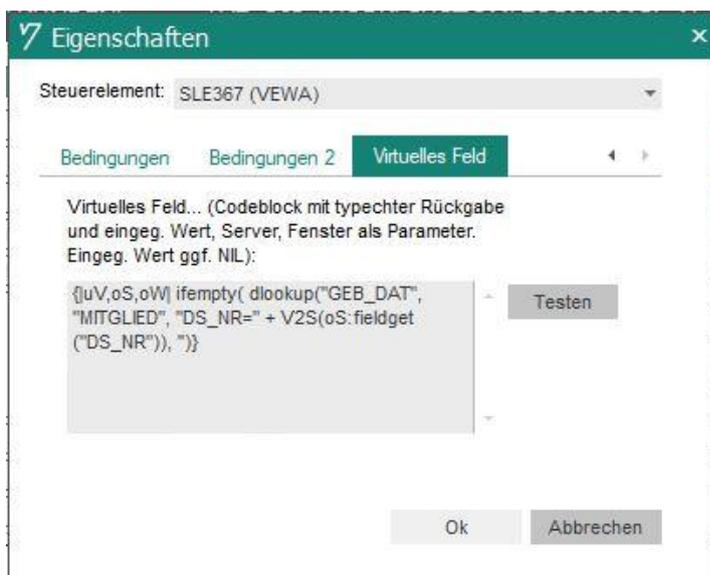


Codeblöcke in Masken (VEWA6 und VEWA7)

Mit Codeblöcken kann man eine logische Abfrage in der Maske ausführen, um entweder Daten aus anderen Tabellen anzuzeigen, Berechnungen auszuführen oder Anzeigebedingungen zu steuern.

1. Virtuelles Feld

Ein Virtuelles Feld ermöglicht es, beliebige MySQL-Befehle auszuführen, oder Informationen darzustellen:



Bei einem Virtuellem Feld ist es sinnvoll, in den Eigenschaften die „Nur lesen“ Checkbox anzuhaken. Es können zwar keine Daten in der Datenbank verändert werden, aber VEWA-User könnten verunsichert sein, wenn sie aus Versehen einen Eintrag entfernt haben.

Der Codeblock baut sich wie folgt auf:

```
{|uV,oS,oW| ifempty( dlookup("concat(NAME, ', ', VORNAME)", "MITGLIED", "DS_NR=" +  
V2S(oS:fieldget("DS_NR")), ")}|}
```

1.1 Erläuterung:

{ } Der Codeblock muss in geschweiften Klammern stehen, um von VEWA verarbeitet werden zu können
|uV,oS,oW| in diesem Pipe-Getrennten Bereich werden Variablen definiert. Es muss bei virtuellen Feldern immer mindestens die angegebenen Variablen definiert werden:

- uV „undefined Variable“ → kann im Codeblock verwendet werden, um Platz zu sparen. Wenn eine typisierte Variable genutzt werden soll, muss man die ungarische Notation benutzen zB dHeute, cVewauser oder nSumme
- oS „Object Server“ → hiermit ist die Datenbanktabelle der Maske angesprochen (alternative Schreibweise oSrv)
- oW „Object Window“ → hiermit ist die aktuell offene Maske angesprochen (alternative Schreibweise oF)

dLookup(„Feld bzw MySQL-Syntax“, „Tabelle“, „Bedingung“) Der dLookup-Befehl wird bei Aufruf der Maske und bei jedem Datensatz-Blättern an die Datenbank gesendet, um die angeforderten Informationen zurückzugeben. Es gibt auch den dSum-Befehl, welcher eine Summe über ein numerisches Feld zurückgeben kann. Die Parameter sind die gleichen wie bei dLookup. Man kann entweder ein beliebiges Feld aus einer Tabelle darstellen, oder mit MySQL-Syntax arbeiten (zB DATE_ADD(), DATEDIFF(), CONCAT(), GROUP_CONCAT()).

