LibreOffice
The Document Foundation

Base

# *Kapitel 4*
# *Formulare*

# Copyright

Dieses Dokument unterliegt dem Copyright © 2015. Die Beitragenden sind unten aufgeführt. Sie dürfen dieses Dokument unter den Bedingungen der GNU General Public License (*http://www.-gnu.org/licenses/gpl.html*), Version 3 oder höher, oder der Creative Commons Attribution License (*http://creativecommons.org/licenses/by/3.0/*), Version 3.0 oder höher, verändern und/oder weitergeben.

Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt.

Fast alle Hardware- und Softwarebezeichnungen und weitere Stichworte und sonstige Angaben, die in diesem Buch verwendet werden, sind als eingetragene Marken geschützt.

Da es nicht möglich ist, in allen Fällen zeitnah zu ermitteln, ob ein Markenschutz besteht, wird das Symbol (R) in diesem Buch nicht verwendet.

## Mitwirkende/Autoren

Robert Großkopf
Jost Lange
Jochen Schiffers
Jürgen Thomas
Michael Niedermair

## Rückmeldung (Feedback)

Kommentare oder Vorschläge zu diesem Dokument können Sie in deutscher Sprache an die Adresse *discuss@de.libreoffice.org* senden.

| **Vorsicht** | Alles, was an eine Mailingliste geschickt wird, inklusive der E-Mail-Adresse und anderer persönlicher Daten, die die E-Mail enthält, wird öffentlich archiviert und kann nicht gelöscht werden. Also, schreiben Sie mit Bedacht! |
|---|---|

## Datum der Veröffentlichung und Softwareversion

Veröffentlicht am 15.08.2021. Basierend auf der LibreOffice Version 7.2.

# Inhalt


Formulare als Eingabeerleichterung ..... 5

Erstellung von Formularen ..... 5

- Einfaches Formular ..... 5
- Symbolleisten des Formularentwurfs ..... 8
  - Formulargründung über den Navigator ..... 8
  - Formulargründung über ein Formularfeld ..... 9
  - Externe Formulare ..... 10

Formular-Eigenschaften ..... 11

- Eigenschaften der Kontrollfelder ..... 14
  - Standardeinstellungen vieler Kontrollfelder ..... 15
  - Textfeld ..... 21
  - Numerisches Feld ..... 22
  - Datumsfeld ..... 23
  - Zeitfeld ..... 24
  - Währungsfeld ..... 25
  - Formatiertes Feld ..... 25
  - Listenfeld ..... 27
  - Kombinationsfeld ..... 32
  - Markierfeld ..... 35
  - Optionsfeld ..... 36
  - Grafisches Steuerelement ..... 37
  - Maskiertes Feld ..... 38
  - Tabellen-Kontrollfeld ..... 39
  - Beschriftungsfeld ..... 41
  - Gruppierungsrahmen ..... 42
  - Schaltfläche ..... 44
  - Grafische Schaltfläche ..... 47
  - Navigationsleiste ..... 47
  - Drehfeld und Bildlaufleiste ..... 49
  - Verstecktes Steuerelement ..... 50
  - Mehrfachselektion ..... 51
- Einfaches Formular komplett erstellt ..... 53
  - Felder als Gruppe hinzufügen ..... 54
  - Felder anpassen ..... 55
  - Felder einzeln hinzufügen ..... 58
  - Tabellenkontrollfeld ..... 59

Hauptformular und Unterformular ..... 64

Eine Ansicht – viele Formulare ..... 78

Fehlermeldungen bei der Eingabe in Formulare ..... 85

Suchen und Filtern in Formularen über die Navigationsleiste ..... 86

- Datensatzsuche mit Parametern ..... 87
- Filterung mit dem Autofilter ..... 88
- Filterung mit dem formularbasierten Filter ..... 89
- Filterung über den Standardfilter ..... 91
- Fazit ..... 93

Dateneingabe und Navigation ..... 93

Drucken aus Formularen ..... 94

Darstellungsgröße von Formularen ..... 95

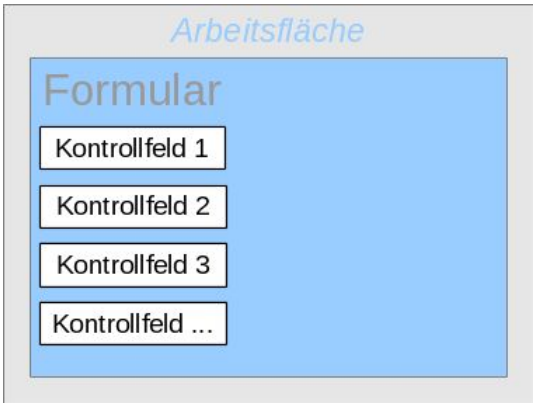
# Formulare als Eingabeerleichterung

Formulare werden dann genutzt, wenn die Eingabe direkt über eine Tabelle zu unübersichtlich wird, eventuelle Fehleingaben rechtzeitig abgefangen werden sollen oder zu viele Tabellen eine direkte Verwaltung der Daten unmöglich machen.

**Hinweis**

Ein *Formular* ist in Base ein für den Nutzer *nicht sichtbares Konstrukt*. Es dient innerhalb von Base dazu, den Kontakt zur Datenbank zu ermöglichen.

*Sichtbar* sind für den Nutzer die *Kontrollfelder*, die dazu dienen, Text, Zahlen usw. einzugeben oder anzuzeigen. Diese Kontrollfelder werden über die GUI in verschiedene Feldarten unterteilt.



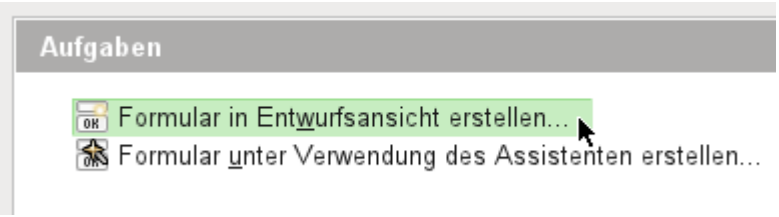

Der Begriff *Formular* hat eine doppelte Bedeutung.
Zum einen steht ein *Formular* für den gesamten Inhalt des Eingabefensters, in dem die Daten für eine oder mehrere Tabellen verwaltet werden. Hier handelt es sich genaugenommen um ein Formulardokument, das in der Datenbankdatei auch als komplette Writer-Datei abgespeichert wird.
Zum anderen enthält ein solches Formulardokument ein Fenster, das wiederum ein oder mehrere Hauptformulare enthalten kann; auch für diese Teilbereiche des Fensters wird der Begriff *Formular* verwendet. Jedes dieser Hauptformulare in dem Fenster des Formulardokumentes kann außerdem noch wieder Unterformulare enthalten.
Aus dem Zusammenhang sollte immer klar werden, welche Art von Formular gemeint ist, sodass es hoffentlich niemals zu Missverständnissen kommt.

# Erstellung von Formularen

Der einfachste Weg zur Erstellung von Formularen ist der über den Formular-Assistenten. Mit diesem Assistenten lassen sich auch Formulare mit Unterformularen erstellen. Dieser Weg wurde bereits im Kapitel «Eingabeformular» im Kapitel «Einführung in Base» beschrieben.

## Einfaches Formular

Als Start dient uns aus dem Formularbereich die Aufgabe **Formular in Entwurfsansicht erstellen**.




Rufen wir damit den Formulareditor auf, so zeigt sich erst einmal das Fenster *Abbildung 1*.

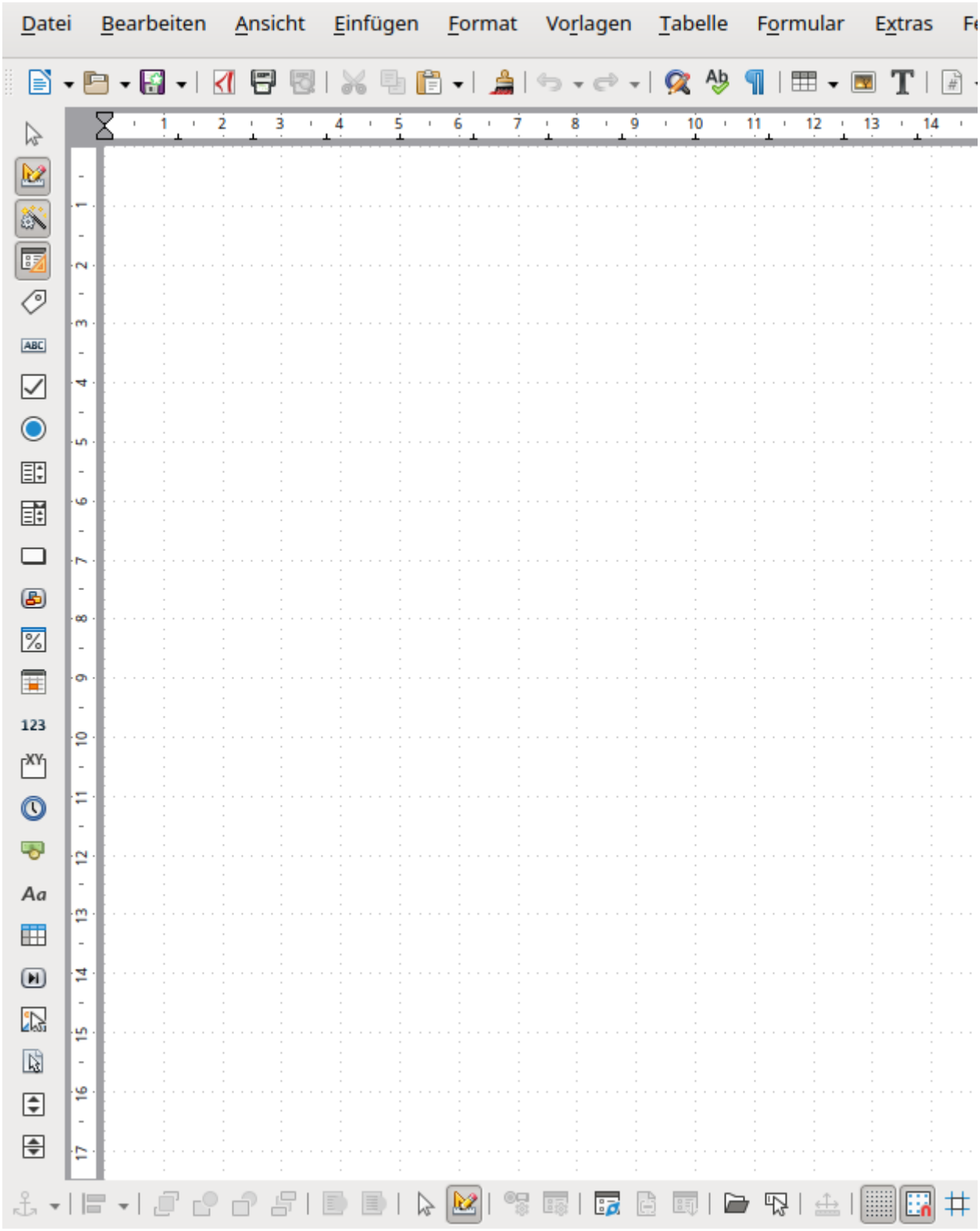



*Abbildung 1: Formular in der Entwurfsansicht*

Am linken Rand ist die Symbolleiste «Formular-Steuerelemente» eingeblendet. Am unteren Rand ist die Symbolleiste «Formular-Entwurf» angedockt. Sollten diese Symbolleisten nicht automatisch erscheinen, so können sie über **Ansicht → Symbolleisten** angewählt werden. Ansonsten steht noch im Hauptmenü der Menüpunkt **Formular** zur Verfügung.

Die weiße Fläche weist ein gepunktetes Raster auf. Dies dient dazu, die Elemente möglichst genau positionieren zu können – vor allem im Verhältnis zueinander. Dass das Raster sichtbar und

eingeschaltet ist, ist an den Symbolen ganz rechts in der Leiste zum Formular-Entwurf zu erkennen.

*Abbildung 2: Alle verfügbaren Schaltflächen der Symbolleiste «Formular-Entwurf»*

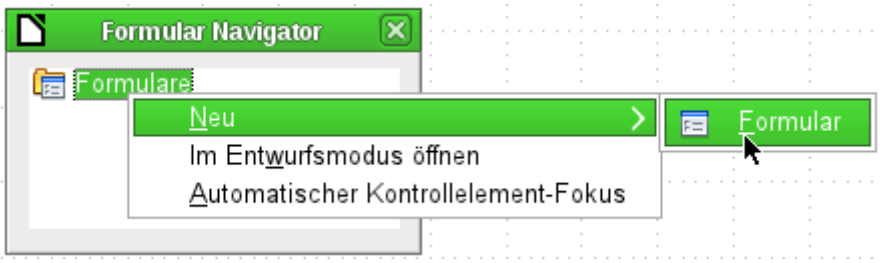
## Symbolleisten des Formularentwurfs

Auf der leeren Fläche soll nun ein Formular entstehen. Dies kann auf verschiedene Arten geschehen:

- Aufruf des Formular-Navigators, von dort Gründung eines Formulars sowie
- Erstellung von Formularfeldern und Gründung des Formulars über das dortige Kontextmenü.

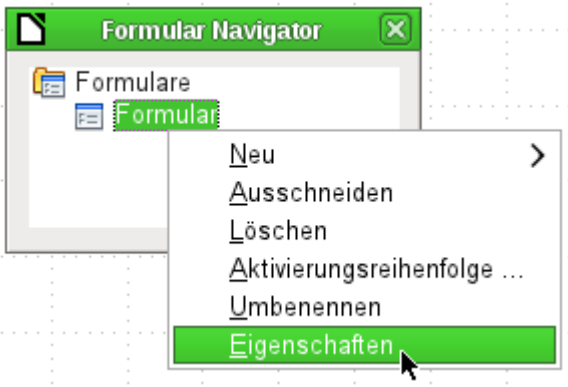
### *Formulargründung über den Navigator*

Mit dem in *Abbildung 2* abgebildeten Button Formular-Navigator wird der Navigator gestartet. Es erscheint ein Fenster, das auf lediglich ein Verzeichnis hinweist. Dies ist die höchste Ebene der Fläche, die jetzt bearbeitet wird. Sie ist mit **Formulare** benannt. Dies weist darauf hin, dass nicht nur ein, sondern ohne weiteres mehrere Formulare auf der angezeigten Fläche untergebracht werden können.




Mit einem Rechtsklick auf das Verzeichnis Formulare öffnet sich ein Kontextmenü, in dem über den Menüpunkt **Neu** ein neues **Formular** erstellt werden kann. Die weiteren Befehle des Kontextmenüs entsprechen denen der Buttons in der *Abbildung 2*.

| **Hinweis** | Soll ein Formular automatisch mit dem Cursor im ersten Formularelement starten, so kann dies durch die Auswahl **Automatischer Kontrollelement-Fokus** bewirkt werden. Das erste Formularelement wird über die *Aktivierungsreihenfolge* geregelt.<br>Leider hat diese Funktion zur Zeit einen Bug. Befindet sich in einem Formular ein Tabellenkontrollfeld, so wird der Fokus automatisch auf das erste Feld des Tabellenkontrollfeldes gesetzt. Dieser Bug ist merkwürdigerweise zu beheben, wenn nach der Wahl des automatischen Kontrollelement-Fokus die Sprache der Benutzeroberfläche umgestellt wird: *Bug 87290*. |
|---|---|




Das Formular erhält standardmäßig den Namen **Formular**. Diese Bezeichnung kann direkt oder später geändert werden. Sie hat allerdings für die spätere Funktion nur dann eine Bedeutung, wenn über Makros auf Teile des Formulars zugegriffen werden soll. Spätestens dann sollten nicht zwei Elemente mit gleicher Bezeichnung in der gleichen Ebene des Verzeichnisbaums auftauchen.

Über das Kontextmenü des Formulars geht schließlich der Weg zu den Formulareigenschaften.

### *Formulargründung über ein Formularfeld*

Über die Symbolleiste für die Formularsteuerelemente (*Abbildung 3*) stehen direkt alle Formularfelder zur Verfügung. Während die ersten vier Elemente identisch zu den entsprechenden Elementen des Formular-Entwurfs sind, folgen anschließend sämtliche Formularfelder.

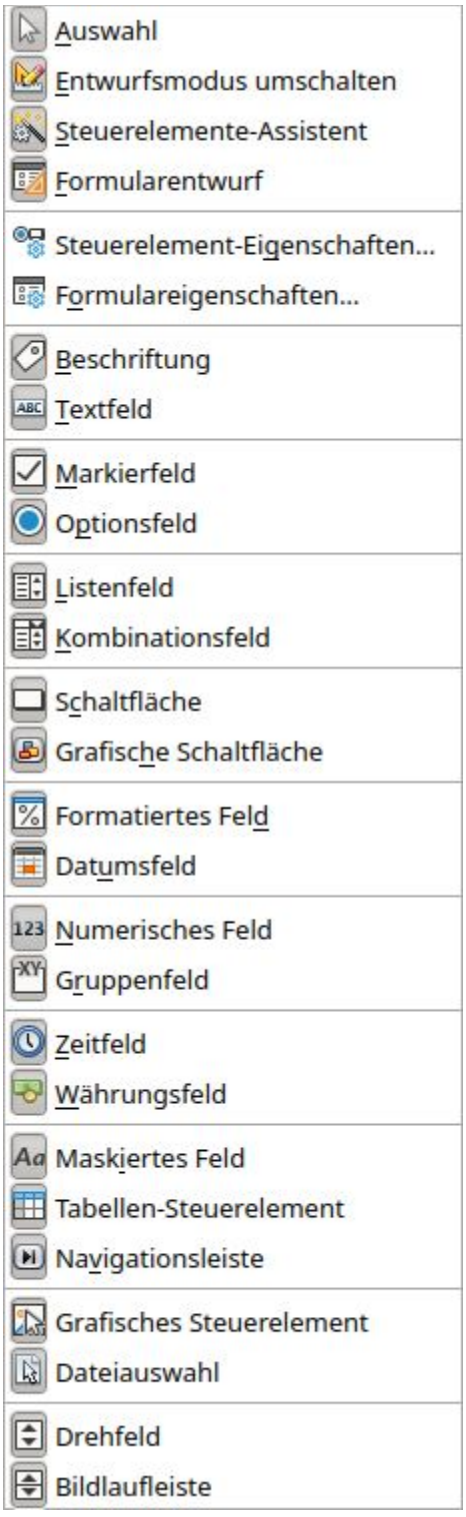
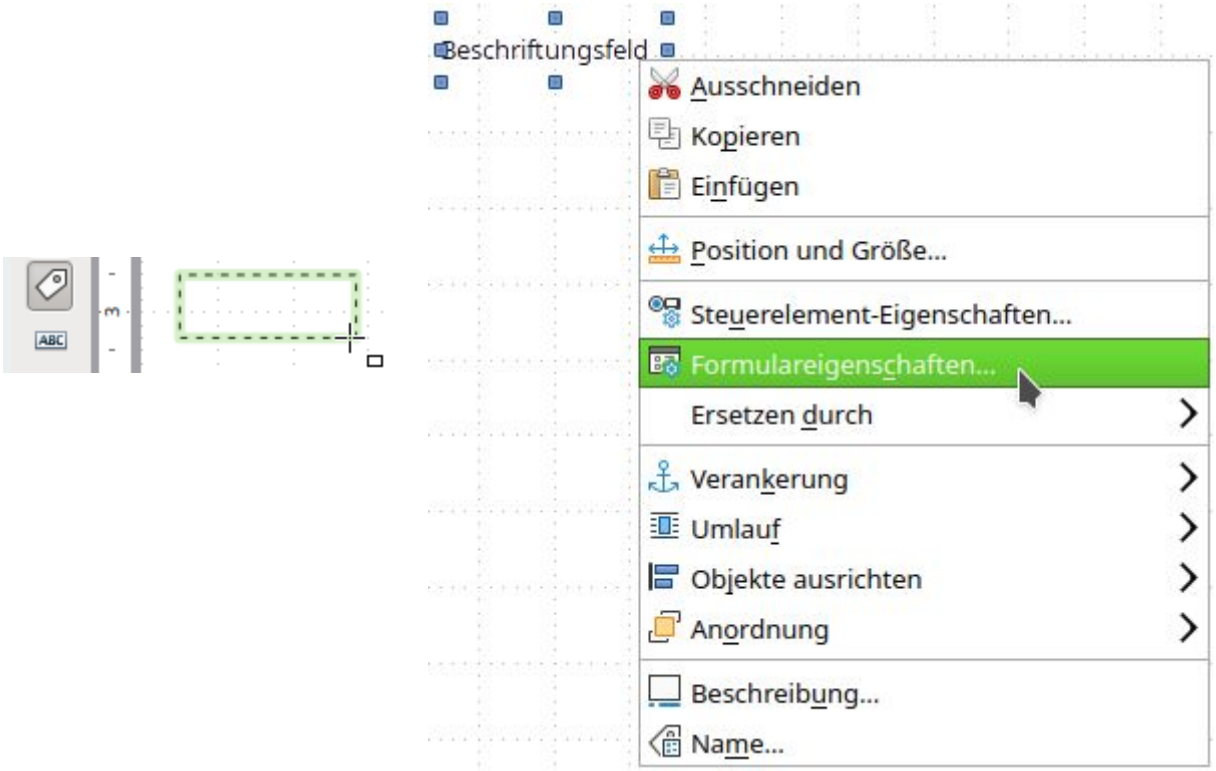



*Abbildung 3: Alle verfügbaren Schaltflächen der Symbolleiste «Formular-Steuerelemente»*

| Hinweis | Der Inhalt der Symbolleiste «Formular-Steuerelemente» ist mit der Zeit geändert worden. Vor der Version LO 6.0 gab es eine Symbolleiste «Weitere Steuerelemente», die aus der Symbolleiste «Formular-Steuerelemente» als frei schwebende Leiste angezeigt wurde. Zur Version LO 6.3 schließlich gibt es die Symbolleiste «Weitere Steuerelemente» nicht mehr.<br>Neben der Symbolleiste stehen alle Steuerelemente über den neuen Menüeintrag **Formular** zur Verfügung. |
|---|---|

Über den Aufruf eines Formularfeldes wird automatisch ein Formular mit gegründet:



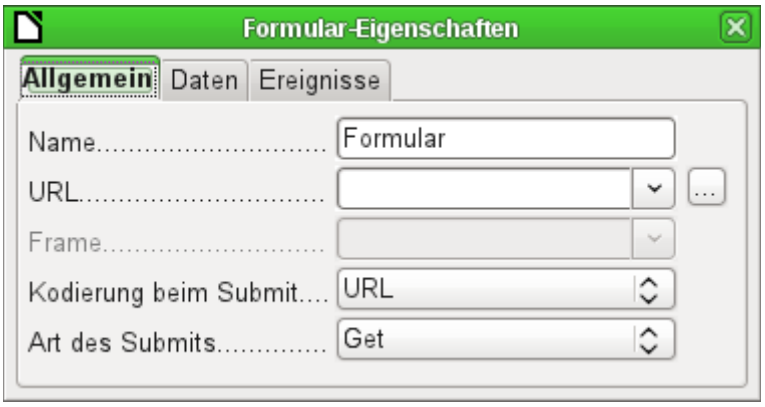

Dazu wird z. B. ein Beschriftungsfeld aufgerufen. Der Cursor verändert sein Erscheinungsbild. Es kann eine rechteckige Form auf die weiße Oberfläche des Formulars gezogen werden. Aus der gestrichelten Form entsteht anschließend ein Beschriftungsfeld. Jetzt wird zur Erstellung des Formulars das **Kontextmenü** des Kontrollfeldes aufgerufen.

Über den Menüpunkt **Formular** werden hier die Eigenschaften des nebenher gegründeten Formulars aufgerufen. Das Formular wurde mit dem Standardnamen «Formular» erstellt.

### *Externe Formulare*

Neben den Formularen, die direkt in Base erstellt werden, gibt es auch die Möglichkeit, im Writer oder in Calc Formulare zu erstellen. Auch Formulare, die in Base erstellt wurden, lassen sich zu externen Formularen umwandeln. Dies wird im Kapitel «Datenbank-Anbindung» beschrieben.

# Formular-Eigenschaften

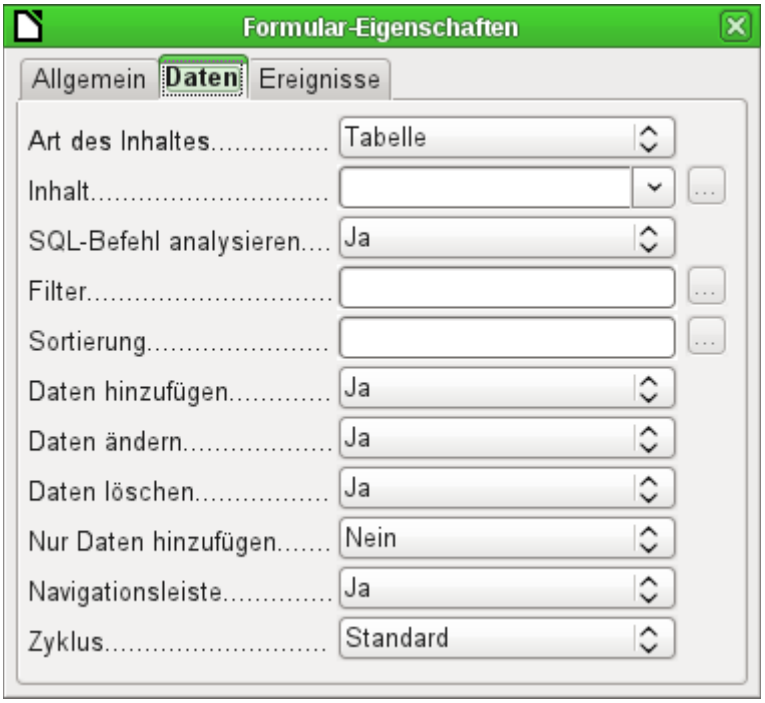



Werden die Formulareigenschaften über das Kontextmenü im Formular-Navigator oder das Kontextmenü eines Formularfeldes aufgerufen, so erscheint ein Fenster mit drei Registerreitern: **Allgemein**, **Daten** und **Ereignisse**

**Allgemein**

Hier kann der **Name** des Formulars geändert werden. Außerdem finden sich Einstellungsmöglichkeiten, die innerhalb von Base keine weitere Bedeutung haben. Sie zeigen lediglich die universelle Einsatzmöglichkeit des Formulareditors; bei Erstellung von Webformularen ist hier eine Einstellung notwendig. (URL: Ziel, an das die Daten gesandt werden sollen | Frame: Teil der Zielwebseite, der eventuell separat angesprochen werden muss | Kodierung beim Submit: Neben dem normalen Kodieren der Zeichen für eine Weitergabe über die URL sind hier Textkodierungen und Multipartkodierungen (z. B. zum Versenden von Dateien …) möglich. | Art des Submits: Get (über die URL sichtbar an den Dateinamen angehängt) oder POST (nicht sichtbar, auch für größere Datenmengen geeignet).

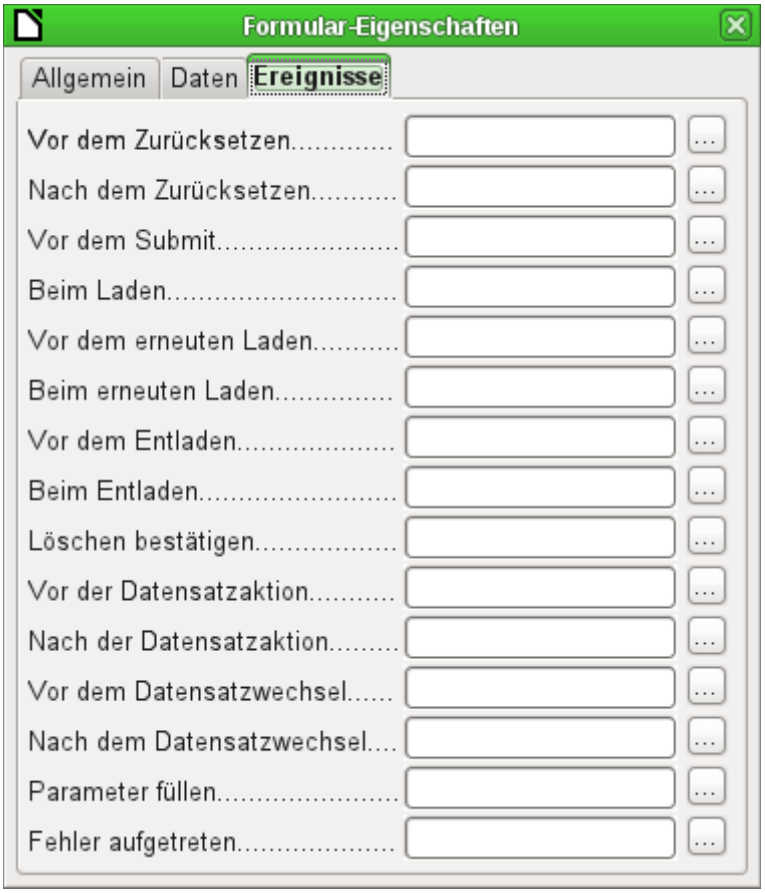
**Daten**




Für die Erstellung interner Formulare von Base ist dies der wichtigste Registerreiter. Hier wird zuerst der Inhalt des Formulars festgelegt.

- *Art des Inhalts:* Hier besteht die Wahl zwischen *Tabelle*, *Abfrage* und *SQL-Befehl*. Während Tabellen in der Regel immer für Eingaben in einem Formular genutzt werden können, so ist dies bei Abfragen eventuell nicht der Fall. Näheres dazu im Kapitel 'Abfragen'. Gleiches wie für Abfragen gilt auch für die direkten Eingabe eines SQL-Befehls. Hier handelt es sich dann lediglich um eine Abfrage, die nicht im Abfragecontainer von Base sichtbar ist, aber vom Prinzip her die gleiche Struktur aufweist.
- *Inhalt:* Wird unter Art des Inhaltes Tabelle oder Abfrage gewählt, so werden hier alle *verfügbaren Tabellen und Abfragen* gelistet. Soll ein SQL-Befehl erstellt werden, so besteht die Möglichkeit, den Abfrageeditor dazu über den Button mit den drei Punkten rechts von dem Inhaltsfeld aufzurufen.
- *SQL-Befehl analysieren:* Wird die Analyse des SQL-Befehls nicht zugelassen (weil z. B. mit Code gearbeitet wird, den die GUI eventuell nicht richtig deuten kann), so ist hier 'Nein' zu wählen. Allerdings schließt dies aus, dass das Formular weiterhin noch mit der Filterung oder mit der Sortierung auf die zugrundeliegenden Daten zugreifen kann.
- *Filter:* Hier kann ein Filter gesetzt werden. Hilfe dazu bietet ein Klick auf den Button rechts von dem Eingabefeld. Das Filtern entspricht dem «Filtern von Tabellen».
- *Sortierung:* Hier kann eine Sortierung der Daten festgelegt werden. Hilfe dazu bietet ein Klick auf den Button rechts von dem Eingabefeld. Das Sortieren entspricht dem «Sortieren von Tabellen».
- *Daten hinzufügen:* Sollen neue Daten erstellt werden können? Standardmäßig ist dies auf Ja eingestellt.
- *Daten ändern:* Sollen Daten geändert werden können? Ebenfalls Standard Ja.
- *Daten löschen:* Auch das Löschen von Daten wird standardmäßig ermöglicht.
- *Nur Daten hinzufügen:* Ist dies gewählt, so erscheint immer ein leeres Formular. Auf die alten Datensätze besteht kein Zugriff, sie können nicht bearbeitet oder auch nur angesehen werden.
- *Navigationsleiste:* Das Erscheinen der Navigationsleiste am unteren Bildschirmrand kann angeschaltet oder ausgeschaltet werden. Außerdem besteht die Möglichkeit, bei einem Unterformular immer die Navigationsleiste des darüber liegenden Hauptformulars anzeigen zu lassen, so dass eine Betätigung der Navigationsleiste direkte Auswirkung auf das Hauptformular hat.
  Diese Einstellung zur Navigationsleiste betrifft nicht die Leiste, die gegebenenfalls als Formularfeld eingefügt werden kann.
- *Zyklus:* Standard bedeutet hier für Base-Datenbanken, dass nach der Eingabe im letzten Feld innerhalb eines Formulars mit dem Tabulator zum ersten Feld des nächsten Datensatzes, also gegebenenfalls eines neuen Datensatzes, gesprungen wird. Dies ist für die Datenbanken gleichbedeutend mit «Alle Datensätze». Wird hingegen bei Datenbanken «Aktueller Datensatz» gewählt, so bewegt sich der Cursor nur innerhalb des einen Datensatzes, beim Erreichen des letzten Feldes also zum ersten Feld des gleichen Datensatzes. Aktuelle Seite bezieht sich wieder besonders auf HTML-Formulare. Hier springt dann der Cursor vom Ende eines Formulars gegebenenfalls zum nächsten Formular auf der Seite, das weiter unten liegt.

**Ereignisse**




*Ereignisse* können Makros auslösen. Durch einen Klick auf den rechts stehenden Button (…) können Makros mit dem Ereignis verbunden werden.

*Zurücksetzen*: Das Formular wird von allen neuen Einträgen geleert, die noch nicht abgespeichert sind.

*Vor dem Submit*: Bevor die Formulardaten gesendet werden. Dies hat nur bei Webformularen eine Bedeutung.

*Beim Laden*: Nur beim Öffnen des Formulars. Nicht beim Laden eines neuen Datensatzes in das Formular.

*Erneutes Laden*: Dies erfolgt, wenn der Inhalt des Formulars z. B. über einen Button in der Navigationsleiste aktualisiert wird.

*Entladen*: Nach einigen Tests scheint dies ohne Funktion zu sein. Erwartet würde der Ablauf eines Makros beim Schließen des Formulars.

*Datensatzaktion*: Dies ist z. B. das Abspeichern mittels Button. Im Test ergibt sich, dass die Aktion **Vor der Datensatzaktion** regelmäßig doppelt erscheint, d. h. Makros werden direkt nacheinander zweimal abgearbeitet. Dies liegt daran, dass hier unterschiedliche Funktionen (Implementationen) ausgeführt werden. Beide sind mit Namen versehen: **`org.openoffice.comp.svx.FormController`** und **`com.sun.star.comp.forms.ODatabaseForm`**. Wenn innerhalb des Makros mit **`oForm.ImplementationName`** der entsprechende Name abgefragt wird, so kann das Makro auf eine Ausführung begrenzt werden.

*Datensatzwechsel*: Bereits das Öffnen des Formulars stellt einen Datensatzwechsel dar. Beim Wechsel von einem Datensatz zum anderen innerhalb eines Formulars taucht diese Aktion

ebenfalls zweimal auf. Makros werden also auch hier zweimal hintereinander ausgeführt. Auch hier kann aber zwischen den Ursachen des Ereignisses unterschieden werden.

*Parameter füllen:* Dieses Makro springt ein, wenn eine Parameterabfrage in einem Unterformular aufgerufen werden soll, aber aus irgendeinem Grund der Parameter vom Hauptformular nicht richtig weiter gegeben wird. Ohne das Ereignis abzufangen, erfolgt dann beim Laden des Formulars eine Parameterabfrage.

*Fehler aufgetreten:* Dieses Ereignis lässt sich nicht nachvollziehen.

## Eigenschaften der Kontrollfelder

Ist ein Formular erstellt, so kann es mit den sichtbaren Kontrollfeldern bestückt werden. Die Kontrollfelder sind für verschiedene Aufgaben gedacht:

- Die meisten zeigen den Inhalt aus der Datenbank an oder nehmen die Daten entgegen, die in die Datenbank eingefügt werden.
- Andere Kontrollfelder dienen zur Navigation, zum Suchen und zur Ausführung von Befehlen (Interaktion).
- Weitere Kontrollfelder sorgen für eine zusätzliche grafische Aufarbeitung des Formulars.

