LibreOffice
The Document Foundation

Base

Kapitel 1
Einführung in Base

Copyright

Dieses Dokument unterliegt dem Copyright © 2014. 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 Michael Niedermair

Jochen Schiffers

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 15.8.2014. Basierend auf der LibreOffice Version 4.3.

Anmerkung für Macintosh Nutzer

Einige Tastenbelegungen (Tastenkürzel) und Menüeinträge unterscheiden sich zwischen der Macintosh Version und denen für Windows- und Linux-Rechnern. Die unten stehende Tabelle gibt Ihnen einige grundlegende Hinweise dazu. Eine ausführlichere Aufstellung dazu finden Sie in der Hilfedatei des jeweiligen Moduls.

Windows/Linux	***entspricht am Mac***	***Effekt***
Menü-Auswahl **Extras → Optionen**	**LibreOffice → Einstellungen**	Zugriff auf die Programmoptionen
Rechts-Klick	*Control*+Klick	Öffnen eines Kontextmenüs
Ctrl (Control) oder *Strg* (Steuerung)	⌘ *(Command)*	Tastenkürzel in Verbindung mit anderen Tasten
F5	*Shift*+⌘+*F5*	Öffnen des Dokumentnavigator-Dialogs
F11	⌘+*T*	Öffnen des Formatvorlagen-Dialogs

Inhalt

Einführung 4

Base – ein Container für Datenbankinhalte 4

- Formulare – Start für die Dateneingabe 6
- Tabellen – Grundlagen für die Dateneingabe 7
- Abfragen – Auswertungsmöglichkeiten für eingegebene Daten 9
- Berichte – Präsentationen der Datenauswertung 9

Eine einfache Datenbank – Testbeispiel im Detail 12

- Tabellenerstellung 12
- Eingabeformular 20
  - Mit dem Tabulator zum Unterformular 27
  - Navigationsleiste des Hauptformulars auch im Unterformular aktivieren 28
  - Eingabe eines Kontrollfeldes einschränken 28
- Abfrage 30
- Ausdruck im Bericht 34
  - Abstände zwischen den Berichtsbereichen einstellen 39
  - Beeinflussung eines Textfeldinhaltes durch eine Formel 40
  - Umstellung der Formatierung eines Textfeldes 40
  - Verschieben von Feldern im Report-Builder 41
- Erweiterungen der Beispieldatenbank 41

Einführung

Im täglichen Büroeinsatz werden häufig Tabellenkalkulationen dazu benutzt, Datensammlungen zu erstellen um anschließend damit eventuell noch kleine Berechnungen durchzuführen. Die Tabellensicht ist sofort da, der Inhalt einfach einzugeben – da fragen sich dann viele Nutzer, warum es denn eine Datenbank sein sollte. Dieses Handbuch versucht den Unterschied zwischen Tabellenkalkulation und Datenbank herauszuarbeiten. Zu Beginn wird erst einmal kurz dargestellt, was eine Datenbank denn überhaupt leisten kann.

Hinweis	In der Fachsprache wird statt von Datenbanken von einer Benutzeroberfläche und einem «Datenbanksystem» gesprochen. Dieser Begriff umfasst das «Datenbankmanagementsystem» (DBMS) und den eigentlichen Datenbestand, die «Datenbank». Base hingegen bietet einen Zugriff auf verschiedene Datenbanksysteme über eine grafische Benutzeroberfläche. Base arbeitet standardmäßig mit dem eingebetteten Datenbanksystem «HSQLDB»

Das gesamte Handbuch bezieht sich, wie auch dieses erste Kapitel, in der Hauptsache auf zwei Beispieldatenbanken. Die eine Datenbank hat die Bezeichnung «Medien_ohne_Makros.odb», die andere Datenbank ist entsprechend mit Makros erweitert worden und trägt die Bezeichnung «Medien_mit_Makros.odb». Beide Datenbanken sollen einen Bibliotheksbetrieb ermöglichen: Medienaufnahme, Nutzeraufnahme, Medienverleih und alles, was damit verbunden ist wie z. B. das Anmahnen säumiger EntleiherInnen.

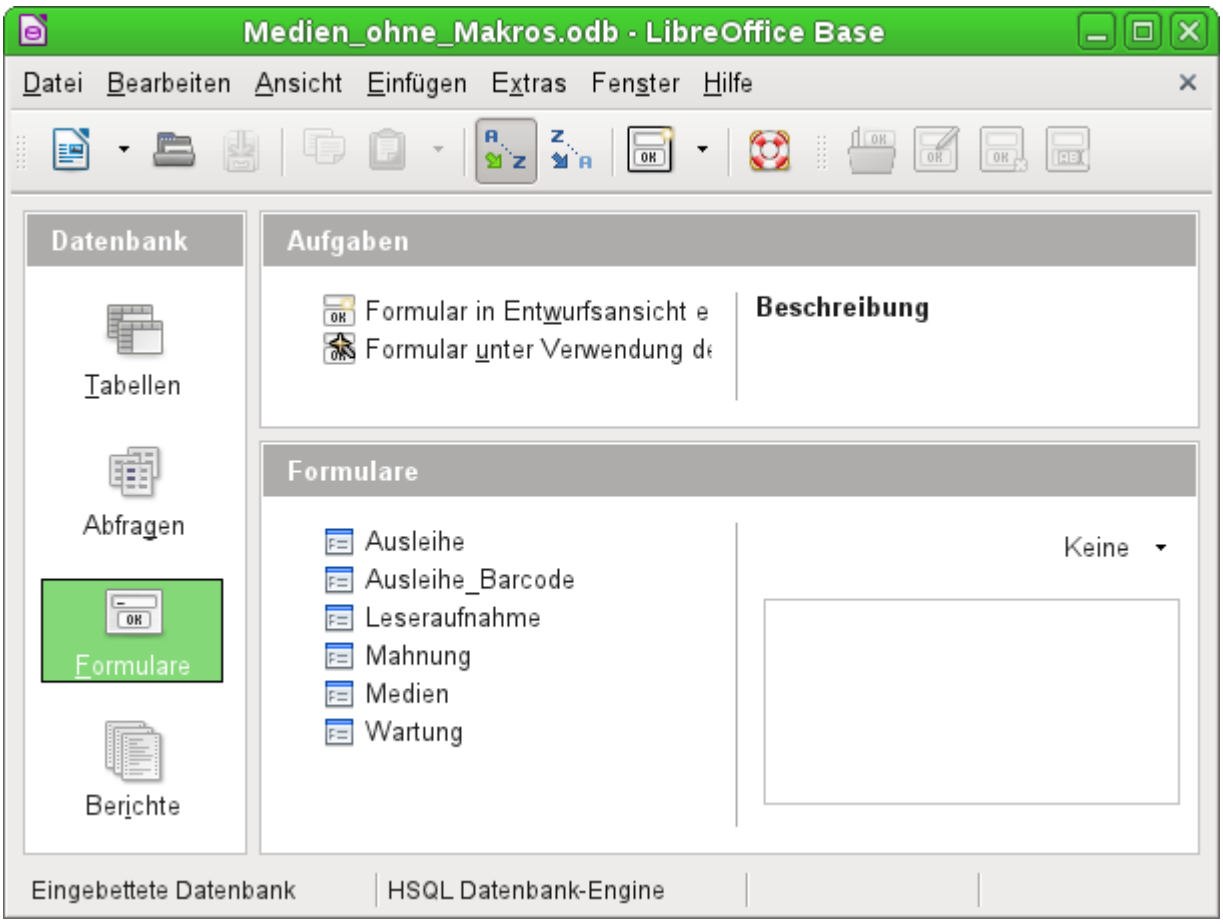
Zur Einführung in den Umgang mit Base ist das entsprechende Kapitel aus dem «Erste Schritte Handbuch» «*Einführung in Base*» gedacht.

Hinweis	Wie jede Software läuft auch LO-Base nicht vollkommen fehlerfrei. Besonders ärgerlich sind hier die «Regressionen», also Rückschritte von einer vorhergehenden Version zur gerade aktuellen Version. Der folgende Link führt zu den momentan noch offenen Regressionen: *https://bugs.freedesktop.org/buglist.cgi?keywords=regression%2C&keywords_type=allwords&bug_status=UNCONFIRMED&bug_status=NEW&bug_status=ASSIGNED&bug_status=REOPENED&bug_status=NEEDINFO&component=Database&product=LibreOffice* Ein Blick auf die Bug-Liste kann also helfen, Unterschiede zwischen Dokumentation und eigener Programmversion zu verstehen.

Hinweis	In diesem Handbuch werden Anführungszeichen mit unterschiedlichen Intentionen verwendet: "Tabellenname"."Feldname" → Tabellen und Felder in Datenbanken werden in doppelten Anführungszeichen oben wieder gegeben. 'Wert' → Werte, die in die Datenbank eingegeben oder aus ihr gelesen werden, sollen in einfachen Anführungszeichen erscheinen. «Hervorhebungen» → Alles, was sonst hervorgehoben werden soll, erscheint in der französischen Form der Anführungszeichen.

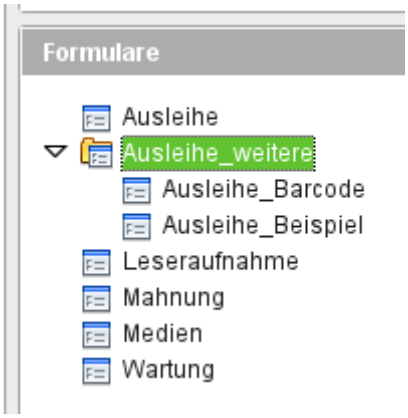
Base – ein Container für Datenbankinhalte

Eine Base-Datei ist eigentlich nur ein gepacktes Verzeichnis, in dem Informationen für die verschiedenen Arbeitsbereiche von Base stecken. In der täglichen Nutzung startet eine Base-Datei erst einmal mit der folgenden Ansicht:



Zur Arbeitsumgebung von Base gehören insgesamt vier Arbeitsbereiche: Tabellen, Abfragen, Formulare und Berichte. Je nach gewähltem Arbeitsbereich können bestimmte Aufgaben zur Neuerstellung der Elemente in Angriff genommen werden oder entsprechen fertiggestellte Elemente aufgerufen werden.