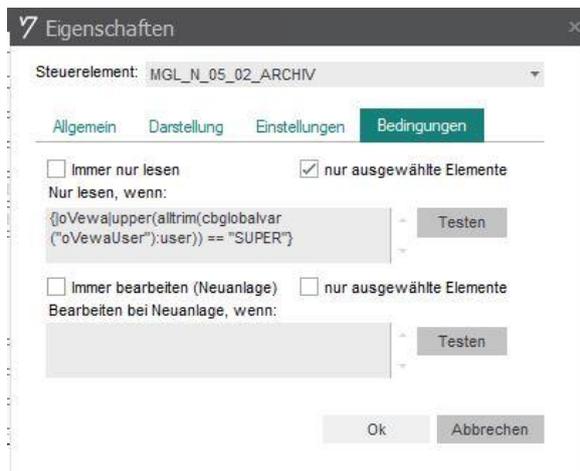
Ifempty(dlookup(...), ,') Sollte der dLookup-Befehl kein Ergebnis zurückgeben (NULL-Wert), kann es im Virtuellen Feld zu einer Fehlermeldung kommen. Deswegen wird mit ifempty() ein ggf zurückgegebener NULL-Wert in einen Leerstring umgewandelt (den die Maske problemlos darstellen kann).

V2S() Dieser Befehl sorgt für den Wechsel aus Vewascript-Syntax zu MySQL-Syntax (im Beispiel wird die DS_NR des Adressdatensatzes aus dem gerade geöffneten Datensatz mittels Vewascript ausgelesen). Es kann auch mit zB V2S(vewatoday()) das VEWA-Tagesdatum abgegriffen werden. Die alte Schreibweise Value2Statement() kann noch bei älteren VEWA6-Versionen vorkommen.

Die nötigen Informationen wie Feldname und Datenbanktabelle der Maske erhalten Sie, indem Sie auf das entsprechende Elemente in VEWA gehen, und mit der Tastenkombination strg+shift+rechte Maustaste sich die entsprechenden Elementinformationen anzeigen lassen.

2. Anzeigebedingung

Mittels Anzeigebedingungen können einzelne Elemente oder ganze (Sub)Masken entweder in den „nur lesen“-Modus gesetzt, oder komplett nicht dargestellt werden.



Anzeigebedingung Maske



Anzeigebedingung Element

Die Syntax ist in beiden Fällen leicht unterschiedlich zu der eines Virtuellen Feldes:

```
{|oVewa| upper(alltrim(cbglobalvar("oVewaUser"):user)) == "SUPER"}
```

2.1 Erläuterung:

- `|oVewa|` „Object Vewa“ → Es wird die aktuelle Sitzung in VEWA angesprochen
- `cbglobalvar("oVewaUser"):user` → Hier wird der Username vom aktuell angemeldeten VEWA-User genutzt. Alternativ kann statt `:user` auch zB `:abteilung` verwendet werden.
- `upper(alltrim(...))` → Durch „upper“ wird der folgende String in Großbuchstaben gesetzt; durch „alltrim“ werden führende und folgende Leerzeichen abgetrennt.

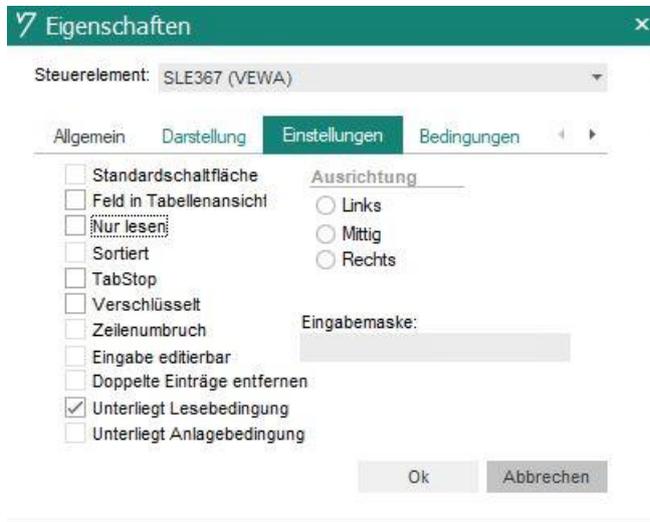
→ Der Abgleich muss identisch sein. Es kann auch z.B. `!=` für „ist nicht“ verwendet werden.

Wenn man auf einen Teilstring abfragen will, kann man die Bedingung umstellen:

```
{|oVewa| „SUP“ $ upper(alltrim(cbglobalvar("oVewaUser"):user))}
```

\$ → Das Dollar-Symbol entspricht einem „ist enthalten in“

Wenn man nur bestimmte Elemente der Lesebedingung unterwerfen will, muss bei dem Element in den Einstellungen die Checkbox „Unterliegt Lesebedingung“ angehakt sein:



Lesebedingung