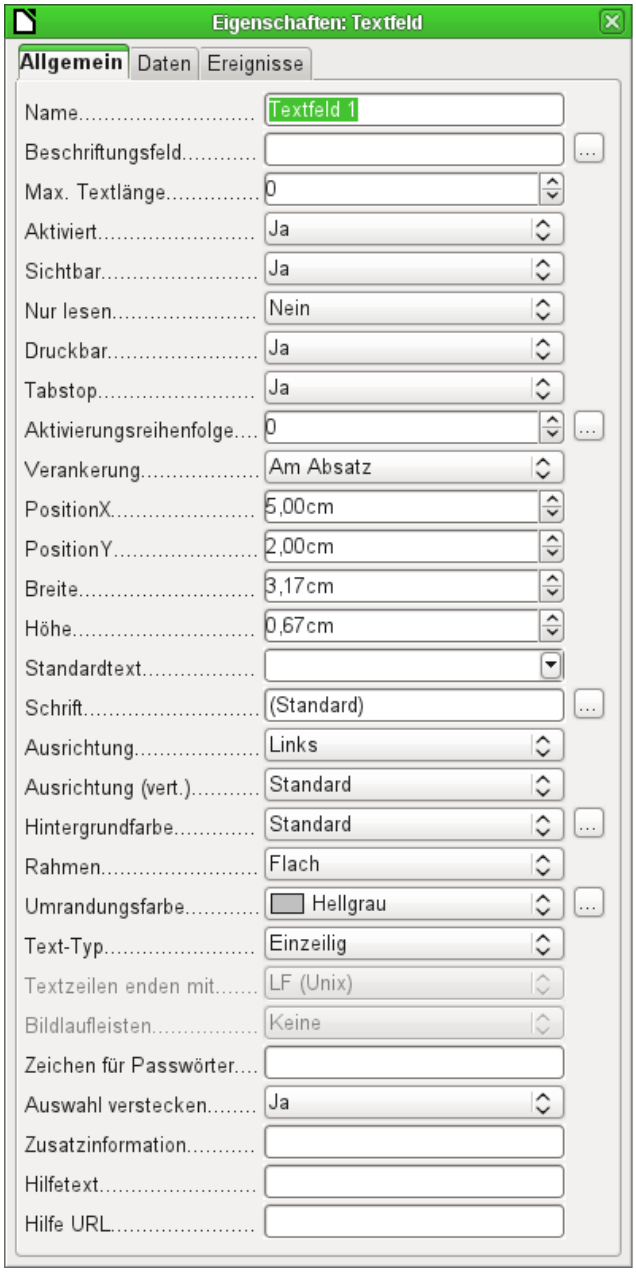
| ***Dateneingabe und Datenanzeige*** | |
|---|---|
| ***Kontrollfeld*** | ***Anwendungsgebiet*** |
| Textfeld | Texteingaben |
| Numerisches Feld | Zahleneingabe |
| Datumsfeld | Datumseingabe |
| Zeitfeld | Zeiteingabe |
| Währungsfeld | Zahleneingabe, vorformatiert für Währungen |
| Formatiertes Feld | Anzeige und Eingabe mit zusätzlicher Formatierung wie z. B. Maßeinheiten |
| Listenfeld | Auswahl zwischen vielen verschiedenen Möglichkeiten, Weitergabe eines anderen als des angezeigten Wertes an die Datenbank. |
| Kombinationsfeld | Wie Listenfeld, nur Weitergabe des angezeigten Wertes und dazu noch die Möglichkeit, auch neue Werte einzugeben. |
| Markierfeld | Ja/Nein-Felder |
| Optionsfeld | Auswahl zwischen verschiedenen, stark begrenzten Möglichkeiten. |
| Grafisches Kontrollfeld | Anzeige von Bildern und seit LO 5.0 auch *.pdf-Dateien (1. Seite) aus einer Datenbank und Neueingabe von Bildern und seit LO 5.0 auch anderen Dateien, abhängig von der Einstellung **Extras → Optionen → LibreOffice → Allgemein→ Dialoge zum Öffnen/Speichern** und den Systemdialogen des Betriebssystems, in eine Datenbank über eine Pfadangabe |
| Maskiertes Feld | Eingabe in eine vorgefertigte Maske; grenzt die Eingabemöglichkeiten auf bestimmte Zeichenkombinationen ein. |
| Tabellen-Kontrollfeld | Universelles Eingabemodul, das eine ganze Tabelle bedienen kann. Integriert in dieses Kontrollfeld sind wieder viele der obigen Kontrollfelder |

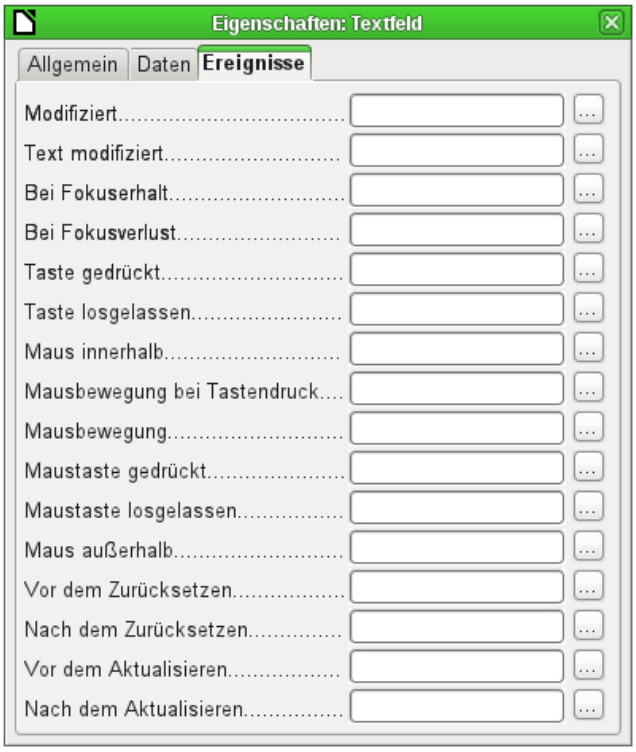
| Gestaltung | |
|---|---|
| ***Kontrollfeld*** | ***Anwendungsgebiet*** |
| Beschriftungsfeld | Überschrift über das Formular, Beschriftung anderer Kontrollfelder |
| Gruppierungsrahmen | Linienzug um z. B. verschiedene Optionsfelder |

| Interaktion | |
|---|---|
| ***Kontrollfeld*** | ***Anwendungsgebiet*** |
| Schaltfläche | Button mit Beschriftung |
| Grafische Schaltfläche | Wie der Button, nur mit einer zusätzlich auf dem Button erscheinenden Grafik |
| Navigationsleiste | Leiste mit geringen Abweichungen zu der, die am unteren Bildschirmrand angezeigt wird. |
| Dateiauswahl | Auswahl von Dateien z. B. zum Hochladen in HTML-Formularen. - nicht weiter beschrieben |
| Drehfeld | Nur über Makroprogrammierung verwendbar – nicht weiter beschrieben |
| Bildlaufleiste | Nur über Makroprogrammierung verwendbar – nicht weiter beschrieben |
| Verstecktes Steuerelement | Hier kann ein Wert über Makros eingespeichert und wieder ausgelesen werden. |

### *Standardeinstellungen vieler Kontrollfelder*

Die Eigenschaften werden wie beim Formular in drei Kategorien unterteilt: Allgemein, Daten und Ereignisse. Unter Allgemein wird all das eingestellt, was für den Nutzer sichtbar ist. In der Kategorie Daten wird die Verbindung zu einem Feld der Datenbank hergestellt. Die Kategorie Ereignisse schließlich regelt Auslösemomente, die mit irgendwelchen Makros verbunden werden können. Für eine Datenbank ohne Makros spielt diese Kategorie keine Rolle.






**Hinweis**

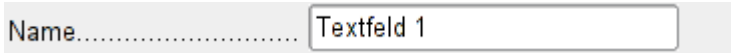
In den folgenden Übersichten stehen die deutschen Begriffe vor allem der Eigenschaften – genau wie im Formular-Entwurf. In Makros werden zur Beeinflussung der Felder die englischen Begriffe benötigt; diese sind in [Klammern] angegeben. So bedeutet z. B. [Printable] «Druckbar».

Beispiel:

```
oDoc = thisComponent
oDrawpage = oDoc.drawpage
oForm = oDrawpage.forms.getByName("Formular")
oFeld = oForm.getByName("Formularfeld")
oFeld.Printable = True 'Das Feld wird auf "Druckbar"→"Ja" gesetzt.
```

Die Eigenschaften werden einfach mit einem Punkt verbunden an das entsprechende Objekt des Formularfeldes angehängt. Weitere Hinweise stehen im Kapitel «Makros».

**Allgemein**

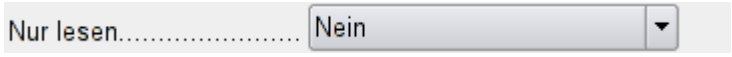



Die Bezeichnung für das Feld sollte innerhalb eines Formulars nur einmal vorkommen – Anwendung bei Zugriff über Makros.
[Name]

Gehört zu dem Feld ein Beschriftungsfeld? Hiermit wird eine Gruppierung festgelegt. Über das Beschriftungsfeld kann dann das Formularfeld mit einer Tastenkombination direkt erreicht werden.
[LabelControl]

Nicht aktivierte Felder sind nicht verwendbar und werden grau hinterlegt. Sinnvoll bei Steuerung über Makros (Entscheidung: Wenn in Feld 1 ein Wert eingegeben wird, darf in Feld 2 kein Wert eingegeben werden – Feld 2 wird deaktiviert)
[Enabled]

In der Regel 'Ja'; nicht sichtbare Felder können Werte zwischenspeichern. Anwendung z. B. bei der Erstellung von Kombinationsfeldern mit Makros.
[EnableVisible]

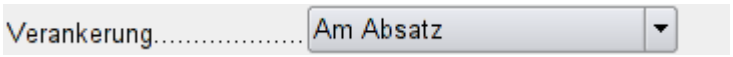



'Ja' würde eine Veränderung des Wertes ausschließen; z. B. für die Anzeige eines automatisch erstellten Primärschlüssels sinnvoll.
[ReadOnly]

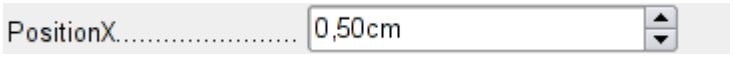
Manchmal sind Seitenausdrucke aus einem Formular sinnvoller als ein separater Bericht. Hier sollen dann eventuell nicht alle Felder erscheinen.
[Printable]

Durch ein Formular wird in der Regel mit dem Tabulator navigiert.
Ein Feld, das z. B. nur gelesen wird, braucht keinen Stop des Tabulators, würde also übersprungen.
[Tabstop]

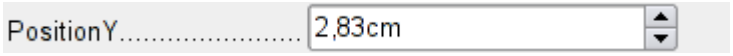
Hat das Feld einen Tabstop? Hier wird die Reihenfolge innerhalb des Formulars eingestellt.
[Tabindex]



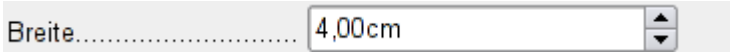

Verankerung der Grafik, die das Textfeld darstellt.



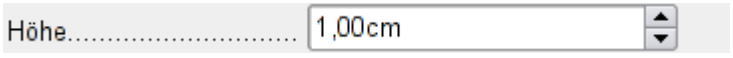

Position links oben vom linken Rand aus.
[PosSize.X]



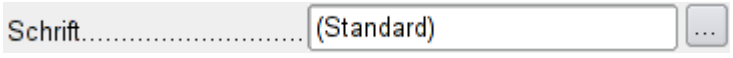

Position von oben aus.
[PosSize.Y]




Breite des Feldes
[PosSize.Width]



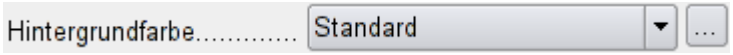

Höhe des Feldes
[PosSize.Height]



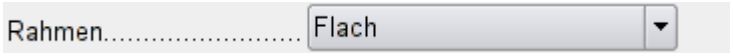

Schriftart, Schriftschnitt, Schriftgrad und Schrifteffekt sind hier einstellbar.
[Fontxxx]

Der Eintrag wird linksbündig dargestellt.
[Align]

Standard | Oben | Mitte | Unten
[VerticalAlign]




Hintergrundfarbe des angezeigten Textfeldes
[BackgroundColor]




Rahmenform: Ohne Rahmen | 3D-Look | Flach
[Border]

Wenn ein Rahmen, dann kann hier die Umrandungsfarbe eingestellt werden.
[BorderColor]

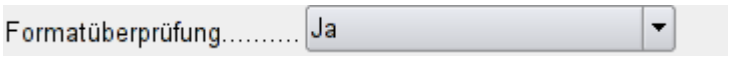
Markierter Text wird so nicht mehr als markiert angezeigt, wenn das Textfeld den Fokus verliert.
[HideInactiveSelection]

Gut nutzbar für Informationen, die mittels Makros ausgelesen werden sollen, siehe «Zusatzinformationen eines Feldes» im Kapitel «Makros».
[Tag]

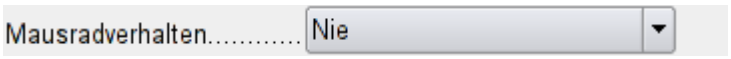
Erscheint als sogenannter Tooltip, wenn mit der Maus über das Textfeld gefahren wird.
[HelpText]

Verweist auf eine Hilfedatei – eher für HTML-Zwecke geeignet. Durch F1 abrufbar, wenn der Fokus auf dem Feld liegt.
[HelpURL]

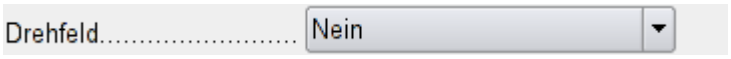
Zusätzlich sind bei Zahlenfeldern, Datumsfeldern u.ä. üblich:



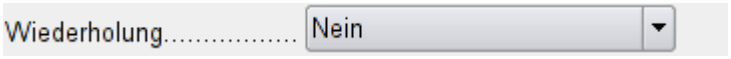

Mit eingeschalteter Überprüfung ist nur die Eingabe von Ziffern und Komma möglich.
[EnforceFormat]




*Nie* erlaubt keine Änderung mit dem Mausrad; *Wenn ausgewählt* lässt eine Änderung zu, wenn das Feld ausgewählt ist und die Maus sich über dem Feld befindet; *Immer* bedeutet, dass sich die Maus über dem Feld befinden muss.
[MouseWheelBehavior]




Ein Drehsymbol wird an der rechten Seite des Feldes eingeblendet.
[Spin]




Wenn ein Drehfeldpfeil länger gedrückt wird lässt sich hier einstellen, ob nicht nur zum nächsten Wert gedreht werden soll.
[Repeat]

Stellt die Verzögerungszeit ein, nach der der Druck auf die Maustaste beim Drehfeld Wiederholung interpretiert wird.
[RepeatDelay]

**Daten**

*Datenfeld*: Hier wird die Verbindung zur Tabelle hergestellt, die dem Formular zugrunde liegt.
[Model.DataField]

*Leere Zeichenfolge ist NULL*: Soll ein leeres Feld geleert werden (NULL) oder nur der Inhalt gelöscht werden?

*Eingabe erforderlich*: Dieser Eintrag sollte sich mit dem in der Tabelle decken. Dann fragt die GUI gegebenenfalls nach einer Eingabe, wenn vom Nutzer kein Wert eingegeben wurde. [Model.InputRequired]

*Filtervorschlag*: Bei einer Filterung der Daten werden die Inhalte dieses Feldes als Vorschläge zwischengespeichert. Achtung – dies ist bei umfangreichen Inhalten recht speicherintensiv. [Model.UserValueFilterProposal]

**Ereignisse**

*Modifiziert*: Dieses Ereignis tritt ein, wenn das Kontrollfeld geändert wurde und anschließend den Fokus verloren hatte. Das Ereignis geht verloren, wenn direkt zum nächsten Datensatz gewechselt wird. Unter den Umständen ist also eine Änderung gespeichert, ohne vorher wahrgenommen zu werden. [com.sun.star.lang.EventObject]

*Text modifiziert*: Direkt auf den Inhalt bezogen; kann Text, Zahl o.a. sein, tritt also nach jeder Tastatureingabe auf. [com.sun.star.awt.TextEvent]

*Fokuserhalt*: Der Cursor kommt in das Feld hinein. Hier sollte auf keinen Fall über das Makro eine Messagebox auf dem Bildschirm erzeugt werden. Durch das Anklicken dieser Box verliert das Formularfeld den Fokus und erhält ihn direkt danach zurück – eine Schleife wird dadurch erzeugt. Sie kann nur durch Tastatureingaben unterbrochen werden.

*Fokusverlust*: Der Cursor bewegt sich aus dem Feld heraus. Auch dies kann zu einem Wechselspiel führen, wenn eine zu bestätigende Bildschirmausgabe erfolgt.

*Taste*: Bezieht sich auf die Tastatur. Die Taste muss ausgelöst werden, wenn der Cursor in dem Kontrollfeld steht. Auch der Tabulator zum Verlassen des Kontrollfeldes löst dieses Ereignis aus. Dem Ereignis wird über den **`KeyCode`** bzw. **`KeyChar`** die auslösende Taste (Buchstabe, Zahl, Spezialtaste) mitgegeben. [com.sun.star.awt.KeyEvent]

*Maus*: selbsterklärend; Ereignisse treten nur ein, wenn vorher die Maus innerhalb des Feldes ist oder war («außerhalb» entspricht javascript onMouseOut). [com.sun.star.awt.MouseEvent]

*Zurücksetzen*: Das Formular wird von allen Daten geleert (Anlegen eines neuen Datensatzes) oder auf den alten Datenstand zurück gesetzt (Änderung eines bestehenden Datensatzes). Bei einem Formularfeld wird das Ereignis nur ausgelöst, wenn über den Button in der Navigationsleiste die Dateneingabe rückgängig gemacht wird. [com.sun.star.lang.EventObject]
Wenn ein Formular aufgerufen wird, wird nacheinander das Ereignis *Vor dem Zurücksetzen* und *Nach dem Zurücksetzen* abgearbeitet, bevor das Formular für eine Eingabe verfügbar ist.

*Aktualisieren*: Ist dies Ereignis an ein Kontrollfeld des Formulars gebunden, so tritt die Aktualisierung bei Fokusverlust und Sprung zu einem anderen Formularfeld auf, wenn der Inhalt des Kontrollfeldes geändert wurde. Änderungen in dem Formular werden übernommen und angezeigt. Bei Schließen eines Formular werden nacheinander die Ereignisse *Vor dem Aktualisieren* und *Nach dem Aktualisieren* abgearbeitet. [com.sun.star.lang.EventObject]

## *Textfeld*

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

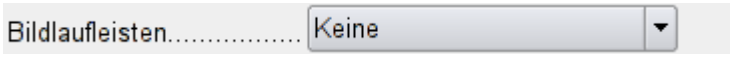
### Allgemein

Solange der Wert 0 ist wird die Einstellung nicht berücksichtigt. In der Praxis wird hier die Zeichenlänge des Feldes aus der Datenbank übernommen, auf das sich das Textfeld bezieht.
[MaxTextLen]

Soll in einem leeren Feld ein Standardtext erscheinen? Dieser Text muss gelöscht werden, wenn ein anderer Eintrag erfolgen soll.
[DefaultText]

Mögliche Typen: Einzeilig | Mehrzeilig | Mehrzeilig mit Formatierungen (wobei sich die beiden letzten nur im Tabulator unterscheiden – und ein Feld mit Formatierungen nicht an eine Datenbank angebunden werden kann). Bei mehrzeiligen Feldern ist die vertikale Ausrichtung nicht aktiv.
[MultiLine]

Unix oder Windows? Prinzipiell funktionieren beide Endungen. Intern müssen für Windowszeilenenden aber zwei Steuerzeichen verwendet werden (CR und LF).
[LineEndFormat]




Nur bei mehrzeiligen Feldern: Horizontal | Vertikal | Beide
Nur bei eingeschalteter Bildlaufleiste funktioniert das Scrollen mit der Maus.
[HScroll], [VScroll]

Aktiv nur bei einzeiligen Feldern.
[EchoChar]

**Daten**

keine weiteren Besonderheiten

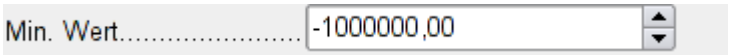
**Ereignisse**

keine weiteren Besonderheiten

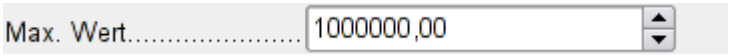
## *Numerisches Feld*

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**




Mindestwert, den dieses Feld einnehmen kann. Sollte mit dem Mindestwert übereinstimmen, der in der Tabelle erwartet wird.
[ValueMin]

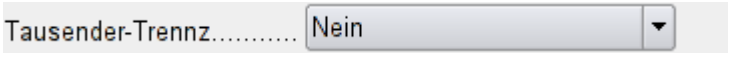



Maximalwert
[ValueMax]

Intervall-Wert für die Funktion als Scrollelement per Mausrad bzw. Drehfeld
[ValueStep]

Wert, der beim Erstellen eines neuen Datensatzes angezeigt wird.
[DefaultValue]

Anzahl Nachkommastellen, bei Integer-Feldern auf 0 zu stellen
[DecimalAccuracy]




Trennzeichen für Tausenderstellen, in der Regel der Punkt
[ShowThousandsSeparator]

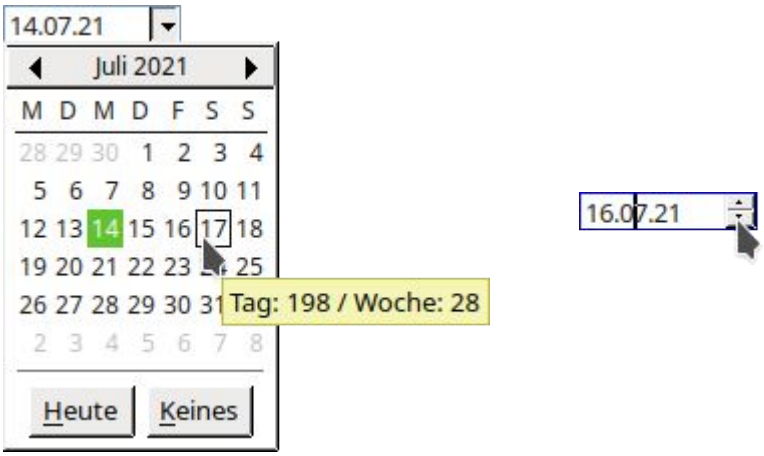
**Daten**

Es erfolgt keine Nachfrage, ob das Feld NULL sein soll. Eine fehlende Eingabe belässt dennoch das Feld auf NULL, nicht auf 0.
Ein Filtervorschlag wird ebenfalls nicht erstellt.

**Ereignisse**

Es fehlt das Feld «Modifiziert». Änderungen werden über «Text modifiziert» (hier wohl nicht wörtlich zu nehmen) angesprochen.

## *Datumsfeld*



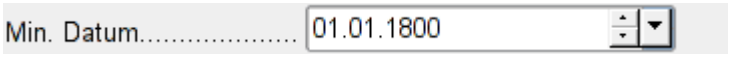

Das Datumsfeld kann als aufklappbares Feld Feld und als Drehfeld definiert werden.

Beim aufklappbaren Feld ist ein Monatskalender eingeblendet. Ein Klick auf «Heute» gibt das aktuelle Datum ein. In der Tipp-Hilfe erscheint dann auch noch genau der Tag im Jahr und die entsprechende Woche.

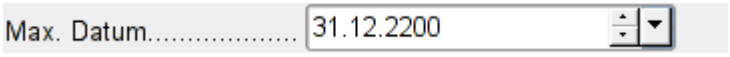
Beim Drehfeld entscheidet der Stand des Cursors darüber, welcher Wert geändert wird. Steht der Cursor, wie im Screenshot, im Wert für den Monat, so verändert ein Klick auf das Drehfeld den Wert für den Monat.

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**



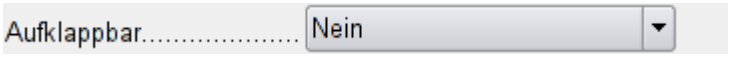

Mindestwert für das Feld, einstellbar über ein Aufklappfeld, das einen Kalender bereitstellt.
[DateMin]




Maximalwert.
[DateMax]

Kurzform wie 10.02.12 sowie unterschiedliche Formen auch mit `'/'` statt `'.'` oder `'-'` in amerikanischer Schreibweise.
[DateFormat]

Hier kann ein festes Datum vorgegeben werden. Das aktuelle Datum (Heute) beim Aufruf des Formulars muss leider (noch) durch ein Makro eingetragen werden.
[DefaultDate]




Ein Monatskalender zur Auswahl des Tages kann eingeblendet werden.
[DropDown]

**Daten**

Es erfolgt keine Nachfrage, ob das Feld NULL sein soll. Eine fehlende Eingabe belässt dennoch das Feld auf NULL, nicht auf 0.
Ein Filtervorschlag wird ebenfalls nicht erstellt.

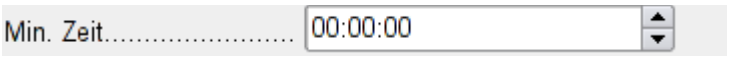
**Ereignisse**

Es fehlt das Feld «Modifiziert». Änderungen werden über «Text modifiziert» (hier wohl nicht wörtlich zu nehmen) angesprochen.

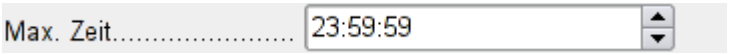
## *Zeitfeld*

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

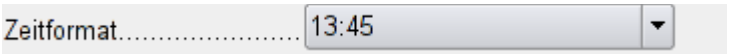
**Allgemein**




Mindestwert für das Feld, standardmäßig auf 0 gesetzt.
[TimeMin]




Maximalwert, standardmäßig auf 1 Sekunde unter 24 Uhr gesetzt.
[TimeMax]




Kurzform ohne Sekunden, Langform mit Sekunden sowie Benennungen mit PM (post meridiem / Nachmittag)
[TimeFormat]

Eine feste Zeit ist voreinstellbar, die aktuelle Zeit beim Abspeichern des Formulars leider (bisher) nur mit Makro.
[DefaultTime]

**Daten**

Es erfolgt keine Nachfrage, ob das Feld NULL sein soll. Eine fehlende Eingabe belässt dennoch das Feld auf NULL, nicht auf 0.
Ein Filtervorschlag wird ebenfalls nicht erstellt.

**Ereignisse**

Es fehlt das Feld «Modifiziert». Änderungen werden über «Text modifiziert» (hier wohl nicht wörtlich zu nehmen) angesprochen.

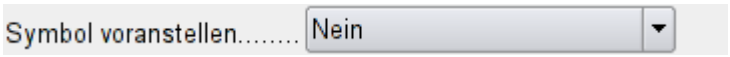
## *Währungsfeld*

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Min. Wert, Max. Wert, Intervall, Standardwert, Nachkommastellen und Tausender-Trennzeichen entsprechen den allgemeinen Eigenschaften *Numerisches Feld*. Daneben gibt es lediglich:

Das Symbol wird angezeigt, aber nicht in der dem Formular zugrundeliegenden Tabelle mit abgespeichert. [CurrencySymbol]




Soll das Symbol vor oder hinter der Zahl erscheinen? [PrependCurrencySymbol]

**Daten**

Es erfolgt keine Nachfrage, ob das Feld NULL sein soll. Eine fehlende Eingabe belässt dennoch das Feld auf NULL, nicht auf 0.
Ein Filtervorschlag wird ebenfalls nicht erstellt.

**Ereignisse**

Es fehlt das Feld «Modifiziert». Änderungen werden über «Text modifiziert» (hier wohl nicht wörtlich zu nehmen) angesprochen.

## *Formatiertes Feld*

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Min. Wert und Max. Wert und Standardwert hängen von der Formatierung ab. Hinter dem Button zur Formatierung versteckt sich ein «Allroundfeld", das Währungsfeld und Zahlenfeld meist überflüssig macht. Im Gegensatz zum einfachen Währungsfeld kann das formatierte Feld negative Beträge auch in roter Farbe darstellen.

Über den rechten Button mit den drei Punkten erscheint die Auswahl zum Zahlenformat, die auch in LibreOffice-Calc üblich ist. [FormatKey]

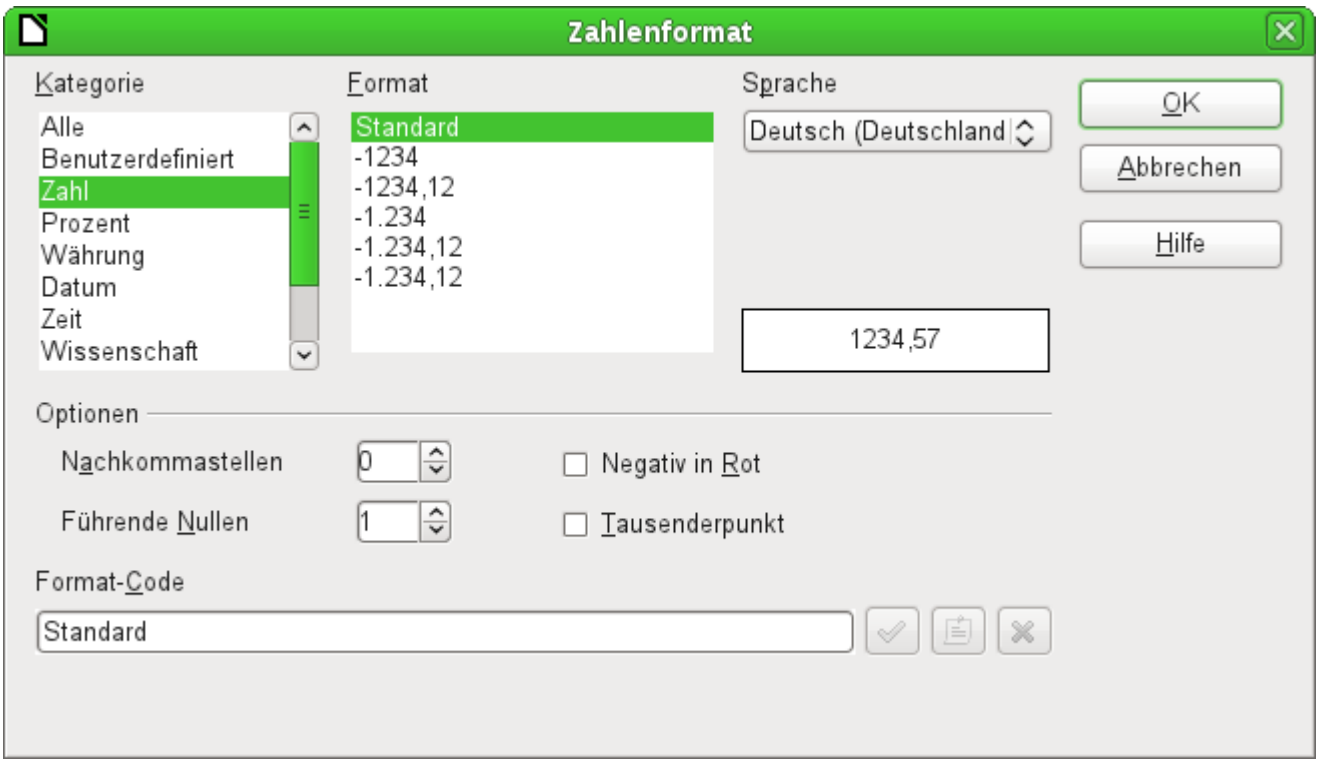



*Abbildung 4: Formatiertes Feld mit allgemeinen Zahleneinstellungen*

In dem Zahlenformat sind dann neben Datum, Zeit, Währung oder normalen Zahlenformaten auch Möglichkeiten gegeben, die Felder mit einer Maßeinheit wie z. B. **kg** zu bestücken. Siehe dazu die allgemeine Hilfe zu Zahlenformat-Codes.

Mit dem formatierten Feld ist es auch möglich, Timestamp-Felder aus Tabellen mit nur einem Feld darzustellen und zu beschreiben. Der Formularassistent nutzt dazu sonst ein Datumsfeld und ein Zeitfeld.

Nur mit Hilfe von Makros ist es bei Timestamp-Feldern möglich, Eingaben im Format Minuten:Sekunden,Hundertstelsekunden zu erreichen.

Das Zahlenformat erlaubt auch mit Einstellungen im Format-Code wie z.B.

```
[<20][BLAU]0" °C";[>30][ROT]0" °C";[SCHWARZ]0" °C "
```

Werte entsprechend farbig darzustellen. Hier im Beispiel Temperaturen unter 20°C in Blau und über 30°C in Rot, sonst in Schwarz.

| **Hinweis** | Werden Formulare mit Hilfe des Assistenten erstellt, dann wird der Schrift in den Formularfeldern eine Farbe fest zugewiesen. Die formatierten Felder richten sich dann nicht nach dem vorgesehenen Format-Code. Dies kann in dem Steuerelement unter **Eigenschaften → Allgemein → Schrift → Button ... → Schrifteffekte → Schriftfarbe** mit der Einstellung **Automatisch** rückgängig gemacht werden. Das Steuerelement übernimmt die Eigenschaft, auch wenn es nachher wieder statt «Automatisch» «Schwarz» anzeigt. |
|---|---|

**Daten**

keine weiteren Besonderheiten.

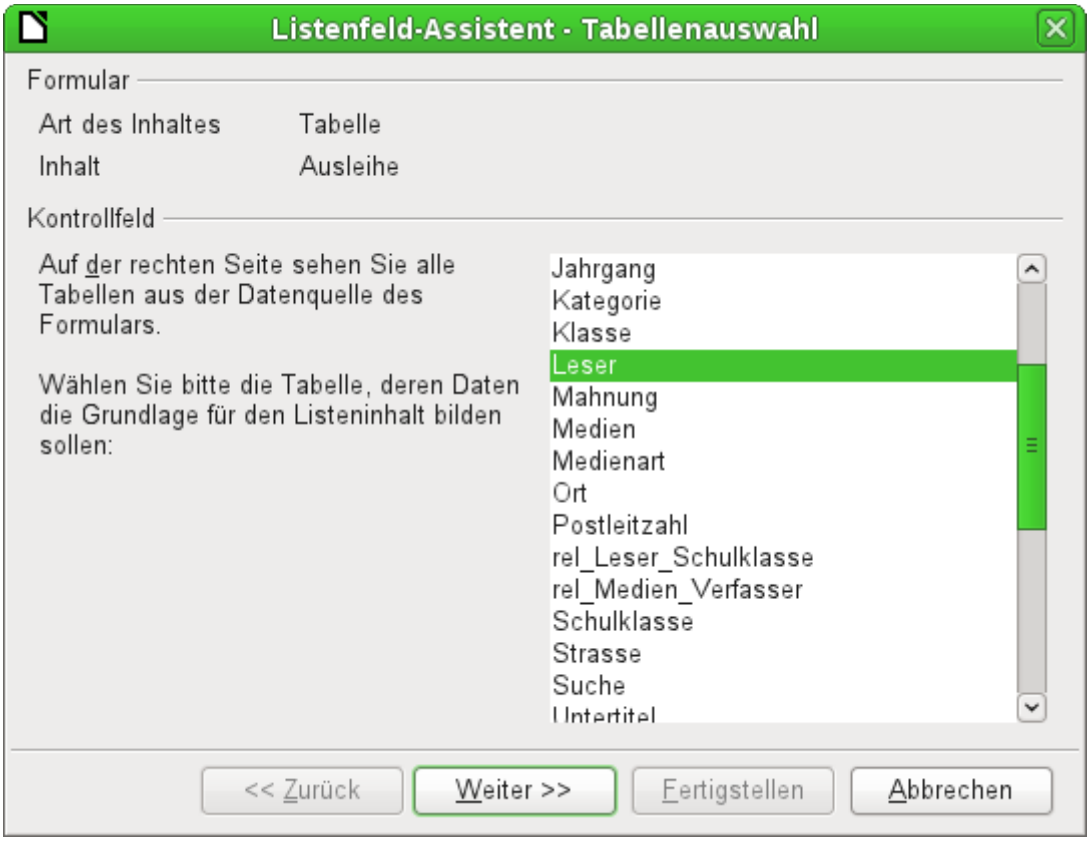
**Ereignisse**

Es fehlt das Feld «Modifiziert». Änderungen werden über «Text modifiziert» (hier wohl nicht wörtlich zu nehmen) angesprochen.

### *Listenfeld*

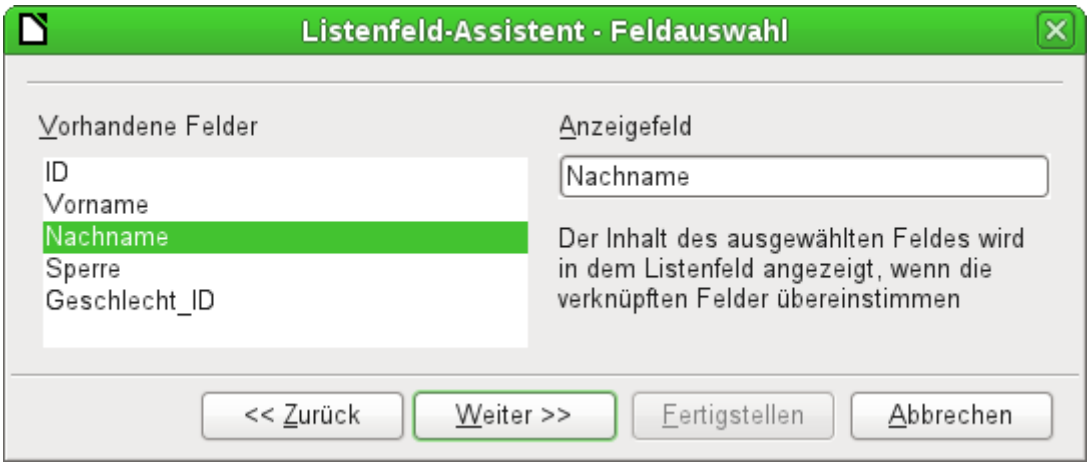
Sobald ein Listenfeld erstellt wird, erscheint standardmäßig der Listenfeldassistent. Diese Automatik lässt sich gegebenenfalls über den Steuerelemente-Assistenten an/aus (*Abbildung 3*) abschalten.

**Assistent**

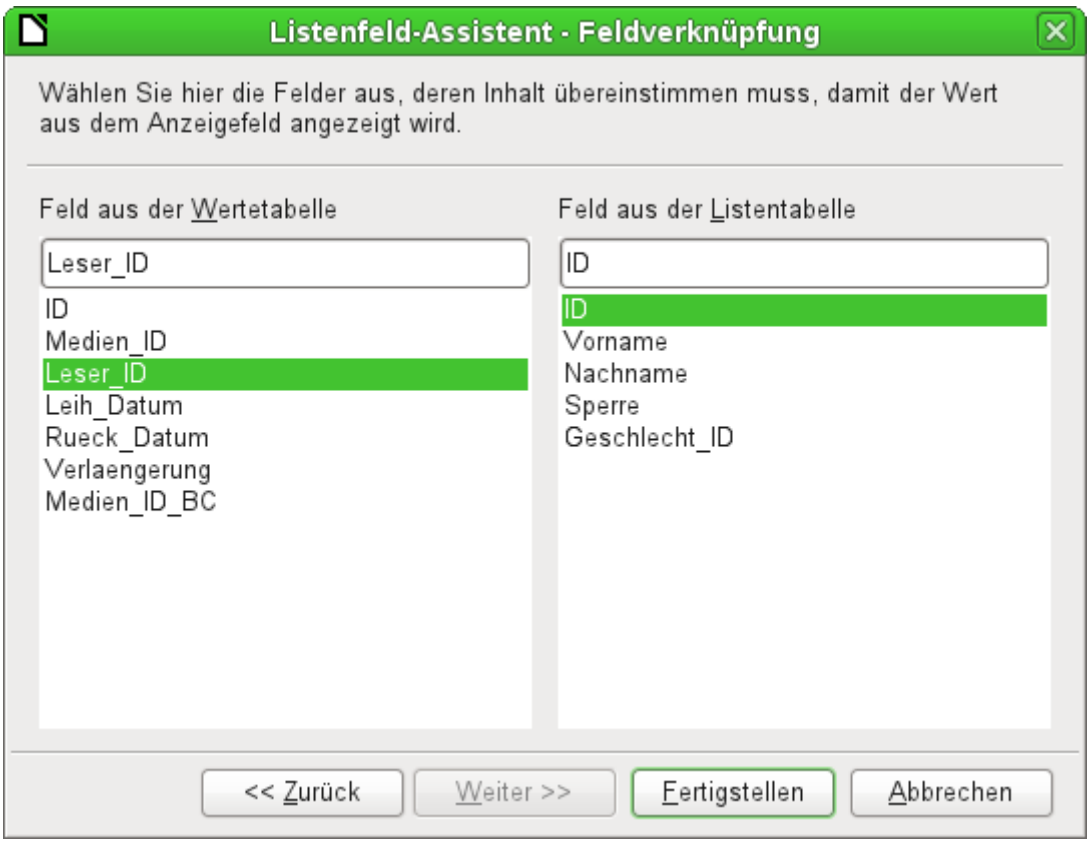



Das Formular ist bereits definiert. Es verbindet mit einer *Tabelle*, die den Namen "Ausleihe" trägt. Ein Listenfeld zeigt andere Daten für den Nutzer an als solche, die an die *Tabelle* weitergegeben werden. Diese Daten stammen bei Datenbanken in der Regel aus einer anderen Tabelle als der, mit der das Formular verbunden ist.

In der Tabelle "Ausleihe" soll verzeichnet werden, welcher "Leser" welche Medien entliehen hat. Allerdings wird in dieser Tabelle nicht der Name des Lesers gespeichert sondern der Primärschlüssel aus der Tabelle "Leser". Die Tabelle "Leser" bildet also die Grundlage für das Listenfeld.



Das Feld "Nachname" aus der Tabelle "Leser" soll in dem Listenfeld sichtbar sein. Es ist also das *Anzeigefeld*.