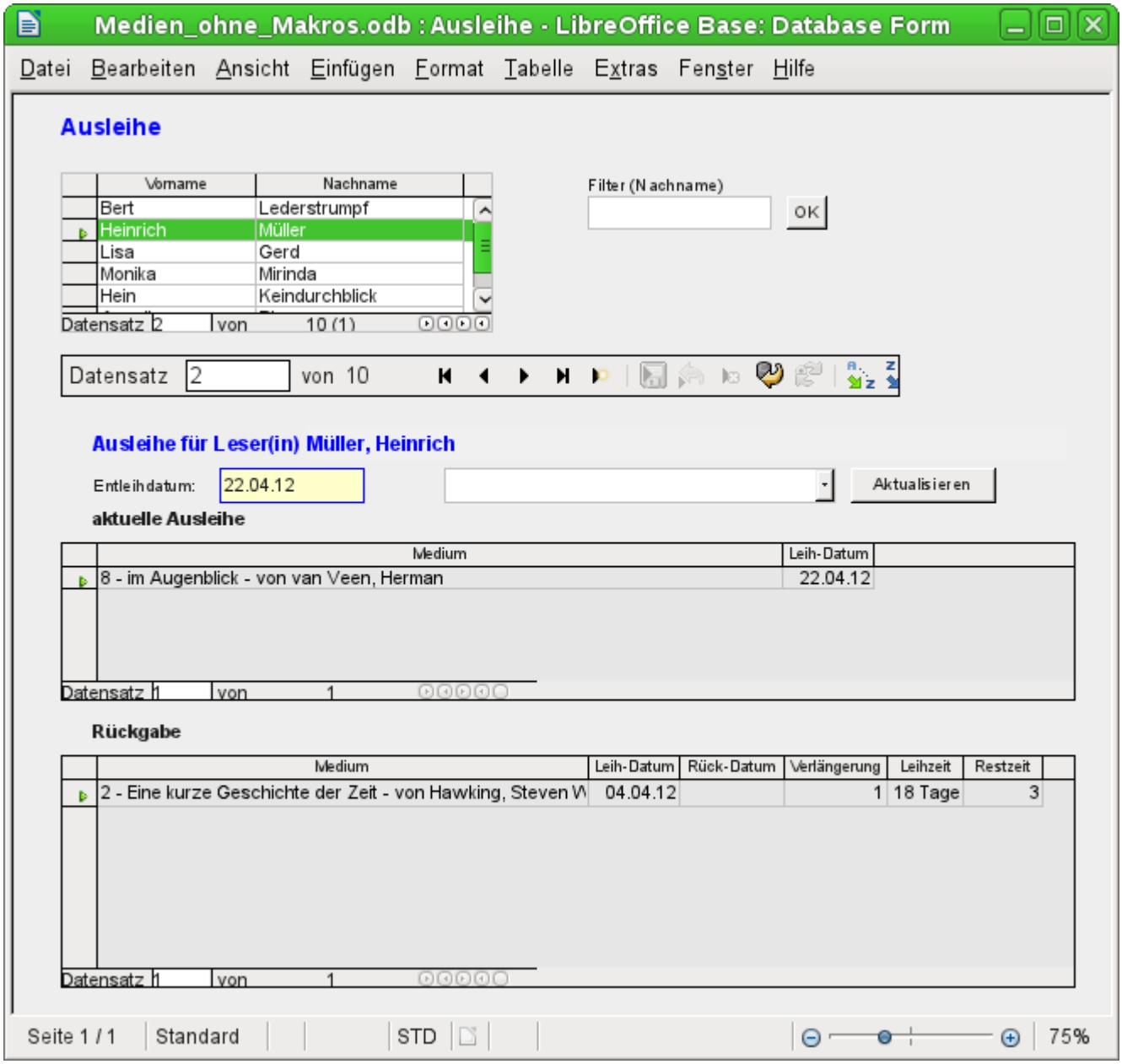
In den Arbeitsbereichen «Formulare» und «Berichte» können die jeweiligen Elemente auch innerhalb einer Verzeichnisstruktur angeordnet werden:



Dies geht entweder direkt beim Abspeichern über den Speicherdialog oder durch Neugründung von Verzeichnissen über **Einfügen → Ordner**.

Obwohl die Basis für eine Datenbank durch Tabellen gebildet wird, startet Base mit der Formularansicht, weil Formulare in der Regel die Elemente sind, mit denen die tägliche Datenbankarbeit vonstatten geht. Über die Formulare werden Einträge in die Tabellen vorgenommen und Inhalte aus den Tabellen ausgewertet.

Formulare – Start für die Dateneingabe



Einfache Formulare bilden lediglich eine Tabelle wie im oben sichtbaren Tabellenkontrollfeld mit den Namen ab. Dieses Formular erfüllt durch seine Struktur einige zusätzliche Punkte.

- Die Auswahl der Personen kann mit Filtern eingeschränkt werden. Die einfache Eingabe der Buchstaben «G» erzeugt eine Auswahl aller Personen, deren Nachname mit «G» beginnt.
- Neue Mediennutzer können direkt in das Tabellenkontrollfeld eingegeben werden.
- Die Ausleihdaten des aktuell markierten Nutzers werden in den darunterliegenden Feldern angezeigt. Dabei wird auch noch einmal der Name des Nutzers deutlich hervorgehoben. Hat ein Nutzer ein Buch entliehen und müsste es bereits zurückgegeben werden, so steht das Tabellenkontrollfeld «aktuelle Ausleihe» erst einmal nicht zur Verfügung. Stattdessen wird angegeben, dass die Ausleihe zur Zeit gesperrt ist.

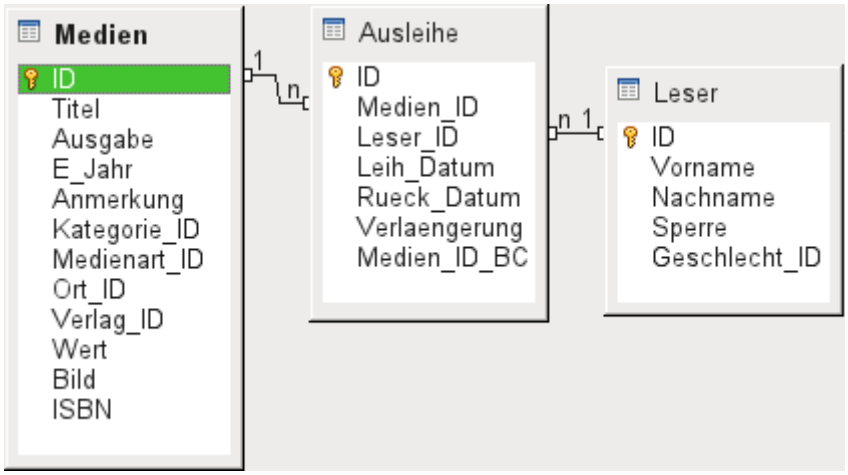
- Das Entleihdatum wird mit dem aktuellen Datum vorgegeben. In dem daneben stehenden Listenfeld werden die zu entleihenden Medien ausgewählt. Dabei können keine Medien ausgewählt werden, die zur Zeit noch entliehen sind.
- Die ausgewählten Daten werden mit dem Button Aktualisieren in das Tabellenkontrollfeld für den aktuellen Ausleihvorgang übernommen.
- Im Tabellenkontrollfeld für die Rückgabe ist es nicht möglich, einen Datensatz einfach zu löschen. Nur die Felder «Rück-Datum» und «Verlängerung» können bearbeitet werden. War ein Mediennutzer vorübergehend gesperrt und hat seine entliehenen Medien zurückgegeben, so lässt sich die Ausleihe über den Button Aktualisieren wieder freischalten.

Alle diese Funktionen können ohne den Einsatz von zusätzlicher Programmierung mit Makros gewährleistet werden, wenn entsprechend an den Formularen gefeilt wird.

Tabellen – Grundlagen für die Dateneingabe

Die Tabellen in einer Datenbank hängen in einem großen Geflecht zusammen. Eine Tabelle bezieht Informationen aus einer anderen oder gibt Informationen an andere Tabellen weiter. Dies wird als «Relation» bezeichnet.

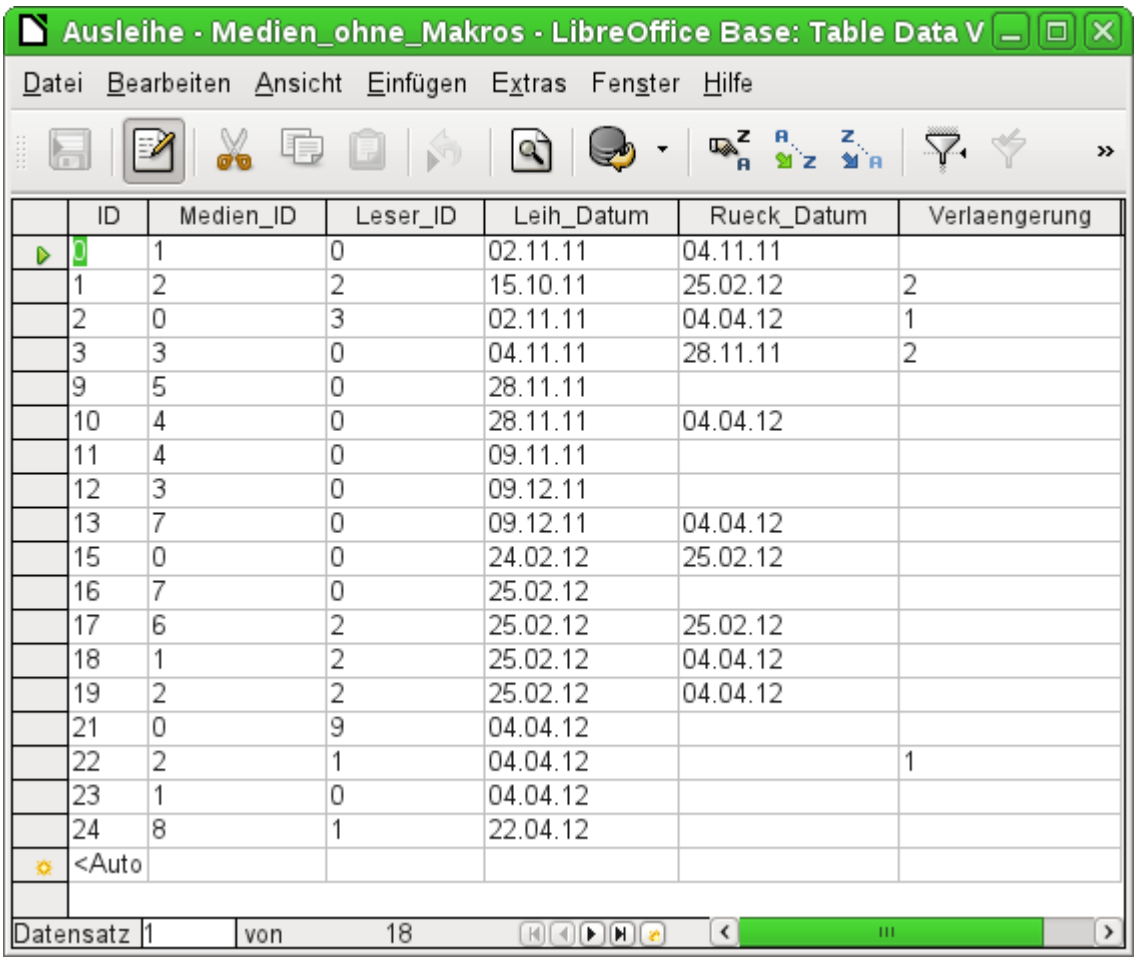
Die Tabelle "Ausleihe" steht in direkter Beziehung zu den Tabellen "Medien" und "Leser".



Statt in der Ausleihe den Titel eines Buches abzuspeichern, wird dort nur eine Zahl abgespeichert. Das eindeutige Kennzeichen eines Datensatzes der Tabelle "Medien" ist in dem Feld "ID" gespeichert. Das Feld ist das Schlüsselfeld der Tabelle "Medien", der Primärschlüssel.

In der Tabelle "Ausleihe" wird auch nicht jedes Mal der Lesername eingetragen. Hier gibt die Tabelle "Leser" Auskunft. Auch sie hat ein Primärschlüsselfeld. Der Wert dieses Feldes kann dann in die Tabelle "Ausleihe" eingetragen werden.

Die Beziehungen zwischen den Tabellen haben hier den Vorteil, dass die Schreibarbeit im Formular erheblich reduziert wird. Statt bei jeder Ausleihe ein Medium zumindest unverwechselbar und für jeden erkennbar aufzuschreiben und Leser mit Vor- und Nachnamen zu notieren, wird einfach im Inhalt der anderen Tabellen nachgesehen. Schließlich wird dasselbe Medium später noch öfter ausgeliehen und derselbe Leser kann schon beim ersten Entleihvorgang mehrere Medien entleihen.



ID	Medien_ID	Leser_ID	Leih_Datum	Rueck_Datum	Verlaengerung
0	1	0	02.11.11	04.11.11	
1	2	2	15.10.11	25.02.12	2
2	0	3	02.11.11	04.04.12	1
3	3	0	04.11.11	28.11.11	2
9	5	0	28.11.11		
10	4	0	28.11.11	04.04.12	
11	4	0	09.11.11		
12	3	0	09.12.11		
13	7	0	09.12.11	04.04.12	
15	0	0	24.02.12	25.02.12	
16	7	0	25.02.12		
17	6	2	25.02.12	25.02.12	
18	1	2	25.02.12	04.04.12	
19	2	2	25.02.12	04.04.12	
21	0	9	04.04.12		
22	2	1	04.04.12		1
23	1	0	04.04.12		
24	8	1	22.04.12		
<Auto					

Die Tabellengrundlage für das oben vorgestellte Formular sieht erst einmal recht nüchtern aus. In der oben abgebildeten Tabelle werden bis auf die Nutzerneueingabe alle Eingaben gemacht, die in dem Formular möglich sind. Dabei werden im Formular die Verbindungen dieser Tabelle zu den anderen Tabellen der Datenbank genutzt.

