Info:

Der **Rückgabewert** ist jeweils **einer der beiden Übergabewerte**.

Die Funktionen arbeiten folgendermaßen:

- **"\$Min"** liefert jeweils **den kleinsten aller Werte zurück** - vergleichbar mit: **\$IF(A>B,A,B)**
- **"\$Max"** liefert jeweils **den größten aller Werte zurück** - vergleichbar mit: **\$IF(A<B,A,B)**

Zweck der Funktionen "Min" und "Max"

Diese Funktionen können genutzt werden, um etwa die Komplexität von Formeln an verschiedenen Stellen zu reduzieren.

Sind die genutzten Werte A und B von der Funktion nicht einfache Variablen, sondern hochkomplexe Formeln, lässt sich mit der Verwendung dieser Funktionen die Lesbarkeit und Wartbarkeit der Formel verbessern, beispielsweise wenn die Formel länger ist, um z. B. Berechnungen in **Freien Datenbanktabellen** durchzuführen. Die Funktion lässt sich natürlich auch für einfachere Beispiele nutzen.

**Beachten Sie:**

Die Implementierung der Funktionen Min & Max ist **nicht in den Filtern / Bedingungen** erlaubt.

Die Funktion steht **exklusiv für Formeln** zur Verfügung **und** kann auch für **verzweigte Bedingungen** genutzt werden, da die verzweigten Bedingungen Formeln abarbeiten und dann das Ergebnis vergleichen.

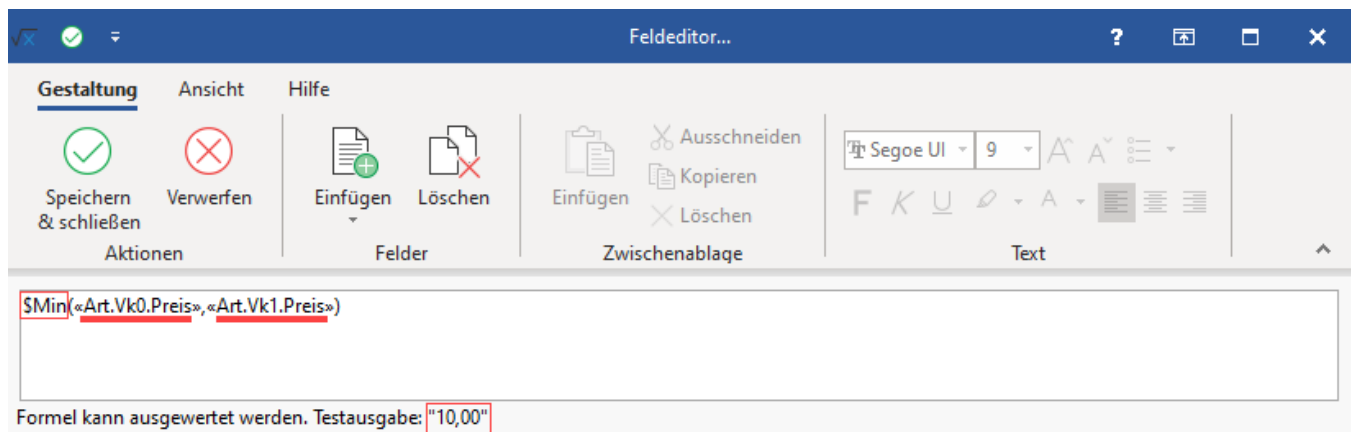
Einfaches Beispiel:

Durch die Funktion **\$Min** sollen Artikelverkaufspreis **VK0 und VK1 verglichen** werden (mit Rückgabe des kleinsten Wertes). Im DB-Manager wurde für die Artikel-Datenbank ein **Selektionsfeld** namens "**Günstigster VK: VK0 vs. VK1**" angelegt.

Mittels Feldzuweisung in einer Regel soll der kleinste der beiden Werte in das Selektionsfeld im Artikel geschrieben werden.

Unter PARAMETER - ARTIKEL - REGELN wird eine Feldzuweisung für dieses Selektionsfeld angelegt, welche "**Vor dem Ändern (vor einer Änderung oder einem Import)**" das Selektionsfeld nach folgender Formel befüllen soll:

```
$Min(«Art.Vk0.Preis»,«Art.Vk1.Preis»)
```



Im Beispiel beträgt **VK0 = 10,00 EUR**; **VK1 = 20,00 EUR**. Der **Rückgabewert** beider Werte ist in diesem Fall: **10,00 EUR** aus VK0, da dies der **kleinste** der beiden **Übergabewerte** ist.

Durch die eingetragene **Feldzuweisung** wird dieser **Feldwert** nun bei Änderungen oder Import mittels Formel **in das Selektionsfeld 1 des Artikels** geschrieben.

Zuweisungen		
Feld	Art der Feldzuweisung	Vorgabewert
Sel1: Günstigster VK: VK0 vs. VK1 [Sel1]	Feldwert über Formel füllen	\$Min(«Art.Vk0.Preis»,«Art.Vk1.Preis»)

Übersicht:

- [Bsp. zu \\$IncWhour\(\)](#) - Reaktionszeiten berechnen