Das Feld "Leser_ID" befindet sich in der dem Formular zugrundeliegenden Tabelle "Ausleihe". Diese Tabelle wird hier als *Wertetabelle* bezeichnet. Der Primärschlüssel "ID" aus der Tabelle "Leser" soll mit diesem Feld verbunden werden. Die Tabelle "Leser" wird hier als *Listentabelle* bezeichnet.

Das Listenfeld ist jetzt komplett mit Daten und Standardeinstellungen erstellt und funktionsfähig.

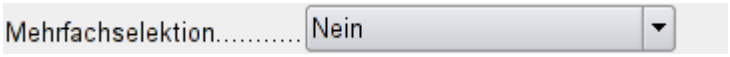
Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Die Listeneinträge wurden bereits durch den Assistenten erstellt. Hier könnten sonst auch Einträge stehen, die in keiner Tabelle der Datenbank verzeichnet sind. Mit Listeneinträgen sind hier die sichtbaren Einträge gemeint; nicht die Einträge, die über das Formular an die Tabelle weitergegeben werden. [StringItemList]

Wird das Feld als nicht aufklappbar festgelegt, so erscheinen bei Aufruf des Formulars am rechten Rand des Listenfeldes Scrollpfeile. Das Listenfeld wird dann automatisch zu einem mehrzeiligen Feld, bei dem die aktuelle Auswahl markiert ist.
[Dropdown]

Falls aufklappbar wird hier die maximal sichtbare Anzahl der Zeilen eingegeben. Geht der Inhalt über mehr Zeilen, so erscheint beim Aufklappen ein Scrollbalken.
[LineCount]




Sollen mehrere Werte selektiert werden können? Im obigen Beispiel ist das nicht möglich, da ein Fremdschlüssel abgespeichert wird.
In der Regel wird diese Funktion in Verbindung mit Datenbanken nicht genutzt werden, da jedes Feld eigentlich nur einen Wert einnehmen sollte. Gegebenenfalls kann eine Auswertung des Eintrags in das Listenfeld über Makros hier Abhilfe schaffen.
[MultiSelection] [MultiSelectionSimpleMode]

Schon der deaktivierte Button macht deutlich, dass auch eine Standardselektion bei einer Verbindung mit einer Datenbanktabelle, wie über den Listenfeldassistenten erzeugt, wenig Sinn macht. Schließlich kann es passieren, dass der der Standardselektion entsprechende Datensatz in der Beispieltabelle "Leser" gar nicht mehr vorhanden ist.
[DefaultSelection]

**Daten**

Neben den üblichen Daten-Eigenschaften Datenfeld und Eingabe erforderlich sind hier Eigenschaften von Bedeutung, die die Verbindung von anzuzeigenden Daten und in die dem Formular zugrundeliegende Tabelle einzutragenden Daten herstellen.

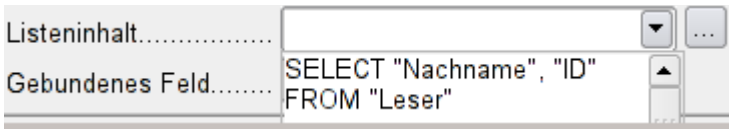
*Art des Listeninhaltes*: Werteliste | Tabelle | Abfrage | SQL | SQL (Native) | Tabellenfelder [ListSourceType]

*Listeninhalt Werteliste*: Sind unter 'Allgemein' Listeneinträge gemacht worden, so werden hier die entsprechenden Werte eingegeben, die mit dem Formular abgespeichert werden sollen. Der Listeninhalt wird bestückt, indem bei der Eingabe die Inhalte über **`Shift – Enter`** aneinandergehängt werden. Es erscheint dann in dem Listeninhalt **`"Wert1";"Wert2";"Wert3"`** **…** Die Eigenschaft *'Gebundenes Feld'* ist *inaktiv*.

*Listeninhalt Tabelle*: Hier wird eine der Datenbanktabellen ausgesucht. Dabei muss aber berücksichtigt werden, dass dann immer das erste Feld der Tabelle in dem Listenfeld angezeigt wird.
Üblicherweise sind Tabellen so aufgebaut, dass zuerst der Primärschlüssel erscheint und anschließend die Inhalte. Solche Tabellen eignen sich nur dann für ein Listenfeld, wenn tatsächlich der Primärschlüssel angezeigt werden soll.

*Listeninhalt Abfrage*: Hier kann extern eine Abfrage vorformuliert werden, die dann auch als Abfrage sichtbar abgespeichert wird. Die Gestaltung solcher Abfragen wird im Kapitel 'Abfragen' erklärt. Durch die Abfrage ist es dann möglich, das Feld "ID" von der ersten Position in der ursprünglichen Tabelle an die zweite Position zu bewegen, was hier dem gebundenen Feld 1 entspricht.

*Listeninhalt SQL*: Hiermit bestückt der Listenfeldassistent das Listenfeld. Die von dem Assistenten konstruierte Abfrage sieht dann so aus:




Die Abfrage ist die einfachste Möglichkeit, die sich bietet. Das Feld "Nachname" erscheint an Position 0, das Feld "ID" an Position 1. Beide werden aus der Tabelle "Leser" ausgelesen. Da das gebundene Feld das Feld 1 ist reicht diese SQL-Formulierung aus. Schon vom Standard her sollte aber hier ergänzt werden **`ORDER BY "Nachname" ASC`**, denn ohne diese Formulierung werden die Nachnamen so wiedergegeben, dass ihre Reihenfolge der Eingabe in die Tabelle Leser entspricht. Zusätzliches Problem dürfte sein, dass "Nachname" bei einigen Lesern gleich ist. Hier muss dann "Vorname" hinzugezogen werden können und, als letztes Mittel, wohl auch der Primärschlüssel "ID". Wie so etwas genauer formuliert wird, kann im Kapitel 'Abfragen' genauer nachgelesen werden.

*Listeninhalt SQL (Nativ)*: Die SQL-Formulierung wird direkt eingegeben, kein Assistent wird dazu aufgerufen. Base wertet die Abfrage nicht aus. Dies bietet sich an, wenn in der Abfrage Funktionen stecken, die eventuell von der GUI von Base nicht verstanden werden. Dann wird die Abfrage nicht weiter auf Fehler untersucht. Genaueres zu dem direkten SQL-Modus im Kapitel 'Abfragen'.

*Listeninhalt Tabellenfelder*: Hier werden die *Feldnamen* einer Tabelle aufgelistet, nicht die Inhalte. Für die Tabelle "Leser" erscheint dort also die Liste "ID", "Vorname", "Nachname", "Sperre", "Geschlecht_ID".

**Hinweis**

Für Zeitfelder, die auch Zeiten im Millisekunden-Bereich darstellen sollen, sind, wie unter «Zeitfelder in Tabellen» beschrieben, Timestamp-Felder erforderlich. Die Darstellung der Millisekunden funktioniert bei der Zusammenfügung von Zeichen in einem Listenfeld nicht. Hier muss der Timestamp in Text umgewandelt werden:

```
SELECT REPLACE(LEFT(RIGHT(CAST("Zeit" AS
VARCHAR(30)),15),8),'.',',') AS "Listinhalt", "ID"
FROM "Leistung_erforderlich"
```

Dies erzeugt die Anzeige in Minuten:Sekunden,Hundertstelsekunden. (HSQLDB)

*Gebundenes Feld*: Im Listenfeld wird ein Inhalt angezeigt, der nicht mit dem Inhalt identisch sein muss, der auch im Formular abgespeichert werden soll. Üblicherweise wird z. B. Ein Name angezeigt und der Primärschlüssel zu diesem Namen als abzuspeicherndes Feld bestimmt.

```
SELECT "Name", "ID" FROM "Tabelle" ORDER BY "Name"
```

Das Feld "ID" wird als Fremdschlüssel in der Tabelle gespeichert, die dem Formular zugrunde liegt. Die Zählung der Felder in Datenbanken beginnt mit 0, das an das Formular gebundene Feld hat also die Position 1.

Wird stattdessen «0» gewählt, dann wird in dem Formular der Inhalt des Feldes "Name" abgespeichert. Das Feld "ID" könnte in der Abfrage entfallen.

Es ist sogar möglich, die Position «-1» zu wählen. Dann wird nicht der Inhalt der Abfrage sondern die Position des Eintrages in der Liste gespeichert. Der erste Datensatz hat dabei die Position 1.

Die oben aufgeführte Abfrage ergäbe z. B. die folgenden Daten:

| ***Name*** | ***ID*** |
|---|---|
| Anneliese | 2 |
| Dorothea | 3 |
| Sieglinde | 1 |

Im Listenfeld könnte nur der Name ausgewählt werden. Das Listenfeld wird auf den Namen «Dorothea» eingestellt. Folgende Abspeicherungen würden über das Listenfeld möglich:
Gebundenes Feld = 1> «3» wird abgespeichert, da das Feld "ID" gespeichert wird.
Gebundenes Feld = 0 > «Dorothea» wird abgespeichert, da das Feld "Name" gespeichert wird.

Gebundenes Feld = -1 > «2» wird abgespeichert, da «Dorothea» als zweiter Datensatz in der Liste aufgeführt wird.

**Hinweis**

Die Änderung des gebundenen Feldes auf «0» oder «-1» ist erst mit der Version LO 4.1 eingeführt worden. Vorher war nur die Auswahl von Werten >= 1 möglich.

**Hinweis**

Ein Listenfeld zeigt in der Regel nicht nur die Werte an, die z.B. über SQL dargestellt werden. Als erstes wird ein leeres Feld (**NULL**) zur Auswahl angeboten. Dies ist notwendig, damit einmal eingegebene Daten über das Listenfeld auch wieder entfernt werden können.

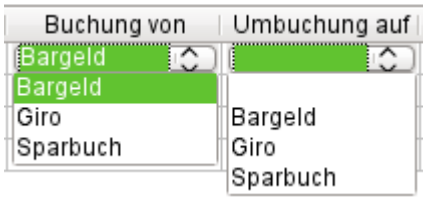



*Abbildung 5: Für das Buchungskonto ist in der Tabelle eine Eingabe erforderlich. Das Umbuchungskonto kann auch leer bleiben.*

Soll von vornherein ausgeschlossen werden, dass das leere Feld angeboten wird, so ist die zugrundeliegende Tabelle für das Formular entsprechend einzustellen. Das Feld muss über **Feldeigenschaften → Eingabe erforderlich → Ja** so eingestellt sein, dass **NULL** nicht erlaubt ist.

**Ereignisse**

Neben den Standardereignissen werden folgende Ereignisse hinzugefügt:

Aktion ausführen: Wird durch Tastatureingabe ein Listenfeld dazu veranlasst, einen entsprechenden Inhalt anzuzeigen, so ist dies eine Aktion, die das Listenfeld ausführt.
Auch die Auswahl des Wertes aus einem Listenfeld entspricht dieser Aktion. Je nach System kann hier auch ein Doppelklick mit der Maus erforderlich sein.

Status geändert: Dies kann die Änderung des angezeigten Inhaltes eines Listenfeldes durch Betätigen des Aufklappbuttons sein. Es kann auch lediglich ein Klicken auf den Aufklappbutton des Feldes sein.

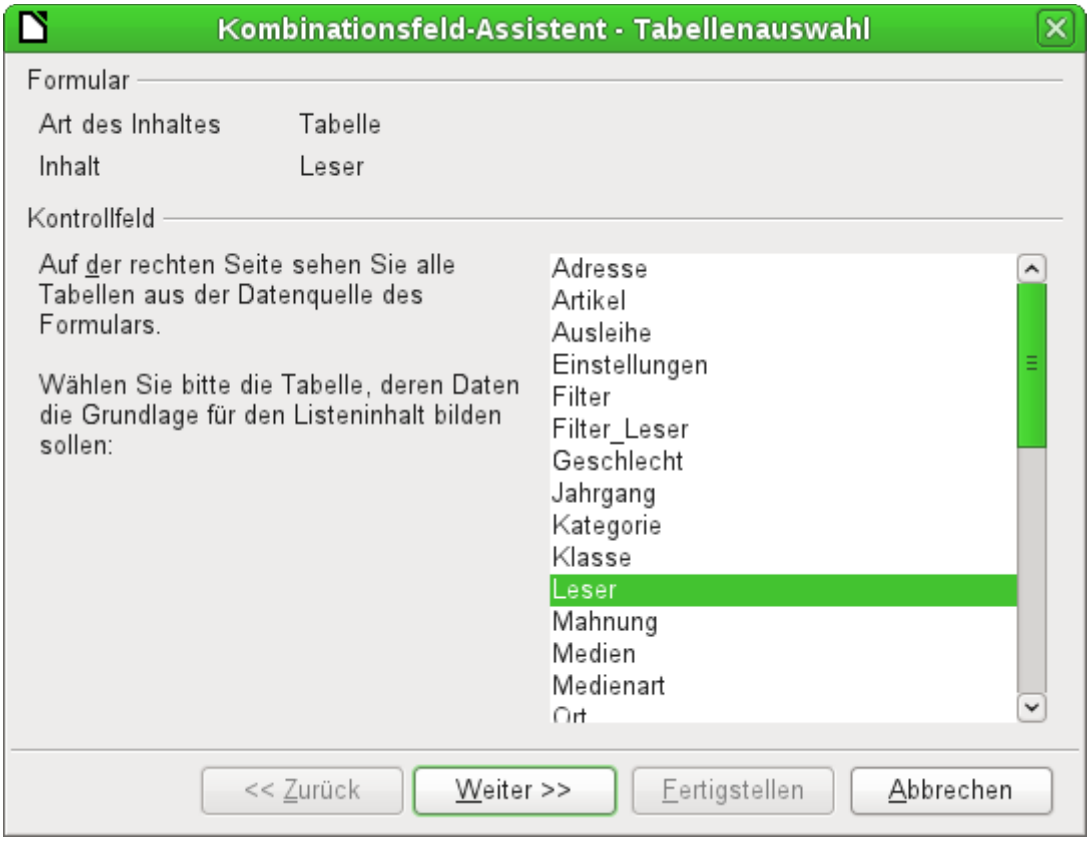
Fehler aufgetreten: Das Ereignis lässt sich leider beim Listenfeld nicht nachvollziehen.

### *Kombinationsfeld*

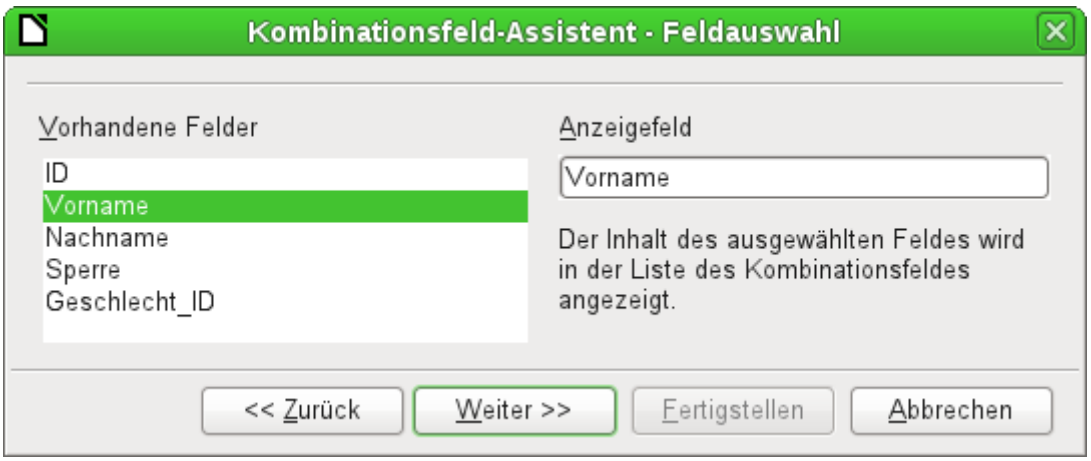
Sobald ein Kombinationsfeld erstellt wird erscheint wie beim Listenfeld standardmäßig ein Assistent. Diese Automatik lässt sich gegebenenfalls über Steuerelemente-Assistenten an/aus (*Abbildung 3*) abschalten.

Kombinationsfelder schreiben direkt den ausgewählten Wert in die dem Formular zugrundeliegende Tabelle. Deshalb ist im folgenden Beispiel die dem Formular zugrundeliegende Tabelle die Tabelle "Leser" und die für das Kontrollfeld gewählte Tabelle ebenfalls die Tabelle "Leser".

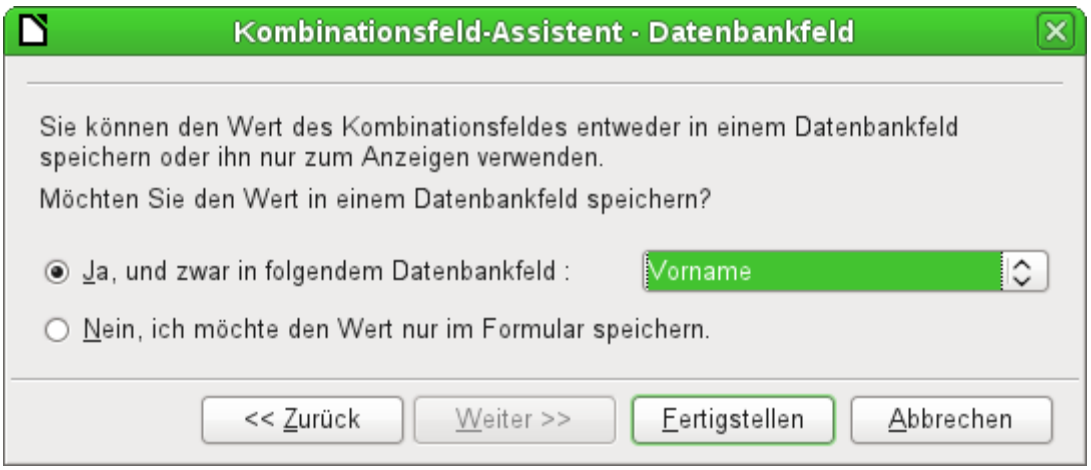
**Assistent**




Das Formular ist wieder vordefiniert, diesmal als Beispiel mit der Tabelle "Leser". Da die Daten, die in dem Listenfeld angezeigt werden, auch in dieser Tabelle abgespeichert werden sollen, wird als Quelle für die Daten des Listenfeldes ebenfalls die Tabelle "Leser" ausgewählt.




In der Tabelle "Leser" befindet sich das Feld "Vorname". Es soll im Kombinationsfeld angezeigt werden.




Innerhalb einer Datenbank scheint es wenig Sinn zu machen, den Wert des Kombinationsfeldes nicht in einem Feld zu speichern. Aus der Tabelle "Leser" sollen die Vornamen ausgelesen werden und für neue Leser als Vornamen zur Verfügung stehen, damit bei gleichen Vornamen eben nicht immer wieder eine neue Eingabe erfolgen muss. Das Kombinationsfeld zeigt dann hier den Vornamen an und die Texteingabe braucht nicht weiter durchgeführt zu werden.

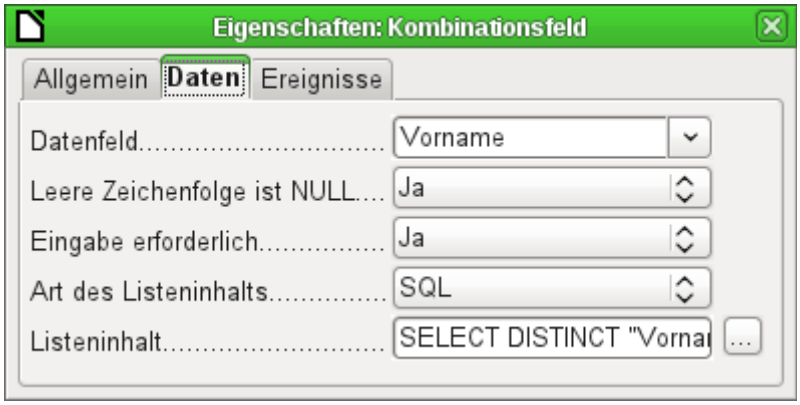
Soll ein neuer Wert eingegeben werden, so ist dies im Kombinationsfeld ohne weiteres möglich, denn das Kombinationsfeld zeigt genau das an, was es selbst an die zugrundeliegende Tabelle des Formulars weitergibt.

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* und beim Listenfeld erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Während der Eingabe von neuen Werten wird eine Liste der eventuell zutreffenden Werte angezeigt, die noch zur Auswahl stehen. [AutoComplete]

**Daten**




Die Datenfeldern entsprechen den bereits vorgestellten Standardeinstellungen und den Einstellungen bei einem Listenfeld. Der SQL-Befehl weist hier allerdings eine Besonderheit auf:

```
SELECT DISTINCT "Vorname" FROM "Leser"
```

Durch den Zusatz des Begriffes **`DISTINCT`** werden gleiche Vornamen nur einmal angezeigt. Wieder fehlt allerdings bei der Erstellung durch den Assistenten eine Sortierung der Inhalte.

**Ereignisse**

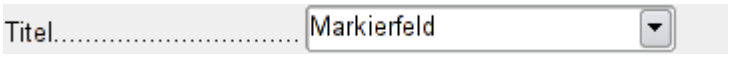
Die Ereignisse entsprechen denen beim Listenfeld.

## *Markierfeld*

Das Markierfeld erscheint direkt als eine Kombination von Markierbox und Beschriftung dieser Box.

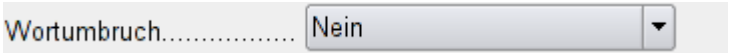
Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**




Die Beschriftung dieser Box, erscheint standardmäßig rechts von der Box. Zusätzlich ist noch eine Verbindung zu einem Beschriftungsfeld möglich.
[Label]

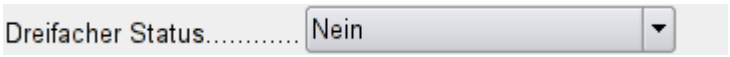
Hier gibt es, abhängig von der Auswahl in 'Dreifacher Status', bis zu drei Möglichkeiten: *Nicht ausgewählt* | *Ausgewählt* | *Nicht definiert*. *Nicht definiert* entspricht als Eintrag in der dem Formular zugrundeliegenden Tabelle *NULL*.
[State]




Standardmäßig wird der Titel nicht umbrochen. Ein zu langer Titel wird abgeschnitten, wenn das Feld nicht groß genug ist.
[MultiLine]

Hier kann eine Grafik statt bzw. zusätzlich zum Titel eingefügt werden. Dabei sollte beachtet werden, dass die Grafiken standardmäßig erst einmal nicht in das *.odb-Dokument eingebunden werden. Sinnvoll bei kleinen Grafiken ist es, die Grafiken einzubetten und nicht zu verknüpfen.
[Graphic]

Ist eine Grafik ausgewählt, so kann sie im Verhältnis zum Titel ausgerichtet werden.
[ImagePosition] (0=links | 1=zentriert | 2=rechts)




Standardmäßig gibt es für Markierfelder nur *Ausgewählt* (Wert: 1) und *Nicht ausgewählt* (Wert: 0). Beim dreifachen Status kommt die Definition als *leeres Feld* (NULL) hinzu. [TriState]

**Daten**

Dem Markierfeld kann ein Referenzwert [RefValue] zugewiesen werden. In der Regel ist dies der Wert 1 für (ein) und 0 für (aus). Schließlich werden Markierfelder meist als Felder für die Auswahl von 'Ja' und 'Nein' genutzt.. Die Referenzwerte können auch andere Zahlen und Texte enthalten.

Wird ein Markierfeld an ein Textfeld gebunden, so wird an das Textfeld ab der Version LO 4.2 standardmäßig «true» und «false» weiter gegeben. Allerdings liest das Markierfeld auch die Werte 1 und 0 korrekt aus. Eine klare Wahl des Referenzwertes als 1 und 0 dient dann dazu, auch 1 und 0 in die Felder zu schreiben.

**Ereignisse**

Es fehlen die Felder «Modifiziert», «Text modifiziert». «Vor der Aktualisieren» und «Nach dem Aktualisieren».

Zusätzlich erscheinen «Aktion ausführen» (siehe Listenfeld) und «Status geändert» (entspricht «Modifiziert»).

## *Optionsfeld*

Das Optionsfeld entspricht bis auf eine Ausnahme in den allgemeinen Eigenschaften und die äußerlich andere Form (rund) dem eben beschriebenen Markierfeld.
Werden mehrere Optionsfelder mit gleichem Namen oder unterschiedlichem Namen, aber gleichem Gruppennamen, mit einem Tabellenfeld über das Formular verbunden, so kann von den Optionen immer nur eine Option gewählt werden. Gleiche Namen sollten möglichst vermieden werden, da dann die Navigation mit Tabulatoren zur Zeit nicht korrekt funktioniert.

**Allgemein**

Das Optionsfeld wird bevorzugt für Gruppierungen verwandt. Zwischen mehreren Optionen kann dann nur eine ausgewählt werden. Deswegen erscheint hier auch der Gruppenname, unter dem die Option angesprochen werden kann.
[GroupName]

**Daten**

Siehe Markierfeld, hier werden aber die Referenzwerte [RefValue], die eingetragen wurden, auch tatsächlich an das Datenfeld weitergegeben.

**Ereignisse**

Siehe Markierfeld

### *Grafisches Steuerelement*

Ein grafisches Steuerelement übernimmt die Eingabe und Ausgabe von Bildmaterial in die Datenbank. Das darunterliegende Datenfeld muss ein Binärfeld sein, wenn es das Bild direkt speichern soll. Es kann auch ein Textfeld sein. Dann wird lediglich der relative Pfad zur Datei abgespeichert. Hierbei sollte dann aber darauf geachtet werden, dass der Pfad zu der Datei auch bei einer Weitergabe der Datenbank stimmig bleibt.

| **Hinweis** | Wird über einen Assistenten ein Formular erstellt, so erstellt der Assitent nur für Binärfelder ein grafisches Kontrollfeld. Bei Feldern, die den Pfad speichern sollen, muss dann das Textfeld mit der rechten Maustaste über **Ersetzen durch → Grafisches Steuerelement** zu einem grafischen Steuerelement umgewandelt werden. |
|---|---|

Seit LO 5.0 können über dieses Feld beliebige Dateien, abhängig von den Dialogen der Benutzeroberfläche des Betriebssystems, in Base eingebunden werden. Angezeigt werden in dem Fall aber nur Bilder und von *.pdf-Dateien die erste Seite.

| **Hinweis** | FIREBIRD: Der Feldtyp Bild [BLOB] funktioniert nicht mit dem grafischen Kontrollfeld. Hier wird der Feldtyp Blob [BLOB] benötigt. |
|---|---|

| **Vorsicht** | Bilder sollten auf jeden Fall verkleinert werden, wenn sie schon in der Datenbank gespeichert werden. Hat ein Kamerabild z.B. 3 MB, so passiert es sehr schnell bei Nutzung der internen HSQLDB, dass mit Angabe einer Java-Fehlermeldung (**`NullPointerException`**) die Abspeicherung der Daten nicht mehr möglich ist. Dies kann zum Verlust sämtlicher Daten führen, die seit dem Öffnen der Base-Datenbank in die Tabelle eingegeben wurden. |
|---|---|

Die Eingabe in das grafische Steuerelement erfolgt entweder über einen Doppelklick mit der Maus; dann erscheint ein Dateiauswahlfenster. Oder sie erfolgt über einen Klick der rechten Maustaste; dann kann ausgewählt werden, ob die Grafik nur gelöscht oder eine andere geladen werden soll.

Das grafische Steuerelement wird in Graustufen angezeigt, wenn das Formular z.B. auf einer Abfrage beruht, die nicht bearbeitet werden kann. Dann kann natürlich auch über das Kontrollfeld keine Eingabe erfolgen.

Ein grafisches Steuerelement erhält standardmäßig erst einmal keinen Tabstop.

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Die hier ausgesuchte Grafik wird nur im Kontrollfeld angezeigt, wenn das Formular bearbeitet wird. Für die spätere Eingabe ist sie ohne Bedeutung.
[Graphic]

*Nein*: Das Bild wird nicht an das Kontrollfeld angepasst. Ist das Bild zu groß, so wird ein Ausschnitt des Bildes gezeigt. Das Bild wird nicht verzerrt.
*Seitenverhältnis beibehalten*: Das Bild wird in das Kontrollfeld eingepasst, aber nicht verzerrt dargestellt.
*Autom. Größe*: Das Bild wird der Größe des Kontrollfeldes angepasst und gegebenenfalls verzerrt dargestellt.
[ScaleImage] [ScaleMode]

**Daten**

keine weiteren Besonderheiten

**Ereignisse**

Es fehlen die Ereignisse «Modifiziert», «Text modifiziert», «Vor dem Aktualisieren» und «Nach dem Aktualisieren».

## *Maskiertes Feld*

Mittels einer Eingabemaske wird hier die Eingabe in das Feld gesteuert. Zeichen werden für eine bestimmte Position vorgegeben, einzugebende Zeichen in ihrer Eigenschaft festgelegt. Die vorgegebenen Zeichen werden dann mit abgespeichert.

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Hier wird festgelegt, welche Inhalte eingegeben werden können.
[EditMask]

Diese Ansicht wird dem Formularnutzer präsentiert.
[LiteralMask]

Ist die Formatüberprüfung eingeschaltet, so wirkt sie direkt bei der Eingabe. Ansonsten sind erst einmal Falscheingaben möglich, die beim Verlassen des Feldes korrigiert werden.
[StrictFormat]

Der folgende Inhalt entstammt der LibreOffice-Hilfe:

Die Länge der Eingabemaske bestimmt, wie viele Stellen die mögliche Eingabe haben darf. Sollten die vom Benutzer eingegebenen Zeichen der Eingabemaske nicht entsprechen, so wird die Eingabe beim Verlassen des Kontrollfelds verworfen. Zur Definition der Eingabemaske stehen die folgenden Zeichen zur Verfügung:

| ***Zeichen*** | ***Bedeutung*** |
|---|---|
| L | Eine Textkonstante. Diese Stelle kann nicht editiert werden. Das Zeichen wird an der entsprechenden Position der Zeichenmaske angezeigt. |
| a | Es können die Buchstaben a-z und A-Z eingegeben werden. Großbuchstaben werden nicht in Kleinbuchstaben konvertiert. |
| A | Es können die Buchstaben A-Z eingegeben werden. Eingegebene Kleinbuchstaben werden automatisch in Großbuchstaben konvertiert. |
| c | Es können die Buchstaben a-z und A-Z sowie die Ziffern 0-9 eingegeben werden. Großbuchstaben werden nicht in Kleinbuchstaben konvertiert. |
| C | An dieser Stelle können die Zeichen A-Z und 0-9 angegeben werden. Wird ein Kleinbuchstabe angegeben, wird automatisch in Großschrift umgewandelt. |
| N | Es können nur die Zeichen 0-9 angegeben werden. |
| x | Es können alle druckbaren Zeichen angegeben werden. |
| X | Es können alle druckbaren Zeichen angegeben werden. Wird dabei ein Kleinbuchstabe verwendet, wird automatisch in Großschrift umgewandelt. |

Definieren Sie z. B. für die Zeichenmaske «__.__.2000» die Eingabemaske «NNLNNLLLLL», um dem Benutzer nur noch die Eingabe von 4 Ziffern zur Datumsangabe zu ermöglichen. Schalten sie unbedingt die Formatüberprüfung des Feldes ein.

**Daten**

keine weiteren Besonderheiten

**Ereignisse**

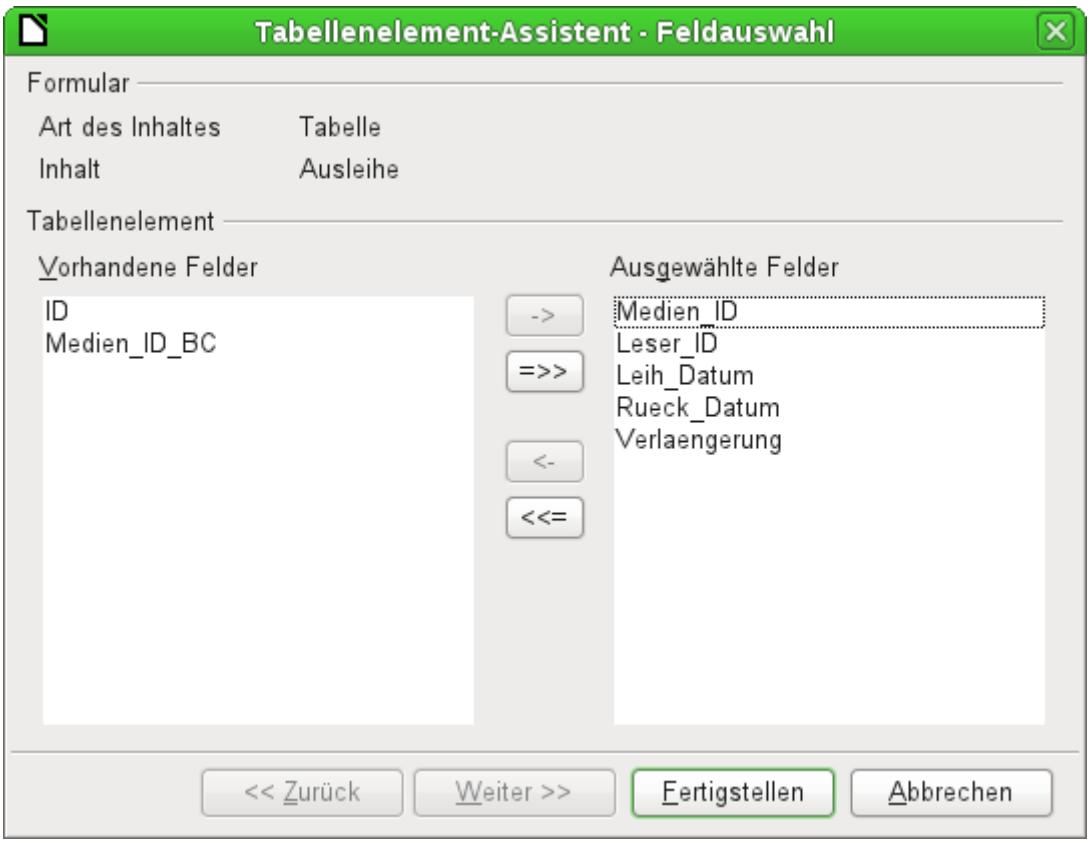
Es fehlt das Ereignis «Modifiziert»

### *Tabellen-Kontrollfeld*

Dieses Kontrollfeld ist das umfassendste Feld. Es stellt eine Tabelle bereit, die wiederum durch Kontrollfelder für die einzelnen Spalten beschickt werden kann. Damit ist bei der Eingabe nicht nur der Blick auf die aktuell einzugebenden Daten, sondern auch auf vorher eingegebene Daten möglich, ohne mit der Navigationsleiste durch die Datensätze zu scrollen.

Nicht alle Felder, die auch im Formular möglich sind, können im Tabellen-Kontrollfeld wieder ausgewählt werden. So fehlen z. B. Schaltflächen, das grafische Kontrollfeld und das Optionsfeld.

Ein Assistent für das Tabellen-Kontrollfeld stellt in einem Fenster die Felder zusammen, die anschließend in der Tabelle zur Verfügung stehen sollen.




In dem Kontrollfeld soll die Tabelle "Ausleihe" zur Bearbeitung zur Verfügung stehen. Bis auf die Felder "ID" (Primärschlüssel) und das Feld "Medien_ID_BC" (Eingabe der Medien über einen Barcodescanner) sollen alle Felder in dem Kontrollelement benutzt werden.

| Medien_ID | Leser_ID | Leih_Datum | Rueck_Datum | Verlaengerung |
|---|---|---|---|---|
| 1,00 | 0,00 | 02.11.11 | 04.11.11 | |
| 2,00 | 2,00 | 15.10.11 | 25.02.12 | 2,00 |
| 0,00 | 3,00 | 02.11.11 | 04.04.12 | 1,00 |
| 3,00 | 0,00 | 04.11.11 | 28.11.11 | 2,00 |
| 5,00 | 0,00 | 28.11.11 | | |
| 4,00 | 0,00 | 28.11.11 | 04.04.12 | |
| 4,00 | 0,00 | 09.11.11 | | |
| 3,00 | 0,00 | 09.12.11 | | |

Datensatz 1 von 18

*Abbildung 6: Ergebnis des Tabellen-Kontrollfeld-Assistenten*

Das erst einmal erstellte Tabellen-Kontrollelement muss hier noch weiter bearbeitet werden, da ja nur die Eingaben der Tabelle "Ausleihe" ermöglicht werden. Für Felder wie "Leser_ID" oder "Medien_ID" wäre es aber sinnvoller, statt einer Nummer den Leser bzw. das Medium selbst auswählen zu können. Hierzu werden innerhalb des Tabellen-Kontrollelementes z. B. Listenfelder eingesetzt. Entsprechendes ist weiter unten beschrieben. Auch die Formatierung des Feldes "Verlängerung" mit zwei Nachkommastellen ist sicher nicht gewollt.

| **Hinweis** | Wird ein Tabellenkontrollfeld ohne eingeschalteten Assistenten aufgezogen, so fehlen erst einmal die Spaltenköpfe. Die Spaltenköpfe können anschließend über einen rechten Mausklick auf die Kopfleiste (Position, an der die Spaltenköpfe erscheinen sollen) mit Hilfe eines Kontextmenüs erstellt werden. |
|---|---|

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

In welcher Höhe sollen die einzelnen Zeilen angezeigt werden. Ohne einen Wert wird die Höhe automatisch der Schriftgröße angepasst. Mehrzeilige Textfelder würden so nur einzeilig zum Scrollen angezeigt. [RowHeight]

Wie bei Tabellen werden am unteren Rand des Kontrollfeldes die Datensatznummer und Navigationshilfen angezeigt. [HasNavigationBar]

Am linken Rand des Kontrollfeldes befindet sich standardmäßig der Datensatzmarkierer. Er zeigt den aktuellen Datensatz an. Über den Datensatzmarkierer wird die Löschfunktion für den gesamten Datensatz zur Verfügung gestellt. [HasRecordMarker]

**Daten**

Da es sich um ein Feld handelt, das selbst keine Daten anzeigt oder abspeichert, sondern stattdessen andere Felder verwaltet, die dies tun, gibt es keine Eigenschaft Daten.

**Ereignisse**

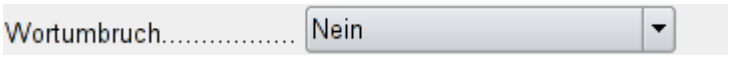
Es fehlt das Ereignis «Modifiziert» und «Text modifiziert». Das Ereignis «Fehler aufgetreten» kommt hinzu.