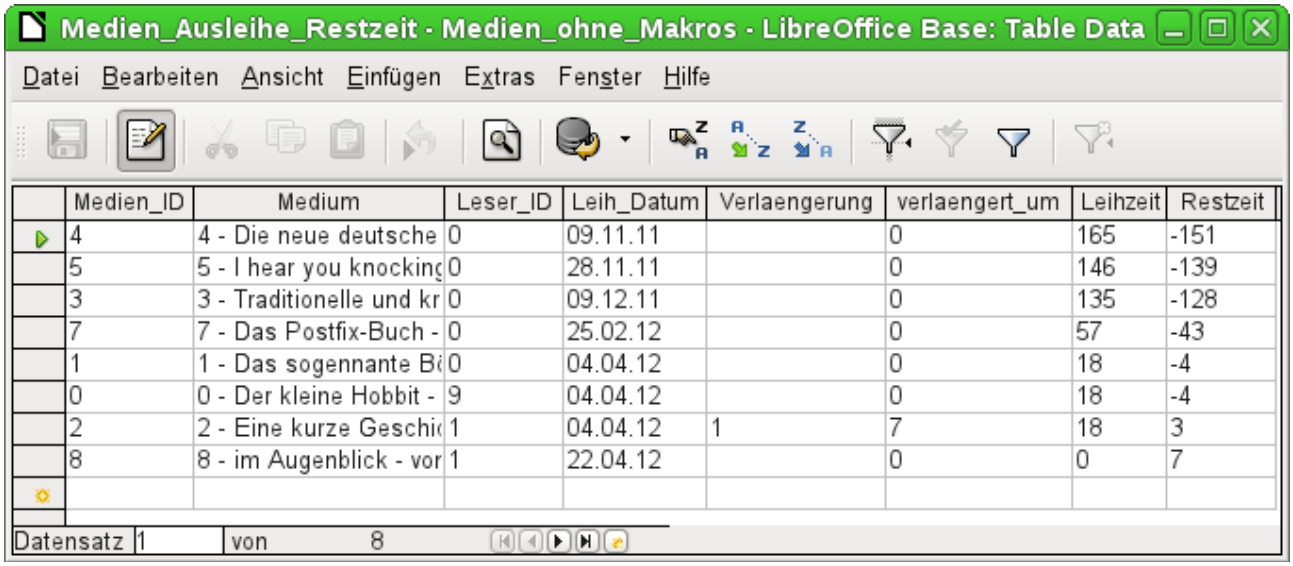
- Das erste Feld zeigt den für die meisten Datenbanken unabdingbaren, unverwechselbaren Inhalt, den Primärschlüssel ("ID"), der automatisch geschrieben wird.
  Mehr dazu im Kapitel «Beziehungen zwischen Tabellen allgemein».
- Das zweite Feld "Medien_ID" speichert den Primärschlüssel der Tabelle "Medien". Es verweist über die Nummer auf den entsprechenden Inhalt in dieser Tabelle. Solch ein Verweis auf einen Primärschlüssel wird als Fremdschlüssel bezeichnet. Im Formular werden statt des Fremdschlüssels über ein Listenfeld der Fremdschlüssel, der Titel und der Verfasser angezeigt. Das Listenfeld gibt im Hintergrund den Fremdschlüssel an die Tabelle weiter.
- Das dritte Feld "Leser_ID" speichert den Primärschlüssel der Tabelle "Leser", also nur eine Nummer, die auf den Leser verweist. Im Formular wird aber der Nachname und der Vorname angegeben. Wie in der Tabelle zu sehen ist, hat der Leser mit der Primärschlüsselnummer '0' sehr viele Medien entliehen. Die Tabelle kann beliebig oft den einzigartigen Primärschlüssel der Tabelle "Leser" als Fremdschlüssel "Leser_ID" abspeichern. Auf keinen Fall darf aber ein Leser, der in dieser Tabelle über den Fremdschlüssel verzeichnet ist, gelöscht werden. Sonst wäre nicht mehr nachvollziehbar, wer denn nun das jeweilige Medium entliehen hat. Die Datenbank macht in den Standardeinstellungen so ein Löschen unmöglich. Der Fachbegriff dafür ist die Gewährung der «referentiellen Integrität».
- Im dritten Feld wird das Ausleihdatum abgespeichert. Ist dieses Datum abgespeichert und entspricht das Datum nicht dem aktuellen Datum, so erscheint der entsprechende Datensatz zu dem entsprechenden Leser im Formular im untersten Tabellenkontrollfeld zu Rückgabe der Medien.

- Im letzten Feld wird eine Verlängerung ermöglicht. Was hier eine 1, 2 usw. bedeutet wird an anderer Stelle festgelegt. Dafür enthält die Datenbank eine gesonderte Tabelle mit der Bezeichnung "Einstellungen".

Diese Eingaben reichen aus, um letztlich einen Bibliotheksbetrieb in Gang zu halten.

Abfragen – Auswertungsmöglichkeiten für eingegebene Daten

Abfragen zeigen eine Sicht auf die Tabellen. Sie bringen Inhalte mehrerer Tabellen zusammen in einer Übersicht. Abfragen werden nur in der Abfragesprache «SQL» gespeichert. Sie sind also keine neuen Tabellen, auch wenn sie von der Ansicht her in Base erst einmal gleich erscheinen



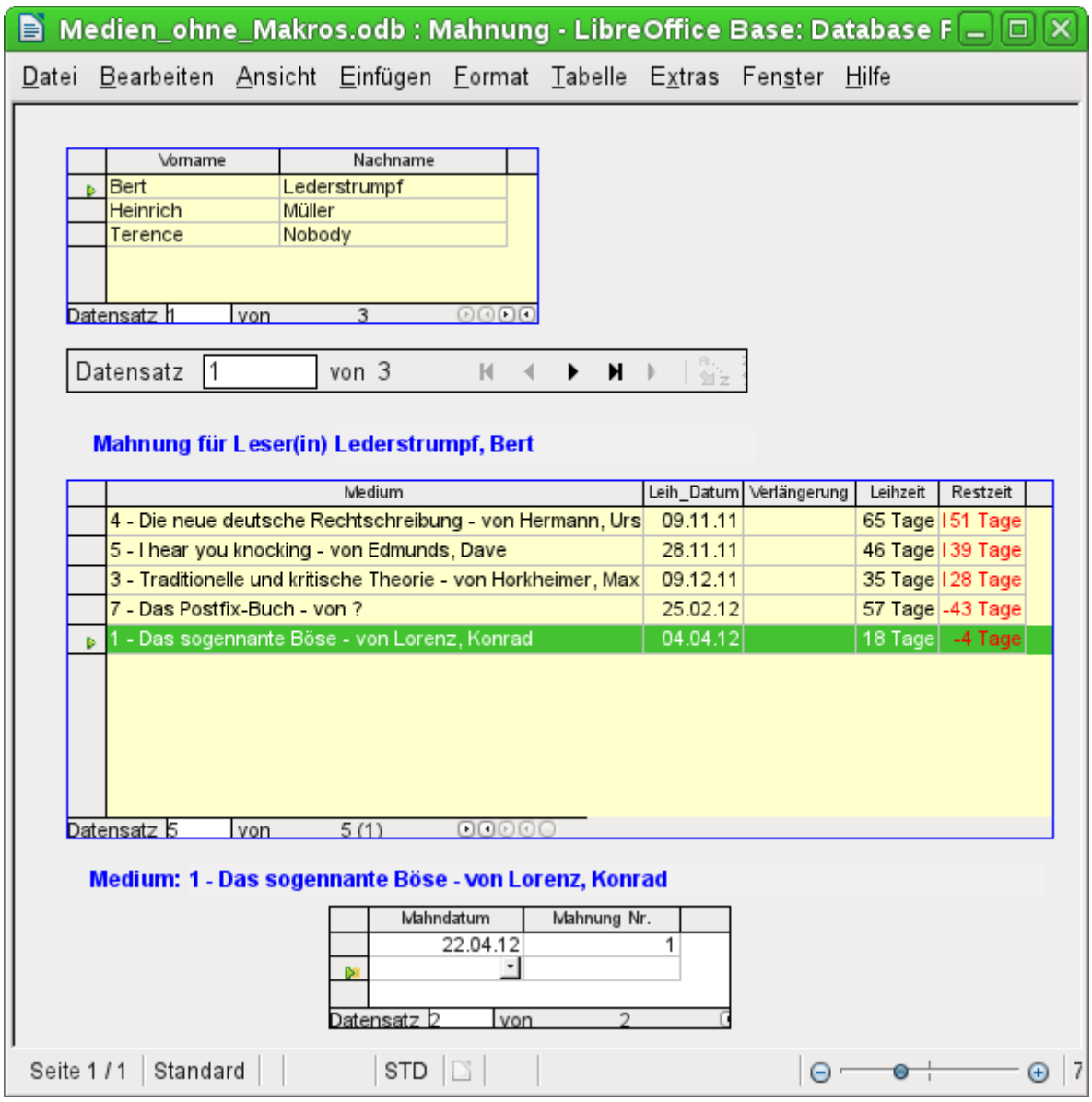
Medien_ID	Medium	Leser_ID	Leih_Datum	Verlaengerung	verlaengert_um	Leihzeit	Restzeit
4	4 - Die neue deutsche	0	09.11.11		0	165	-151
5	5 - I hear you knocking	0	28.11.11		0	146	-139
3	3 - Traditionelle und kr	0	09.12.11		0	135	-128
7	7 - Das Postfix-Buch -	0	25.02.12		0	57	-43
1	1 - Das sogennante Bi	0	04.04.12		0	18	-4
0	0 - Der kleine Hobbit -	9	04.04.12		0	18	-4
2	2 - Eine kurze Geschic	1	04.04.12	1	7	18	3
8	8 - im Augenblick - vor	1	22.04.12		0	0	7

Diese Abfrage listet alle Medien auf, die zur Zeit entliehen sind. Darüber hinaus berechnet sie, wie lange die Medien bereits entliehen sind und wie lange sie noch entliehen werden dürfen. Während das Feld "Medien_ID" das Fremdschlüsselfeld ausliest, wird in dem Feld "Medium" aus dem Primärschlüssel, dem Titel und dem Autor des Mediums ein zusammenhängender Text gebildet. Dieses Feld wird dann im Formular unter dem Untertitel «Rückgabe» benötigt. Einige Felder der Abfrage dienen dabei als Verbindungsfelder zu dem eigentlichen Ausleihformular mit der Tabelle "Ausleihe", nämlich die Felder "Medien_ID" und "Leser_ID".

- Alle Medien, bei denen das Rückgabedatum in der Tabelle "Ausleihe" nicht ausgefüllt ist, werden aufgelistet. Die Medien sind hier als zusätzliche Übersicht auch noch entsprechend benannt.
- Der Bezug zu den Lesern ist hergestellt über den Primärschlüssel der Tabelle "Leser"
- Aus dem Ausleihdatum "Leih_Datum" und dem aktuellen Datum wird die Zeitdifferenz in Tagen als "Leihzeit" angegeben.
- Von der Ausleihzeit, die je nach Medienart unterschiedlich sein kann, wird die Leihzeit abgezogen. Daraus wird die verbleibende Restzeit gebildet.
- Bei einer Verlängerung wurde in der Tabelle "Einstellungen" vermerkt, dass der Wert '1' für eine Verlängerung von 7 Tagen steht. Im vorletzten Datensatz mit der "Medien_ID" '2' ist so eine Verlängerung bei der Restzeit berücksichtigt worden.