## *Beschriftungsfeld*

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Der Titel dient als Beschriftung für ein anderes Kontrollfeld. Wird der Titel mit einem Kontrollfeld verbunden, dann kann er auch als Sprungziel für dieses Kontrollfeld festgelegt werden. Mit einem eingefügten «~» kann ein bestimmter Buchstabe des Titels als Sprungziel angegeben werden. «Na~me» definiert das «m» als Sprungziel. Befindet sich der Cursor in dem Formular, so wird die Schaltfläche über *Alt* + *m* erreicht. [Label]




Standardmäßig wird die Beschriftung nicht umbrochen. Ein zu langer Titel wird abgeschnitten, wenn das Feld nicht groß genug ist. Der Wortumbruch hat allerdings keine Trennfunktion, so dass bei zu kleinen Feldern innerhalb eines Wortes ein Umbruch erfolgt.
[MultiLine]

**Daten**

keine

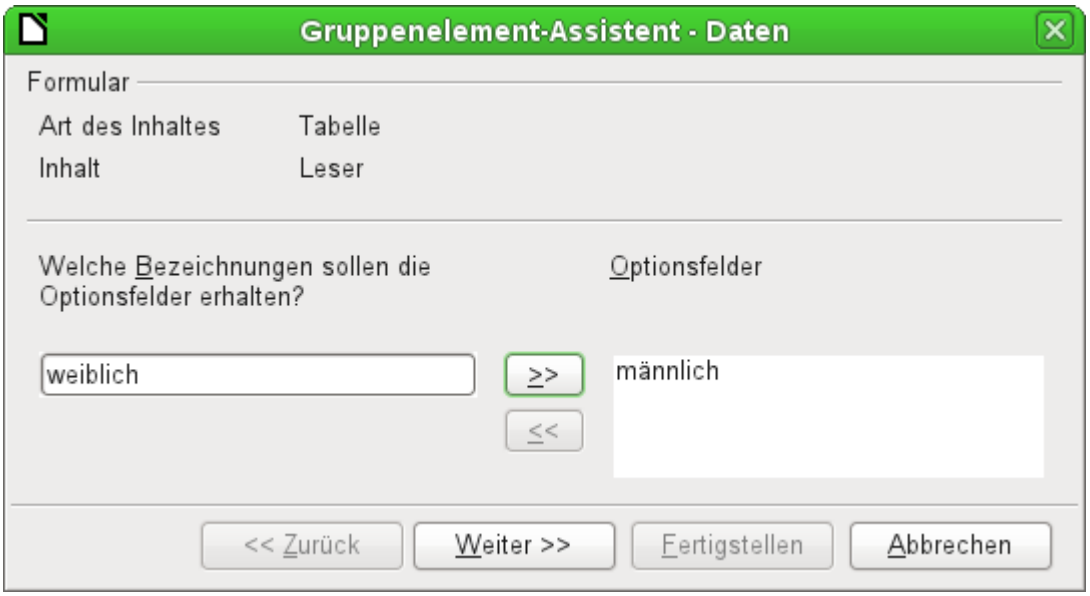
**Ereignisse**

Selbst das Beschriftungsfeld reagiert noch auf Ereignisse, die mit der Maus einer Taste oder dem Fokuserhalt zusammenhängen.

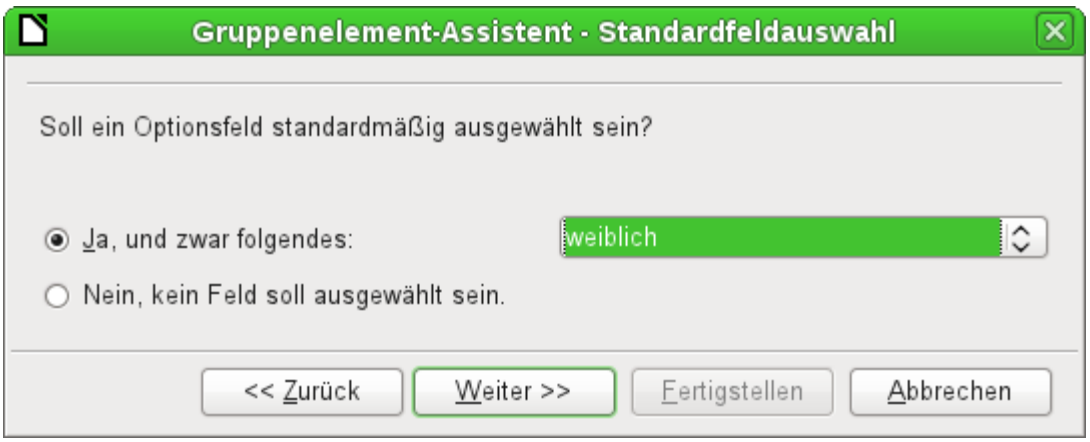
### *Gruppierungsrahmen*

Der Gruppierungsrahmen fasst lediglich grafisch mehrere Kontrollfelder zu einer Gruppe zusammen und versieht sie mit einem zusätzlich grafisch eingebundenen Titel.

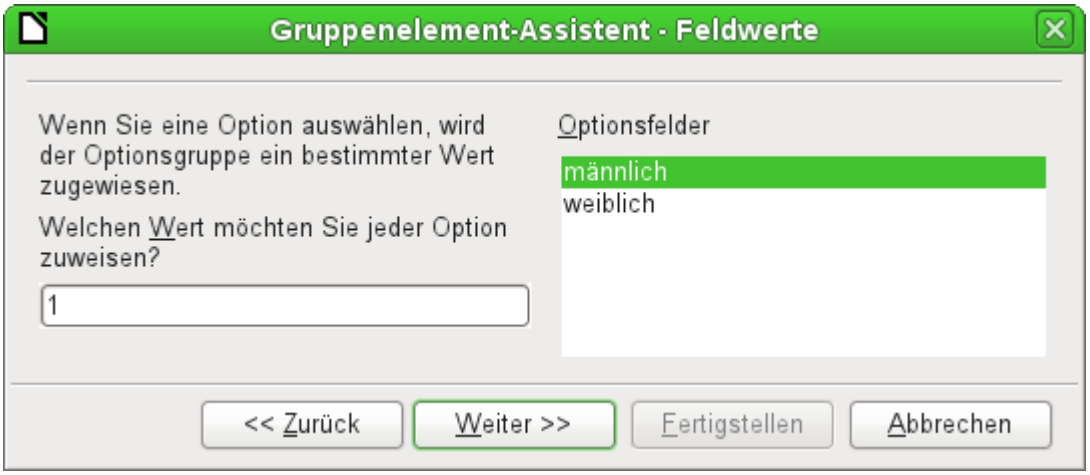
Wird ein Gruppierungsrahmen mit eingeschaltetem Assistenten aufgezogen, so geht der Assistent davon aus, dass mehrere Optionsfelder zusammen in diesem Rahmen erscheinen sollen.



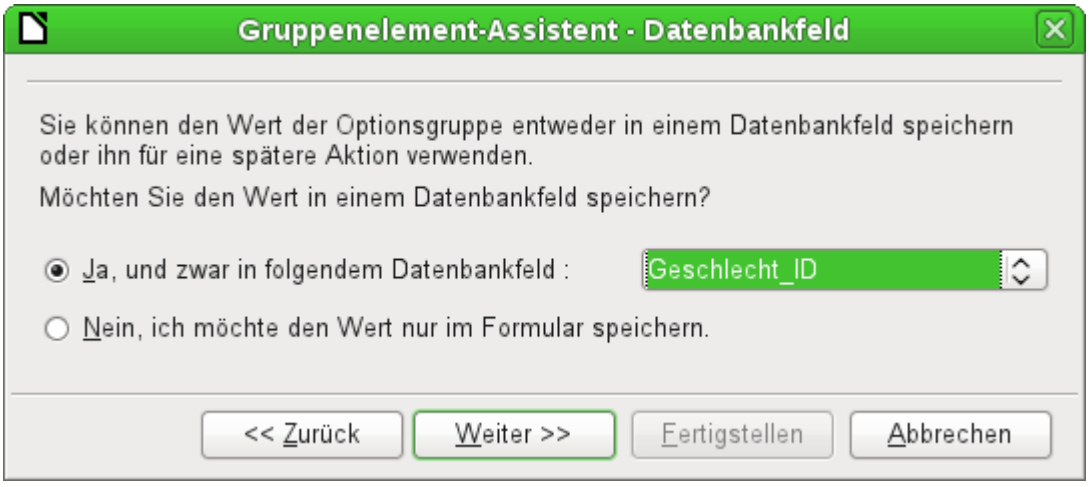

Dem Formular liegt die Tabelle Leser zugrunde. Hier wird gerade die Auswahl des Geschlechtes zusammengestellt. Die Einträge sind anschließend die Beschriftungen der Optionsfelder.



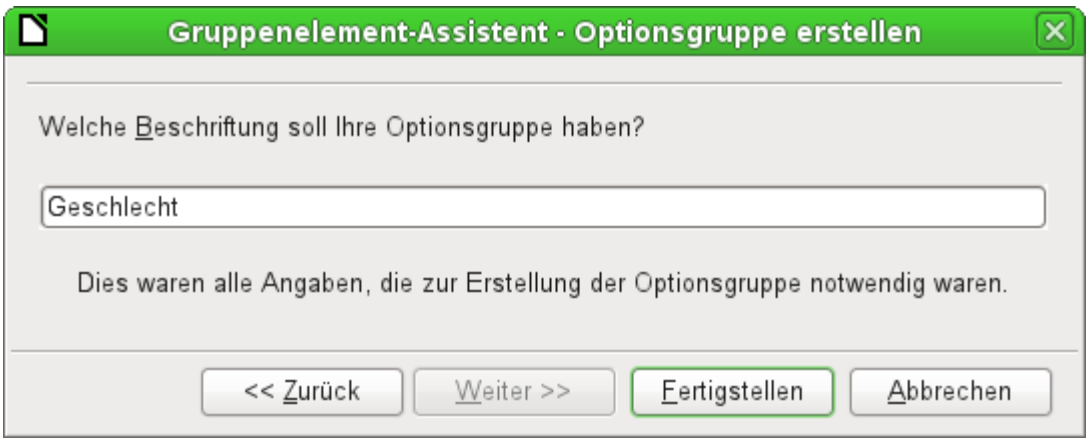

Standardmäßig wird hier vorgewählt, dass das Geschlecht «weiblich» angewählt wird. Wird kein Feld vorgewählt, so ist der Standardeintrag in die zugrundeliegende Tabelle *NULL*.




Den Optionsfeldern wird bereits standardmäßig durch den Assistenten ein jeweils separater Wert zugeteilt. Da dieser Wert in unserem Fall nicht dem des Primärschlüssels entspricht, wird er geändert, und zwar für «männlich» auf den Wert 'm', für «weiblich» auf den Wert 'w'. Diese Werte entsprechen in dem Beispiel den Primärschlüsselfeldern der Tabelle "Geschlecht"




Der jeweils in den Optionsfeldern angeklickte Wert wird an das Feld "Geschlecht_ID" der dem Formular zugrundeliegenden Tabelle "Leser" übertragen. Die Tabelle "Leser" wird also über ein Optionsfeld mit den entsprechenden Fremdschlüsseln der Tabelle "Geschlecht" versorgt.




Die Gruppe der Optionsfelder erhält einen Rahmen, in den die Beschriftung 'Geschlecht' eingefügt wird.

Wird jetzt bei aktivem Formular 'männlich' ausgewählt, so wird gleichzeitig 'weiblich' abgewählt. Dies ist ein Kennzeichen der Optionsfelder, wenn sie mit gleichem Namen versehen und mit dem gleichen Feld der dem Formular zugrundeliegenden Tabelle verbunden sind. In dem obigen Beispiel ersetzen die Optionsfelder so ein Listenfeld mit zwei Einträgen.

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Der Titel kann neben den Standardeinstellungen verändert werden. Eine Veränderung des Rahmens (Liniendicke, Linienfarbe) ist zur Zeit nicht vorgesehen, wohl aber eine Formatierung der Schrift.

**Daten**

Keine, fasst schließlich Eingabefelder nur optisch zusammen.

**Ereignisse**

Auch der Gruppierungsrahmen reagiert noch auf Ereignisse, die mit der Maus einer Taste oder dem Fokuserhalt zusammenhängen.

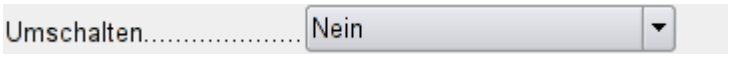
### *Schaltfläche*

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

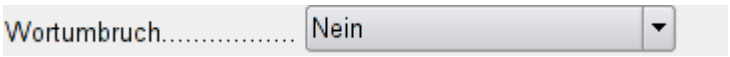
Beschriftung auf der Schaltfläche. Mit einem eingefügten «~» kann ein bestimmter Buchstabe des Titels als Sprungziel angegeben werden. «Schalt~fläche» definiert das «f» als Sprungziel. Befindet sich der Cursor in dem Formular, so wird die Schaltfläche über *Alt* + *f* erreicht.
[Label]

Die Schaltfläche erhält den Fokus, wenn auf sie geklickt wird.
Dies kann manchmal nicht erwünscht sein, wenn z.B. über die Schaltfläche Inhalt in ein anderes Formularfeld eingefügt werden soll. Dann soll bei Betätigung der Schaltfläche der Cursor in dem anderen Feld stehen bleiben.
[FocusOnClick]

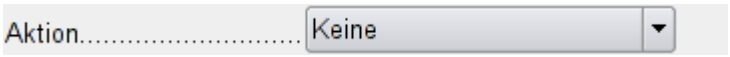



Wenn ja, so kann die Schaltfläche auch eingedrückt dargestellt werden. Der Schaltzustand wird wie bei einem Schalter angezeigt – im Gegensatz zu einem Taster wie sonst bei Buttons.
[Toggle]

Aktiv, wenn Umschalten auf *Ja* gesetzt ist. Ausgewählt ist dann der eingedrückte Zustand.
[DefaultState]



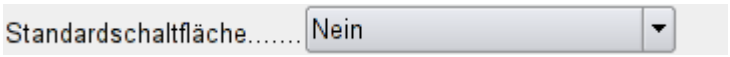

Wortumbruch, wenn die Schaltfläche zu schmal ist.
[MultiLine]




Verschiedene Aktionen ähnlich denen der Navigationsleiste stehen zur Verfügung.
Es kann hier auch ein Dokument oder eine Webseite geöffnet werden. Hierzu ist die URL in dem folgenden Feld einzugeben.

HTML: Datei, die hiermit aufgerufen werden soll. Hierzu muss unter **Aktion → Dokument/Webseite öffnen** gewählt werden.
Neben HTML kann dies auch genutzt werden, um bestimmte Programmmodule von LO aufzurufen. So wird mit dem hier einzutragenden Befehl **`.uno:RecSearch`** z. B. die Suchfunktion aus dem Formularnavigator auf einen Button gelegt. Außerdem lassen sich auch z. B. Writerdateien öffnen, die anschließend per Druck einen mit der Datenbank zusammenhängenden Serienbrief ausgeben.
Welche Befehle hier zur Verfügung stehen, lässt sich über den Makrorecorder gegebenenfalls feststellen.
[TargetURL]

Nur für HTML-Formulare: Der Ziel-Frame (Rahmeneinteilung für verschiedene HTML-Seiten), in dem die Datei geöffnet werden soll.
[TargetFrame]




Schaltfläche wird bei *Ja* zusätzlich umrandet. Bei mehreren alternativen Schaltflächen auf einem Formular soll dies die übliche Schaltfläche kennzeichnen. Sie wird durch Betätigung der Eingabetaste ausgelöst, wenn keine andere Aktion gerade auf die Eingabetaste reagieren muss. Nur eine Schaltfläche sollte im Formular die Standardschaltfläche sein.
[DefaultButton]

Soll eine Grafik auf der Schaltfläche erscheinen?
[Graphic] [ImageURL]

Nur aktiv, wenn eine Grafik ausgewählt wurde. Die Grafik wird im Verhältnis zum Text angeordnet.
[ImagePosition] [ImageAlign]

**Daten**

Keine, es werden nur Aktionen ausgeführt.

**Ereignisse**

«Aktion bestätigen», «Aktion ausführen» und «Statusänderung»

### Grafische Schaltfläche

Neben den bereits unter *Eigenschaften der Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Entspricht der normalen Schaltfläche. Allerdings wird diese Schaltfläche ohne Text und ohne einen sichtbaren Button dargestellt. Lediglich eine Rahmen um die Grafik kann eingestellt werden.

Eine grafische Schaltfläche erhält standardmäßig erst einmal keinen Tabstop.

| Hinweis | Bereits in OpenOffice funktionierte diese Schaltfläche nicht zusammen mit den Aktionen, die auch der normalen Schaltfläche zugeordnet werden können. Die Schaltfläche lässt sich aber so nutzen, dass z. B. unter **Allgemein → Aktion → Dokument/ Webeite öffnen** und anschließend unter **Allgemein → URL** `.uno:NextRecord` angegeben wird, Damit ist ein Sprung zum nächsten Datensatz möglich. Siehe dazu auch die uno-Befehle im Angang des Handbuches.<br>Ansonsten ist die Schaltfläche nur mit Makros nutzbar. |
|---|---|

**Daten**

Keine, es werden nur Aktionen ausgeführt.

**Ereignisse**

«Aktion bestätigen» sowie alle Ereignisse, die die Maus, eine Taste oder den Fokus betreffen.

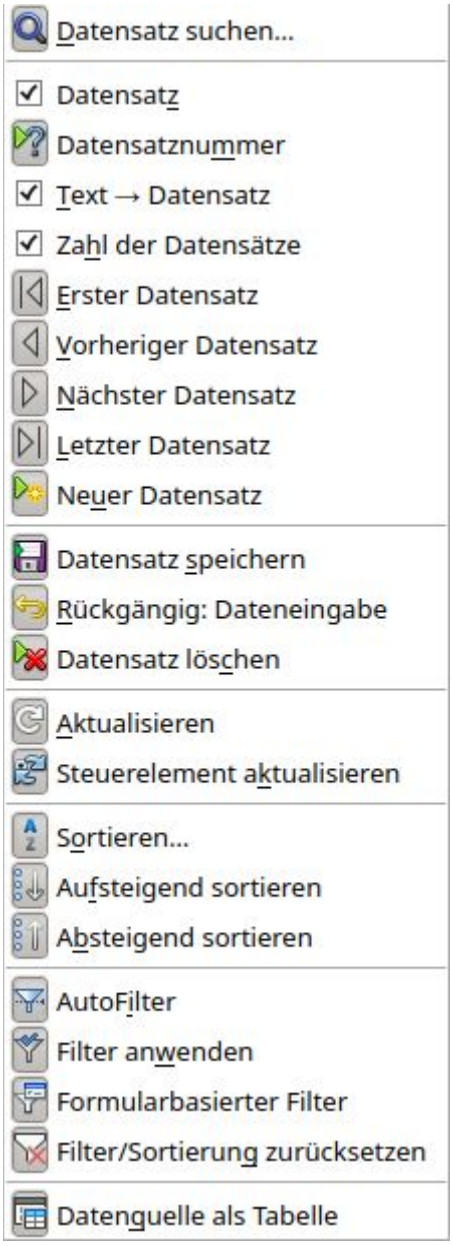
### Navigationsleiste

*Abbildung 7: Formularkontrollelement Navigationsleiste*

Die Standardnavigationsleiste wird beim Formular am unteren Bildrand eingeblendet. Durch das Einblenden dieser Leiste kann es beim Bildschirmaufbau zu einem kurzen Zurechtschieben des Formulars kommen. Dies ist vor allem dann störend, wenn die Navigationsleiste eventuell in Teilen des sichtbaren Formulars ausgeschaltet ist.

Eine separat zu den jeweiligen Elementen eines Formulars angeordnete Navigationsleiste macht dagegen klar, durch welche Elemente mit der Leiste navigiert wird.

Die Standardnavigationsleiste bietet etwas mehr Elemente als das Formularkontrollelement Navigationsleiste. Die folgende Übersicht zeigt den gesamten Umfang der Standardnavigationsleiste.




Die Such- und Filterfunktionen werden zum Schluss des Kapitels erklärt.

Die Navigation durch die Datensatznummern dürfte von der Bedienung von Multimedia-Geräten bekannt sein. Wird bei der Datensatznummer eine konkrete Nummer eingegeben und mit der *Eingabetaste* bestätigt, so bedeutet dies einen Sprung zum entsprechenden Datensatz.

Beim Aktualisieren gibt es auch den Button «Steuerelement aktualisieren». Werden Inhalte von Listenfeldern durch ein anderes Formular beeinflusst, so muss der Cursor in das Listenfeld gesetzt werden und der Inhalt kann über diesen Button neu eingelesen werden.

Neben den bereits unter *Standardeinstellungen vieler Kontrollfelder* erklärten Eigenschaften gibt es die folgenden Besonderheiten:

**Allgemein**

Die Größe der Symbole kann eingestellt werden. Außerdem ist auswählbar, welche Gruppen angezeigt werden. Dies sind in *Abbildung 7* von links nach rechts, bei den Symbolen durch eine senkrechte Linie getrennt, die Positionierung, die Navigation, die Datensatzaktionen sowie Filterung / Sortierung.
[IconSize]
[ShowPosition]
[ShowNavigation]
[ShowRecordActions]
[ShowFilterSort]

**Daten**

Keine, es werden nur Aktionen ausgeführt.

**Ereignisse**

Alle Ereignisse, die die Maus, eine Taste oder den Fokus betreffen.

Unabhängig von diesem Formularkontrollelement existiert natürlich auch die unten **einblendbare Navigationsleiste** mit den Elementen der obigen Abbildung.

Diese einblendbare Navigationsleiste bietet neben den Elementen des entsprechenden Formularkontrollelementes noch die *allgemeine Datensatzsuche*, den *formularbasierten Filter* und die Darstellung der dem Formular zugrundeliegenden *Datenquelle in der Tabellenansicht* oberhalb des Formulars. Im Gegensatz zu einer einfachen Tabellenansicht des Formularinhaltes übernimmt diese Tabellenansicht die Definition von Listenfeldern aus dem Formular. Sie ist daher eher vergleichbar mit einem fertig gestalteten Tabellen-Kontrollfeld.

Wird nicht nur mit einem Formular, sondern mit Unterformularen und Nebenformularen gearbeitet, so ist darauf zu achten, dass diese einblendbare Navigationsleiste nicht beim Formularwechsel je nach Einstellung verschwindet. Das erzeugt nur Unruhe auf dem Bildschirm.

### *Drehfeld und Bildlaufleiste*

Beide Felder haben keine direkte Verbindung zu Daten des Formulars. Sie lassen sich nur über Makros entsprechend nutzen, wobei das Drehfeld von der Anwendung her bereits z.B. in das Zahlenfeld integriert ist.

Die Bildlaufleiste unterscheidet sich von dem Drehfeld dadurch, dass sie neben den Buttons für steigende und fallende Werte innerhalb einer vorher festgelegten Grenze auch einen Schiebeknopf haben. Der Name «Bildlaufleiste» ist dabei etwas irritierend, da über dieses Schieberegler nicht irgendwelche Bilder verschoben werden, sondern lediglich Werte in einem Feld verändert werden können.

Beim Start eines Formulars muss der Bildlaufleiste zuerst mitgeteilt werden, welchen Wert das Feld hat, das mit ihr verbunden sein soll. Dieses Makro ist an die **Formular-Eigenschaften → Ereignisse → Nach dem Datensatzwechsel** gebunden.

```
SUB Bildlaufleiste_Formular(oEvent AS OBJECT)
    DIM oForm AS OBJECT
    DIM oFeld AS OBJECT
    oForm = oEvent.Source
    oFeld = oForm.getByName("Bildlaufleiste")
    oFeld.ScrollValue = oForm.getInt(oForm.findColumn("Zahlenwert"))
END SUB
```

Zuerst werden die Variablen deklariert. Das auslösende Ereignis weist dabei auf das Formular hin, da das Makro schließlich an das Formular gebunden wurde. Die Bildlaufleiste wird nach dem entsprechenden Namen in dem Formular gefunden. Der Wert der Bildlaufleiste wird auf den Wert eingestellt, der in dem Feld "Zahlenwert" der dem Formular zugrundeliegenden Tabelle liegt.

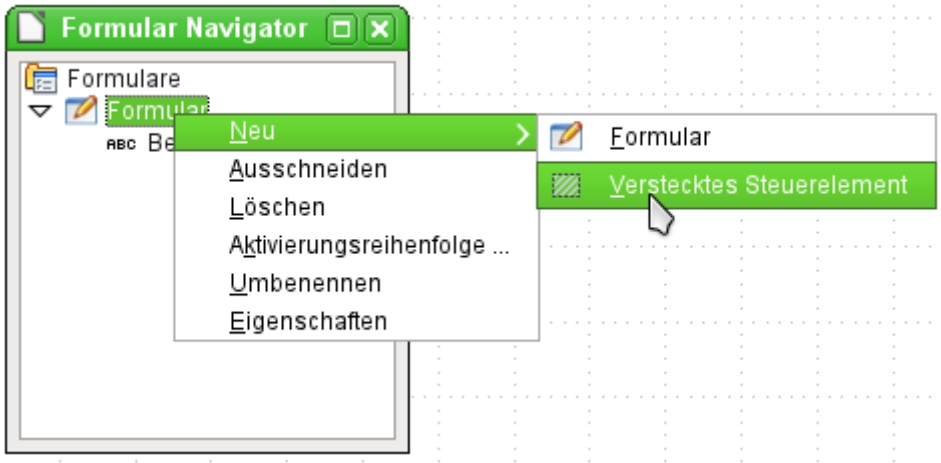
Wird der Wert der Bildlaufleiste geändert, so muss entsprechend der Wert in dem verbundenen Feld geändert werden. Dieses Makro wird an **Eigenschaften: Bildlaufleiste → Ereignisse → Beim Justieren** gebunden

```
SUB Bildlaufleiste_Aenderung(oEvent AS OBJECT)
    DIM oForm AS OBJECT
    DIM oFeld AS OBJECT
    DIM inValue AS INTEGER
    oFeld = oEvent.Source.Model
    inValue = oEvent.Value
    oForm = oFeld.Parent
    oForm.updateInt(oForm.findColumn("Zahlenwert"),inValue)
END SUB
```

Zuerst werden die Variablen deklariert. Aus dem Feld wird der gewählte Wert als Ganzzahlenwert ausgelesen. Das dem Formular zugrundeliegende Tabellenfeld "Zahlenwert" wird auf den entsprechenden Wert eingestellt.

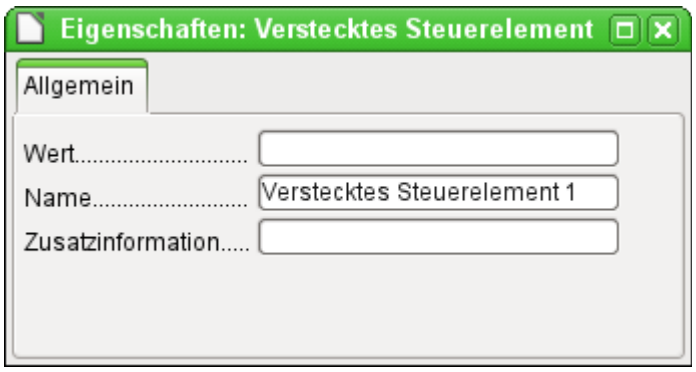
Details zu einem Code wie diesem sind in dem Kapitel «Makros» zu finden.

### *Verstecktes Steuerelement*




Ein Kontrollfeld, das nicht über die Symbolleisten einfügbar ist, ist das versteckte Steuerelement. Es wird über den Formularnavigator im Kontextmenü eines Formulars neu erstellt.

Das versteckte Steuerelement erscheint wie jedes andere Kontrollfeld als Teil des Formulars. Es ist auf der Benutzeroberfläche nicht sichtbar.




Die kompletten Eigenschaften des versteckten Steuerelementes sind recht spärlich. Der Name und die Zusatzinformationen sind von den anderen Steuerelemente bekannt. Zusätzlich kann hier ein Wert angegeben werden. [HiddenValue]

Für die Nutzung ohne Makros ist dies sicher nicht weiter von Bedeutung. Bei der Arbeit mit Makros kann es aber sinnvoll sein, an einer Stelle des Formular Werte zwischenspeichern zu können, um auf sie bei einer weiteren Aktion wieder zugreifen zu können. Ein entsprechendes Anwendungsbeispiel findet sich bei den Makros im Kapitel «Hierarchische Listenfelder».

### *Mehrfachselektion*

Wird mit dem Auswahlpfeil (siehe *Abbildung 3*) ein größerer Bereich oder sogar sämtliche Elemente eines Formulars ausgewählt, so sind über diese Mehrfachselektion die Änderungen nach *Abbildung 8* möglich.

*Abbildung 8: Allgemeine Eigenschaften von Formularfeldern in der Mehrfachselektion*

Den Namen sollte hier möglichst nicht geändert werden. Schließlich lauten so plötzlich alle Elemente gleich. Das Auffinden eines einzelnen Elementes über den Formularnavigator wird erschwert, die Verarbeitung des Formulars nach Namensbezeichnung der Kontrollfelder in Makros unmöglich.

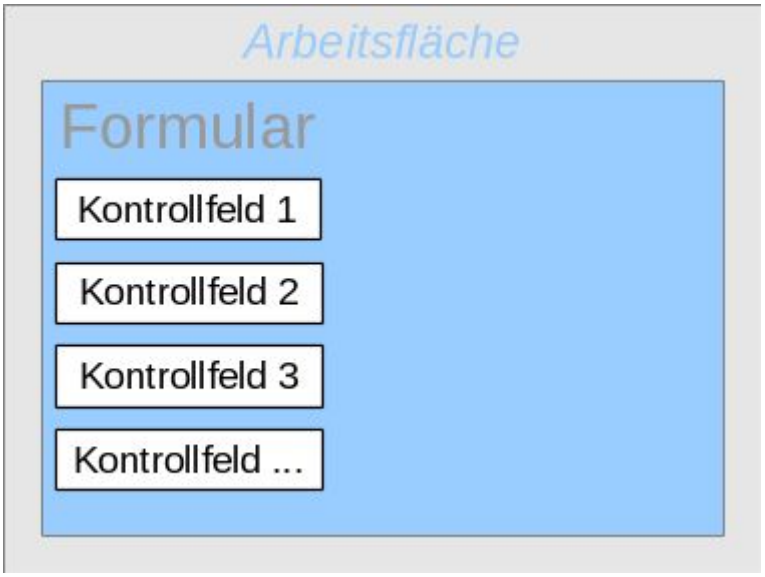
Eher bietet sich eine Mehrfachselektion an, wenn insgesamt die Schriftart, die Höhe oder die Hintergrundfarbe gewechselt werden soll. Allerdings wirkt sich natürlich die Hintergrundfarbe auch auf die Beschriftungsfelder aus.

Sollen z. B. nur die Beschriftungsfelder geändert werden, so bietet es sich an, bei gedrückter *Strg*-Taste mit der linken Maustaste diese Felder direkt oder im Navigator anzuklicken, mit der rechten Maustaste über eins der Felder zu gehen und die Kontrollfeldeigenschaften aufzurufen. Jetzt wird die Auswahl der änderbaren Eigenschaften größer, da ja nur gleiche Felder ausgewählt wurden. Entsprechend kann hier auch all das geändert werden, was sonst in einem Beschriftungsfeld zur Verfügung steht.

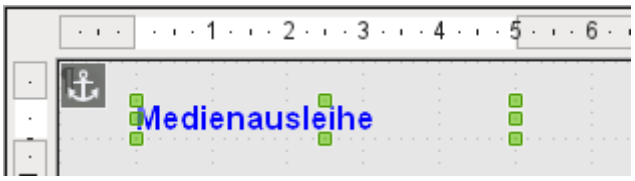
Die Möglichkeiten der Mehrfachselektion richten sich also nach der Auswahl der Felder. Kontrollfelder der gleichen Art können in allen allgemeinen Eigenschaften, die das einzelne Kontrollfeld bietet, auf diese Art gemeinsam angepasst werden.

## Einfaches Formular komplett erstellt

Ein einfaches Formular stellt Formularkontrollfelder für das Schreiben oder Lesen von Datensätzen aus einer einzigen Tabelle oder Abfrage zur Verfügung. Seiner Konstruktion entspricht dem folgenden Schaubild:




Am Beispiel eines einfachen Formulars für die Bibliotheksausleihe sollen hier verschiedenen Varianten eines Formulars vorgestellt werden.



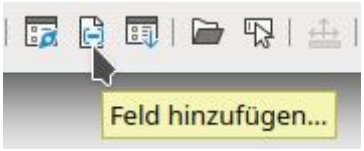

Die Überschrift für das Formular wurde über ein Beschriftungsfeld erzeugt. Die Schrift wurde geändert. Das Beschriftungsfeld ist am Absatz verankert, der sich in dem Dokument an der linken oberen Ecke befindet. Über das Kontextmenü des Beschriftungsfeldes wurde ein Formular erzeugt, das mit der Tabelle Ausleihe verbunden wurde (siehe: *Formulargründung über ein Formularfeld*). Die Seite wurde außerdem mit einem einfarbigen Hintergrund versehen.

| | |
|---|---|
| **Tipp** | Das Formular sollte grundsätzlich nur aus Formularfeldern erstellt werden. Text gehört in die Beschriftungsfelder. Alle Felder werden an dem Absatz der linken oberen Ecke verankert.<br>Werden Zeilenumbrüche auf der Hintergrundseite eingefügt, so kann dies bei einem Wechsel des Zeichensatzes des Systems zu Verschiebungen der Formularelemente führen. Die Formularelemente sind dann nämlich an den nächstliegenden Absatz gebunden, nicht an die linke obere Ecke des Formularhintergrundes. |

| | |
|---|---|
| **Hinweis** | Das Einfügen von Bildern im Hintergrund kann dazu führen, dass die *.odb-Datei unnötig groß wird. Die Bilder werden nämlich in dieser Datei gespeichert. Manchmal lassen sich die Bilder zwar aus dem Formular löschen, werden aber nicht aus der *.odb-Datei entfernt. Hier hilft dann nur ein Öffnen der Datei mit einem Packprogramm. Siehe die Beschreibung dazu im Anhang. |

### Felder als Gruppe hinzufügen

Eine schnelle Variante, direkt Felder mit Beschriftungen einzufügen, bietet die Funktion *Feld hinzufügen*.




Über diese auf der Symbolleiste Formular-Entwurf (siehe *Abbildung 2*) erreichbaren Funktion lassen sich alle Felder der dem Formular zugrundeliegenden Tabelle auswählen

Über einen Doppelklick auf die Felder werden diese als Gruppierung zusammen mit einer Beschriftung immer in der Mitte des Formulars eingegliedert. Die Gruppen müssen also auf jeden Fall noch verschoben werden, damit anschließend das Formular im Einsatz wie das folgende Bild aussehen kann. Für die bessere Übersicht wurden alle unnötigen Leisten des Fensters entfernt und das Fenster entsprechend schmal zusammengeschoben, so dass nicht mehr alle Elemente der Navigationsleiste sichtbar sind. Es wurden alle Felder bis auf "Medien_ID_BC" ausgewählt, da dies speziell für die Bedienung mit einem Barcodescanner gedacht ist.

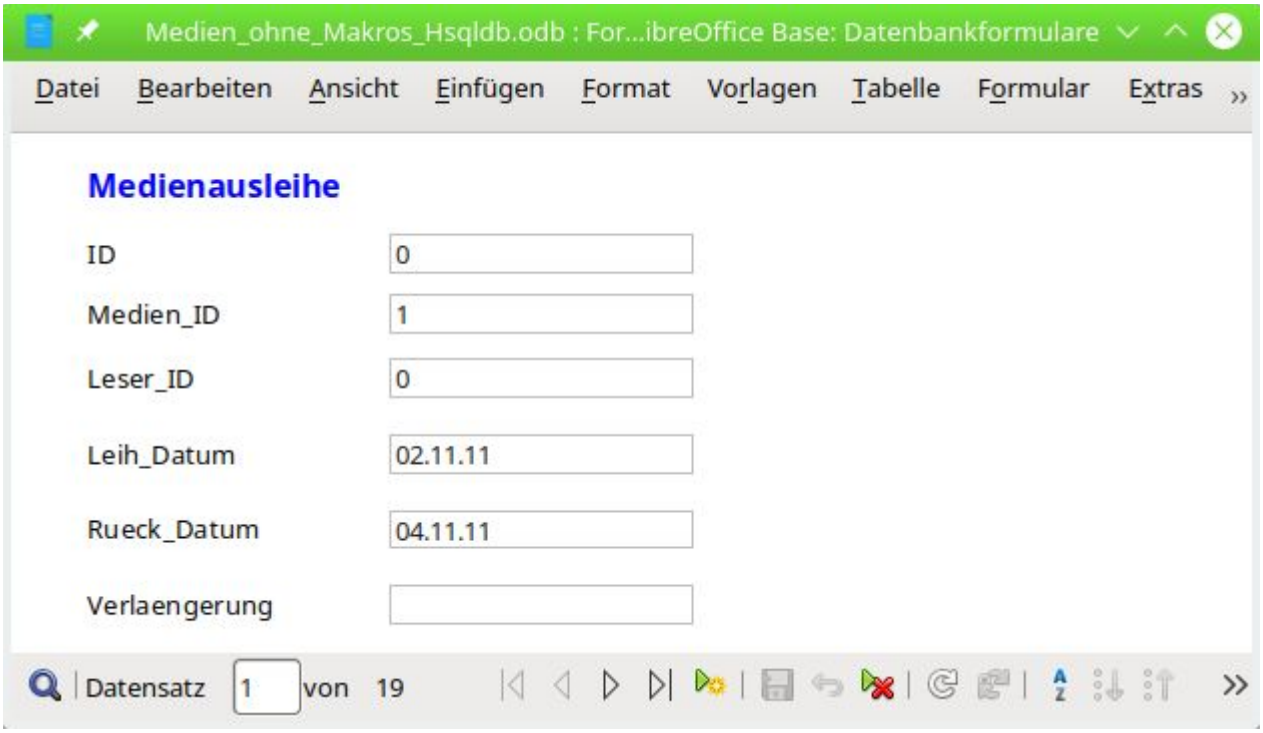



*Abbildung 9: Einfaches Formular über «Feld hinzufügen"*

Für alle Tabellenfelder wurden die korrekten Formularkontrollfelder ausgesucht. Zahlen werden in numerische Felder gesetzt und gleich als Ganzzahlen ohne Nachkommastellen erkannt. Datums-

felder werden ebenfalls korrekt als Datumsfelder wiedergegeben. Alle Felder werden in gleicher Breite dargestellt. Würde ein Grafisches Kontrollfeld eingesetzt, so würde hier ein quadratisches Feld erzeugt.

### Felder anpassen

Gestalterisch kann jetzt einiges getan werden, indem die Felder von der Länge her angepasst werden und die Datumsfelder aufklappbar gemacht werden. Wichtiger ist aber, dass die Felder für die "Medien_ID" und die "Leser_ID" für den Nutzer lösbar werden – es sei denn, jeder Bibliotheksbesucher muss einen Ausweis mit der "ID" mitbringen und jedes Medium wird bei der Aufnahme mit der "ID" versehen. Hiervon wurde im Folgenden nicht ausgegangen.

Um einzelne Felder anzupassen, muss zuerst einmal die Gruppe betreten werden. Dies kann über einen rechten Mausklick auf die Gruppe und das entsprechende Kontextmenü erfolgen.

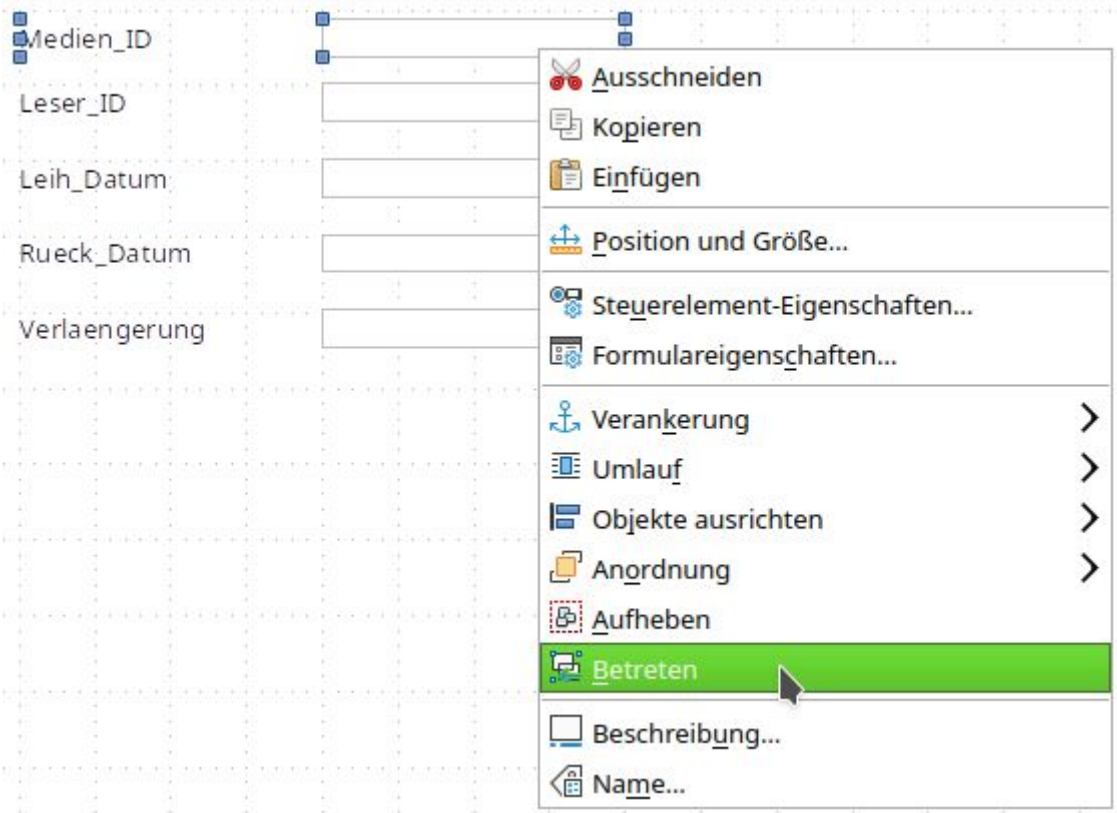
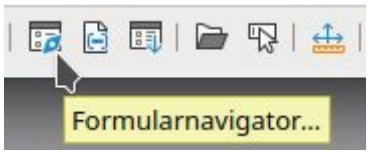



*Abbildung 10: Formularkontrollelemente: Gruppe betreten*

Die Gruppe kann auch direkt betreten werden, wenn die *Strg* mit einem Linksklick der Maus kombiniert wird. Mehrere Felder aus unterschiedlichen Gruppen werden mit *Strg* + *Shift* und einem Linksklick der Maus markiert.

Übersichtlicher für die späteren Verfahren ist allerdings die Arbeit mit dem Formularnavigator:



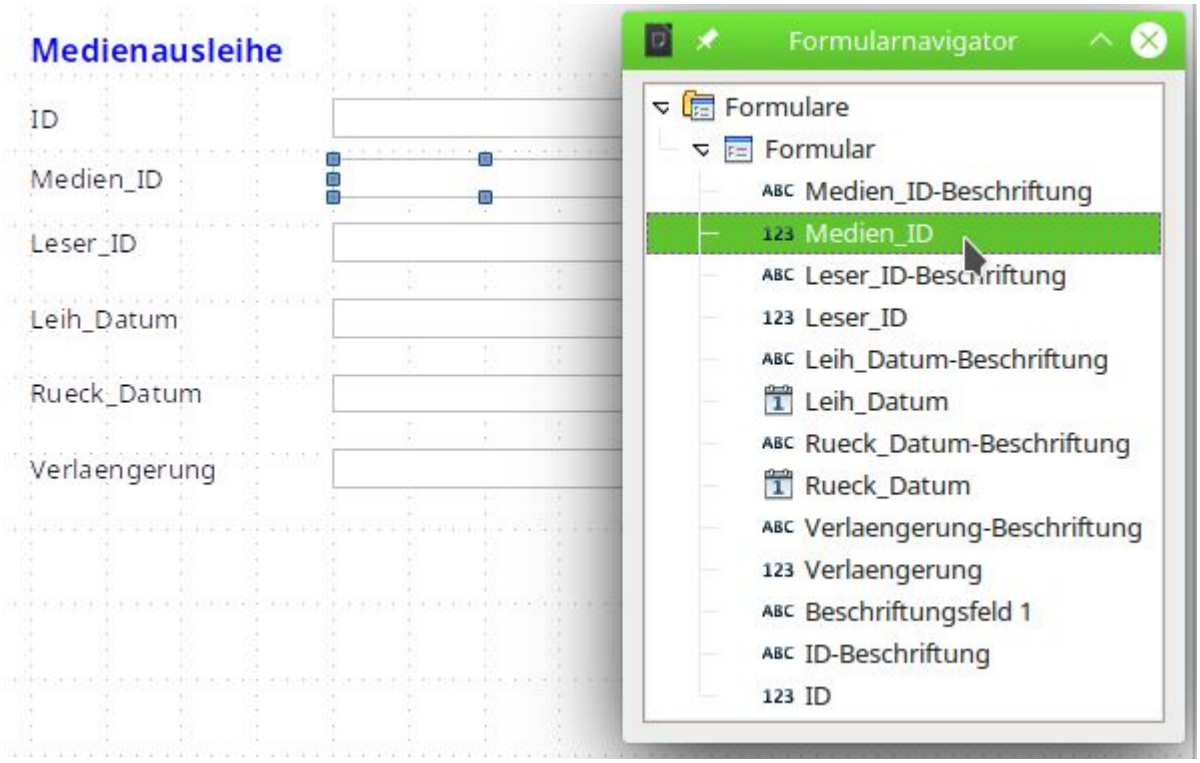



*Abbildung 11: Formularkontrollelemente über den Formularnavigator direkt anwählen*

Der Formularnavigator stellt alle Elemente des Formulars mit ihren Bezeichnungen dar. Für die Kontrollfelder wurden als Bezeichnungen direkt die Namen der Felder aus der dem Formular zugrundeliegenden Tabelle genommen. Die Beschriftungselemente haben den Zusatz «Beschriftung».

Durch einen Klick auf "Medien_ID" ist dieses Feld ausgewählt. Mit einem Rechtsklick wird es über das Kontextmenü möglich, das ausgewählte Feld durch eine andere Feldart zu ersetzen:

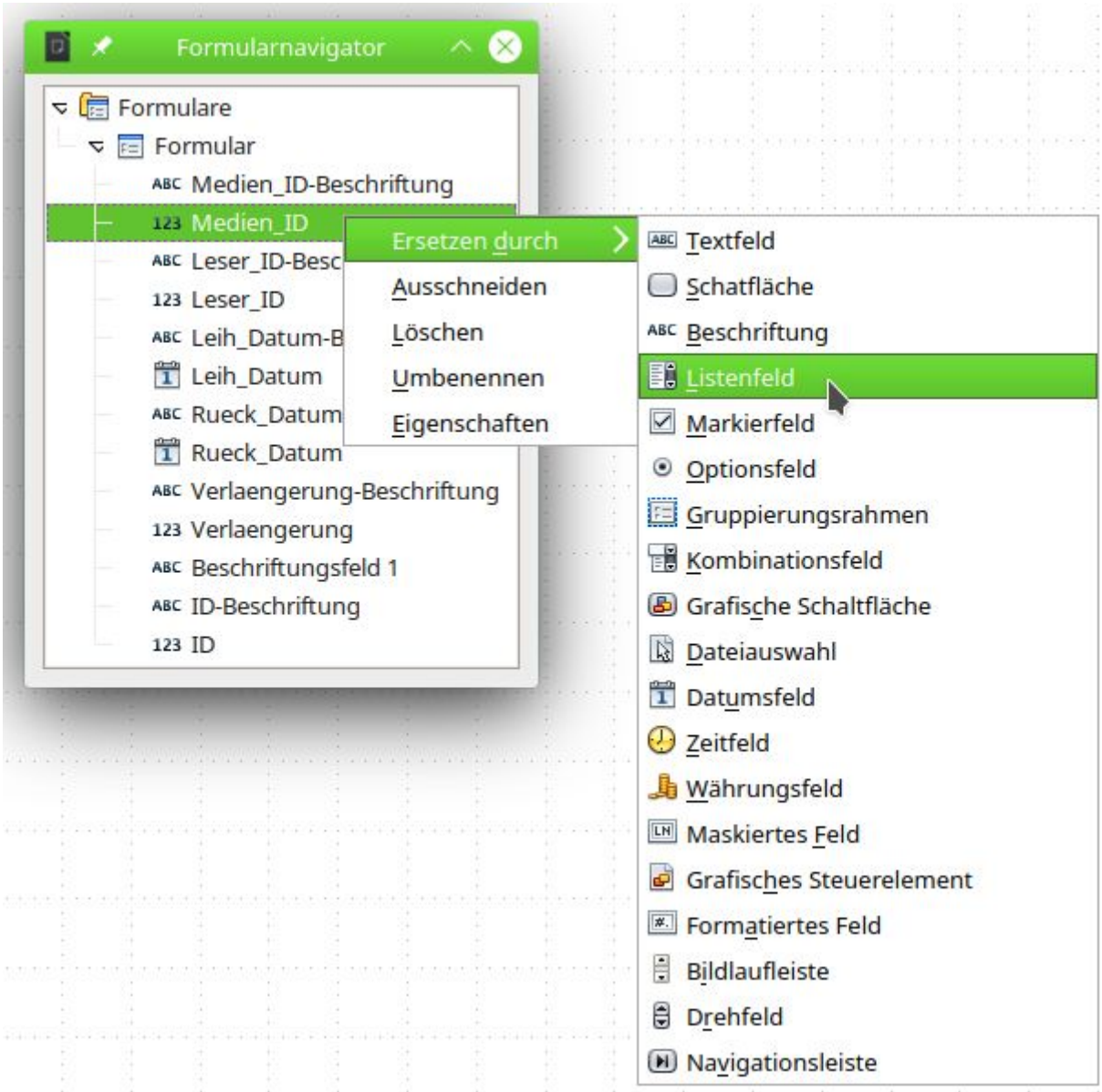
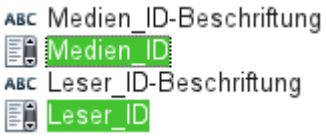



*Abbildung 12: Kontrollelemente über den Formularnavigator durch andere Kontrollelemente ersetzen*

Diese Ersetzung wird in dem Formular für "Medien_ID" und "Leser_ID" vorgenommen.




Die Änderung ist im Formularnavigator über die Symbole sichtbar.

Die SQL-Abfragen für die Listenfelder können jetzt über den rechten Button mit der grafischen Benutzeroberfläche erstellt werden. Der Listenfeldassistent steht hier nicht zur Verfügung. Er springt nur automatisch ein, wenn ein Listenfeld direkt neu gegründet, nicht aber durch Umwandlung von einem anderen Feld aus erstellt wird. Zum SQL-Befehl siehe das Kapitel «Abfragen für die Erstellung von Listenfeldern».

Nachdem die Listenfelder als aufklappbar angepasst wurden, können noch die folgenden Mängel behoben werden:

- Die Beschriftung der Listenfelder sollte auf «Medien» statt «Medien_ID» und «Leser» statt «Leser-ID» geändert werden.
- Das Kontrollfeld «ID» sollte als nicht beschreibbar erklärt werden.
- Alle Felder, die nicht beim Aufruf der Ausleihe auf jeden Fall durchlaufen werden müssen, wenn ein neues Medium ausgeliehen wird, benötigen keinen Tabstop. Ohne den Tabstop geht es schneller durch das Formular. Eventuell muss der Tabstop auch über die Aktivierungsreihenfolge (siehe *Abbildung 2*) nachjustiert werden. Lediglich die Felder «Medien», «Leser» und «Leihdatum» müssen für eine Ausleihe auf jeden Fall über den Tabulator erreichbar sein.
- Soll über das Formular die Ausleihe erfolgen, so ist es eigentlich unnötig und auch unübersichtlich, bereits zurückgegebene Medien angezeigt zu bekommen. Medien mit einem Rückgabedatum sollten ausgefiltert werden. Außerdem könnte die Reihenfolge der Anzeige nach dem Leser sortiert werden, damit Medien, die von der gleichen Person entliehen wurden, nacheinander aufgezeigt werden. Siehe dazu die Hinweise unter *Formular-Eigenschaften*. Speziell die Lesersortierung hat hier allerdings den Haken, dass lediglich nach der ID, nicht aber nach dem Alphabet sortiert werden kann, da die Tabelle für das Formular ja nur die ID enthält.

### *Felder einzeln hinzufügen*

Das Hinzufügen einzelner Felder gestaltet sich zuerst einmal etwas aufwändiger. Die Felder müssen ausgewählt, auf der Formularfläche aufgezogen und dem Feld der dem Formular zugrundeliegenden Tabelle zugewiesen werden. Außerdem ist noch die Art des Feldes richtig einzustellen, da z. B. numerische Felder standardmäßig erst einmal 2 Nachkommastellen haben.

Lediglich beim Aufziehen der Listenfelder kommt der Assistent zum Einsatz, der die Schritte zur Erstellung eines korrekten Feldes für Ungeübte vereinfacht. Nach kurzer Anwendungsphase wird der Assistent allerdings den Ansprüchen nicht mehr gerecht, da er

- Die Einträge nicht automatisch sortiert.
- Eine Zusammenfassung von mehreren Feldern im aufzulistenden Inhalt nicht ermöglicht.

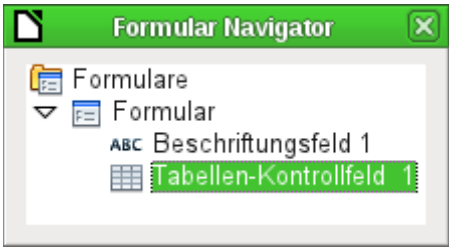
Hier muss dann immer wieder nachgebessert werden, so dass ganz schnell der SQL-Code direkt über den eingeblendeten Abfrageeditor erstellt wird.

Beim Hinzufügen einzelner Felder müsste eine Gruppierung von Feld und Beschriftung gesondert vorgenommen werden (siehe *Standardeinstellungen vieler Kontrollfelder*). In der Praxis kann sich aber die fehlende Verknüpfung auch positiv bemerkbar machen, da der Zugang zu den Eigenschaften eines Kontrollfeldes über das Kontextmenü sofort möglich ist und nicht erst ein Betreten der Gruppe erfordert.

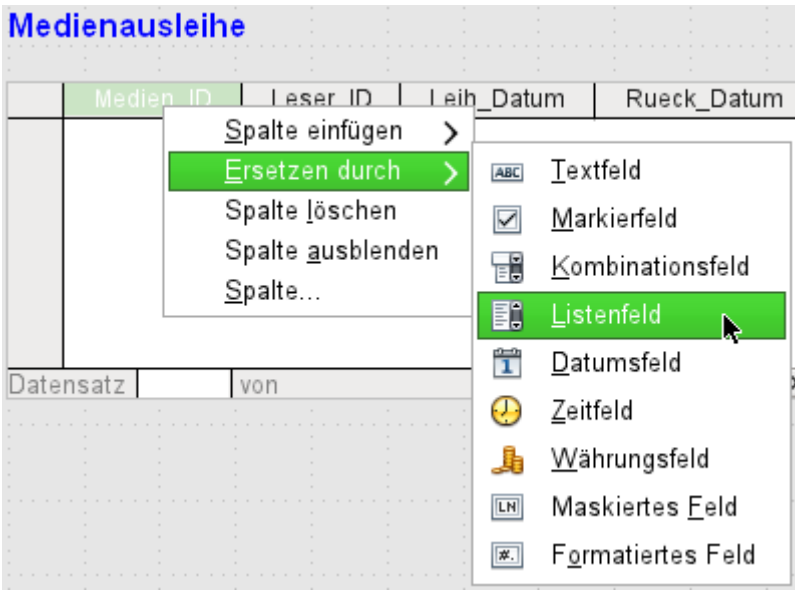
### Tabellenkontrollfeld

Unter der Beschreibung zum *Tabellen-Kontrollfeld* wurde bereits über den Tabellenassistenten das entsprechende Tabellenkontrollfeld erstellt. Es hat allerdings noch einige Nachteile, die zu verbessern sind:

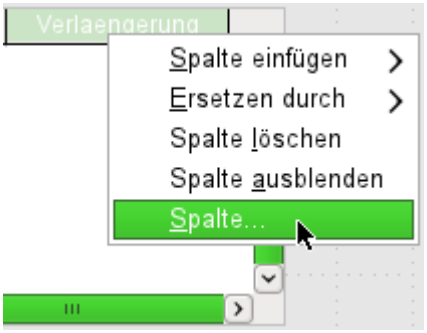
- Aus den Feldern «Medien_ID» und «Leser_ID» muss ein Listenfeld erstellt werden.
- Numerische Felder müssen gegebenenfalls auf Felder ohne Nachkommastellen umgestellt werden, da der Assistent hier grundsätzlich 2 Nachkommastellen belässt.




Die Änderung von Feldern innerhalb des Tabellen-Kontrollfeldes ist allerdings nicht auf die gleiche Art möglich, wie es oben für andere Kontrollfelder beschrieben wurde. Im Navigator endet die Beschreibung der Felder beim Tabellen-Kontrollfeld. Von den im Tabellen-Kontrollfeld liegenden Kontrollfeldern für die dem Formular zugrundeliegenden Tabelle weiß der Navigator nichts.
Dies gilt in gleichem Maße übrigens später auch, wenn mittels Makros auf die Felder zugegriffen werden soll. Sie sind mit Namen nicht ansprechbar.




Die Kontrollfelder innerhalb des Tabellenkontrollfeldes werden als Spalten bezeichnet. Über das Kontextmenü ist es jetzt möglich, Felder durch andere Felder zu ersetzen. Allerdings steht nicht die ganze Palette an Feldern zur Verfügung. So fehlen Buttons, Optionsfelder oder das grafische Kontrollfeld.




Die Eigenschaften der Felder sind über das Kontextmenü unter dem Begriff Spalte verborgen. Hier kann dann z. B. das numerische Feld *Verlängerung* geändert werden, so dass keine Nachkommastellen mehr angezeigt werden. Auch der dort standardmäßig eingetragene minimale Wert von –1.000.000,00 macht wohl für Verlängerungen wenig Sinn. Die Anzahl dürfte wohl im positiven einstelligen Bereich bleiben.

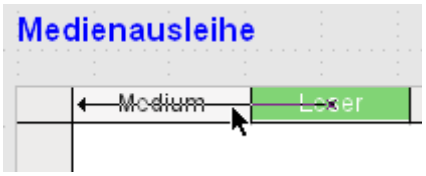
Sobald die Eigenschaften einer Spalte aufgerufen sind, lässt sich ohne das Eigenschaftsfeld zu schließen auch eine andere Spalte aufrufen. Ohne eine separate Speicherung ist es so möglich, alle Felder nacheinander abzuarbeiten.

Die Speicherung erfolgt schließlich im gesamten Formular und letztlich mit dem Datenbankdokument selbst.

Die Eigenschaften dieser in das Tabellenkontrollfeld eingebauten Felder sind übrigens nicht ganz so umfangreich wie die der Felder außerhalb. Die Schrift z. B. lässt sich nur für das ganze Tabellenkontrollfeld einstellen. Außerdem gibt es hier nicht die Möglichkeit, einzelne Spalten ohne Tabulatorstop zu überspringen.

| **Tipp** | Durch ein Formular erfolgt die Bewegung entweder mit der Maus oder mit dem Tabulator.<br>Gerät ein Tabulator einmal in ein Tabellenkontrollfeld, so bewegt sich der Cursor dort mit jeder Tabulatorbewegung ein Feld weiter nach rechts und bei Zeilenende schließlich zum ersten Feld des nächsten Datensatzes im Tabellenkontrollfeld.<br>Aus dem Tabellenkontrollfeld heraus geht es mittels **`Strg + Tab`** . |
|---|---|

Die Anordnung der Spalten lässt sich über das Verschieben der Spaltenköpfe ändern:




Wird diese Verschiebung im Formularentwurf gemacht, so ist sie dauerhaft wirksam. Vorübergehend lässt sich so eine Verschiebung auch im Formular erledigen, wenn es zur Eingabe von Daten geöffnet ist.

| **Hinweis** | Leider enthält diese Funktion zur Zeit einen Bug. Verschiebungen sind seit der Version LO 3.6 nicht mehr möglich. ( *Bug 54021* )<br>Zur Zeit können also nur Felder neu eingefügt und auf diese Weise auch positioniert werden. Ein Kopieren eines Feldes ist bei gedrückter linker Maustaste möglich, nachdem vorher mit der rechten Maustaste auf den Tabellenkopf geklickt wurde. Damit werden aber nur die ursprünglichen Formate kopiert, die beim Erstellen des Tabellenkontrollfeldes über den Assistenten erstellt wurden. Anschließende Änderungen berücksichtigt die Kopie zur Zeit nicht. |
|---|---|

Sollen nur bestimmte Felder zum Bearbeiten offen stehen, so bietet es sich an, das Formular gleich mit mehreren Tabellenkontrollfeldern zu bestücken, denn der Tabulator wird standardmäßig von einem Tabellenkontrollfeld gefangen.

In dem in *Abbildung 14* aufgezeigten Formular wird oben die Ausgabe von Medien geregelt. Hier sind nur die direkt notwendigen Felder.

Im unteren Tabellen-Kontrollfeld erscheinen noch einmal alle Felder, damit auch ersichtlich ist, für welche Person und welches Medium denn die Rückgabe erfolgt.

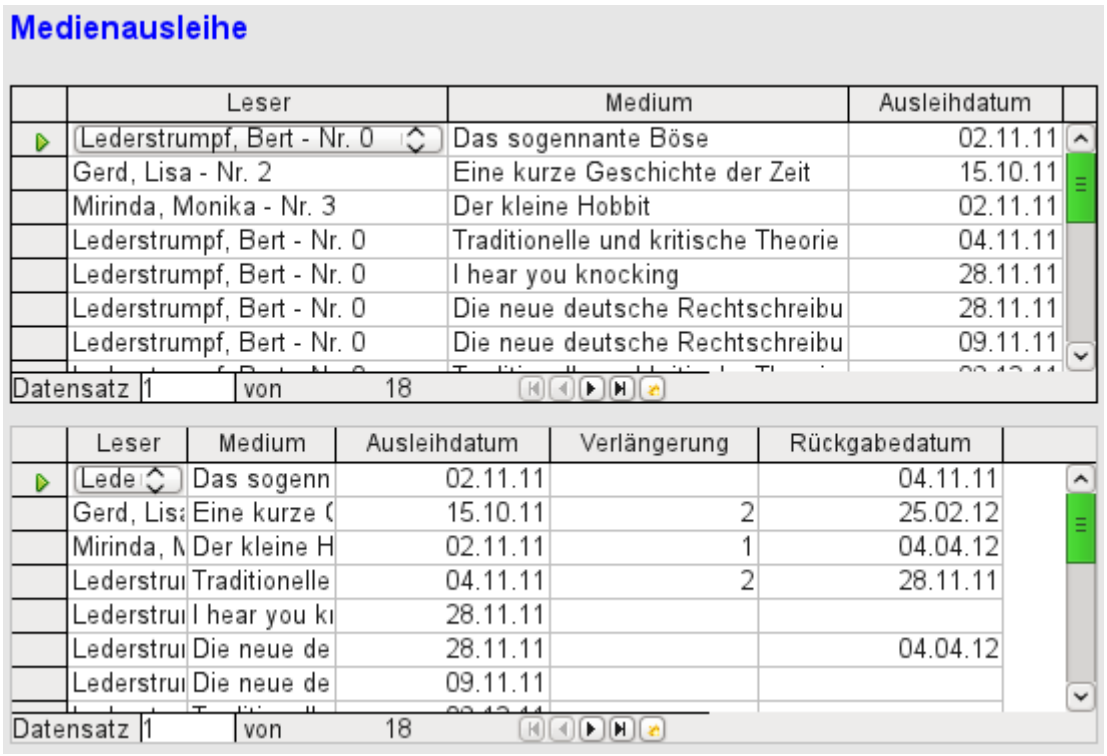
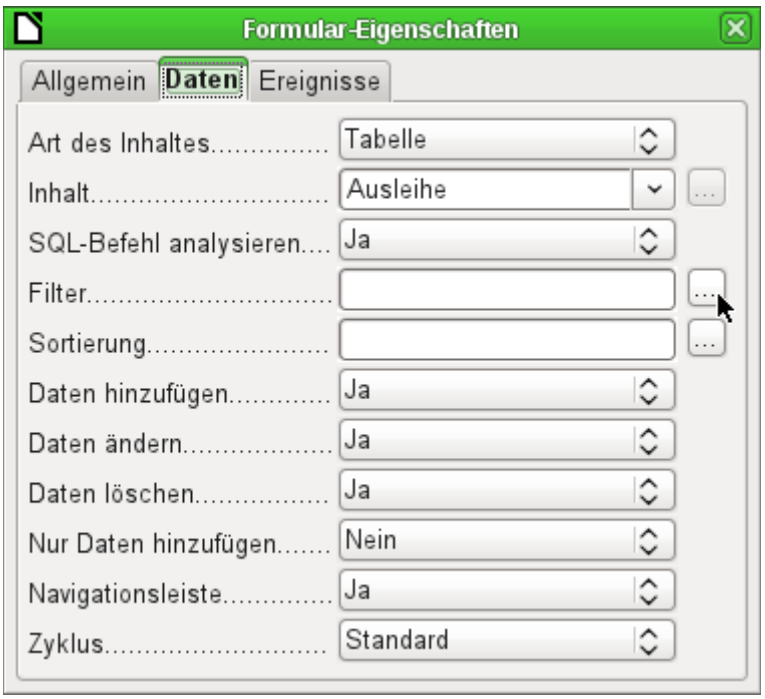



*Abbildung 13: Ein Formular - mehrere Tabellen-Kontrollfelder*

Diese Abbildung zeigt allerdings noch einen Schönheitsfehler, der dringend behoben werden sollte. In dem oberen Tabellenkontrollfeld fällt auf, dass das gleiche Medium mehrmals auftaucht. Dies liegt daran, dass auch die Medien angezeigt werden, die eigentlich zurückgegeben wurden. Die Daten müssen für eine saubere Ausleihe also gefiltert werden. Daten mit «Rückgabedatum» brauchen gar nicht erst zu erscheinen.

Diese Filterung ist entweder über eine Abfrage oder direkt in den Formulareigenschaften möglich. Wird in den Formulareigenschaften eine Filterung erstellt, so kann diese auch bei der Eingabe in das Formular vorübergehend ausgeschaltet werden. Die Filterung mittels Abfrage wird in dem entsprechenden Kapitel «Abfragen» aufgegriffen. Hier wird jetzt der Weg über die Formulareigenschaften gezeigt:

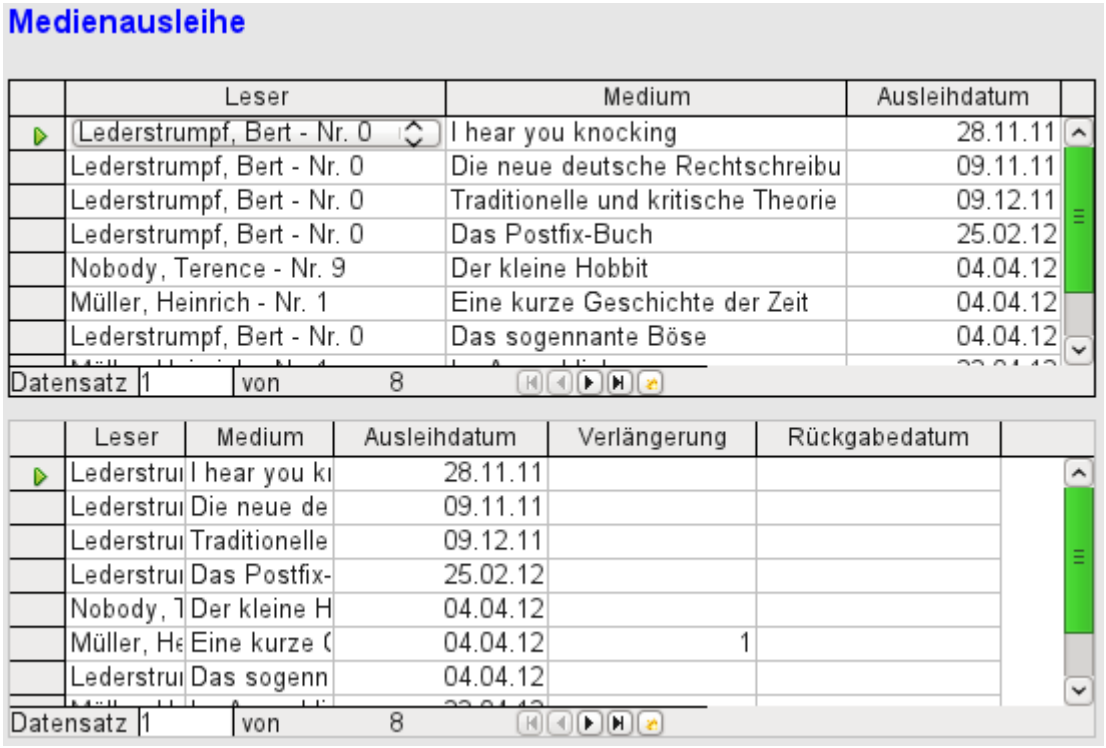



Die Filterung wird über den Button mit den drei Punkten gestartet. Sie könnte allerdings auch direkt in das Textfeld eingegeben werden, wenn die SQL-Formulierung bekannt ist.

In der grafischen Benutzeroberfläche lässt sich jetzt das Feld mit dem Namen "Rueck_Datum" auswählen. Angezeigt werden sollen nur die Datensätze, bei denen diese Feld «leer» ist, wobei «leer» für die SQL-Bezeichnung **`NULL`** steht.

In dem **Formular-Eigenschaften → Daten → Filter** steht dann (**`"Rueck_Datum" IS NULL`**). Mit Hilfe des Filters können alle Formulierungen gewählt werden, die auch in Abfragen erstellt würden. Es muss lediglich darauf geachtet werden, dass der Filter in einfachen Klammern eingetragen wird. Siehe dazu auch «Filtern von Tabellen» und die Erweiterung mit der «WHERE SQL-Expression» aus dem Kapitel «Abfragen».

Das auf diese Art bereinigte Formular sieht dann schon etwas übersichtlicher aus:




Sicher ist dies noch verbesserbar, bietet aber neben der Funktionalität der anderen Formulare den unbestreitbaren Vorteil, dass alle Medien auf einen Blick sichtbar sind.

Die Bearbeitung von Daten mit Tabellenkontrollfeldern ist ähnlich der von Tabellen. Mit einem Rechtsklick auf den Datensatzmarkierer wird bei existierenden Datensätzen die Löschung des Datensatzes angeboten, bei Neueingaben kann die Dateneingabe rückgängig gemacht oder abgespeichert werden.

Wird eine Zeile verlassen, so wird der Datensatz automatisch abgespeichert.

Noch ist das Formular zur Medienausleihe in vielen Bereichen zu verbessern.

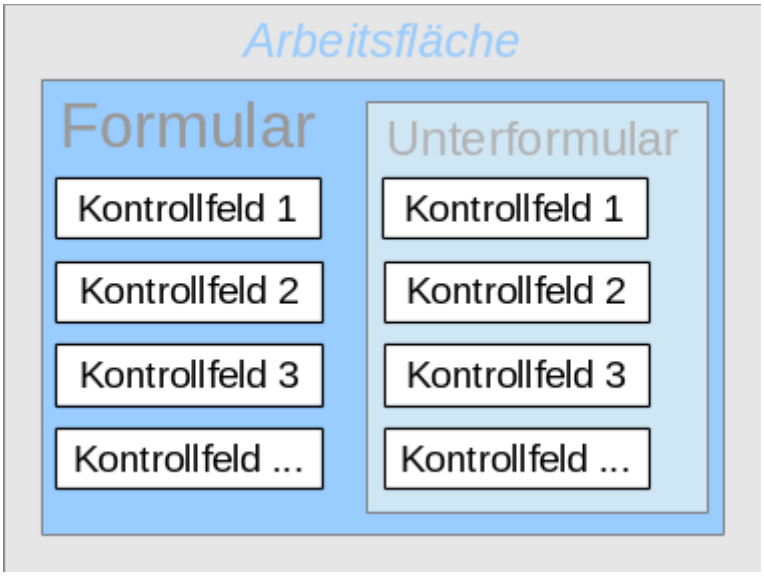
- Es wäre wünschenswert, wenn an einer Stelle der Leser ausgewählt würde und an anderer Stelle zu diesem Leser die entliehenen Medien erscheinen.
- In der oberen Tabelle sind lauter Datensätze zu sehen, die dort eigentlich gar nicht notwendig sind. Die Medien sind ja schon entliehen. Das obere Tabellenblatt ist aber erstellt worden, um die Ausleihe zu ermöglichen. Besser wäre es, wenn hier nur ein leeres Blatt erscheint, das dann mit den neuen Ausleihen ausgefüllt wird.

Solche Lösungen sind mit Hilfe von weiteren Formularen möglich, die hierarchisch aufeinander aufgebaut sind und eine getrennte Sicht auf die Daten ermöglichen.

| **Tipp** | Das Tabellenkontrollfeld kann mit anderen Feldern im Formular kombiniert werden. In dem Tabellenkontrollfeld werden z. B. nur die Medientitel angezeigt, in den darunter stehenden Formularfeldern dann an diesem Datensatz Änderungen vorgenommen.<br><br>Bei der Kombination von Tabellenkontrollfeld und weiteren Formularfeldern gibt es einen kleinen, leicht behebbaren Bug. Wenn beide Feldarten zusammen in einem Formular liegen, so läuft der Cursor von den anderen Formularfeldern automatisch in das Tabellenkontrollfeld, obwohl dieses Feld standardmäßig auf **Tabstop → Nein** eingestellt ist. Das kann behoben werden, indem der Tabstop einmal auf «Ja» und anschließend wieder auf «Nein» eingestellt wird. Dann erst wird «Nein» wirklich übernommen. |
|---|---|

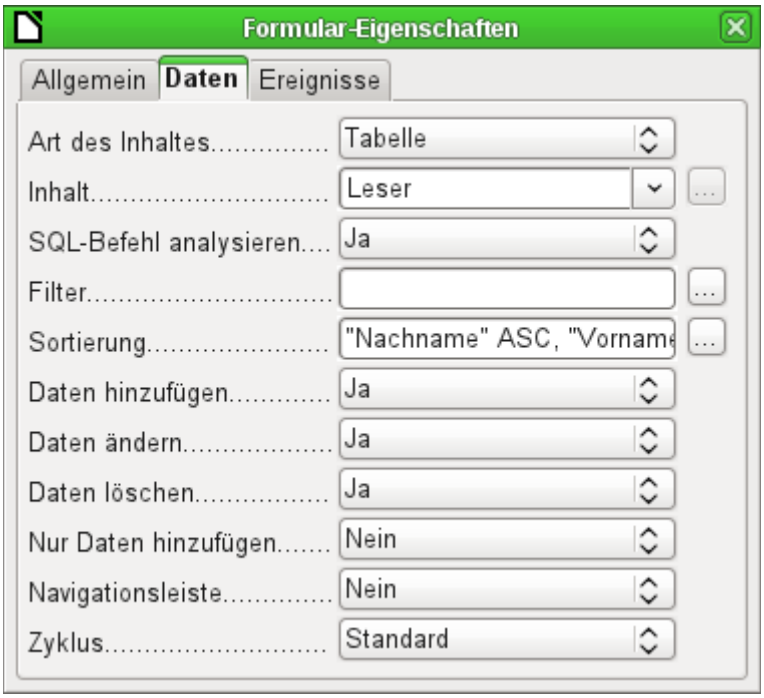
## Hauptformular und Unterformular

Ein Unterformular liegt wie ein Formularkontrollfeld innerhalb eines Formulars. Wie ein Formularkontrollfeld wird es mit den Daten des (Haupt)-Formulars verbunden. Allerdings kann es als Datenquelle eine andere Tabelle oder eine Abfrage (bzw. einen SQL-Befehl) beinhalten. Für ein Unterformular ist nur wichtig, dass seine Datenquelle irgendwie mit der Datenquelle des Hauptformulars verbunden werden kann.

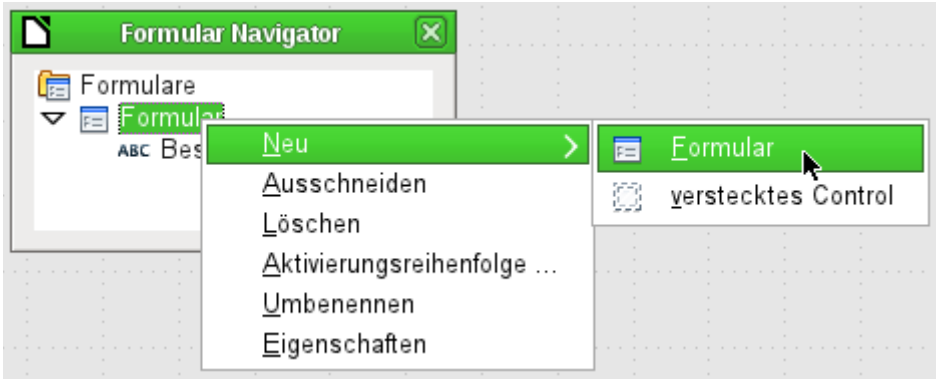



Typische Tabellenkonstruktionen, die sich für Unterformulare eignen, sind Tabellen mit einer Eins-zu-Viele Beziehung. Hauptformulare zeigen eine Tabelle an, zu deren Datensatz dann in den Unterformularen viele abhängige Datensätze aufgezeigt werden.

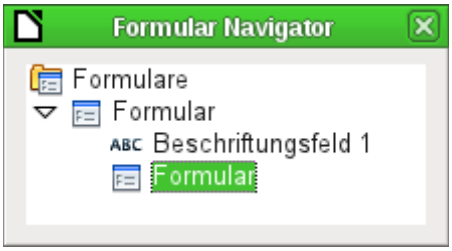
Wir nutzen jetzt erst einmal die Beziehung der Tabelle "Leser" zur Tabelle "Ausleihe" (siehe «Tabellen Ausleihe»). Die Tabelle "Leser" wird Grundlage für das Hauptformular, die Tabelle "Ausleihe" wird in dem Unterformular wiedergegeben.