Berichte – Präsentationen der Datenauswertung

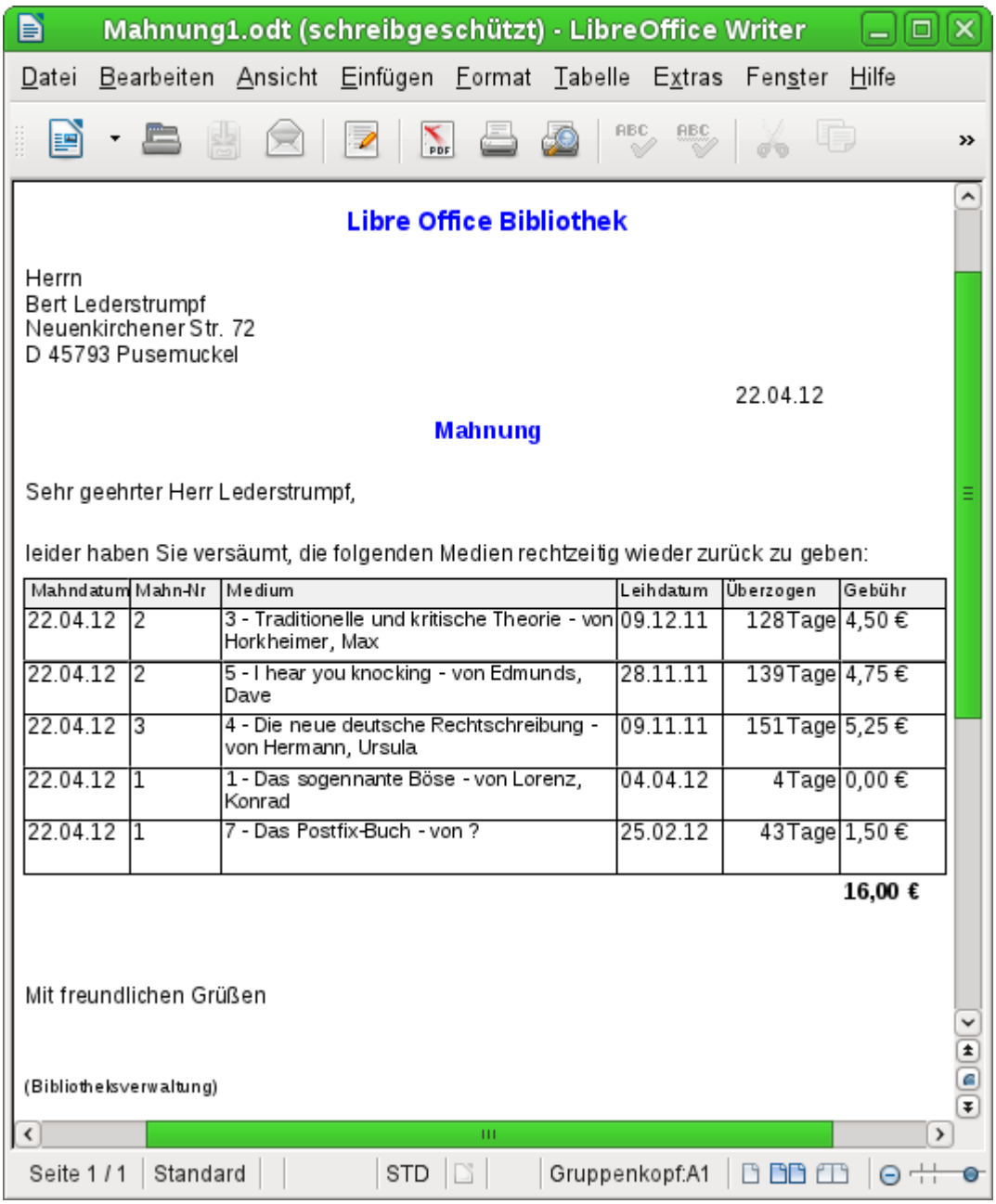
Über Berichte werden die Daten so aufbereitet, dass sie sinnvoll ausgedruckt werden können. Formulare wie das folgende sind nicht dazu geeignet, z. B. einen sauber formatierten Brief auszugeben.



Bevor ein aktueller Bericht in Form einer Mahnung ausgedruckt werden kann müssen in diesem Formular erst einmal die Mahnungen bestätigt werden. Im oberen Tabellenkontrollfeld stehen dafür alle Namen der Leser, die ein Medium entliehen haben und für die eine negative Restzeit vermerkt ist.

Für jedes anzumahnende Buch wird im unteren Tabellenkontrollfeld ein Mahndatum festgelegt. Dies dürfte bei der Abarbeitung von Mahnungen das aktuelle Datum sein. Die Mahnungsnummer wird nicht extra geschrieben. Sie wird anhand der bisher ergangenen Mahnungen einfach durch Addition ermittelt.

Dieses Formular benötigt in der Fassung ohne Makros noch eine Eingabe durch den Nutzer. In der Fassung, die mit Makros arbeitet, wird einfach das Datum automatisch geschrieben und anschließend der Bericht zum Mahnungsdruck aufgerufen.



Mit Hilfe einer Abfrage lässt sich aus den getätigten Eingaben solch eine Mahnung zum Ausdruck fertig erstellen. Der Nutzer der Datenbank braucht dazu lediglich bei den Berichten den Bericht Mahnung auszuwählen und kann dann einen entsprechenden Mahnbrief an alle die Personen schicken, bei denen im vorher angegebenen Formular eine Mahnung bearbeitet wurde.

In so einem Bericht stehen also gegebenenfalls auf den folgenden Seiten jeweils auf einer Seite weitere Mahnungen für andere Personen. Sollte ein Leser so viele Medien entliehen haben, dass der Platz auf einer Seite nicht ausreicht, so wird die Tabelle auf der Folgeseite einfach fortgesetzt.

Ein so erstellter Bericht ist also umfangreicher als ein Serienbrief, der mit Writer erstellt wird: er stellt automatisch alle Datensätze zusammen, die gedruckt werden sollen und ordnet den zusätzlichen Text entsprechend an.

Ein ähnlicher Brief wie in der obigen Abbildung lässt sich sonst nur unter Zuhilfenahme von Makros realisieren, wie es in «Drucken über Textfelder» beschrieben wird.

Eine einfache Datenbank – Testbeispiel im Detail

Die Erstellung einer Datenbank wird in «Neue Datenbank als interne Datenbank» abgehandelt. Dort werden neben der internen HSQLDB verschiedene andere Datenbanksysteme eingebunden.

Das folgende Beispiel basiert auf der Standarddatenbank, die mit LibreOffice als interne Datenbank installiert wird. Es wird also zuerst eine eingebettete Datenbank erstellt, die nicht angemeldet wird.

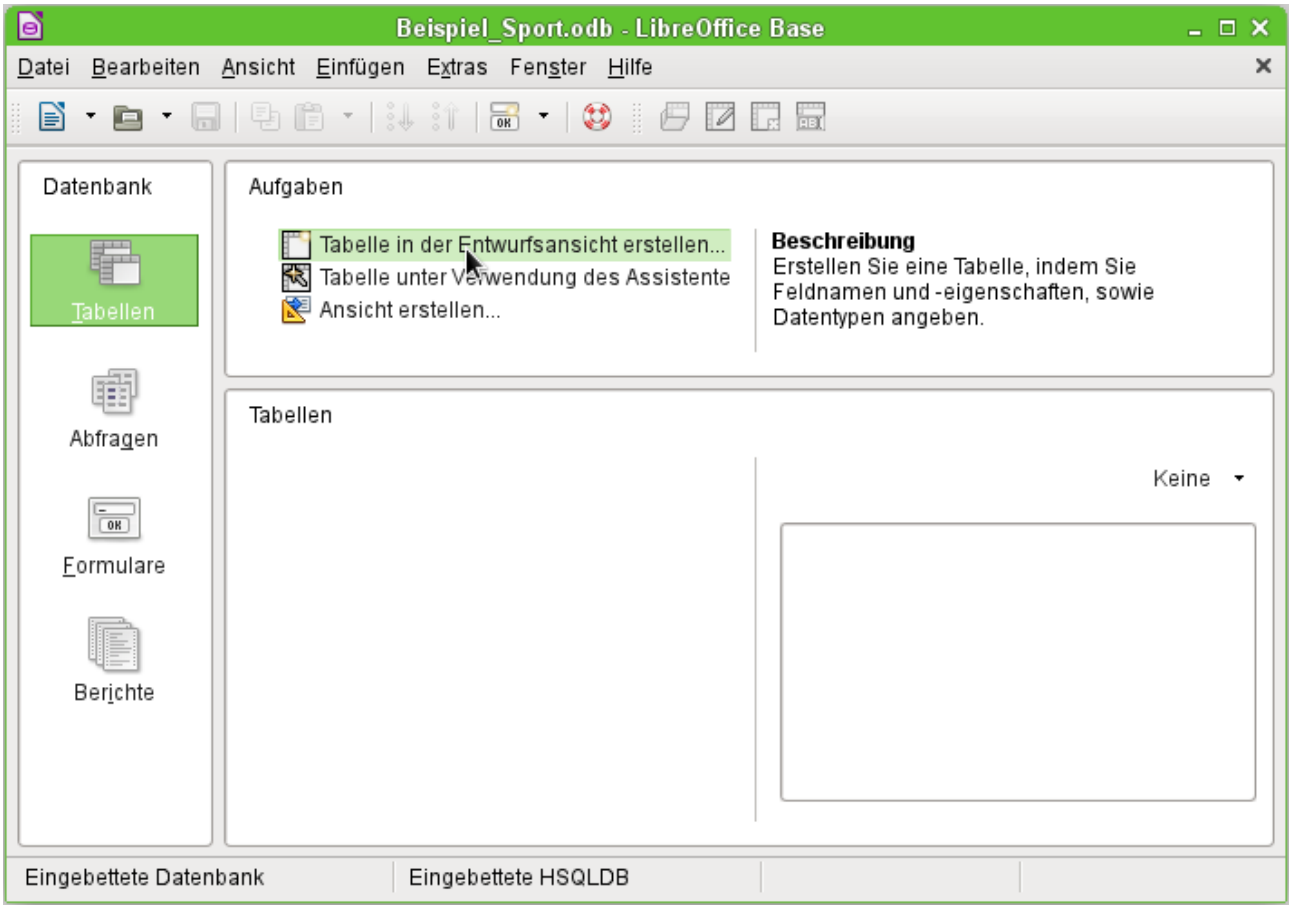
Erstes Kennzeichen bei der Erstellung einer Datenbank ist, dass zum Abschluss des Assistenten zuerst einmal ein Speicherort gesucht wird und die Datei abgespeichert wird.

Die Datenbank soll dazu dienen, einen Sportwettkampf in verschiedenen Disziplinen zu organisieren. Als Bezeichnung wurde deshalb «Beispiel_Sport» gewählt.