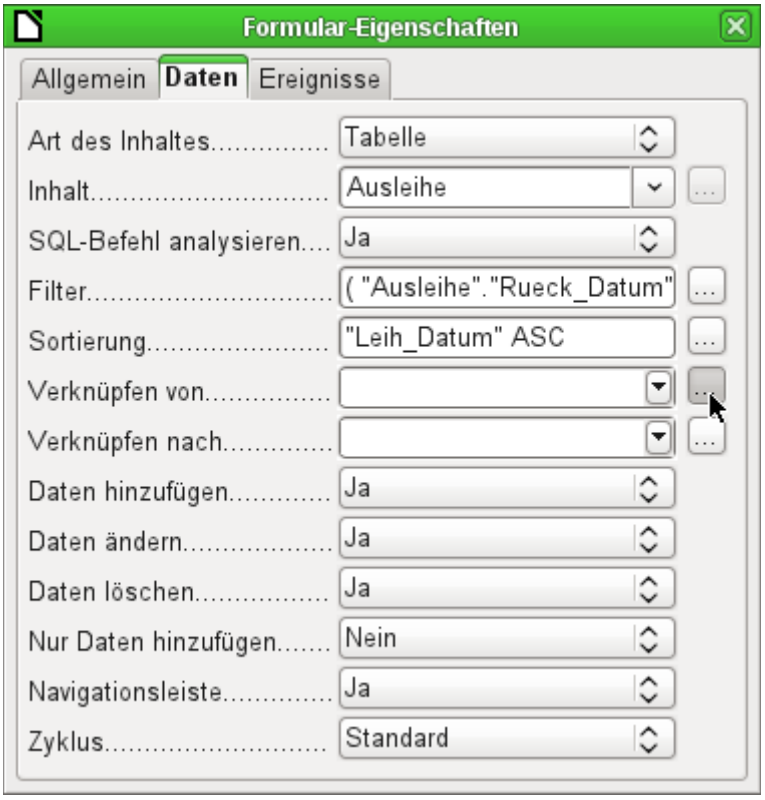
Das Hauptformular wird auf die Tabelle "Leser" eingestellt. Damit die Leser auch schnell genug gefunden werden, wird die Tabelle sortiert wiedergegeben. Auf eine Navigationsleiste wird verzichtet, weil zwischen Hauptformular und Navigationsleiste der Inhalt des Unterformulars angezeigt wird. Stattdessen soll das Formularkontrollelement *Navigationsleiste* eingebaut werden.




Durch Rechtsklick auf das Hauptformular im Formular-Navigator wird über das Kontextmenü ein neues Formular gegründet. Das Formular hat standardmäßig wieder den Namen Formular, ist jetzt aber ein Element im Unterverzeichnis des Hauptformulars.

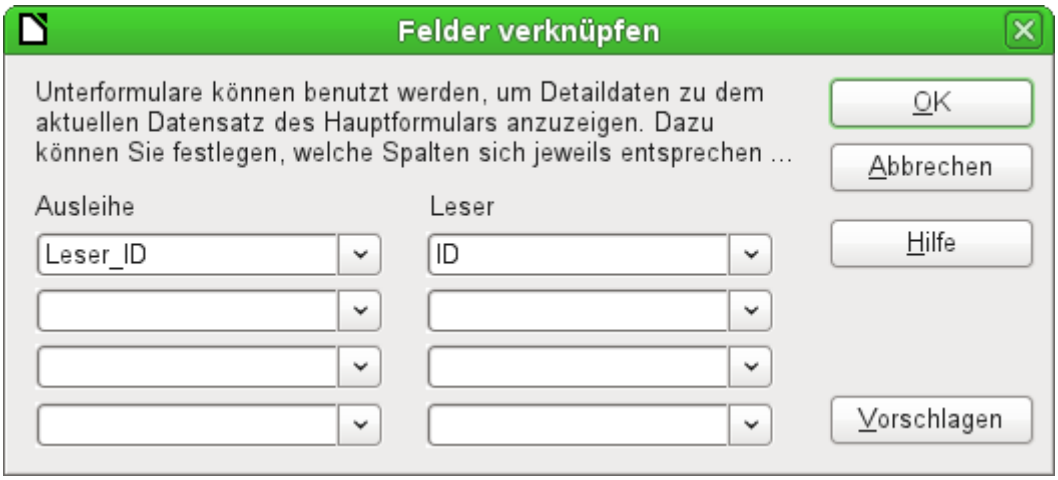



Die Eigenschaften des Unterformulars müssen jetzt entsprechend eingestellt werden, damit es die richtige Datenquelle und die zu dem entsprechenden Leser gehörigen Daten wiedergibt.



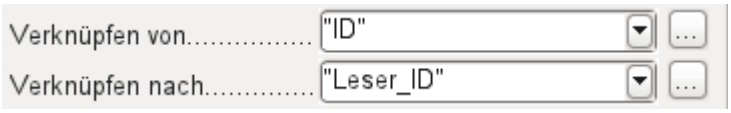
Für das Unterformular wird die Tabelle Ausleihe gewählt. Beim Filter wird eingestellt, dass das Rückgabedatum leer sein soll (**`"Rueck_Datum" IS NULL`**). Dadurch werden keine bereits zurückgegebenen Medien angezeigt. Die Datensätze sollen nach dem Entleihdatum vorsortiert werden. Diese Sortierung zeigt das am längsten entliehene Medium am weitesten oben an.

Über **Verknüpfen von** und **Verknüpfen nach** wird eine Verbindung zum Hauptformular hergestellt, in dem das Unterformular liegt. Der Button mit den drei Punkten zeigt wieder an, dass es hier ein helfendes Fenster für die Auswahl der Einstellungen gibt.




Unter «Ausleihe» werden die Felder der Tabelle "Ausleihe" angezeigt, unter «Leser» die der Tabelle "Leser". Die "Leser_ID" der "Ausleihe" soll gleichbedeutend sein mit der "ID" der Tabelle "Leser".

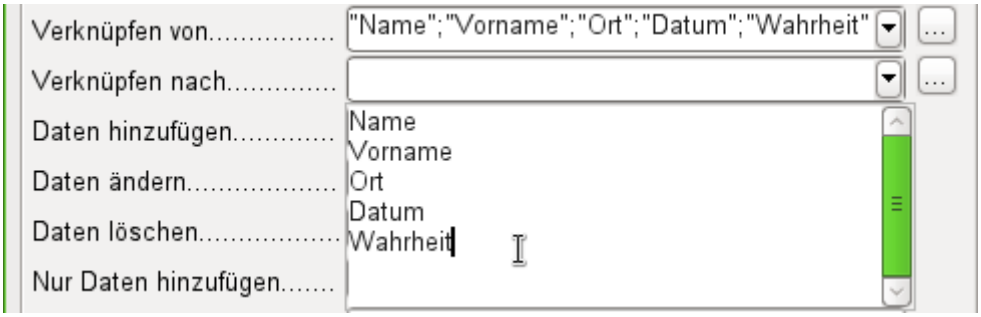
Obwohl diese Verknüpfung bereits in der Datenbank unter **Extras → Beziehungen** erstellt wurde (siehe «Verknüpfung von Tabellen») greift die hinter dem Button Vorschlagen liegende Funktion hierauf nicht zurück und möchte stattdessen den ersten Fremdschlüssel aus der Tabelle "Ausleihe", nämlich "Medien_ID", mit "ID" aus der Tabelle "Leser" verbinden. Dies gelingt dem Assistenten zur Erstellung von Formularen besser.




Die ausgewählte Verknüpfung von der Tabelle des Unterformulars nach der Tabelle des Hauptformulars wird jetzt mit den entsprechenden Feldern der Tabellen angegeben.

**Tipp**

Der Dialog **Felder verknüpfen** zeigt die Möglichkeit an, 4 Felder zu verknüpfen. Es ist aber auch problemlos möglich, mehr Felder in die Verknüpfung mit einzubeziehen:




Am einfachsten ist es, die Bezeichnungen in ein aufgeklapptes Listenfeld zu schreiben. Der Zeilenumbruch wird dabei durch *Shift* + *Enter* erreicht.

Um jetzt ein Tabellenkontrollfeld für das Hauptformular zu erstellen, muss das Hauptformular im Formular-Navigator markiert sein. Dann zeigt bei eingeschaltetem Tabellenkontrollfeld-Assistenten der Assistent die Felder an, die im Hauptformular zur Verfügung stehen. Entsprechend wird mit dem Unterformular verfahren.

Nachdem so die Tabellenkontrollfelder aufgezogen wurden, werden die entsprechenden Änderungen durchgeführt, die schon beim einfachen Formular erklärt wurden:

- Ersetzen des numerischen Feldes «Medien_ID» im Unterformular durch ein Listenfeld.
- Umbenennung des Feldes «Medien_ID» in Medien.
- Anpassung der numerischen Felder an ein Format ohne Nachkommastellen.
- Eingrenzung der minimalen und maximalen Werte.
- Umbenennung anderer Felder, um Platz zu sparen oder Umlaute darzustellen, die bei der Feldbenennung in den Datenbanktabellen vermieden wurden.

Sortier- und Filterfunktion werden für das Hauptformular ergänzt, indem eine Navigationsleiste hinzugefügt wird. Die anderen Felder der Navigationsleiste werden nicht benötigt, da sie vom Tabellenkontrollfeld weitgehend zur Verfügung gestellt werden. (Datensatzanzeige, Datensatznavigation) bzw. durch die Bewegung im Tabellenkontrollfeld erledigt werden (Speicherung von Daten).

Das erstellte Formular könnte schließlich wie in der folgenden Abbildung aussehen:

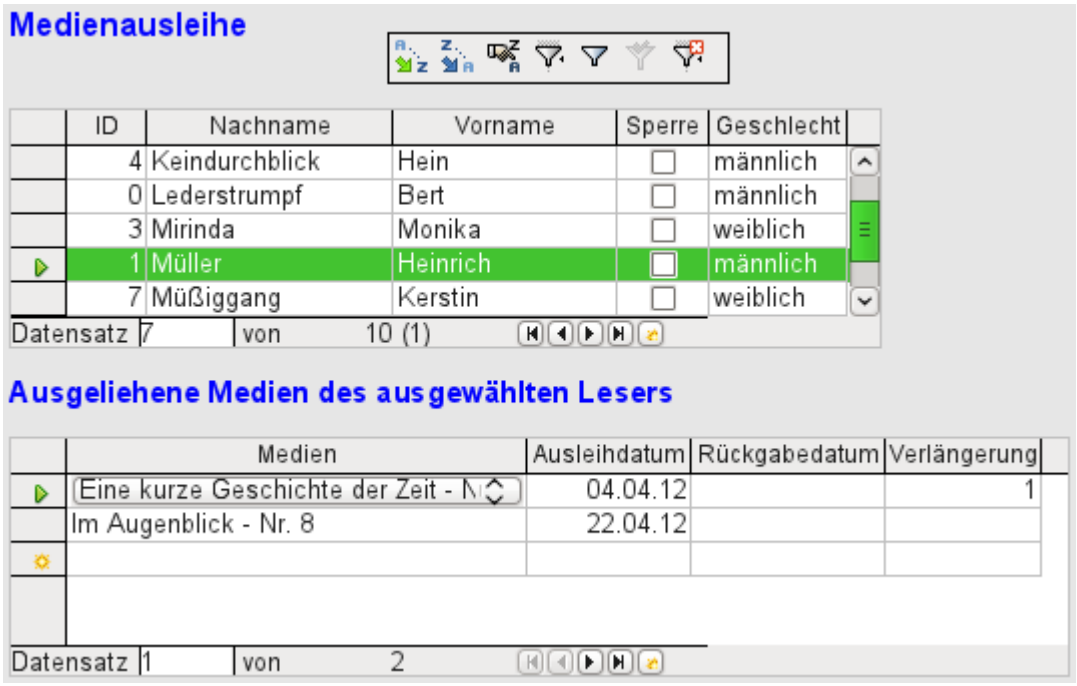



*Abbildung 14: Formular, bestehend aus Hauptformular (oben) und Unterformular (unten).*

Wird jetzt im Hauptformular ein Leser ausgewählt, so werden im Unterformular nur die Medien aufgezeigt, die der Leser zur Zeit entliehen hat. Wird ein Medium zurückgegeben, so erscheint dies noch so lange im Formular, bis das Formular selbst aktualisiert wird. Dies geschieht automatisch, wenn im Hauptformular ein anderer Datensatz gewählt wurde. Bei der erneuten Anwahl des ursprünglichen Lesers sind also die zurückgegebenen Medien nicht mehr in der Anzeige.

Diese verzögerte Aktualisierung ist in diesem Fall wohl auch erwünscht, da so direkt eingesehen werden kann, welche Medien denn jetzt gerade auf der Theke der Mediothek liegen und ob diese schon registriert wurden.

Diese Formularkonstruktion bietet schon deutlich mehr Komfort als die vorherige mit nur einem einzigen Formular. Allerdings gibt es noch Details, die verbesserungswürdig erscheinen:

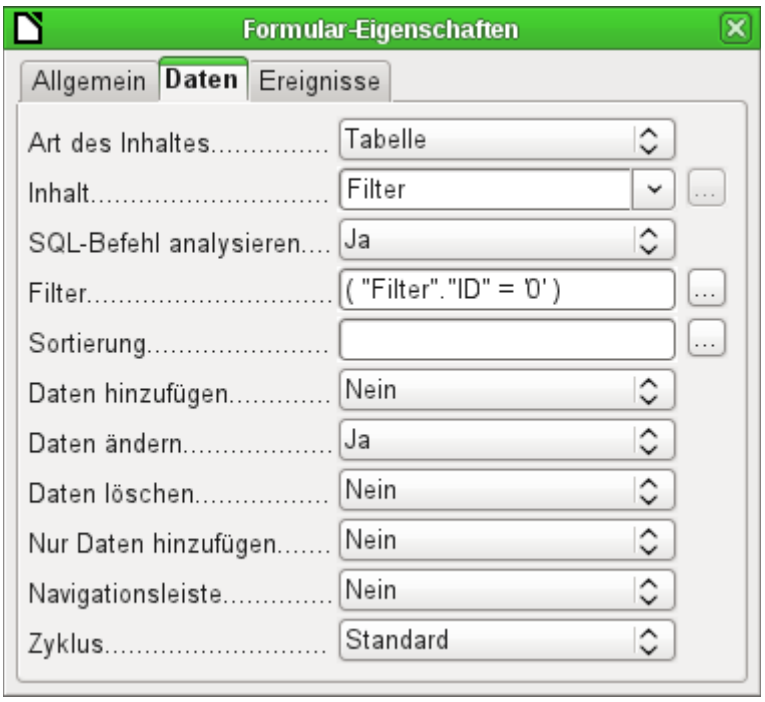
- Medien und Ausleihdaten können geändert werden, wenn die Medien schon länger entliehen sind.
  Eine Änderung der Medien-Daten führt dazu, dass nicht mehr nachvollzogen werden kann, welches Medium denn nun noch in der Mediothek vorhanden ist und welches entliehen wurde.
  Eine Änderung des Ausleihdatums kann zu fehlerhaften bzw. nicht beweisbaren Anmahnungen führen.
- Wird ein Leser nicht durch Klick auf den Datensatzmarkierer markiert, so zeigt nur der kleine grüne Pfeil auf dem Markierer an, welcher Datensatz gerade aktiv ist. Es ist auch möglich, den aktiven Datensatz komplett aus dem Tabellenkontrollfenster zu scrollen.
  Statt des Textes «Ausgeliehene Medien des ausgewählten Lesers» würde hier besser auch der Name erwähnt.
- Es ist möglich, mehrmals das gleiche Medium auszuleihen, ohne dass es zurückgegeben wurde.
- Es ist möglich, die Datensätze für ausgeliehene Medien einfach zu löschen.
- Auch im Hauptformular sind Änderung und Löschung von Daten möglich. Dies kann bei kleinen Mediotheken mit wenig Publikumsbetrieb sinnvoll sein. Sobald aber am Ausgabeschalter größere Hektik entsteht, ist die Bearbeitung von Nutzerdaten nicht an der gleichen Stelle vorzunehmen wie die Ausleihe.
  Eine Vereinfachung wäre schon, wenn eine Neuaufnahme ermöglicht würde, alte Daten aber nicht angerührt werden dürfen. Denn ob nun Löschung oder komplette Veränderung des Namens – das Ergebnis bleibt für die Mediothek das Gleiche.

Zuerst wird einmal die Auswahl der Leser verbessert. Dies soll vor Änderungen in der Ausleihe schützen. Eine einfache Lösung wäre, keine Änderungen zuzulassen, aber neue Datensätze eingeben zu können. Dazu wird immer noch eine Suchfunktion benötigt, wenn ein Leser ein Medium entleihen will. Besser wäre, in einem Listenfeld die Leser auszusuchen und in voneinander getrennten Tabellenkontrollfeldern die Ausgabe und die Rückgabe zu erledigen.

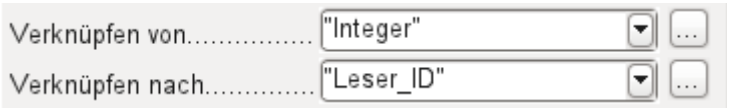
Für das Hauptformular benötigen wir eine Tabelle, in die das Listenfeld seinen mit dieser Tabelle verbundenen Wert schreiben kann. Die Tabelle muss also ein Integer-Feld und einen Primärschlüssel haben. Sie wird beständig nur einen Datensatz enthalten; daher kann das Feld ID als Primärschlüssel ruhig 'Tiny Integer' sein. Die folgende Tabelle mit der Bezeichnung "Filter" soll also gebildet werden:

| ***Tabellenname: Filter*** | |
|---|---|
| ***Feldname*** | ***Feldtyp*** |
| ID | Tiny Integer, Primärschlüssel |
| Integer | Integer |

Die Tabelle wird mit einem Primärschlüsselwert gefüllt, und zwar dem Wert 0. Diesen Datensatz wird das Hauptformular beständig lesen und neu schreiben.

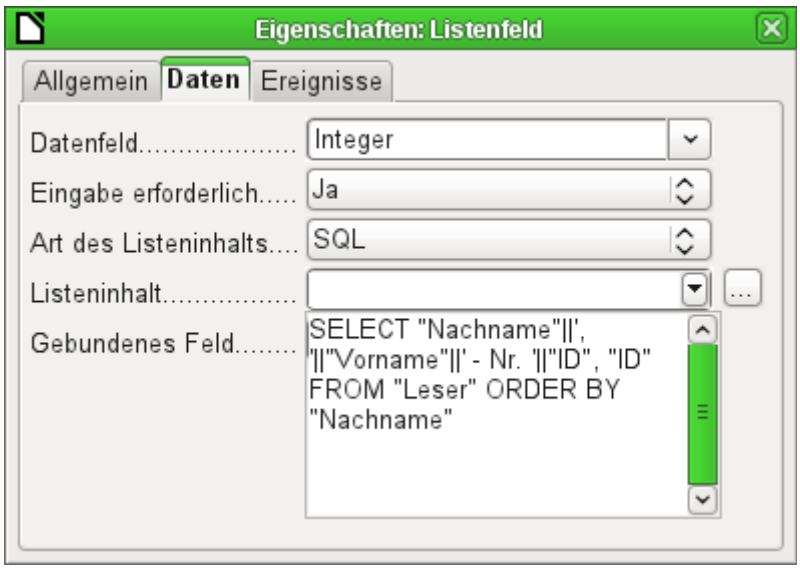



Das Hauptformular beruht auf der Tabelle "Filter". Es wird nur der Wert aus der Tabelle gelesen, bei dem der Primärschlüssel "ID" '0' ist. Es sollen keine Daten hinzugefügt werden, sondern nur der aktuelle Datensatz beständig neu geschrieben werden. Daher ist hier nur das Ändern erlaubt, eine Navigationsleiste sowieso überflüssig.




Das Hauptformular wird mit dem Unterformular so verknüpft, dass der Wert aus dem Feld "Integer" der Tabelle "Filter" gleich dem Wert aus dem Feld "Leser_ID" aus der Tabelle "Ausleihe" ist. Das Unterformular bleibt in seinen Eigenschaften ansonsten gegenüber der Vorversion unberührt.

Bevor jetzt im Hauptformular ein Listenfeld aufgezogen wird, wird erst einmal der Assistent ausgeschaltet. Mit dem Assistenten könnte nur ein Feld erzeugt werden, das lediglich einen Feldinhalt anzeigt; es wäre unmöglich, Nachname und Vorname und zusätzlich noch eine Nummer in dem Anzeigefeld des Listenfeldes zu positionieren. Wie bereits bei dem einfachen Formular wird jetzt das Listenfeld mit *Nachname, Vorname – Nr. ID* bestückt. Das Listenfeld gibt außerdem die ID an die darunterliegende Tabelle weiter.




Neben dem Listenfeld wird ein Button erstellt. Dieser Button ist allerdings Bestandteil des Unterformulars. Er soll gleichzeitig zwei Funktionen übernehmen: Abspeicherung des Datensatzes aus dem Hauptformular und Aktualisierung der Tabelle im Unterformular. Dazu reicht es, dem Button im Unterformular die Aktualisierung zuzuweisen. Der Speichervorgang für das veränderte Hauptformular wird dadurch automatisch ausgelöst.

Der Button kann einfach mit '*OK*' als Aufschrift versehen werden. Als Aktion wird **Formular aktualisieren** zugewiesen.

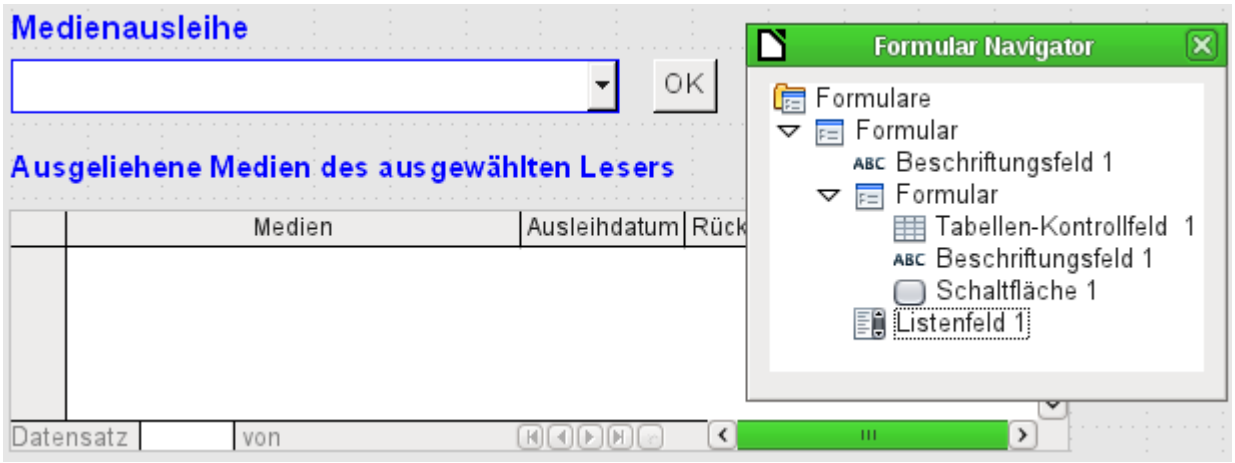


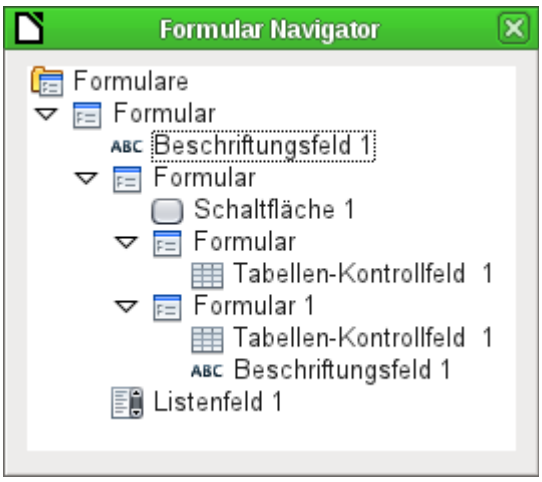

*Abbildung 15: Hauptformular als Filter für ein Unterformular*

Das Hauptformular besteht nur aus der Überschrift und dem Listenfeld, das Unterformular aus einer weiteren Überschrift, dem Tabellen-Kontrollfeld der vorherigen Version und dem Button.

Das Formular funktioniert jetzt schon insofern besser, als

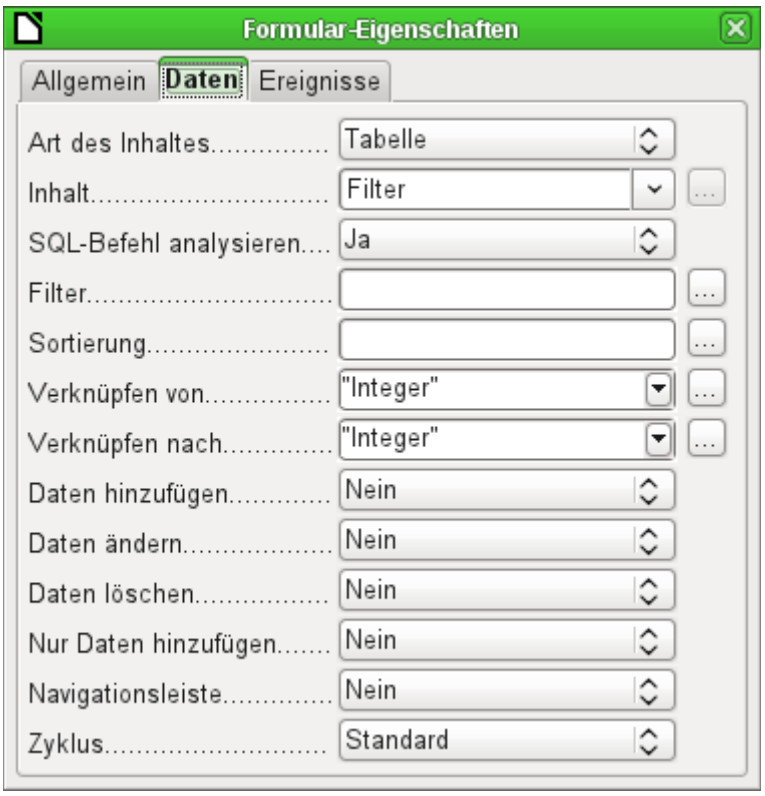
- keine Leser mehr bearbeitet, vor allem verändert und gelöscht werden können und
- Leser schneller über das Eintippen in das Kontrollfeld gefunden werden als über Filter.

Für eine weitreichendere Funktionalität (Rückgabe ohne Änderung der vorherigen Daten) muss ein zweites Unterformular gegründet werden, das sich auf die gleiche Tabelle "Ausleihe" bezieht. Damit dennoch die Funktionalität des Listenfeldes aus *Abbildung 15* gewährleistet wird, müssen beide Unterformulare allerdings noch eine Ebene tiefer gelegt werden, also Unterformulare eines Unterformulars werden. Eine Aktualisierung von Daten verläuft in der Hierarchie nur vom Hauptformular zum Unterformular abwärts. Der Button im letzten vorgestellten Formular würde nur ein Unterformular aktualisieren, nicht aber das zweite, daneben liegende Unterformular.



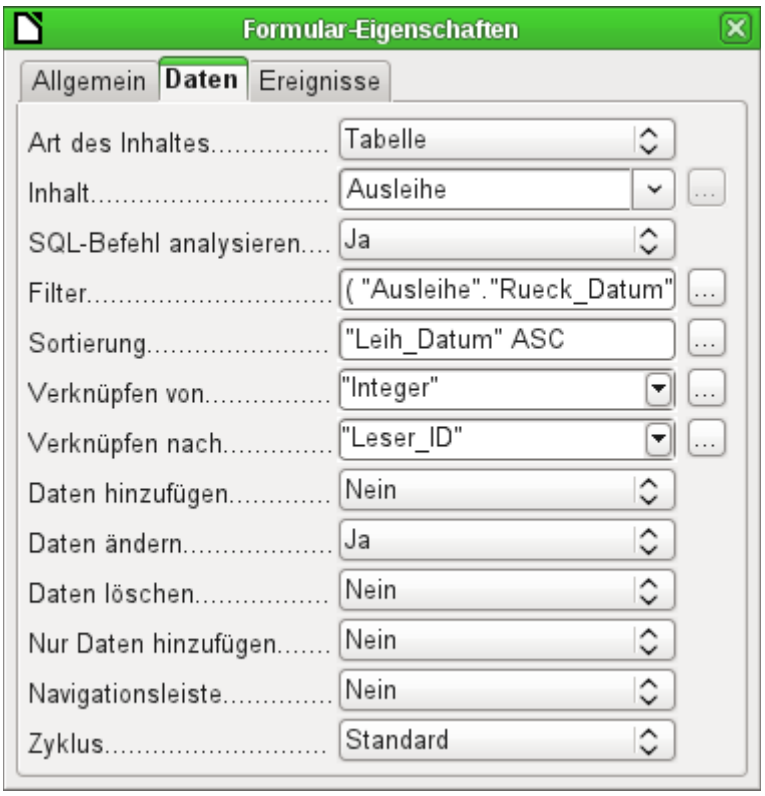

Der Formularnavigator zeigt hier die verschiedenen Ebenen an. Im Hauptformular befindet sich das Beschriftungsfeld für die Formularüberschrift und das Listenfeld, in dem die Leser ausgesucht werden. Das Listenfeld steht in der Ansicht ganz unten, da es nach dem Unterformular gegründet wurde. Diese Reihenfolge der Anzeige lässt sich leider nicht beeinflussen. Das Unterformular hat lediglich eine Schaltfläche, mit der sein Inhalt aktualisiert und der des Hauptformulars gleichzeitig abgespeichert wird. Noch eine Ebene tiefer liegen dann zwei Unter-Unterformulare. Diese werden bei der Gründung bereits unterschiedlich benannt, so dass in keiner Ebene von der Benennung her Verwechselungen auftreten können.

| **Hinweis** | Grundsätzlich sind die Benennungen der Formulare und Kontrollfelder erst einmal ohne Bedeutung. Wenn sie aber über den Namen durch Makros angesprochen werden sollen, müssen sie unterscheidbar sein. Gleiche Namen in der gleichen Formularebene erlauben keine Unterscheidung.<br>Natürlich ist es sinnvoll, bei größeren Formularkonstruktionen aussagekräftigere Namen für die Formulare und ihre Kontrollfelder zu nutzen. Ansonsten dürfte ein Auffinden des richtigen Feldes schnell zum Problem werden. |
|---|---|




Das Hauptformular und das Unterformular nutzen einfach die gleiche Tabelle. Im Unterformular werden keine Daten eingegeben. Deshalb stehen alle diesbezüglichen Felder auf *Nein.* Verknüpft werden Hauptformular und Unterformular durch das Feld, dessen Wert auch an die Unter-Unterformulare weitergegeben werden soll: das Feld "Integer" der Tabelle "Filter".

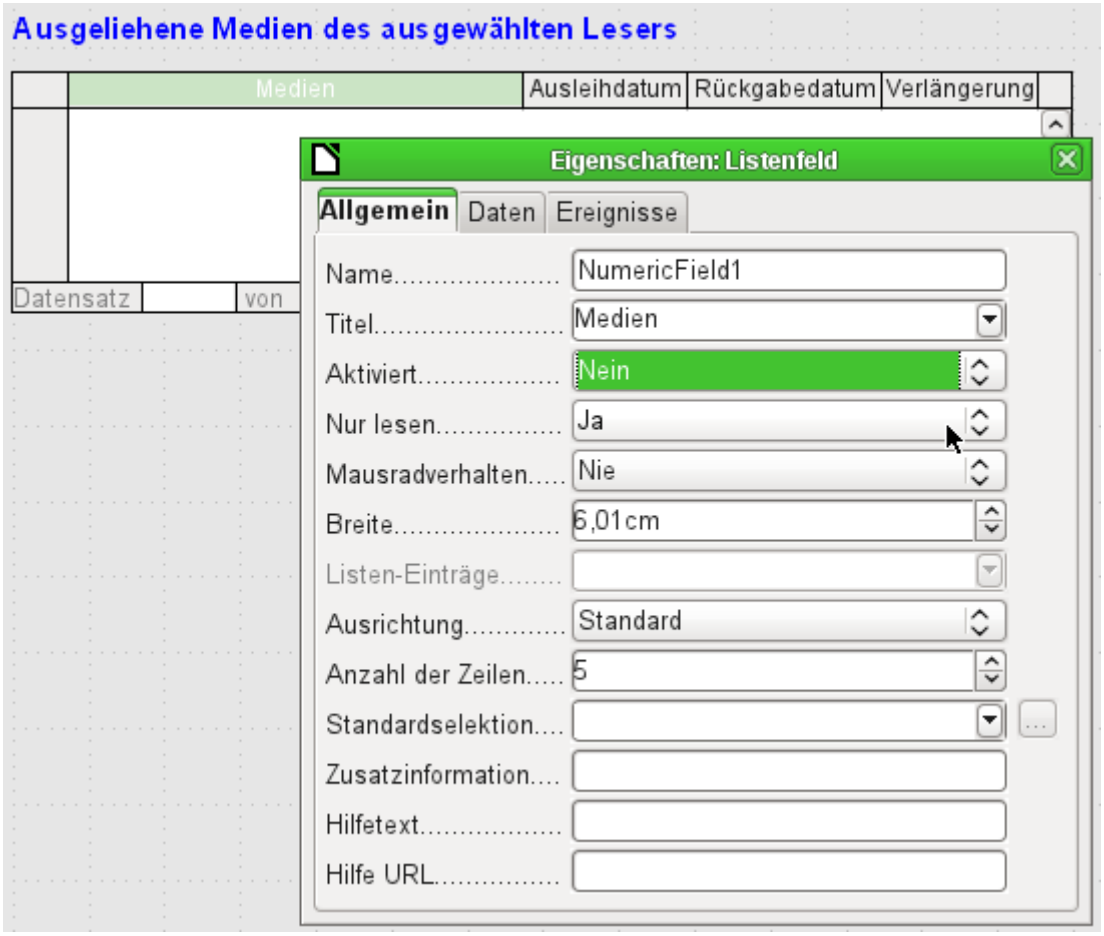
Im ersten Unter-Unterformular werden keine alten Daten angezeigt, sondern nur neue Daten verarbeitet. Hierzu reicht der Filter, der gesetzt wurde. Es werden nur Daten angezeigt, die zu der "Leser_ID" passen und deren Leihdatum leer ist (**`"Leih_Datum" IS NULL`**). Das bedeutet beim Aufruf ein leeres Tabellen-Kontrollfeld. Da das Tabellen-Kontrollfeld zwischendurch nicht laufend aktualisiert wird, bleiben die gerade neu ausgeliehenen Medien so lange in dem Tabellen-Kontrollfeld stehen, bis über den Aktualisierungsbutton '*OK*' entweder ein neuer Name ausgewählt oder auch nur die Übernahme der Daten in das zweite Unter-Unterformular veranlasst wird.




Das zweite Unter-Unterformular erfordert mehr Einstellungen. Auch dieses Formular enthält die Tabelle "Ausleihe". Hier werden aber die Daten gefiltert, bei denen das Rückgabedatum leer ist (**`"Rueck_Dat" IS NULL`**). Die Daten wird wie im vorhergehenden Formular so sortiert, dass die am längsten entliehenen Medien direkt sichtbar sind.

Wichtig sind jetzt auch die weiter unten stehenden Einträge. Alte Datensätze können geändert werden, aber es können keine neuen Datensätze hinzugefügt werden. Ein Löschen ist nicht möglich. Damit ist der erste Schritt gemacht, der notwendig ist, um später nicht Entleihdaten einfach zu löschen. Noch wäre es aber möglich, z. B. das Medium und das Entleihdatum zu ändern. Hier muss in den Eigenschaften der Spalten weiter justiert werden. Schließlich soll das Medium und das Entleihdatum nur angezeigt, aber von der Änderung ausgeschlossen werden.

Das Tabellen-Kontrollfeld wird nach der Erstellung der Formulare einfach verdoppelt. Dazu wird es markiert, anschließend kopiert, danach wird die Markierung aufgehoben und aus der Zwischenablage wieder eingeführt. Das Doppel befindet sich an der gleichen Position wie das Original, muss also noch verschoben werden. Danach können beide Tabellenkontrollfelder entsprechend bearbeitet werden. Das Tabellenkontrollfeld zur Medienrückgabe bleibt nahezu unverändert. Lediglich die Schreibrechte für die Spalten «Medien» und «Ausleihdatum» müssen geändert werden.




Während bei «Ausleihdatum» lediglich **Nur lesen** gewählt werden muss, ist dies bei Listenfeldern nicht ausreichend. Diese Einstellung verhindert nicht, dass das Listenfeld weiterhin betätigt werden kann. Wird aber **Aktiviert → Nein** gestellt, so kann dort eine Auswahl nicht mehr stattfinden. Im Tabellen-Kontrollfeld wird ein enthaltenes Listenfeld dann wie ein nicht veränderbares Textfeld angezeigt.

Im oberen Tabellen-Kontrollfeld werden alle Felder entfernt, die nichts mit der Ausleihe zu tun haben. Es beleibt lediglich das Medium als Auswahlfeld sowie das Ausleihdatum "Leih_Dat" stehen.

Wird schließlich noch die Abfrage für das Listenfeld im oberen Tabellen-Kontrollfeld entsprechend gewählt, so werden dort nur Medien angezeigt, die noch entliehen werden können. Mehr dazu im Kapitel «Abfragen».