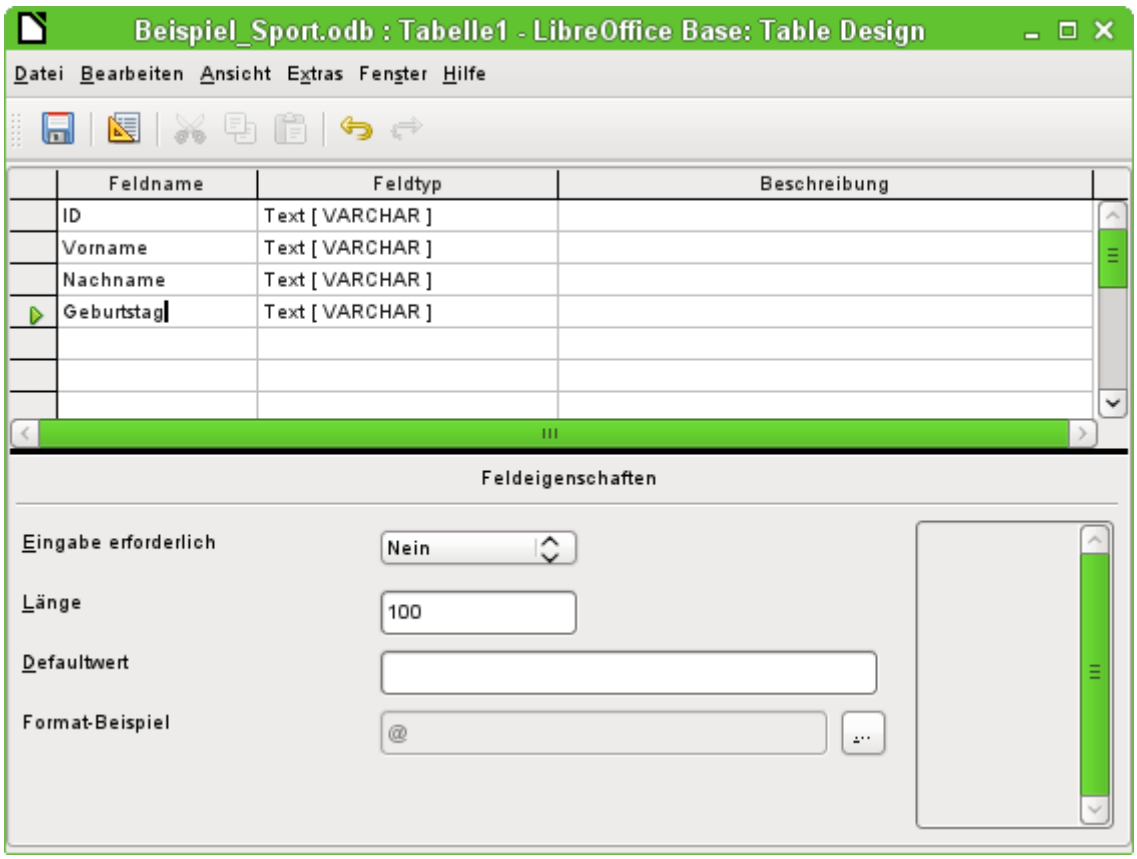
Tabellenerstellung

Sobald die Abspeicherung erfolgt ist erscheint der Startbildschirm der Benutzeroberfläche für die Datenbank. Standardmäßig sind zuerst auf der linken Seite im Bereich «Datenbank» «Tabellen» ausgewählt. Die Tabellen sind das zentrale Speicherelement für die Daten. Ohne Tabellen keine Datenbank.

Während im Handbuch «Erste Schritte» der Zugang über Assistenten gewählt wird, soll hier die direkte Erstellung in der Entwurfsansicht vollzogen werden.



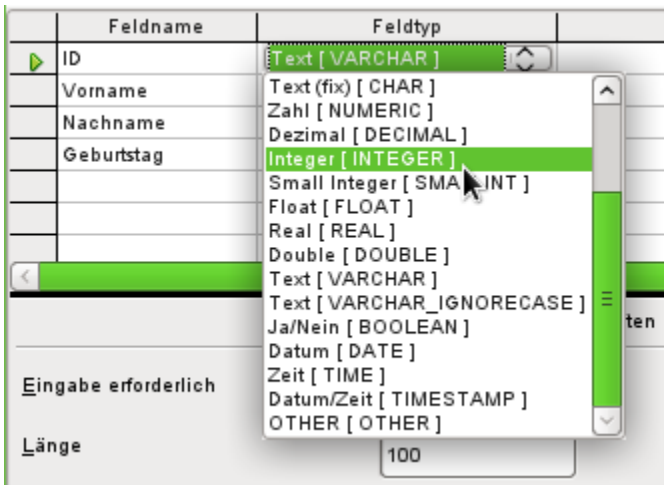
Tabellen → Aufgaben → Tabelle in der Entwurfsansicht erstellen … wird angeklickt. Es öffnet sich der folgende Bildschirm:



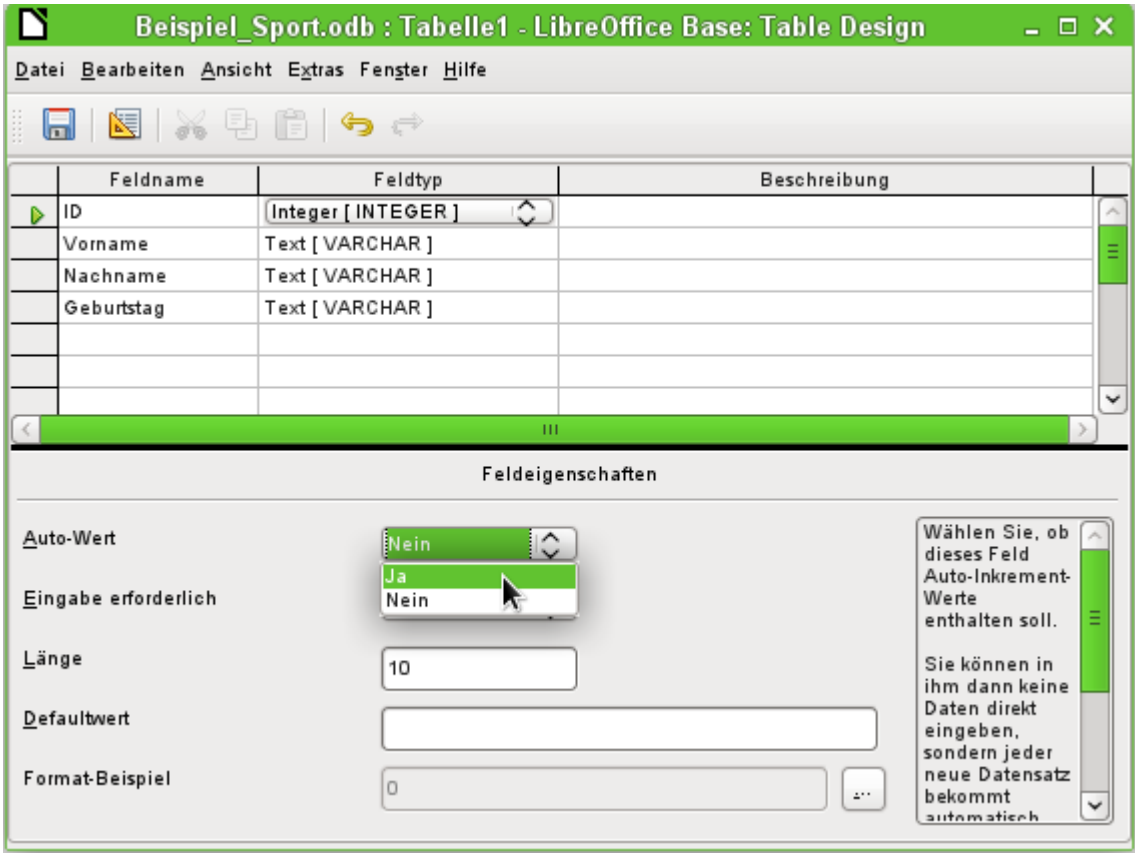
Hier werden zuerst einmal die verschiedenen Feldnamen für die erste Tabelle eingetragen. Die Tabelle soll die Teilnehmer und Teilnehmerinnen enthalten. Sie ist hier erst einmal auf die wesentlichen Bestandteile reduziert.

Die Feldnamen "Vorname", "Nachname" und "Geburtstag" dürften klar sein. Zusätzlich wurde ein Feld mit der Bezeichnung "ID" eingetragen. Dieses Feld soll später einen Wert aufnehmen, der die einzigartige Eigenschaft eines jeden Datensatzes ist. So ein einzigartiges Schlüsselfeld ist für die eingebettete Datenbank notwendig. Ansonsten lassen sich keine Datensätze in die Tabelle eingeben. Dieses Schlüsselfeld wird bei Datenbanken als «Primärschlüssel» bezeichnet.

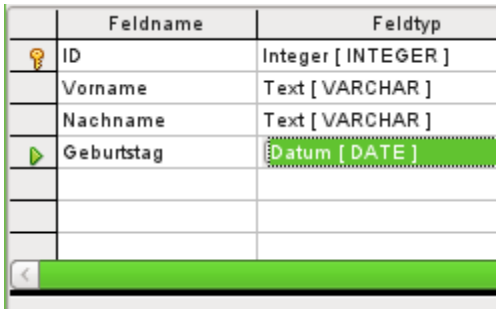
Es könnte natürlich auch ein anderes Feld für diese Eigenschaft genutzt werden. Wird aber z. B. der "Nachname" alleine dafür genutzt, so können nicht zwei Personen mit gleichem Nachnamen abgespeichert werden. In diesem Fall könnte es helfen, zwei Felder zusammen zu einem gemeinsamen Primärschlüssel zu erklären. Eine Garantie, dass das auf Dauer funktioniert, ist das aber nicht. Deshalb wird hier die einfachere Variante bevorzugt.

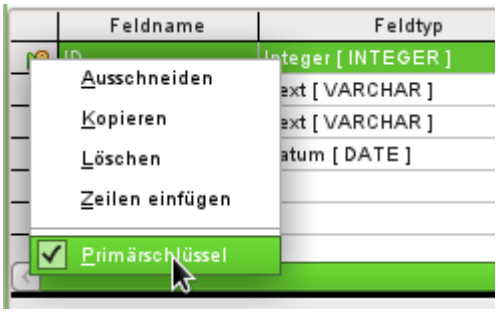


Im zweiten Arbeitsgang werden jetzt die Feldtypen für die bereits benannten Felder aus der Liste ausgewählt. Das Feld "ID" wird auf den Feldtyp «Integer» eingestellt. Es wird dann Ganzzahlen speichern. Dieser Feldtyp hat den Vorteil, dass er bei der eingebetteten HSQLDB automatisch mit der nächst höheren Ganzzahl versehen werden kann.



Die Feldeigenschaft des Feldes "ID" wird bearbeitet. Für dieses Feld wird die automatische Einsetzung von aufsteigenden Zahlenwerten aktiviert: **Feldeigenschaften → Auto-Wert → Ja**.





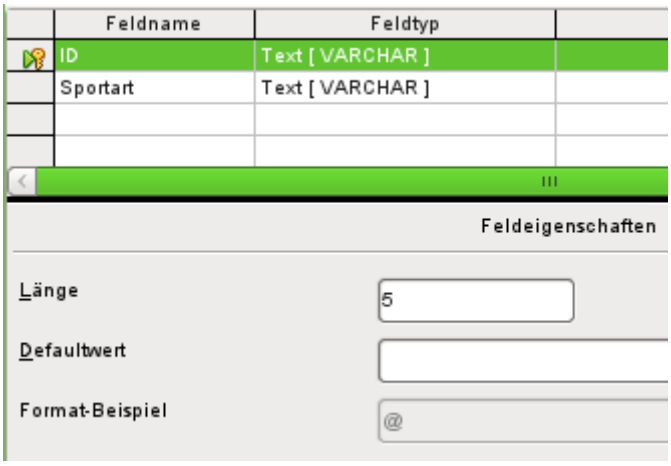
Durch die Einstellung des Auto-Wertes sollte beim Verlassen der Feldtypenauswahl auf dem Zeilenkopf ein Schlüsselsymbol zeigen. Das zeigt an, dass dieses Feld den Primärschlüssel der Tabelle bildet. Wird nicht der Auto-Wert gewählt, so kann der Primärschlüssel auch über das Kontextmenü der Maus (**rechte Maustaste → Primärschlüssel**) ausgewählt werden.

Die Tabelle kann nun unter dem Namen "Teilnehmer" abgespeichert werden. Anschließend können Daten eingegeben werden. Im Feld "ID" ist die Eingabe nicht nötig. Sie wird automatisch beim Abspeichern des Datensatzes erledigt.