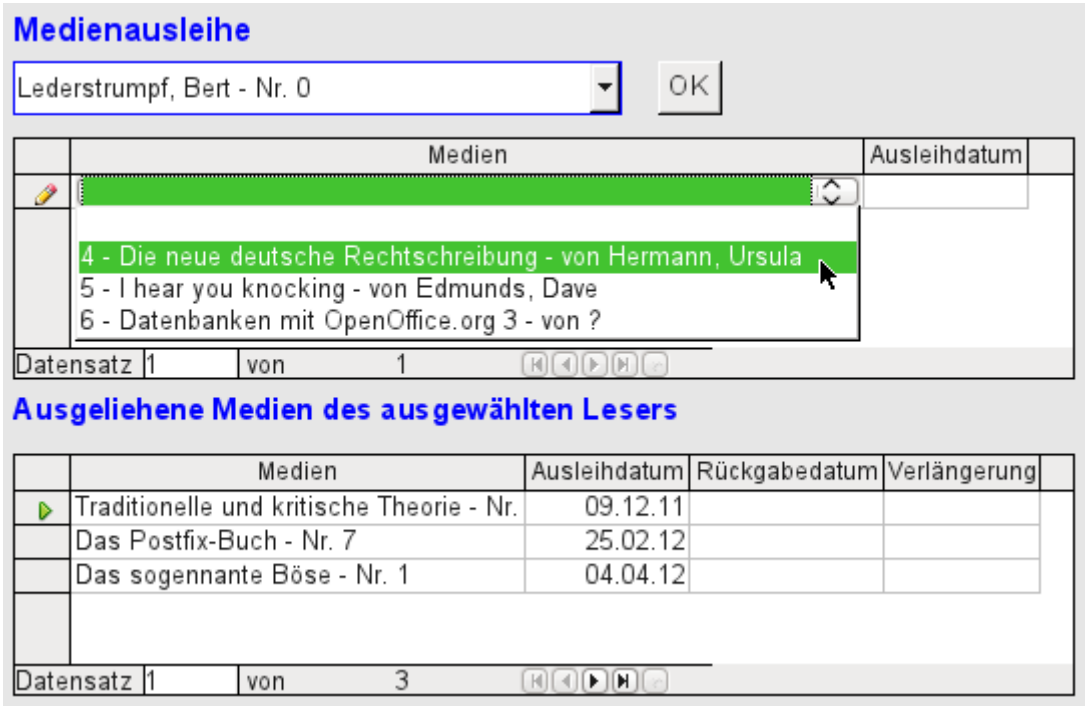
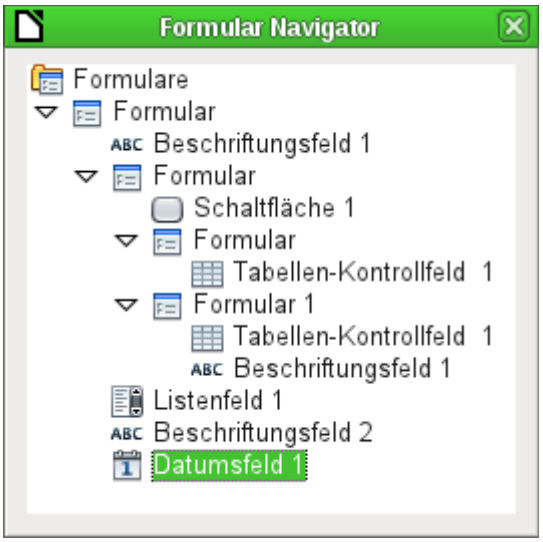



*Abbildung 16: Das Auswahlfeld im oberen Unterformular zeigt nur Medien an, die nicht ausgeliehen sind.*

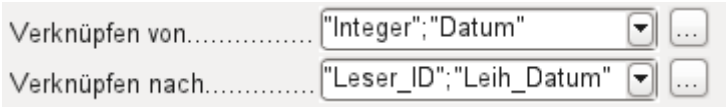
Das Formular zur Medienausleihe ist jetzt schon wesentlich besser zu bedienen. Kommt ein Leser an die Ausleihtheke, so wird der Name herausgesucht. Die zu entleihenden Medien können aus dem Listenfeld gewählt und das Entleihdatum eingestellt werden. Mit dem Tabulator geht es dann zum nächsten Datensatz.

Eine letzte Verbesserung wäre noch wünschenswert: Das Entleihdatum muss jedes Mal gewählt werden. Stellen wir uns einen Tag in der Mediothek mit vielleicht 200 Entleihvorgängen vor, vielleicht auch nur eine Person, die gleich 10 Medien auf einmal entleiht. Das wären mehrmals hintereinander die gleichen Eingabevorgänge für ein Feld. Hier muss eine Einsparmöglichkeit her.

Unser Hauptformular beruht auf einer Tabelle "Filter". Das Hauptformular arbeitet dabei immer nur mit dem Datensatz, der als Primärschlüssel die "ID" '0' hat. In die Tabelle "Filter" können ohne weiteres noch mehr Felder eingebaut werden. Da noch kein Feld enthalten ist, das ein Datum speichern kann, gründen wir einfach ein neues Feld mit dem **Feldname → Datum** und dem **Feldtyp → Datum**. In der Tabelle "Filter" wird jetzt nicht nur die "Leser_ID" ("Filter"."Integer") sondern auch das "Leih_Datum" ("Filter"."Datum") gespeichert.



Im Hauptformular erscheint jetzt zusätzlich ein Datumsfeld, außerdem noch ein Beschriftungsfeld, das auf den Inhalt des Datumsfeldes hinweist. Der Wert aus dem Datumsfeld wird in der Tabelle "Filter" gespeichert und über die Verbindung vom Unterformular zum Unter-Unterformular weitergegeben.



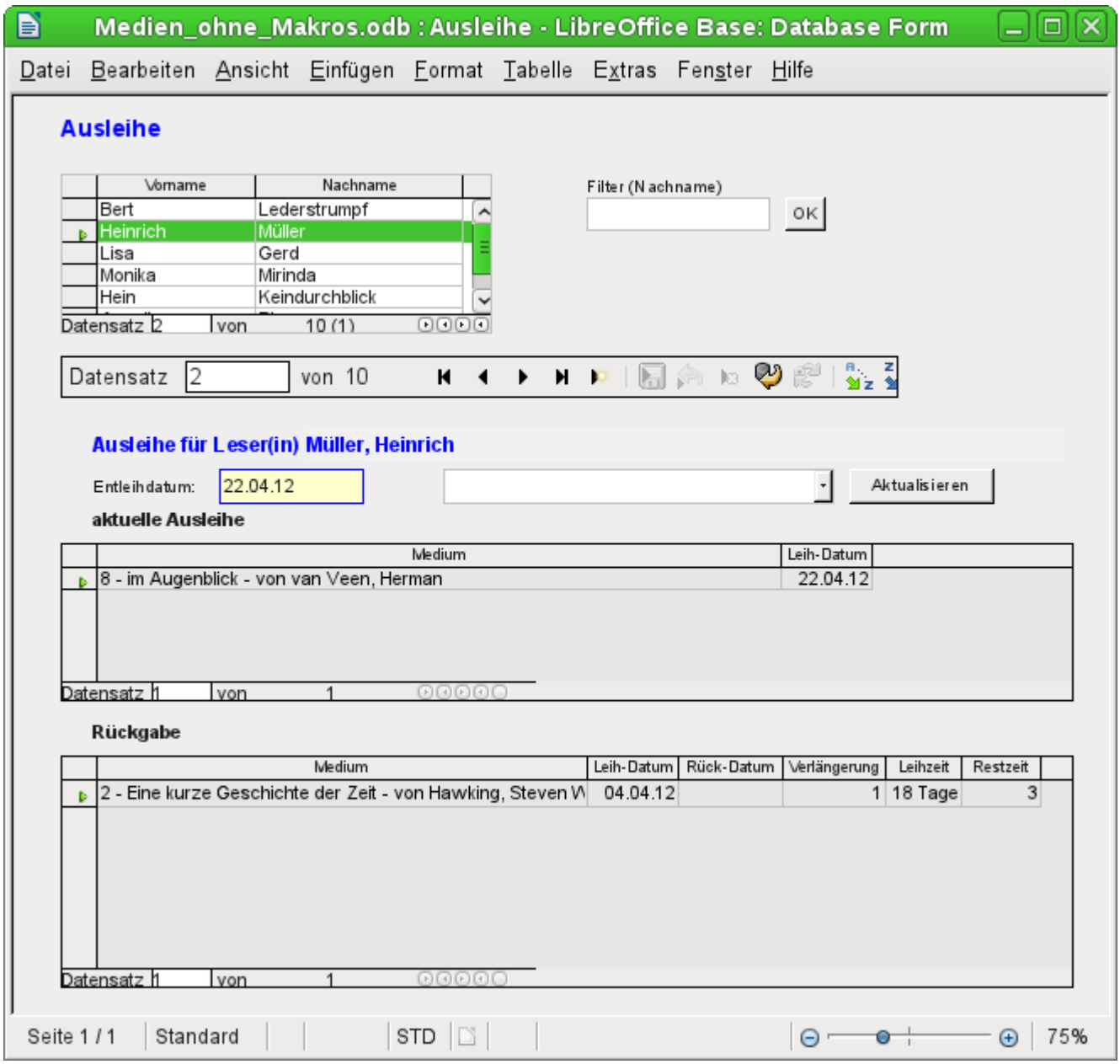

Die Verknüpfung zwischen beiden Formularen weist jetzt zwei Felder auf. Das Feld "Integer" wird mit dem Feld "Leser_ID" des Unter-Unterformulars verbunden. Das Feld "Datum" mit dem Feld "Leih_Datum". Damit wird das "Leih_Datum" automatisch aus der Tabelle "Filter" bei der Ausleihe in die Tabelle "Ausleihe" übertragen.

*Abbildung 17: Das Datum der Ausleihe wird einmal eingestellt. Bei einem Wechsel des Lesers muss es nicht neu eingegeben werden.*

Aus dem Tabellenkontrollfeld wurde jetzt auch noch das Datumsfeld entfernt. Das Tabellenkontrollfeld besteht jetzt lediglich noch aus einem Auswahlfeld. Dies wäre die ideale Voraussetzung um bei einer Mediothek noch an der Beschleunigungsschraube zu drehen. Denn eigentlich haben die Medien ja eine Nummer. Wozu müssen sie also ausgesucht werden. Da könnte doch gleich die Nummer eingetragen werden. Oder, noch besser, die Medien könnten mit Barcode-Etiketten versorgt werden. Ein Scanner dafür ist mit ca. 50.- € mittlerweile recht preisgünstig zu haben. Dann würden die Medien schneller ausgeliehen als der Entleiher sie in die Tasche packen kann.

In der Beispieldatenbank ist dies entsprechend aufgezeigt. Zur Vorstellung des ersten Formularentwurfs sollte das obige Beispiel aber erst einmal ausreichend sein.

Da allerdings das letztlich in der Beispieldatenbank «Medien_ohne_Makros.odb» vorgesehene Formular noch weiter entwickelt wurde, sollen die Erweiterungen hier noch kurz vorgestellt werden.




In die Ausleihe wurden die folgenden Eigenschaften aufgenommen:

- In einem Tabellenkontrollfeld werden die Leser und Leserinnen angezeigt. Hier können auch neue Leser und Leserinnen eingegeben werden.
- Über einen Filter, der mit der Tabelle "Filter" zusammenarbeitet, kann nach den Anfangsbuchstaben des Namens gefiltert werden. So werden bei einem «A» nur die Personen angezeigt, deren Nachname mit «A» beginnt. Die Filterung ist dabei unabhängig von der Eingabe von Groß- und Kleinschreibung.
- Im Untertitel wird noch einmal der Name der Person aufgezeigt, für die die Ausleihe erfolgen soll. Ist die Ausleihe für die Person gesperrt, so wird dies angezeigt.
- Das Entleihdatum ist auf das aktuelle Datum eingestellt. Dazu wurde die Filtertabelle über SQL so eingestellt, dass bei einem nicht eingegebenen Datum der Default-Wert das aktuelle Datum abspeichert.
- Die noch entleihbaren Medien werden in einem Listenfeld ausgewählt. Über den Button 'Aktualisieren' wird die Ausleihe in das darunter stehende Tabellenkontrollfeld übertragen.

- Das mittlere Tabellenkontrollfeld dient lediglich der Anzeige der zu dem angegebenen aktuellen Datum ausgeliehenen Medien. Hier kann auch eine irrtümliche Ausleihe durch Löschen der Zeile rückgängig gemacht werden.
- Im unteren Tabellenkontrollfeld ist wie im vorher gezeigten Beispiel die Änderung von Medien aus Ausleihdatum nicht möglich. Auch eine Löschung ist nicht möglich.
- Neben der Eingabe des Rückgabedatums oder gegebenenfalls einer Verlängerung wird angezeigt, für wie viele Tage das Medium entliehen werden darf und wie viele Tage die restliche Entleihzeit beträgt.
- Geht die Restzeit in den negativen Bereich, so muss das Medium sofort zurückgegeben werden. Die Ausgabe ist deswegen gesperrt. Sie wird dadurch wieder ermöglicht, dass die Medien zurückgegeben werden. Nach der Medienrückgabe muss lediglich einmal auf 'Aktualisieren' gedrückt werden.

Dieses Formular ist mit Hilfe von Abfragen wesentlich komplexer strukturiert als die vorher vorgestellte Variante. Mehr zu den Grundlagen ist deshalb im Kapitel 'Abfragen' zu erfahren.

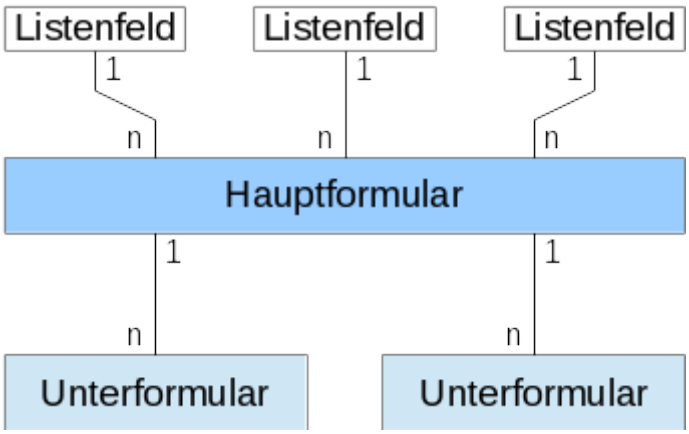
## Eine Ansicht – viele Formulare

Während das Beispiel für die Ausleihe nur Eingaben in eine Tabelle erforderte (Tabelle "Ausleihe") und zusätzlich die Eingabe in der einfacheren Form auch für neue Leser ermöglichte, ist die Eingabe für die Medien wesentlich umfassender. Schließlich spielen rund um die Medien-Tabelle insgesamt 8 zusätzliche Tabellen mit (siehe «Tabellen Medienaufnahme»).

Durch die Zuordnung im Verhältnis n:1 bieten sich die Tabellen "Untertitel" und "rel_Medien_Verfasser" als Unterformulare zum Formular 'Medien' an. Tabellen, die hingegen ein Verhältnis von 1:n zur Tabelle "Medien" haben, müssten eigentlich ein Formular bilden, das über dem der Tabelle Medien liegt. Da es sich aber um mehrere entsprechende Tabellen handelt, werden deren Werte über Listenfelder in das Hauptformular eingetragen.

Die Tabelle eines Hauptformulars steht zur Tabelle eines Unterformulars grundsätzlich im Verhältnis 1:n, in seltenen Ausnahmen im Verhältnis 1:1. Das Hauptformular beherbergt in der Regel nach längerem Gebrauch der Datenbank also eine Tabelle, die deutlich weniger Datensätze hat als die Tabelle des Unterformulars.

Mehrere Hauptformulare können nicht auf ein Unterformular zugreifen. Es ist also nicht möglich, viele 1:n-Beziehungen gleichzeitig über die Formularanordnung zu lösen, bei denen das Unterformular den gleichen Inhalt hat. Gibt es eine 1:n-Beziehung für die Tabelle eines Formulars, so lässt sich dies über ein Listenfeld regeln. Hier stehen wenige Begriffe aus einer anderen Tabelle zur Auswahl, deren Fremdschlüssel auf diese Art in die Tabelle des Hauptformulars eingetragen werden.

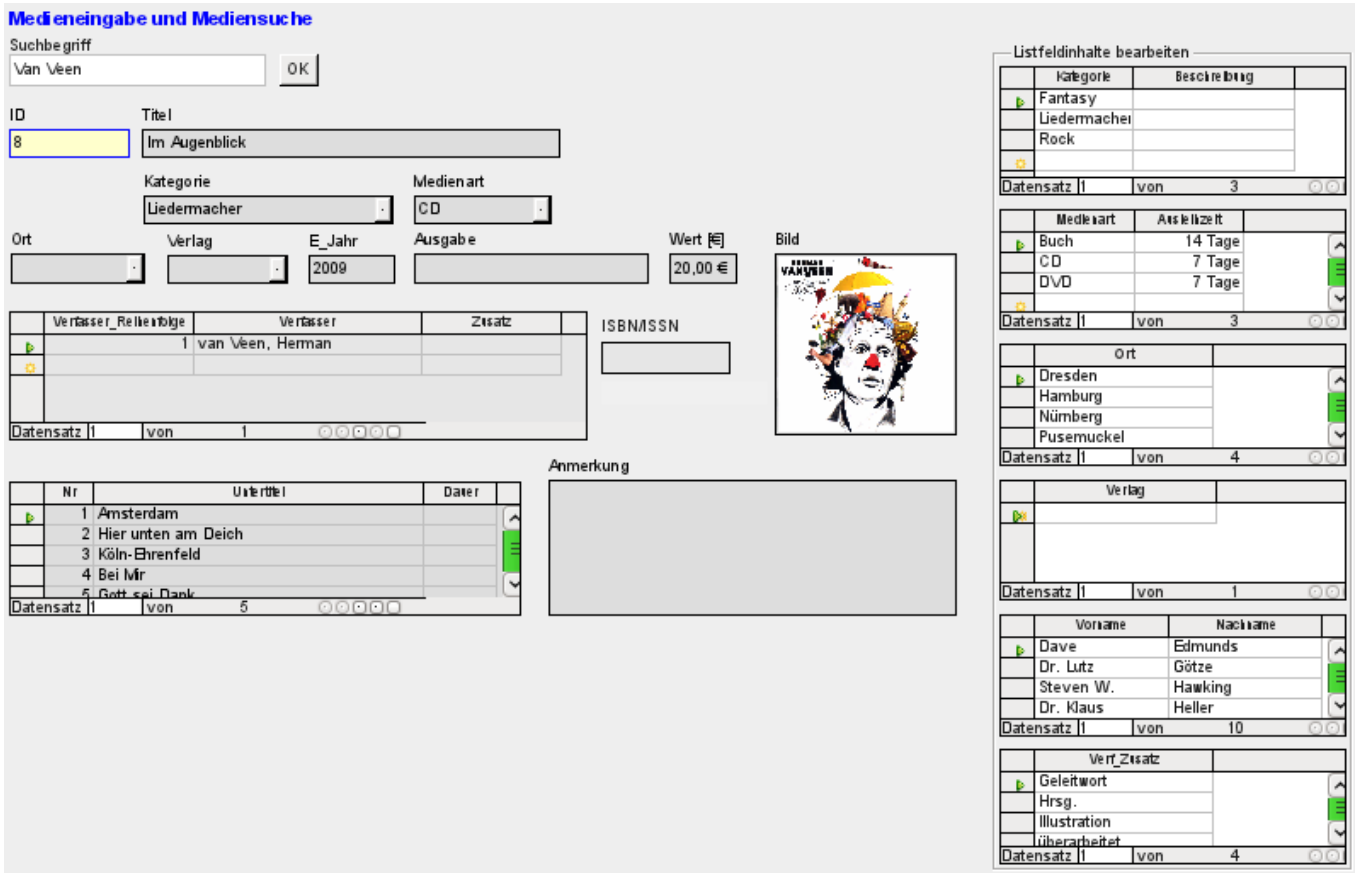


Über Listenfelder werden dem Hauptformular, das auf der Tabelle "Medien" basiert, z. B. die Inhalte der Tabellen "Kategorie", "Ort" oder "Verlag" zugewiesen. Über Unterformulare sind die Tabelle "rel_Medien_Verfasser" und "Untertitel" mit dem Hauptformular und damit mit der Tabelle "Medien" verbunden.

Das Unterformular für die Tabelle "rel_Medien_Verfasser" besteht außerdem wieder aus zwei Listenfeldern, damit nicht die Fremdschlüssel der Tabelle "Verfasser" und "Verf_Zusatz" (Zusätze wie Hrsg., Gesang usw.) direkt als Ziffern eingegeben werden müssen.

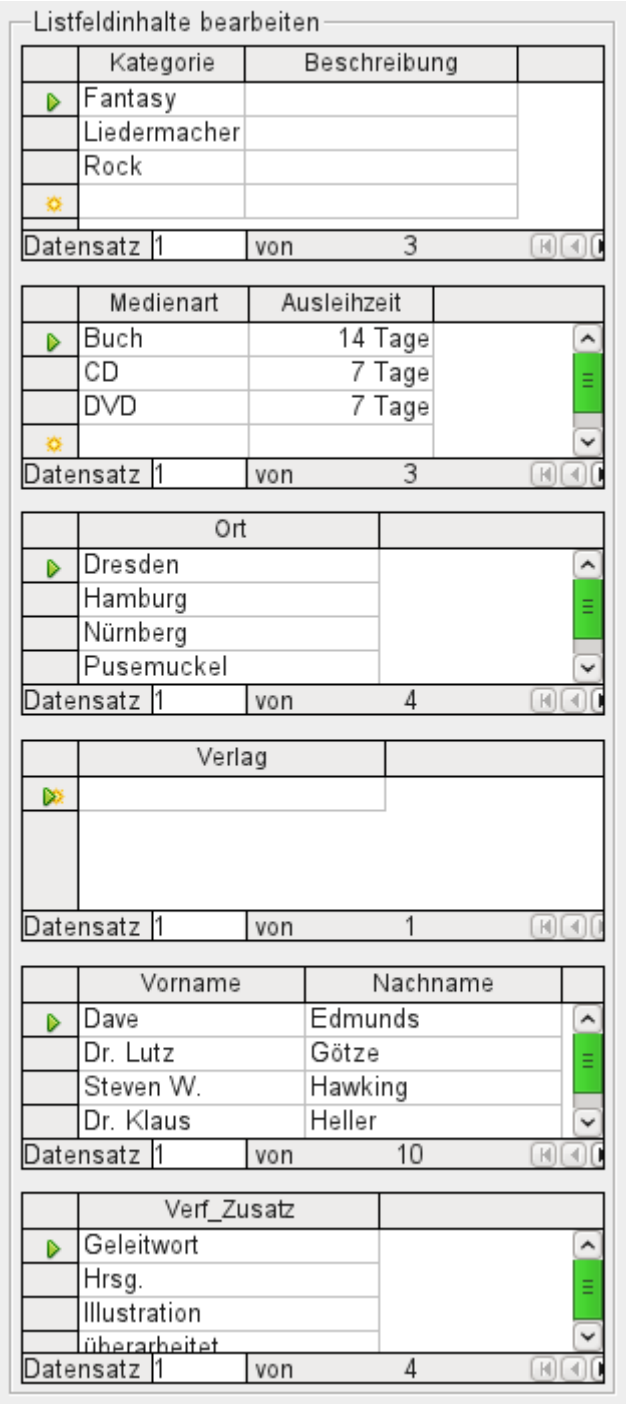
Bei dem Formular für die Medieneingabe müssen die Listenfelder meist während der Eingabe nach und nach aufgefüllt werden. Hierzu werden neben dem Hauptformular weitere Formulare eingebaut. Sie existieren unabhängig vom Hauptformular:

Das gesamte Formular zur Medieneingabe sieht so aus:




Auf der linken Seite befindet sich das Hauptformular mit Blick auf die Suche und Eingabe von neuen Medien. Auf der rechten Seite des Formulars ist durch einen Gruppierungsrahmen mit der Bezeichnung «Listfeldinhalte bearbeiten» eine Bereich abgegrenzt, der zum Auffüllen der Listenfelder (hier verkürzt: «Listfeld») im Hauptformular gedacht ist. Existiert die Datenbank erst kurz, so wird hier wohl häufig eine Eingabe erledigt werden müssen. Je mehr Eingaben allerdings in den Listenfeldern des Hauptformulars zur Verfügung stehen, desto seltener ist ein Zugriff auf die Tabellenkontrollfelder aus dem Gruppierungsrahmen notwendig.

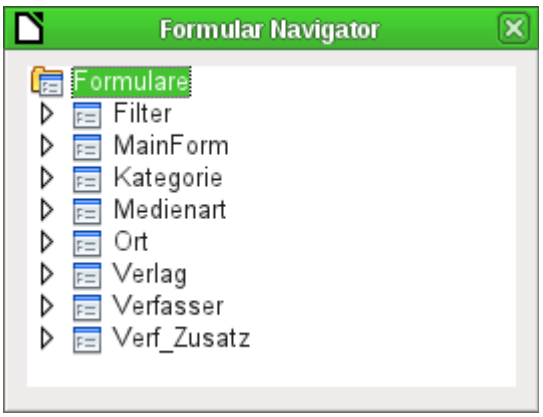
Die folgenden Tabellenkontrollfelder sind alle in einzelnen Nebenformularen zum Hautformular, dem Eingabeformular, untergebracht:




Hier werden jeweils die kompletten Daten für eine Tabelle eingegeben. Am Anfang ist es häufig erforderlich, auf diese Nebenformulare auszuweichen, da z. B. nicht viele Verfasser in der entsprechenden Tabelle bereits abgespeichert wurden.

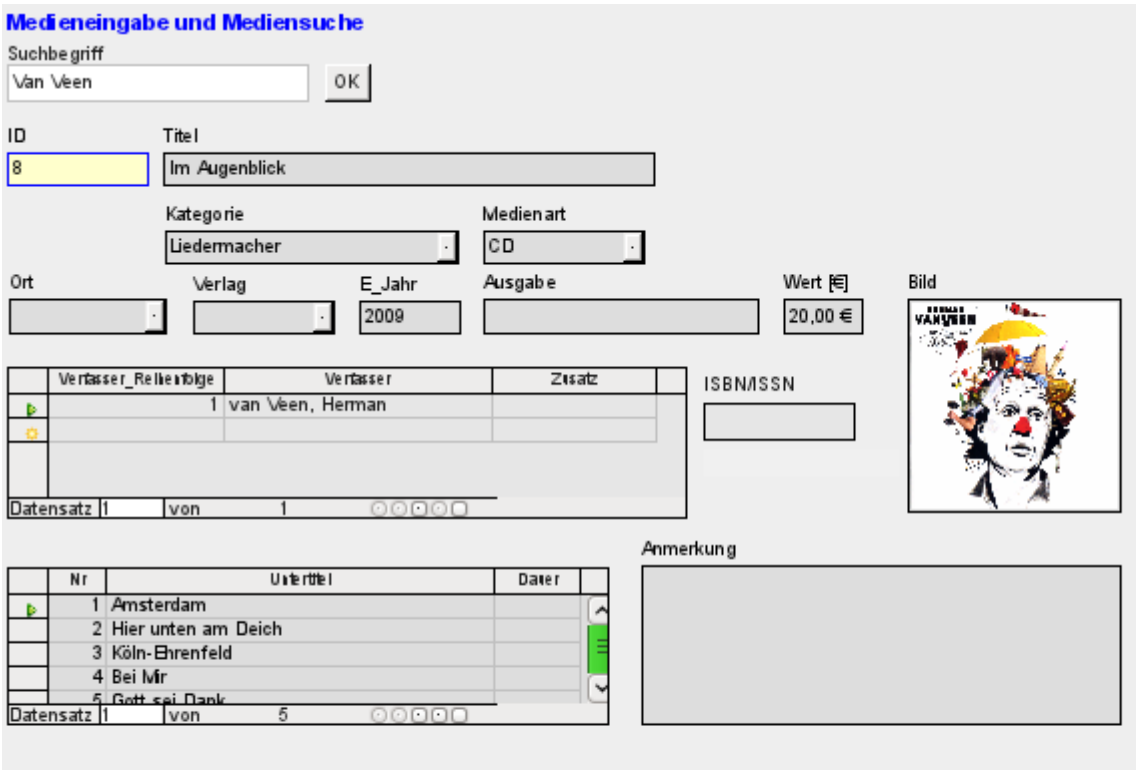
Wurde in einem der Tabellenkontrollfelder ein neuer Datensatz abgespeichert, so ist in dem Hauptformular das entsprechende Listenfeld aufzusuchen und über **Kontrollfeld aktualisieren** (siehe «Navigationsleiste») neu einzulesen.

Der Formularnavigator zeigt entsprechend viele Formulare an:



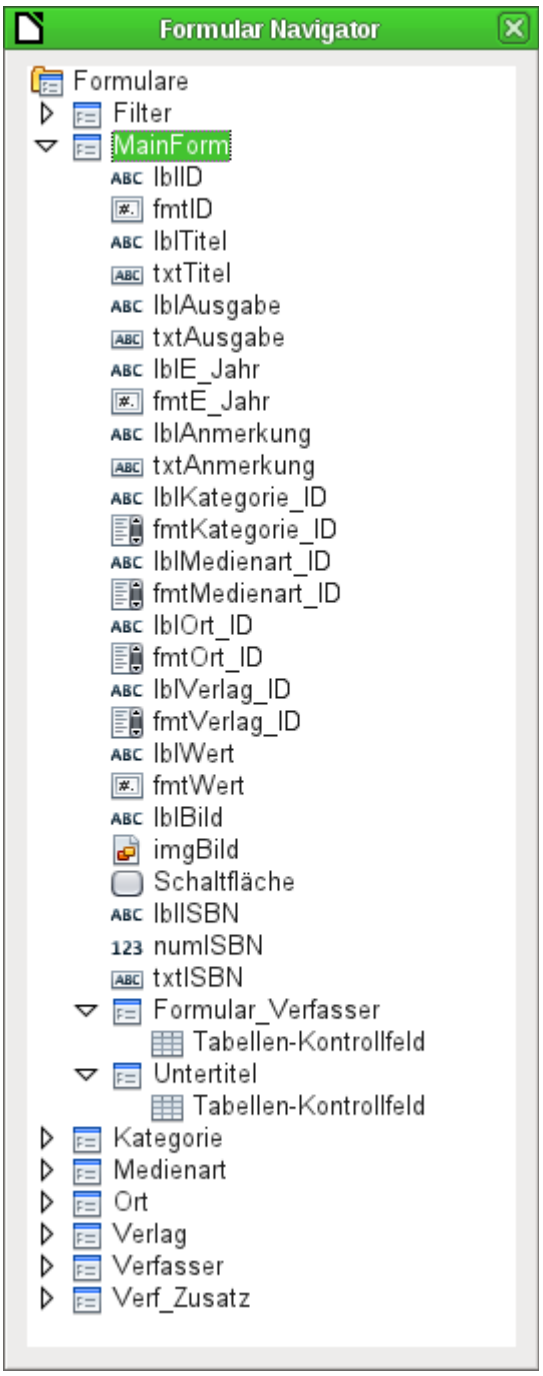

Die Formulare sind einzeln benannt, so dass sie erkennbar bleiben. Lediglich das Hauptformular hat vom Formularassistenten noch die Bezeichnung «MainForm» behalten. Insgesamt existieren also 8 Formulare parallel. Das Formular «Filter» beherbergt eine Suchfunktion, das Formular «MainForm» die Haupteingabefläche. Alle anderen Formulare stehen für je eins der oben abgebildeten Tabellenkontrollfelder.

Ohne die Tabellenkontrollfelder erscheint das Hauptformular schon etwas übersichtlicher:




Das Feld für den Suchbegriff liegt in dem *Nebenformular* «Filter», die beiden *Tabellenkontrollfelder* (für die Verfasser und für die Untertitel) liegen in *Unterformularen* zum Hauptformular der Medieneingabe.

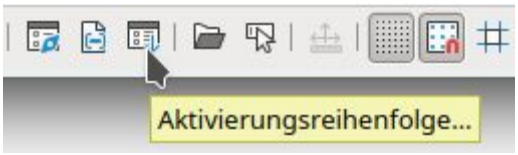
Im Formularnavigator sieht das Formular selbst dann schon wesentlich unübersichtlicher aus, da natürlich alle Kontrollfelder, auch die Beschriftungen, dort auftauchen. In den vorhergehenden Formularen waren ja die meisten Felder als Spalten von Tabellen-Kontrollfeldern im Formularnavigator nicht zu sehen.




Die Reihenfolge innerhalb des Formularnavigators lässt sich leider nicht so einfach verändern. So erscheint es z. B. sinnvoll, die Unterformulare «Untertitel» und «Formular_Verfasser» als Verzweigung direkt zum Beginn des Formulars «MainForm» auftauchen zu lassen. Innerhalb des Formularnavigators werden die einzelnen Kontrollfelder und Unterformulare einfach in der Reihenfolge aufgelistet, in der sie gegründet wurden.

Durch den Formularassistenten werden die Elemente mit bestimmten Kürzeln versehen, die neben den Symbolen andeuten, um welche Art Feld es sich handelt. Mit 'lbl' beginnen Beschriftungsfelder ('label'), mit 'txt' Textfelder usw. Sämtliche Beschriftungsfelder interessieren für die Eingabe von Daten eigentlich nur als zusätzliche Informationen. Sie können auch direkt über Textrahmen erstellt werden und würden dann im Formularnavigator nicht erscheinen.

Die Reihenfolge, in der die Elemente im Navigator auftauchen, hat aber nichts damit zu tun, in welcher Reihenfolge die Elemente durch einen Tabulatorsprung erreicht werden. Dies wird durch die *Aktivierungsreihenfolge* bestimmt.




Die Aktivierungsreihenfolge für ein bestimmtes Formular wird aufgerufen, indem ein Element dieses Formulars markiert wird und dann erst der Button Aktivierungsreihenfolge betätigt wird. Bei einem einzigen Formular ist diese Reihenfolge natürlich nicht notwendig. Die Funktion muss aber bei vielen parallel liegenden Formularen erst einmal wissen, welches Formular denn nun ausgewählt werden soll. Standardmäßig ist es sonst das erste Formular im Formularnavigator – und das enthält im obigen Beispiel nur ein Textfeld.

Über die Aktivierungsreihenfolge werden alle Elemente, die Daten an die dem Formular zugrundeliegende Tabelle weitergeben oder Aktionen hervorrufen können, in ihrer Reihenfolge zueinander festgelegt. Dies entspricht der Einstellung in den Eigenschaften in *Standardeinstellungen vieler Kontrollfelder* zur Aktivierungsreihenfolge.

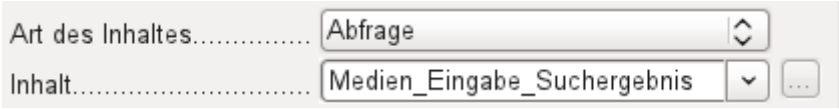
In der Aktivierungsreihenfolge tauchen allerdings auch die Elemente auf, bei denen der Tabstop eigentlich abgeschaltet ist. Sie werden zwar in die Nummerierung aufgenommen, aber tatsächlich bei der Arbeit mit dem Formular über die Tastatur nicht angesprungen.

Automatisch werden die Felder nach der Lage auf dem Formularhintergrund sortiert. Je weiter oben ein Feld liegt, desto eher wird es über die Auto Sortierung angesprungen. Je weiter links ein Feld bei gleicher Höhe liegt, desto eher wird es angesprungen. Diese Sortierung funktioniert dann tadellos, wenn die Elemente wirklich genau ausgerichtet sind (Raster bei der Formularerstellung). Ansonsten muss nachgebessert werden. Hierzu wird ein Element einfach markiert und z. B. über Nach unten weiter nach unten in der Reihenfolge verlegt.

Existiert ein Unterformular, so wird bei der Auto Sortierung das Ganze so eingestellt, dass nach dem Hauptformular direkt in das Unterformular gesprungen wird. Bei einem Tabellenkontrollfeld führt dies über die Tastatureingabe dann dazu, dass der Cursor in diesem Unterformular gefangen ist und nur mit der Maus oder über **`Strg + Tab`** aus dem Subformular wieder herausbewegt werden kann.

| Hinweis | Der Sprung ins Unterformular findet nur dann statt, wenn das Unterformular auch im Formularnavigator unterhalb der Felder des Hauptformulars liegt. Eine Sortierung im Formularnavigator ist leider nicht möglich, so dass gegebenenfalls ein Unterformular nach Erstellung des kompletten Hauptformulars neu erstellt werden muss. Dann können die Elemente des ursprünglichen Unterformulars dorthin verschoben und das ursprüngliche Unterformular gelöscht werden. Jetzt klappt auch der Sprung in das Unterformular, nachdem die Auto Sortierung erneut aufgerufen wurde.<br>Manchmal muss auch bei der Auto Sortierung getrickst werden. Sie funktioniert nur, wenn in einem Formular wenigstens 2 Felder sind, die überhaupt sortiert werden können. Bei lediglich einem Feld im Hauptformular muss dann vorübergehend ein zweites Feld aus der Datenquelle hinzugefügt werde. Wird dann die Auto Sortierung aufgerufen, so klappt der Sprung ins Unterformular auch, wenn das überflüssige Feld anschließend wieder entfernt wird. |
|---|---|

Die Auto Sortierung funktioniert allerdings, bezogen auf das Tabellenkontrollfeld, nur einmal. Ein weiteres Unterformular mit Tabellenkontrollfeld wird nicht mit einbezogen. Parallel liegende Formulare werden also nicht berücksichtigt. Eine Auto-Sortierung kann in Bezug auf ein Unterformular mit Tabellenkontrollfeld auch nicht rückgängig gemacht werden. Hierzu müsste das Unterformular komplett entfernt (oder vorübergehend in ein anderes Formular verschoben) werden.




Die Datengrundlage für das Formular zur Medieneingabe ist in diesem Fall nicht eine Tabelle, sondern eine Abfrage. Dies ist notwendig, da das Formular nicht nur zur Eingabe sondern auch zur Suche benutzt werden soll. Auch wird in dem Formular noch in einem Textfeld nach dem Abspeichern darüber aufgeklärt, ob die eingegebene ISBN-Nummer korrekt ist. Auch dies ist nur über eine umfassende Abfrage möglich. Um diese Hintergründe zu verstehen, ist es also notwendig, sich erst einmal mit den Grundlagen von Abfragen auseinander zu setzen.

## Fehlermeldungen bei der Eingabe in Formulare

Einige Fehlermeldungen treten gerade bei den ersten Formularerstellungen gehäuft auf, so dass sie hier kurz erklärt werden sollen:

**Attempt to insert null into a non-nullable column: column: ID table: Tabellenname in statement …**

Es gibt Felder, die nicht leer sein dürfen. Wird dies nur in der Tabelle definiert (**`NOT NULL`**), so erscheint eine englischsprachige Fehlermeldung. Erfolgt die Definition auch in den Kontrollfeldern des Formulars, dann erscheint eine deutschsprachige Meldung mit dem genauen Hinweis, welches Feld denn nun ausgefüllt werden muss.

Die obige Meldung passiert besonders häufig, wenn das Primärschlüsselfeld, hier **`'ID'`**, nicht in das Formular mit aufgenommen wurde, da es ja als automatisch hoch zählendes AutoWert-Feld gedacht war. Leider wurde aber die entsprechende Definition als AutoWert-Feld schlicht vergessen. Also muss dies entweder nachgeholt werden oder das Feld muss im Formular mit erscheinen, damit ein Wert eingegeben werden kann.

Die nachträgliche Definition eines Feldes als AutoWert-Feld ist manchmal etwas schwierig, wenn die Tabelle mit anderen in der Beziehungsdefinition bereits verknüpft wurde oder wenn bereits mit einer Ansicht auf die Tabelle zugegriffen wird. Hier müssen alle Verbindungen gelöst werden, damit die Tabelle gerade im Bereich des Primärschlüssels wieder editierbar wird. Der Inhalt des SQL-Befehls, mit dem z. B. eine Ansicht erstellt wurde, kann ja gegebenenfalls als Abfrage zwischendurch abgespeichert werden.

**Integrity constraint violation - no parent SYS_FK_95 table: Tabellenname in statement …**

Hier besteht eine Verknüpfung der dem Formular zugrundeliegenden Tabelle mit einer anderen Tabelle. Diese Tabelle soll ein Fremdschlüsselfeld mit seinem Primärschlüssel belegen. Dies geschieht in der Regel durch Listenfelder, die den Inhalt der Tabelle richtig abfragen. Ist aber gar kein Listenfeld vorhanden oder die Konstruktion des Listenfeldes falsch, so kann in das Fremdschlüsselfeld auch irrtümlich ein Wert eingegeben werden, der in der zweiten Tabelle gar nicht als Primärschlüssel vorhanden ist. Dies wird als **'Integritäts-Verletzung'** bezeichnet. **`'no parent SYS_FK_95'`** deutet darauf hin, dass in der zweiten Tabelle, der abgebenden **'Elterntabelle'**, der entsprechende Indexwert mit dem Namen **`'SYS_FK_95'`** nicht vorhanden ist.

**Fehler beim Einfügen des neuen Datensatzes.**
**Fehler in der Funktionsfolge**

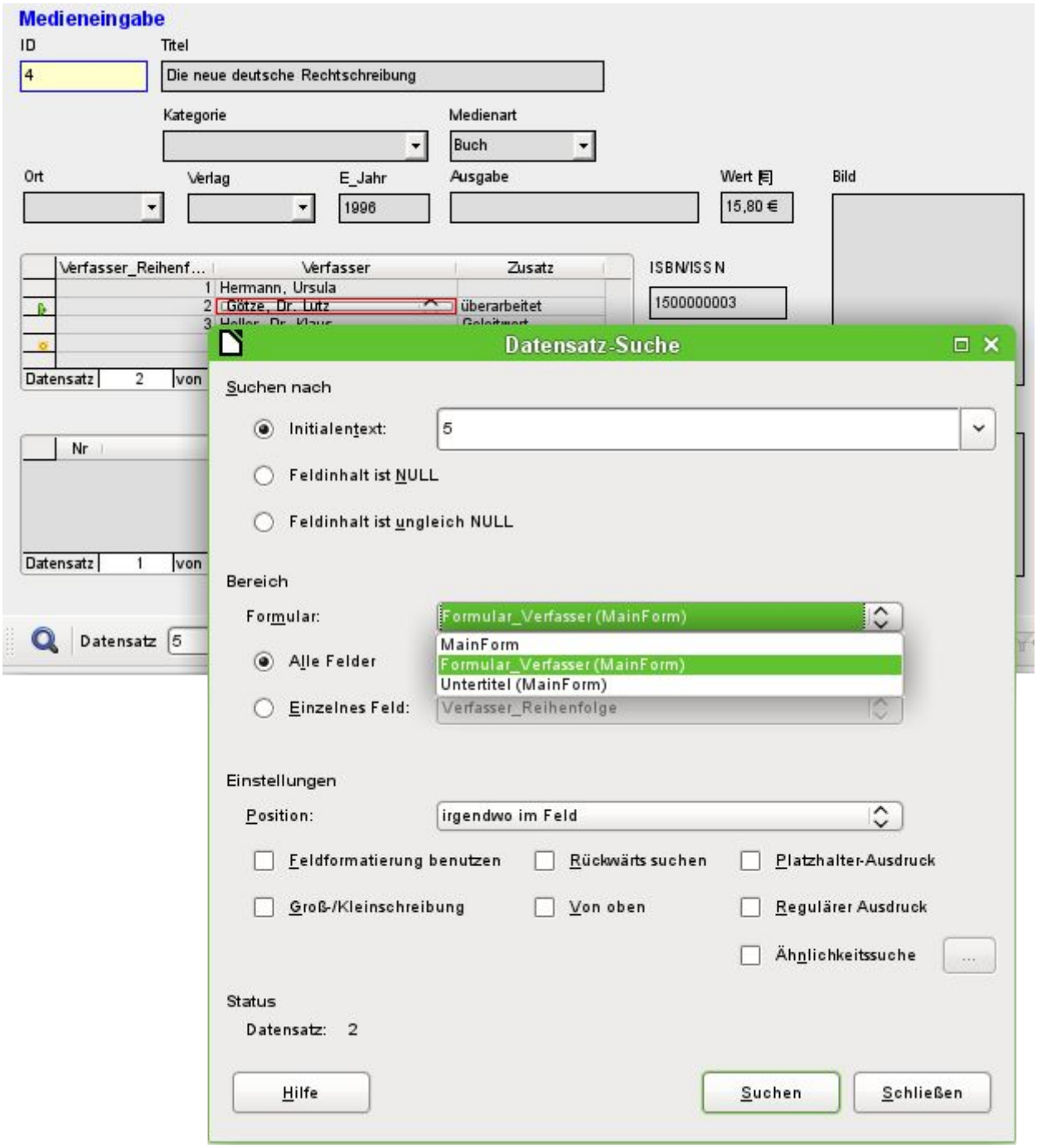
Einem Unterformular werden Daten von den darüber liegenden Formularen weitergegeben. Dies sieht Base nicht als eine Änderung eines Feldes an. Die grafische Benutzeroberfläche bietet zwar das Speichern an – nur scheint der Datensatz dann leer zu sein. Hier hilft es, ein einfaches Feld wie z. B. ein Ja/Nein-Feld in die dem Formular zugrundeliegende Tabelle einzubauen. Jetzt wird vor dem Abspeichern dieses Feld abgehakt und der Datensatz kann problemlos gespeichert werden.

Die über die anderen Formulare weitergegebenen Werte werden anscheinend erst dann in die entsprechenden Felder eingefügt, wenn wenigstens eine zusätzliche Aktion im Formular stattfindet.

## Suchen und Filtern in Formularen über die Navigationsleiste

Die Navigationsleiste der Formulare bietet verschiedene Such- und Filtermöglichkeiten an. Die einzelnen Elemente der Navigationsleiste wurden bereits weiter oben aufgezählt.

## Datensatzsuche mit Parametern




Die einfache Datensatzsuche mit Parametern wurde bereits im Kapitel «Tabellen» vorgestellt. Sie bietet sehr umfangreiche Einstellmöglichkeiten. Charakteristik der Suche ist, dass grundsätzlich alle Datensätze durchsucht werden und nicht die Datensätze so eingeschränkt werden, dass nur Datensätze mit dem Suchwert zu sehen sind.

Die Datensatzsuche im Formular bietet die Möglichkeit, das entsprechende Formular sowie die Unterformulare zu durchsuchen. Es ist allerdings nicht möglich, alle Formulare gleichzeitig zu durchsuchen.

Das Bild zeigt den Zugriff auf das Unterformular «Formular_Verfasser» des Hauptformulars «MainForm». Dabei wird nach einem Listenfeld, nämlich dem Feld für den Verfasser, gesucht.

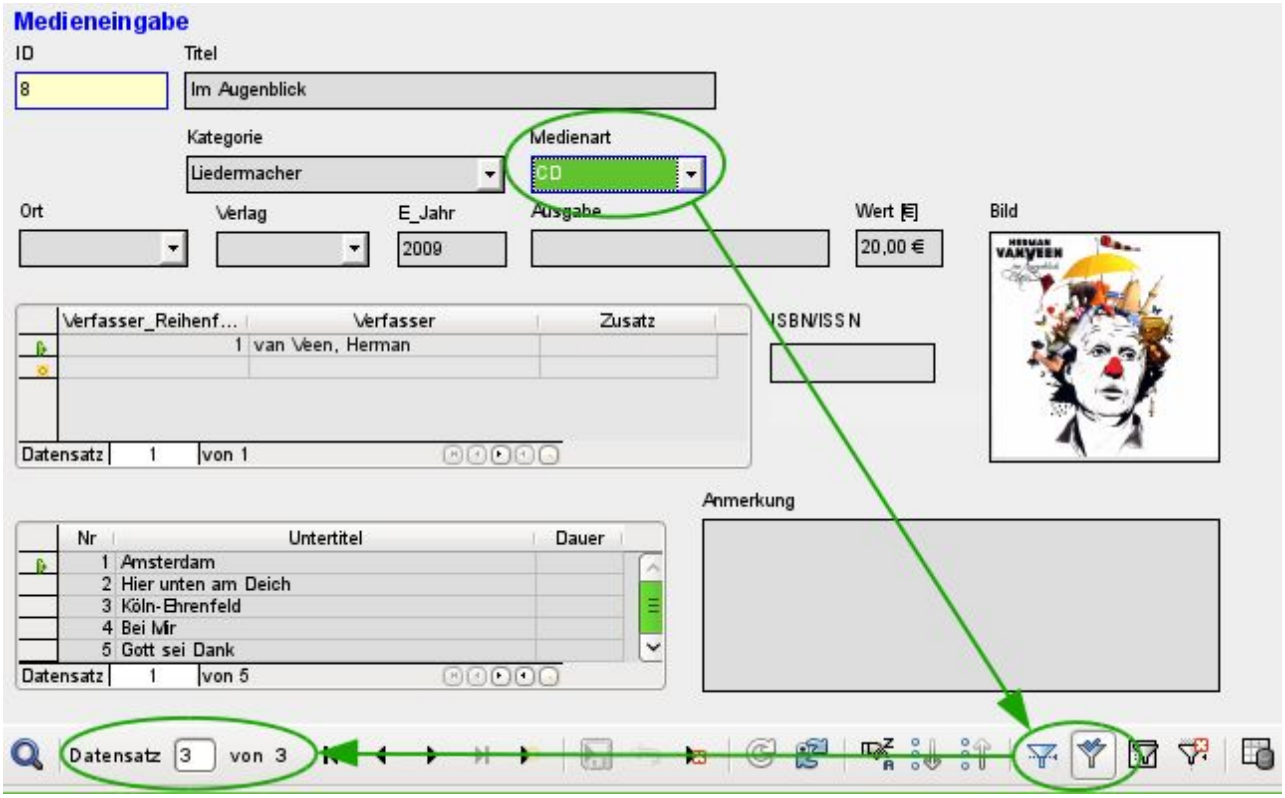
Allerdings wird nicht der Text des Listenfeldes benötigt, sondern der Wert, der als Fremdschlüsselwert an dieser Stelle eingegeben wird.

Leider kennzeichnen diese Suche aber die folgenden Nachteile:

- Die Parametersuche ist für den Normalgebrauch viel zu umständlich zu handhaben.
- Die Suchfunktion selbst ist äußerst langsam, da nicht die Abfragefunktionen der Datenbank genutzt werden.
- Die Suche funktioniert nur innerhalb eines Formulars, dort dann aber auch gegebenenfalls für alle Felder. Unterformulare müssen separat angewählt werden.
- Es wird nur in dem Unterformular gesucht, das zu dem aktuellen Hauptformular passt. Einträge anderer Unterformulare lassen sich nicht ermitteln. So ist es z.B. nicht möglich, einen Untertitel eines anderen Mediums als des gerade angezeigten Mediums zu finden.
- Die Suche funktioniert bei Listenfeldern auf der Basis des Schlüsselfeldes (Fremdschlüssels), das in der Tabelle abgespeichert ist. Es lassen sich nicht die angezeigten Begriffe verwenden.

## Filterung mit dem Autofilter

Der Autofilter kann direkt in der Navigationsleiste angewählt werden. Wird auf ein Feld im Hauptformular geklickt, so wird treffsicher nach diesem Feld gefiltert. Der Autofilter funktioniert auch sicher mit Listenfeldern, hat hier also einen klaren Vorteil gegenüber der händischen Eingabe von Fremdschlüsselwerten zu verbuchen.




In dem obigen Beispiel wurde einfach ein Datensatz gesucht, bei dem im Feld «Medienart» 'CD' steht. Anschließend wurde der Autofilter betätigt. Von den ursprünglich 9 Datensätzen der Medien-Tabelle werden jetzt noch 3 Datensätze in der Datensatzzählung angezeigt.

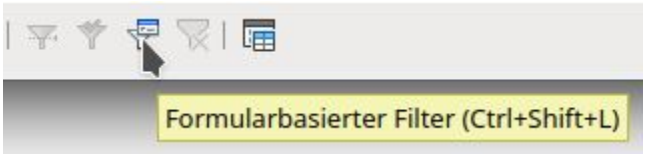
Obwohl nur in zwei Datensätzen im Unterformular eine «Nr.» '3' wie oben existiert, werden alle anderen Datensätze des Hauptformulars weiter angezeigt. Im Unterformular erscheinen hingegen nur die Felder mit «Nr.» '3'.

Der Autofilter kann die folgenden Probleme allerdings nicht lösen:

- Die Suche in Unterformularen führt dazu, dass zwar in den Unterformularen nur der entsprechende Wert angezeigt wird, die Formulare selbst aber weiter komplett angezeigt werden, auch wenn sie den entsprechenden Wert im Unterformular gar nicht verzeichnen können. Wird z.B. nach einem Medium mit einem zweiten Verfasser gesucht, so werden trotzdem auch alle Medien angezeigt, für die nur ein Verfasser verzeichnet ist. Hier bleibt das Feld für die Verfasser schlicht leer.
- Die Wahl des Filterinhaltes muss eindeutig sein. Es gibt nicht die Möglichkeit, über Ähnlichkeitssuche zu einem entsprechenden Wert zu kommen, da eben nur der Originalwert aus dem Formular verwandt wird.

## Filterung mit dem formularbasierten Filter

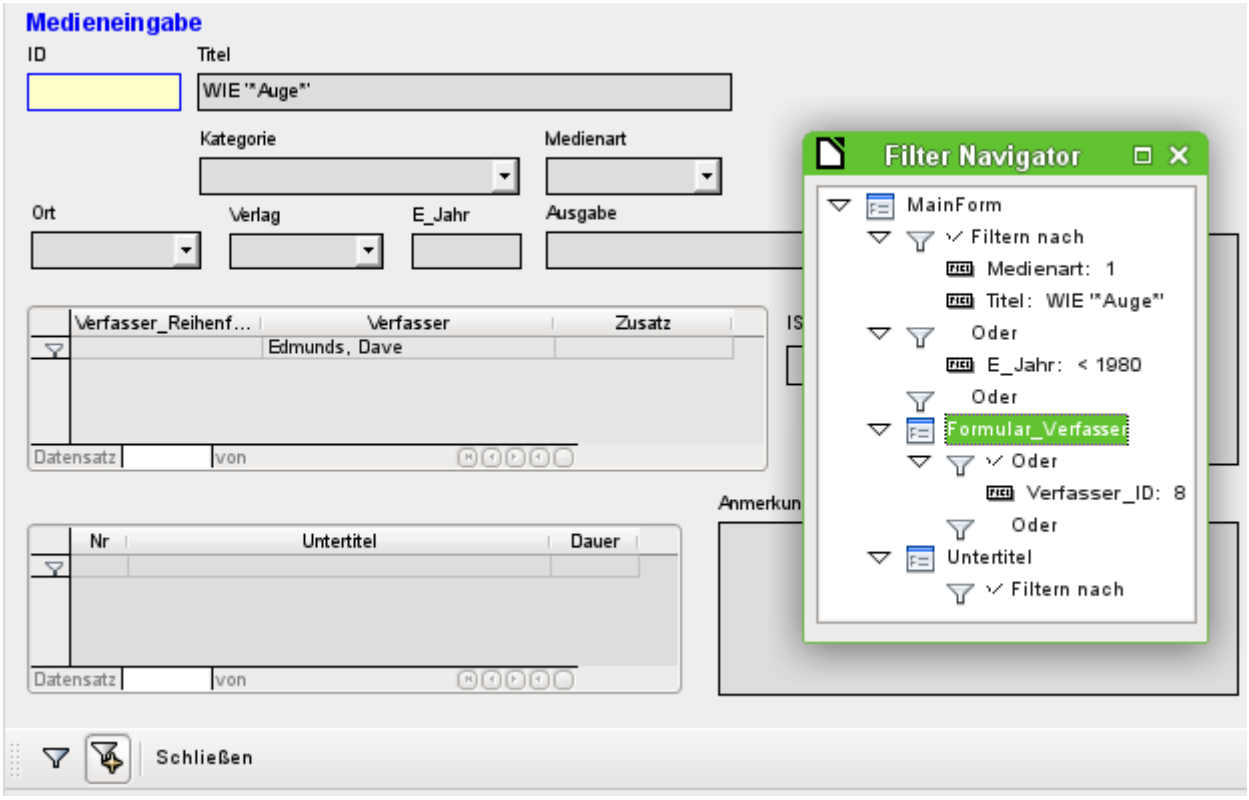
Der formularbasierte Filter bietet statt eines Filtersymbols das Formular zur Eingabe von entsprechenden Filterwerten an. Hier kann gleichzeitig nach mehreren Werten gefiltert werden. Diese Werte können entweder zusammen als gemeinsame Bedingung formuliert werden (UND) oder als unterschiedliche Bedingungen gesetzt werden (ODER).




*Start formularbasierter Filter*

Die Eingabe der Filterwerte muss nicht eindeutig sein. Hier hilft dann die Formulierung von Bedingungen, wie sie bei der Filterung von Tabellen geschildert wurde. So führt ein «WIE '%Auge%'», eingetragen in das Eingabefeld für den Titel des Mediums, direkt zur Auffindung aller Titel, in denen 'Auge' enthalten ist – in der Beispieldatenbank «Medien ohne Makros» z.B. zur Anzeige des Titels «Im Augenblick».

Filter für die Listenfelder werden direkt über die Listenfelder ausgewählt. Eine Eingabe von Fremdschlüsselwerten ist also nicht nötig.



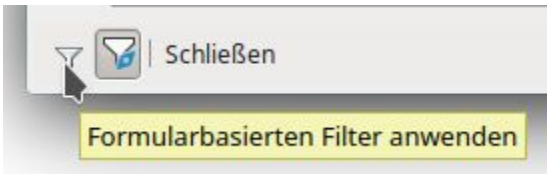

Hier werden alle Datensätze gesucht, die die Medienart 'CD' haben (entspricht dem Fremdschlüsselwert '1') und gleichzeitig im Titel irgendwo die Buchstabenfolge 'Auge' haben. Bei der Eingabe «WIE '%Auge%'» ersetzt die grafische Benutzeroberfläche das in SQL übliche Allroundzeichen für mehrere Werte «%» durch das Zeichen «*», das für viele Nutzer eher gebräuchliche Allroundzeichen.

Neben der Bedingung dürfen auch alle Datensätze des Hauptformulars angezeigt werden, bei denen die Jahresangabe «< 1980» ist.

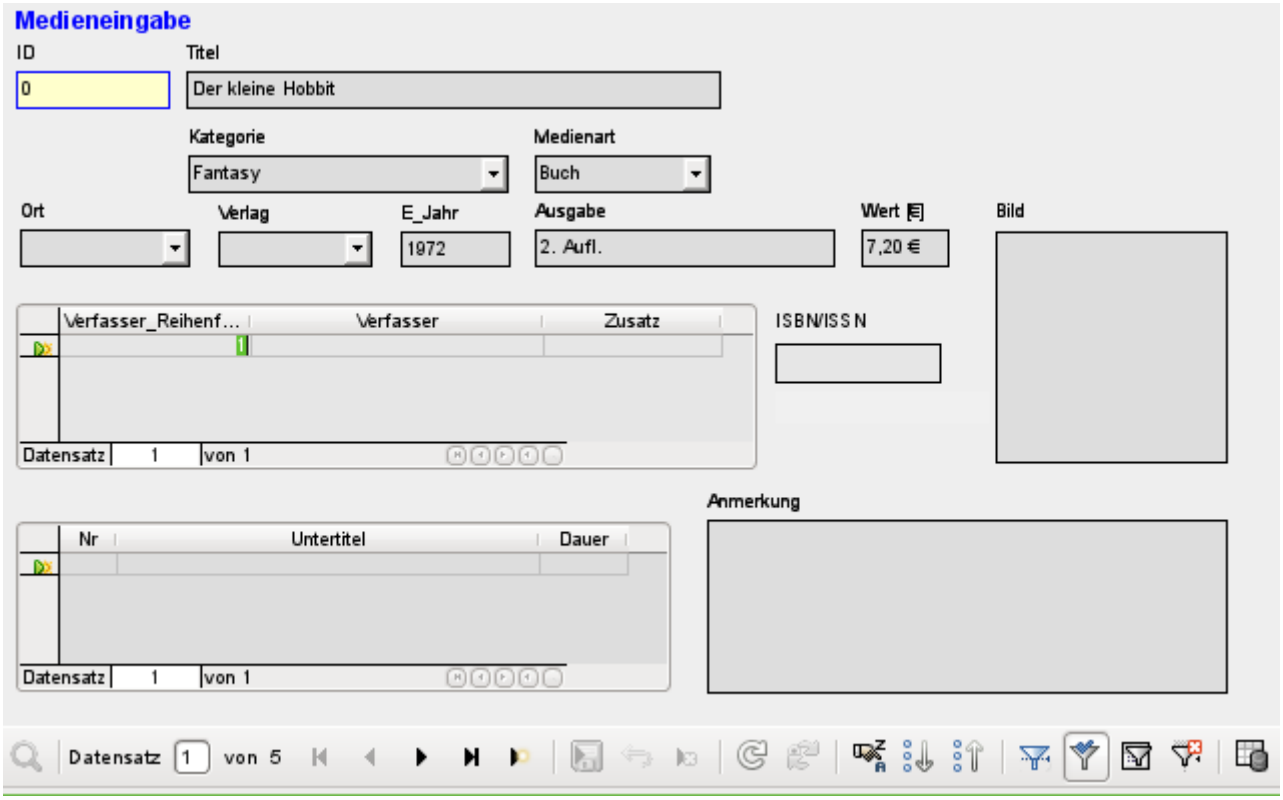
Aus dem Unterformular wird noch der Verfasser 'Edmunds, Dave' aus dem Listenfeld ausgewählt. Der entsprechende Fremdschlüsselwert ist hier die '8'.

Der formularbasierte Filter zeigt bei der Eingabe nur die Felder an, die vor dem ODER mit einem Haken versehen sind. So lassen sich auch mehrere Bedingungen für ein Feld angeben. Leider unterschlägt der Filter dabei die Anzeige des Listenfeldes «Medienart» mit dem entsprechenden Wert im Hauptformular, während dies im Tabellenkontrollfeld des Unterformulars gelingt.

Statt der Navigationsleiste werden während des Erstellens des Filters unter dem Formular lediglich drei Auswahlmöglichkeiten für den formularbasierten Filter dargestellt: Filtern nach den Vorgaben, Filterwerte hinzufügen und die Anzeige des formularbasierten Filters einfach schließen.




Der Filter wird angewandt und das Formular nach den vorgegebenen Werten gefiltert.




Ist die Filterung aktiv, so wird die Zahl der Datensätze auf die Datensätze mit einem Treffer eingeschränkt. Hier gibt es insgesamt 5 Datensätze mit Treffern. In dem obigen Bild stimmt zwar nicht der Titel und die Medienart, dafür aber die Bedingung, dass das Erscheinungsjahr vor 1980 liegen soll.

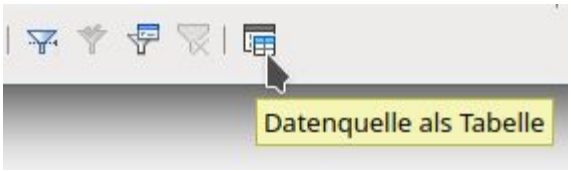
Eine Merkwürdigkeit zeigt sich im Unterformular. Da auch hier eine Filterung vorgenommen wurde, wird der Verfasser des Buches 'Der kleine Hobbit' nicht angezeigt. Der Filterwert war mit dem Verfasser 'Dave Edmunds' verbunden. Dies zeigt wieder die Logik, die hinter der Filterung in allen Fällen steckt: Es werden die Filtereigenschaften des jeweiligen Formulars genutzt. Unterformulare werden separat gefiltert und eine Filterung der Unterformulare wirkt sich nicht auf das Hauptformular aus – und umgekehrt.

Diese schon sehr gelungene Filterfunktion leidet allerdings an den folgenden Nachteilen:

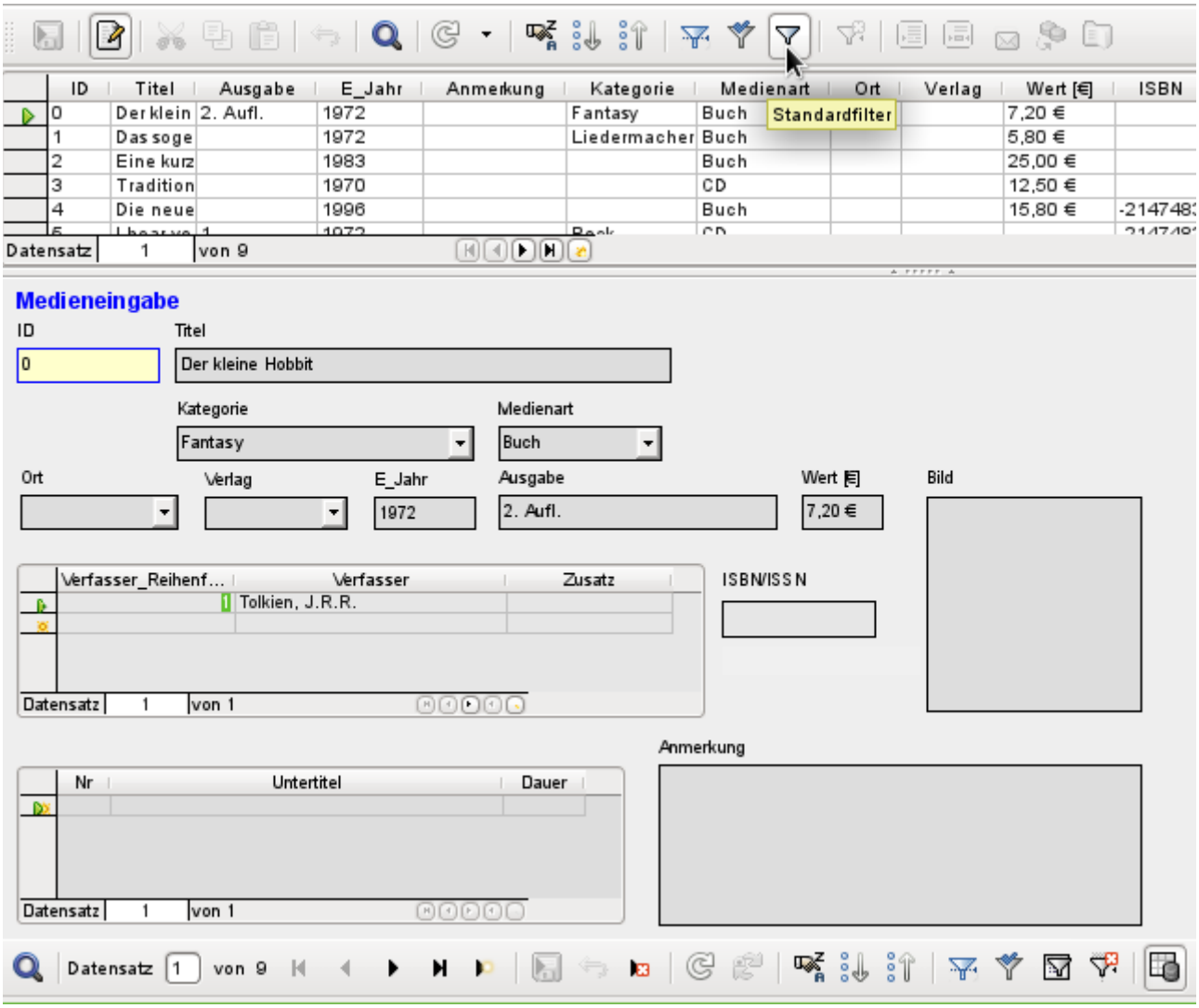
- Die Filterung funktioniert wieder nur im Hauptformular. Im Unterformular werden wie bei Autofilter zwar die fehlenden Werte unterdrückt. Dies wirkt sich aber nicht auf die Anzeige im Hauptformular aus. Es werden also im Hauptformular auch Inhalte angezeigt, die den gefilterten Inhalt aus dem Unterformular nicht vorweisen können.
- Es sind umfangreiche Kenntnisse der Bedingungsformulierung notwendig, wenn mehr als nur die kompletten Begriffe gesucht werden sollen. Hier müssen Kenntnisse über Abfragetechniken vorhanden sein, die aber den meisten suchmaschinenverwöhnten Nutzern fehlen.

## Filterung über den Standardfilter

Die Filterung über den Standardfilter kann durch die Anzeige der Datenquelle als Tabelle erreicht werden. Dort gelten dann die gleichen Regeln wie bei der Filterung mit dem Standardfilter in Tabellen.




*Start des Standardfilters über «Datenquelle als Tabelle»*




Rechts unten im Formularnavigator wird die Datenquelle als Tabelle eingeblendet. In der Tabellensicht steht dann auch der Standardfilter zur Verfügung. Im Gegensatz zur Ansicht der eigentlichen Tabelle "Medien" werden hier die Fremdschlüsselfelder nicht mit ihren Schlüsselwerten, sondern mit den Inhalten dargestellt. Ein Klick auf das Feld «Medienart» zeigt, dass hier ein Listenfeld nach den Vorgaben des Formulars erstellt wurde.

| **Hinweis** | Die obige Ansicht zeigt dann leider auch gleich einen Bug, der seit den Anfängen von Base wohl schon existiert: Die 13-stellige ISBN-Nummer wird nicht korrekt dargestellt. Das entsprechende Feld wird stattdessen mit der niedrigst möglichen Integer-Zahl versehen. Hat eine Zahl mehr als 9 Stellen, so ist hier also Vorsicht geboten: Datenänderung kann zu Datenverlusten führen. (*Bug 82411*)<br><br>Dieser Bug tritt auch in Tabellenkontrollfeldern auf. Die Darstellung ist korrekt, wenn statt eines Zahlenfeldes im Formular ein formatierbares Feld genommen wird. |
|---|---|

Die Tabellenansicht des Formulars hat die folgenden Nachteile für diese Recherchemethode:

- Die Suche funktioniert nur innerhalb der einem Formular zugrundeliegenden Tabelle oder Abfrage, nicht in einem dazugehörigen Unterformular.
- In der Datenansicht werden Listenfelder angezeigt, nach denen aber nicht gefiltert werden kann. Wieder müssen die Fremdschlüssel für die Felder bekannt sein, damit eine Filterung Erfolg hat. Hat z.B. die "Medienart" «Buch» den Primärschlüsselwert «1», so wird in der Tabelle "Medien" der Fremdschlüssel «1» gespeichert. Nach diesem Schlüsselwert ist zu suchen – obwohl die eingeblendete Ansicht «Buch» in einem Listenfeld zeigt.

### Fazit

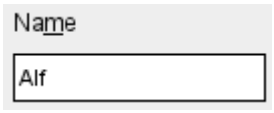
Sämtliche Such- und Filterfunktionen eignen sich für Formulare nur eingeschränkt. Die eingebaute Suche ist viel zu langsam und schränkt die Zahl der Datensätze nicht auf die mit entsprechenden Treffern ein. Die Filter funktionieren grundsätzlich nur innerhalb eines Formulars. Soll nach Werten im Unterformular gefiltert werden, so ist nur das aktuelle Unterformular betroffen. Es werden einfach nicht alle Datensätze korrekt ermittelt.

Aus diesem Grund wurde weiter oben in die Formulare der Beispieldatenbank eine entsprechend einfacher zu bedienende Suchfunktion integriert, die allerdings etwas Handarbeit und SQL-Kenntnis verlangt.

## Dateneingabe und Navigation

Ein Formular kann so eingestellt werden, dass mit der Funktion **Automatischer Steuerelement-Fokus** über den Button im Formularentwurf oder über den Formular-Navigator der Cursor direkt im ersten Eingabefeld steht. Statt mit der Maus von einem Feld zum anderen zu springen, ist bei Formularen die Bewegung mit der Tabulatortaste von einem Feld zum nächsten üblich.

Manchmal erfordern Formulare auch Sprünge zu einem anderen Formularfeld. Diese können natürlich mit der Maus geregelt werden, sind aber auch über Kurzbefehle anspringbar.




Dem Textfeld ist über **Eigenschaften: Textfeld → Beschriftungsfeld** ein Beschriftungsfeld zugewiesen. Die Beschriftung des Beschriftungsfeldes wurde von «Name» auf «Na~me» geändert. Dadurch erscheint im Formular jetzt ein Unterstrich unter dem «m». Das Textfeld kann jetzt bei der Eingabe mit dem Kürzel **`Alt + m`** angesprungen werden. Dies funktioniert allerdings nur, wenn der Cursor bereits in einem Formularfeld steht.

Prinzipiell können alle Buchstaben als Sprungziel genommen werden, da der Sprung innerhalb des Formulars erfolgt. Steht der Cursor aber nicht in einem Formularfeld, so werden mit den Kürzeln eventuell Sprungziele aus der LibreOffice-Oberfläche angesteuert. Mit dem Sprungziel «a» z. B. wäre dies dann statt des Feldes «Name» das Öffnen des Menüs «Ansicht».

Leider funktioniert das Anspringen von Feldern nur innerhalb eines Formulars, nicht aber bei einer Konstruktion mit Unterformularen. Ein Sprung aus einem Unterformular zu einem Hauptformular ist so also zur Zeit nicht möglich.

Aus einem Tabellenkontrollfeld sollte der Cursor mit **`Strg + Tab`** herausspringen. Manchmal kann es passieren, dass die Tastenkombination nicht den gewünschten Erfolg erzielt. Daher hier eine völlig andere Möglichkeit:[1]

1 Siehe hierzu auch die Beispieldatenbank «Beispiel_Cursorsprung_Subform_Mainform.odb»

In dem Unterformular wird eine Schaltfläche angelegt. Der Titel der Schaltfläche heißt z.B. «~neu». Das «~» sorgt, wie oben erwähnt, dafür, dass der Button mit dem Kürzel **Alt + N** ausgelöst werden kann. In den Zusatzinformationen des Buttons wird außerdem noch der Name des Feldes im Hauptformular benannt, zu dem der Cursor springen soll. Die Schaltfläche wird über **Eigenschaften → Ereignisse → Aktion ausführen** mit dem folgenden Makro verbunden:

```
SUB JumpToMainform(oEvent AS OBJECT)
    DIM oField AS OBJECT
    DIM oForm AS OBJECT
    DIM oDoc AS OBJECT
    DIM oController AS OBJECT
    DIM oView AS OBJECT
    DIM stShortcut AS STRING
    oField = oEvent.Source.Model
    stShortcut = Mid(oField.Label, InStr(oField.Label, "~") + 1, 1)
    oForm = oField.Parent.Parent
    SELECT CASE stShortcut
        CASE "n"
            oForm.MoveToInsertRow()
        CASE "ä"
            IF oForm.isLast() THEN
                oForm.MoveToInsertRow()
            ELSE
                oForm.Next()
            END IF
    END SELECT
    oDoc = thisComponent
    oController = oDoc.getCurrentController()
    oView = oController.getControl(oForm.getByname(oField.Tag))
    oView.setFocus
END SUB
```

Das Makro kann dazu genutzt werden, nur ins Hauptformular zurück zu springen. Es kann bei der obigen Einstellung auch dazu dienen, direkt einen neuen Datensatz anzulegen oder auch über einen weiteren Button mit dem Kürzel **Alt + ä** (für «nächster Datensatz») zum nächsten Datensatz zu springen. Befindet sich der Cursor des Hauptformulars allerdings schon auf dem letzten Datensatz, so muss der Sprung zu einem neuen Datensatz erfolgen. Weiteres zu Makros siehe im entsprechenden Kapitel.

Der Sprung auf die Schaltfläche gelingt nur mit einer sichtbaren Schaltfläche. Die Schaltfläche kann aber ruhig eine ganz geringe Ausdehnung oder eine Breite von 0 cm und eine Höhe von 0 cm haben. Bei der Definition einer nicht vorhandenen Breite und einer nicht vorhandenen Höhe kann es allerdings vorkommen, dass der Button als Strich im Formular über die gesamte Bildschirmbreite sichtbar wird.

**Hinweis**

Bei Tabellenkontrollfeldern funktioniert das Setzen des Focus nicht wie bei einfachen Formularfeldern. Hier muss das Tabellenkontrollfeld angesteuert werden und dann über

```
oView.CurrentColumnPosition = 2
```

z. B. das Feld in der 2. Spalte angesteuert werden.

Weitere Möglichkeiten zur Navigation befinden sich im Kapitel «Makros».

## Drucken aus Formularen

Formulare können so aufgebaut werden, dass ein direkter Druck der Ansicht möglich ist. Um dabei zu brauchbaren Ergebnissen zu kommen, muss allerdings darauf geachtet werden, Elemente nicht außerhalb des druckbaren Bereiches zu verschieben. Dazu wird beim Bearbeiten des Formulars **Ansicht → Drucklayout** gewählt. Die Seiteneigenschaften können eingestellt werden. Ein farbiger Hintergrund ist hier sicher nicht von Vorteil.

Jedes einzelne Formularelement kann in seinen **Eigenschaften → Allgemein → Druckbar** von einem Druck ausgeschlossen werden. Bilder, die aus einer Tabelle über grafische Steuerelemente ausgelesen werden, können leider aufgrund des *Bugs 90388* in den Versionen LO 3.4 bis LO 4.4 nicht gedruckt werden.

Ist die Datenbank in LibreOffice registriert worden (über **Extras → Optionen → LibreOffice Base → Datenbanken** oder direkt beim Erstellen der Datenbank), so kann ein Formular auch zum Serienbrief genutzt werden. Das Formular wird zum Bearbeiten geöffnet. Über **Ansicht → Datenquellen** oder mit F4 werden die Datenquellen zugänglich. Felder aus der Datenbank können jetzt über die Tabellenköpfe in das Formular gezogen werden. Das Formular wird anschließend abgespeichert. Wird das Formular anschließend wie zur Eingabe geöffnet, so erkennt LibreOffice, dass Serienbrieffelder enthalten sind und fragt nach, ob ein Serienbrief gedruckt werden soll.

Details zu der Erstellung von Serienbriefen sind in dem Kapitel «Datenbank-Anbindung» enthalten.

## Darstellungsgröße von Formularen

Beim Öffnen von Formularen kann der Nutzer manchmal von einer unüblichen Zoom-Einstellung des Formulars überrascht werden. Die Darstellungsgröße des Formulars richtet sich leider nicht nach der Zoom-Einstellung des Formulars beim Speichern. Vielmehr übernimmt Base die jeweils letzte Einstellung des Zooms von einem vorhergehenden Writer-Dokument.

Der Zoom kann über die Zoomeinstellung oder den Maßstab am unteren rechten Fensterrand eingestellt werden. Eine andere Einstellungsmöglichkeit ist über das Drücken von *Strg* und das gleichzeitige Betätigen des Scrollrades der Maus gegeben.

Wer automatisch bei Start eines Formulars immer die gleiche Größe erreichen möchte, kann dies über das folgende Makro erreichen:

```
SUB Formularzoom
    ThisComponent.CurrentController.ViewSettings.ZoomValue=100
END SUB
```

Dieses Makro soll beim Öffnen des Formulars ausgelöst werden. Das Formular muss zum Bearbeiten geöffnet werden. Unter **Extras → Anpassen → Ereignisse → Dokument öffnen** wird das Makro eingebunden.