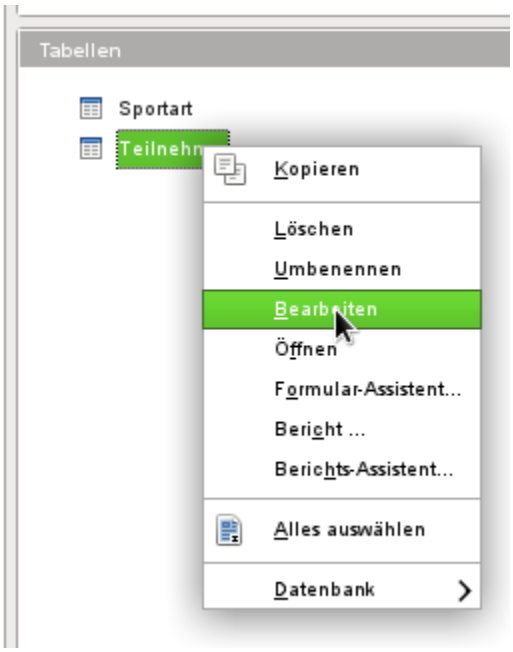
Hinweis	Die Datenbankdatei ist gepacktes Verzeichnis von einzelnen Dateien. Das Speichern eines einzelnen Elementes wie der Tabelle wird deshalb nicht direkt in die Datenbankdatei selbst geschrieben. Deswegen muss nach der Erstellung von Tabellen, Abfragen, Formularen und Berichten anschließend auch der Speicherbutton für die Datenbankdatei selbst betätigt werden. Einzig das Abspeichern von eingegebenen Daten funktioniert beim Verlassen der Datenzeile automatisch.

Die Teilnehmer können jetzt eingegeben werden. Folgende Informationen fehlen allerdings auf den ersten Blick:

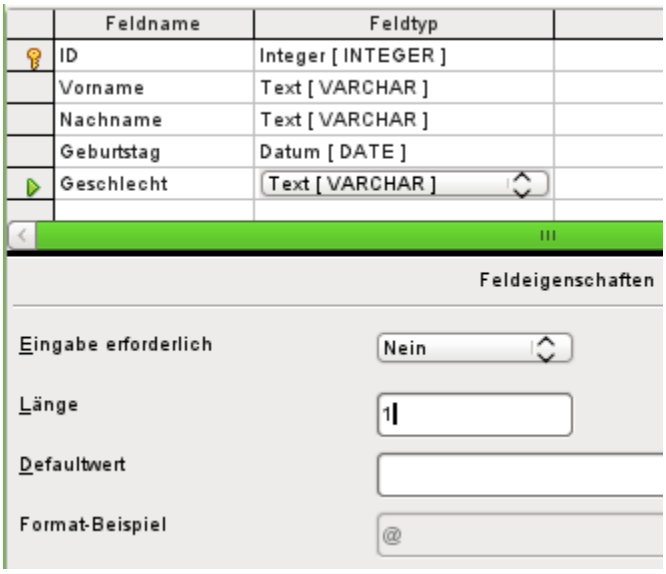
- Eine Liste der Sportarten, die die Teilnehmer belegen wollen.
- Für Wettkämpfe die Unterscheidung von Teilnehmern und Teilnehmerinnen.



Es wird eine Tabelle "Sportarten" erstellt. Da nicht so sehr viele unterschiedliche Sportarten in Frage kommen wird hier nicht der automatisch hoch zählende Primärschlüssel gewählt. Stattdessen wird der Feldtyp auf «Text» belassen, aber auf 5 Zeichen begrenzt. Die 5 Zeichen reichen aus, um damit auch passende Kürzel für die Sportarten zu finden.

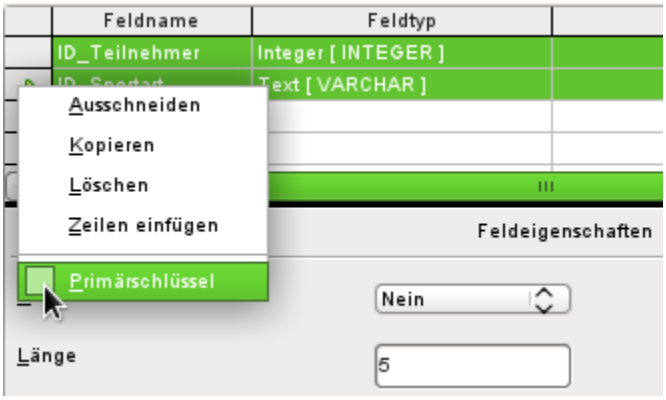


Die Tabelle "Teilnehmer" soll noch einmal zum Bearbeiten, nicht zur Dateneingabe, geöffnet werden. Dies geschieht über das Kontextmenü der Tabelle.



Der Tabelle wird das Feld "Geschlecht" hinzugefügt. Ein neues Feld kann mit der grafischen Benutzeroberfläche nur am Ende der Tabelle hinzugefügt werden. Über SQL ist auch die Möglichkeit gegeben, neue Felder an einer bestimmten Position einzufügen.

Die Länge des Textes in diesem Feld beschränkt sich auf einen Buchstaben. Es reicht also 'm' und 'w' als Eingabe.

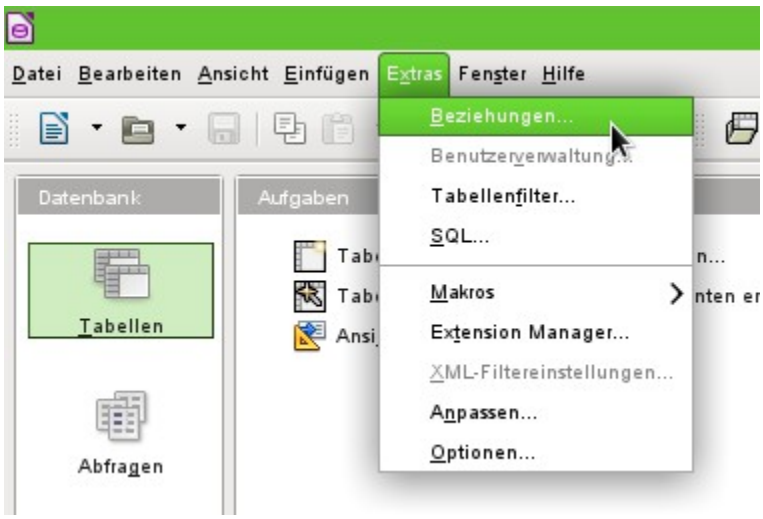


Irgendwie müssen die beiden Tabellen verbunden werden, so dass jedem Teilnehmer mehrere Sportarten und jeder Sportart mehrere Teilnehmer zugeordnet werden können. Dies geschieht durch eine Tabelle, in der die Werte der beiden Primärschlüssel der Tabellen "Teilnehmer" und "Sportart" abgespeichert werden sollen. Da nicht mehr als die Kombination dieser Felder gespeichert werden soll sind beide Felder zusammen der Primärschlüssel für diese Tabelle.

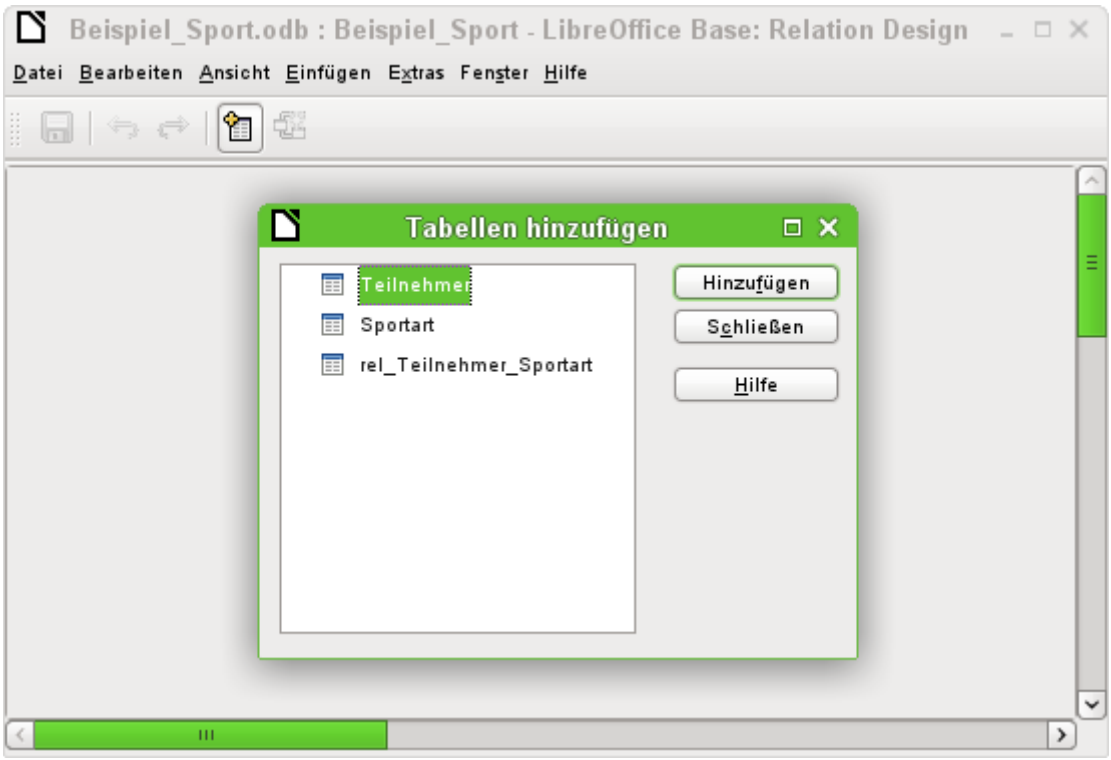
Damit wirklich die entsprechenden Werte aus "Teilnehmer" und "Sportart" aufgenommen werden können, müssen die Felder genau dem Feldtyp entsprechen, den sie speichern sollen. "ID_Teilnehmer" muss also den Feldtyp «Integer» haben. "ID_Sportart" muss den Feldtyp «Text» haben und außerdem, wie das Feld "ID" aus der Tabelle "Sportart", auf 5 Zeichen begrenzt werden.

Die Tabelle wird unter dem Namen "rel_Teilnehmer_Sportart" abgespeichert.

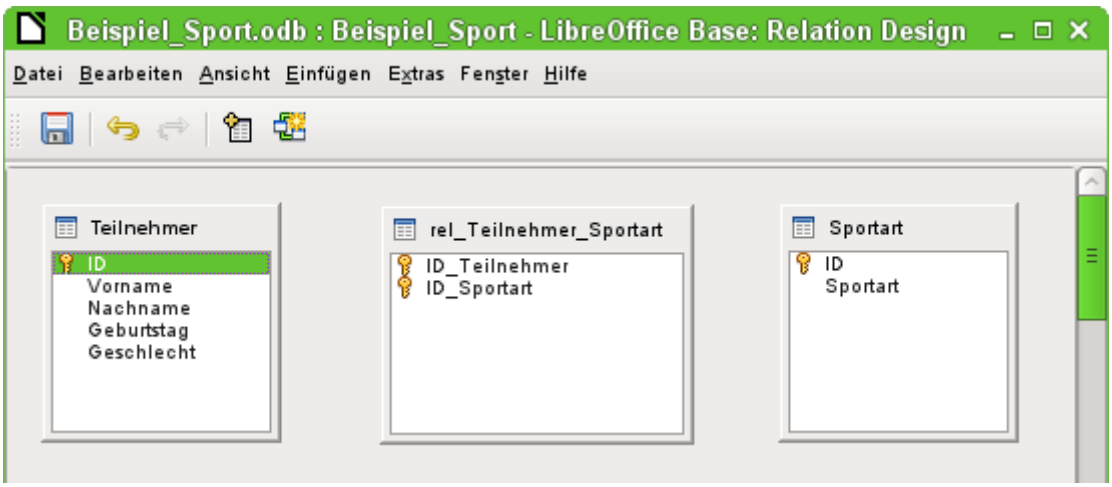
Sind die Tabellen fertiggestellt so sollte eine entsprechende Verbindung der Tabellen fest definiert werden. So kann verhindert werden, dass in der Tabelle "rel_Teilnehmer_Sportart" z. B. eine Nummer für einen Teilnehmer auftaucht, die gar nicht in der Tabelle "Teilnehmer" verzeichnet ist.



Über **Extras** → **Beziehungen** wird das Fenster für die Beziehungsdefinition geöffnet.

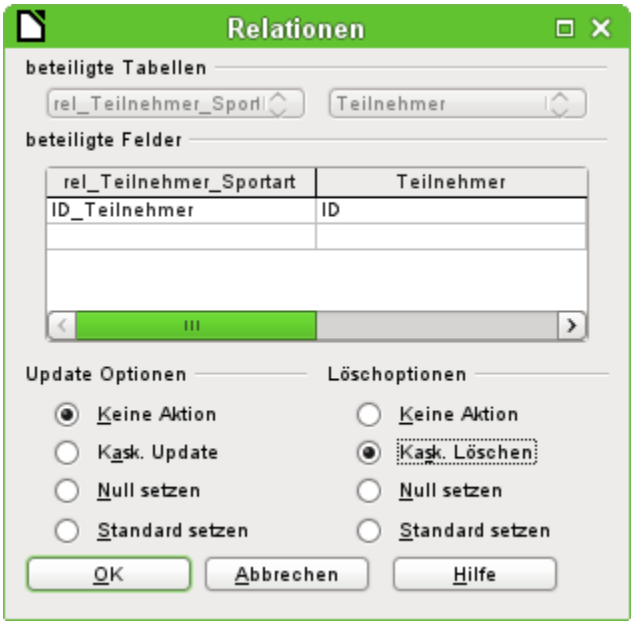


Für die Beziehungsdefinition werden alle bisher erstellten Tabellen notwendig. Die einzelnen Tabellen werden markiert und über *Hinzufügen* in den Beziehungsentwurf aufgenommen. Anschließend wird der Dialog «Tabellen hinzufügen» geschlossen.



Bei den hinzugefügten Tabellen sind alle Felder aufgeführt. Die Primärschlüsselfelder sind zusätzlich mit einem Schlüsselsymbol gekennzeichnet. Die vorgesehenen Rechtecke für die Tabellen können übrigens beliebig verschoben und in der Größe verändert werden.

Das Feld "Teilnehmer"."ID" wird mit der linken Maustaste markiert. Die Maustaste wird gehalten und die Maus auf das Feld "rel_Teilnehmer_Sportart"."ID_Teilnehmer" gezogen. Der Cursor zeigt eine Verknüpfung an. Die Maustaste wird über dem Feld "rel_Teilnehmer_Sportart"."ID_Teilnehmer" los gelassen. Es erscheint der folgende Dialog zur Definition der Beziehung («Relation»).

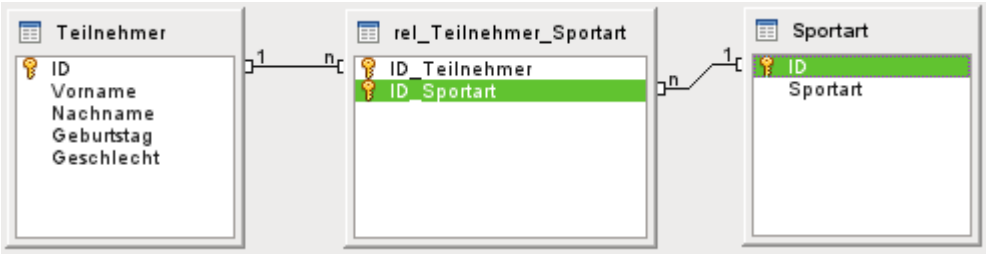


Das Feld "rel_Teilnehmer_Sportart"."ID_Teilnehmer" soll nicht verändert werden, wenn "Teilnehmer"."ID" verändert würde. Das ist die Standardeinstellung. "Teilnehmer"."ID" wird sowieso nicht verändert, da es sich um ein automatisch hoch zählendes Feld handelt und dort kein Eingriff vorgenommen werden soll.

Der Datensatz der Tabelle "rel_Teilnehmer_Sportart" soll gelöscht werden, wenn "ID_Teilnehmer" gleich "Teilnehmer"."ID" ist und "Teilnehmer"."ID" gelöscht wird. Wenn also aus der Teilnehmer-Tabelle ein Teilnehmer entfernt wird, dann werden alle entsprechenden Datensätze aus der Tabelle "rel_Teilnehmer_Sportart" entfernt. Dieses Vorgenehen wird als «kaskadierendes Löschen» bezeichnet.

Im nächsten Schritt werden "rel_Teilnehmer_Sportart"."ID_Sportart" und "Sportart"."ID" durch Ziehen der Maus bei gedrückter linker Maustaste verbunden. Auch hier soll ein Datensatz gelöscht werden, wenn die entsprechende Sportart gelöscht wird.

Für die Datenänderung bei "Sportart"."ID" ist allerdings eine andere Variante gewählt worden. Wird "Sportart"."ID" geändert, so wird entsprechend "rel_Teilnehmer_Sportart"."ID_Sportart" angepasst. So kann gegebenenfalls das Kürzel von maximal 5 Zeichen für eine Sportart problemlos angepasst werden, auch wenn bereits viele Datensätze in der Tabelle "rel_Teilnehmer_Sportart" stehen. Dieser Vorgang wird als «kaskadierendes Update» bezeichnet.

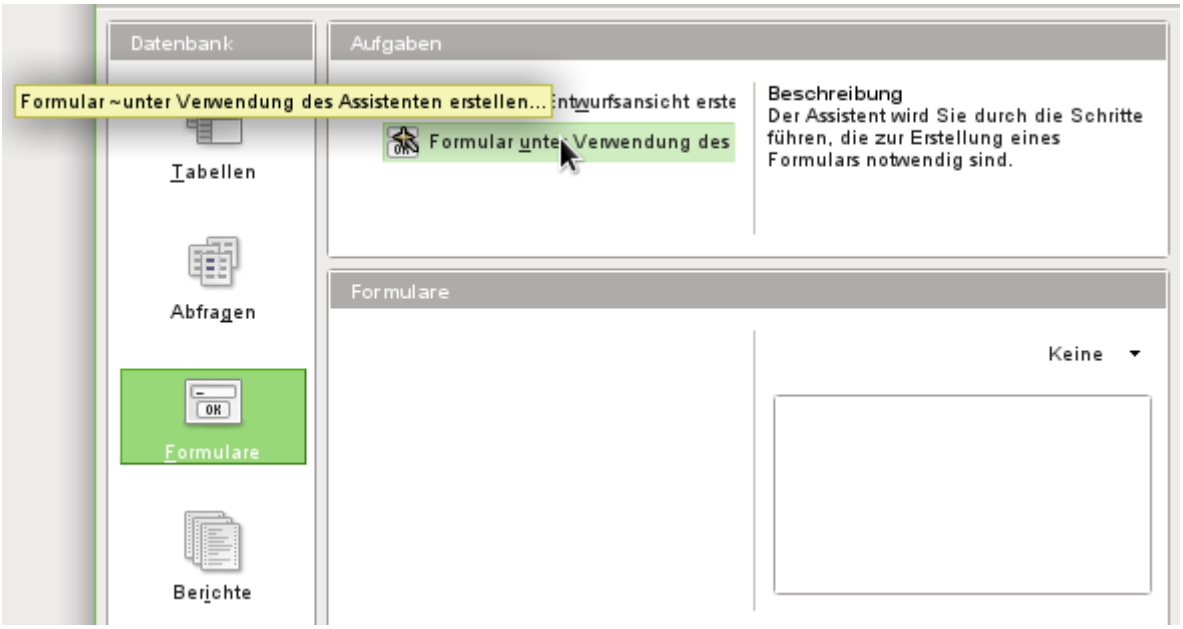


Die Felder sind nun fertig verbunden. An den Enden der Verbindungen erscheint jeweils eine «1» und ein «n». Ein Teilnehmer kann mehrmals in der Tabelle "rel_Teilnehmer_Sportart" auftauchen. Eine Sportart kann ebenfalls mehrmals in der Tabelle "rel_Teilnehmer_Sportart" auftauchen. Nur die Kombination von Teilnehmer und Sportart gibt es in der Tabelle nur maximal 1 Mal. Aus zwei 1:n-Beziehungen wird so über die dazwischen gesetzte Tabelle "rel_Teilnehmer_Sportart" eine n:m-Beziehung.

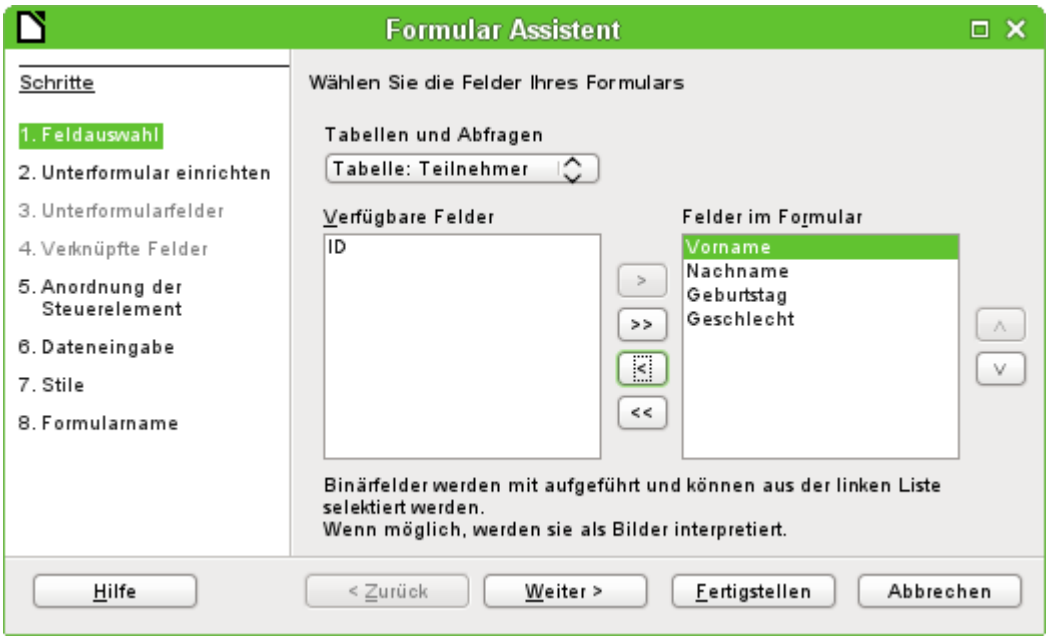
Solch eine Tabellenkonstruktion lässt sich nur schlecht über die Eingabe in Tabellen mit Inhalt füllen. Es müsste alle 3 Tabellen geöffnet sein, wenn einem Teilnehmer eine Sportart zugewiesen werden soll. In der Tabelle "Teilnehmer" muss "Teilnehmer"."ID" gesucht werden und nach "rel_Teilnehmer_Sportart" übertragen werden. In der Tabelle "Sportart" muss "Sportart"."ID" gesucht und entsprechend in "rel_Teilnehmer_Sportart" eingetragen werden. Zu umständlich: Ein Formular löst das eleganter.

Eingabeformular

Formulare können direkt in der Entwurfsansicht oder mit Hilfe des Assistenten erstellt werden. Die Erstellung mit dem Assistenten greift dabei auch alten Hasen so gut unter die Arme, dass diese eben schnell den Assistenten durchlaufen lassen und anschließend ein erst einmal vom Grund her erstelltes Formular ihren Wünschen anpassen. Das ist häufig der deutlich zeit sparendere Weg.



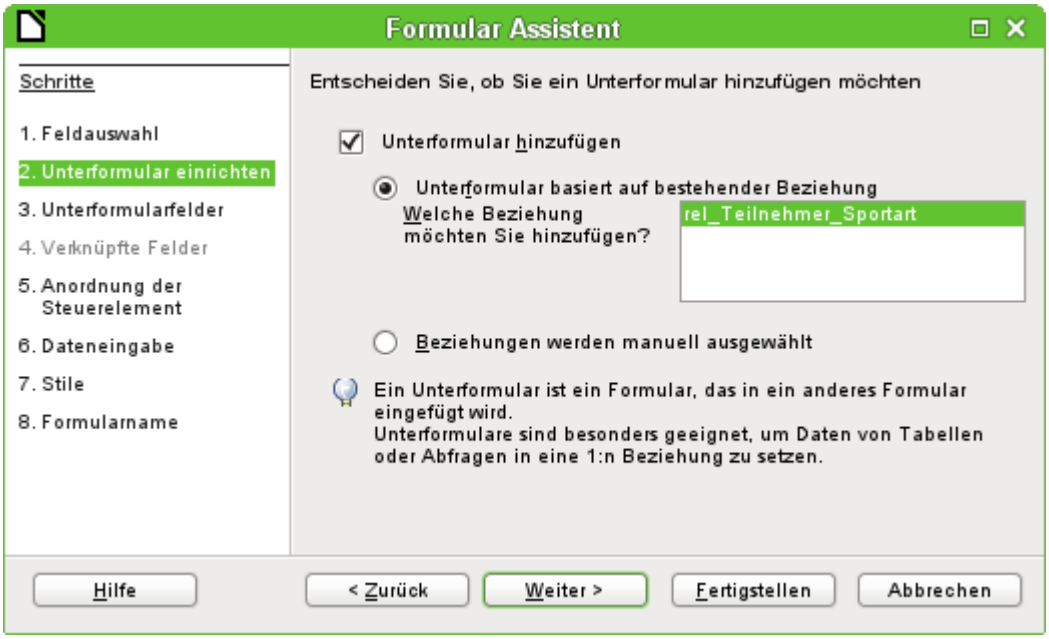
Zuerst wird unter «Datenbank» der Container für «Formulare» gewählt. Dann unter «Aufgaben» «Formular unter Verwendung des Assistenten erstellen ...».



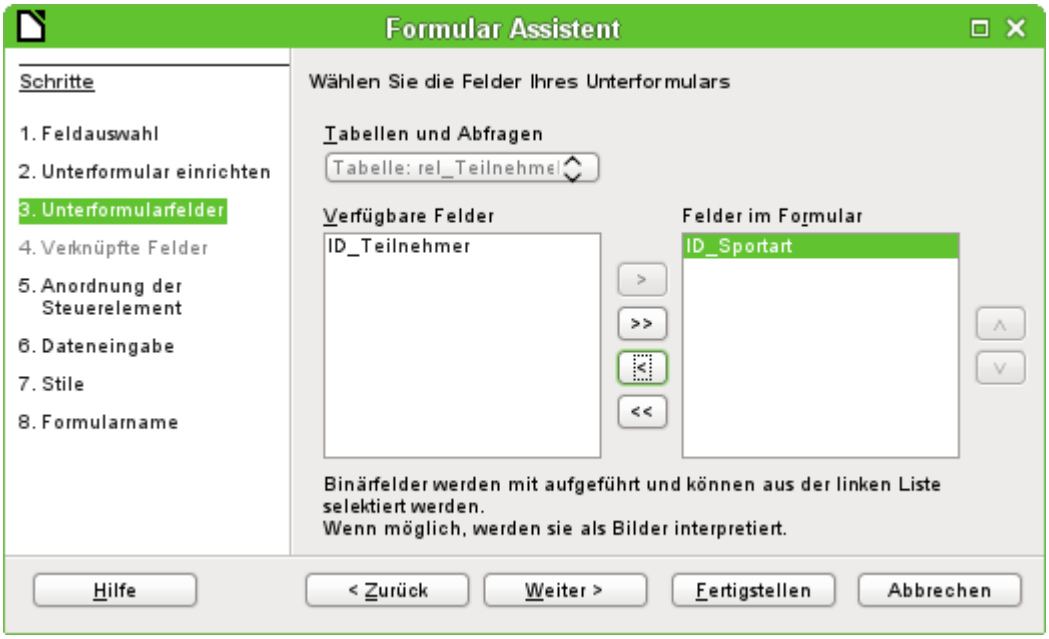
In der Hauptsache sollen in die Tabelle "Teilnehmer" Daten eingegeben werden. Die Tabelle "Sportarten" wird direkt mit den wenigen notwendigen Sportarten bestückt und nur selten ergänzt.

Von der Tabelle "Teilnehmer" werden alle Felder bis auf das Primärschlüsselfeld "ID" benötigt. Das Primärschlüsselfeld wird ja automatisch mit einem entsprechend unverwechselbarem Wert gefüllt.

Die Felder im Formular werden über die Pfeiltasten zwischen den beiden Auswahlkästchen bestückt.

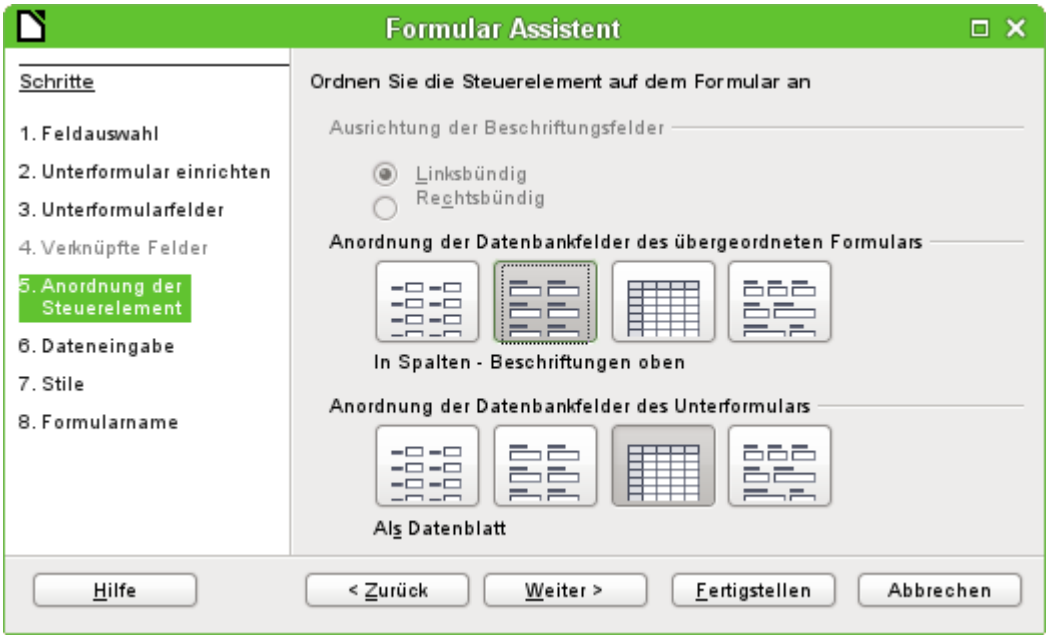


Es soll ein Unterformular eingerichtet werden, in dem die Sportarten dem Teilnehmer zugewiesen werden können. Die Beziehung für das Unterformular wird aus der vorher definierten Beziehung ausgelesen. Allerdings muss sie hier noch durch einen Klick auf die Tabelle "rel_Teilnehmer_Sportart" bestätigt werden.



Aus der Tabelle "rel_Teilnehmer_Sportart" wird nur das Feld "ID_Sportart" benötigt. Das Feld "ID_Teilnehmer" wird über die Verbindung von Hauptformular zu Unterformular mit dem Wert versehen, der in der Teilnehmer-Tabelle für den Primärschlüssel des aktuellen Datensatzes steht.

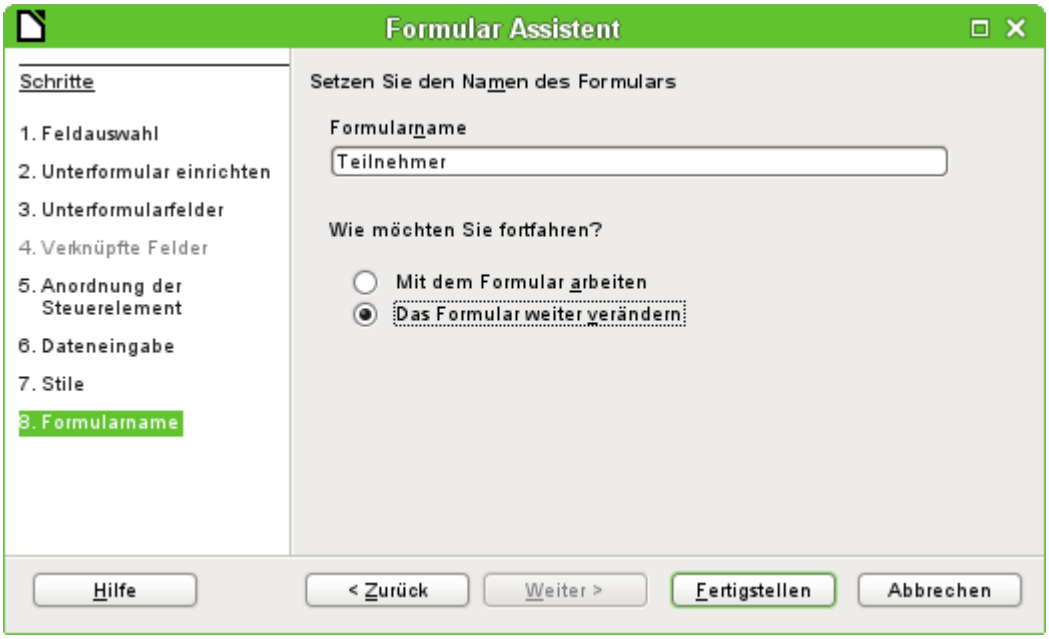
Dass diese Verknüpfung bereits geregelt ist, ist auch an dem 4. Schritt des Assistenten tu erkennen: Der Schritt «Verknüpfte Felder» ist inaktiv.



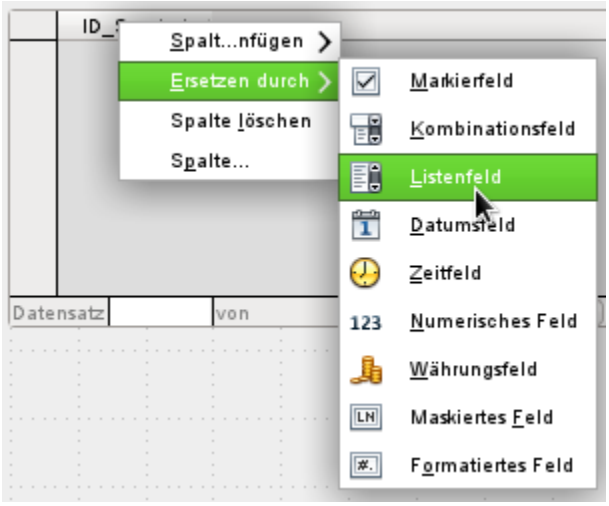
Wie die Elemente im Hauptformular und wie im Unterformular angeordnet sind ist letztlich egal. Damit aber die Zuordnung von Daten auch für ungeübte einfach zu ersehen ist, sollte für das Hauptformular in unserem Beispiel nicht das Tabellenkontrollfeld gewählt werden. Im Unterformular soll das Feld später alle Sportarten des Teilnehmers anzeigen. Deswegen an dieser Stelle am besten das Tabellenkontrollfeld belassen.

Schritt 6 sollte so bleiben, wie er vorgesehen ist, also **Dateneingabe → Das Formular zeigt existierende Daten an**. Damit ist dann eine Neueingabe ebenso möglich wie eine Veränderung bereits bestehender Daten.

Schritt 7 ist reine Geschmackssache. Nur Vorsicht: manche Stile bergen unverhoffte kontrastarme Darstellungen vor allem in Tabellenkontrollfeldern. Hier muss dann gegebenenfalls die Schriftfarbe der Tabellenkontrollfelder nachjustiert werden.



Mit dem Formular soll noch nicht gearbeitet werde, weil das Unterformular noch die Eingabe des Kürzels für die Sportarten erfordert. Hier wäre eine Auswahl der Sportarten nach dem kompletten Namen wünschenswert. Deswegen: **Das Formular weiter verändern**.



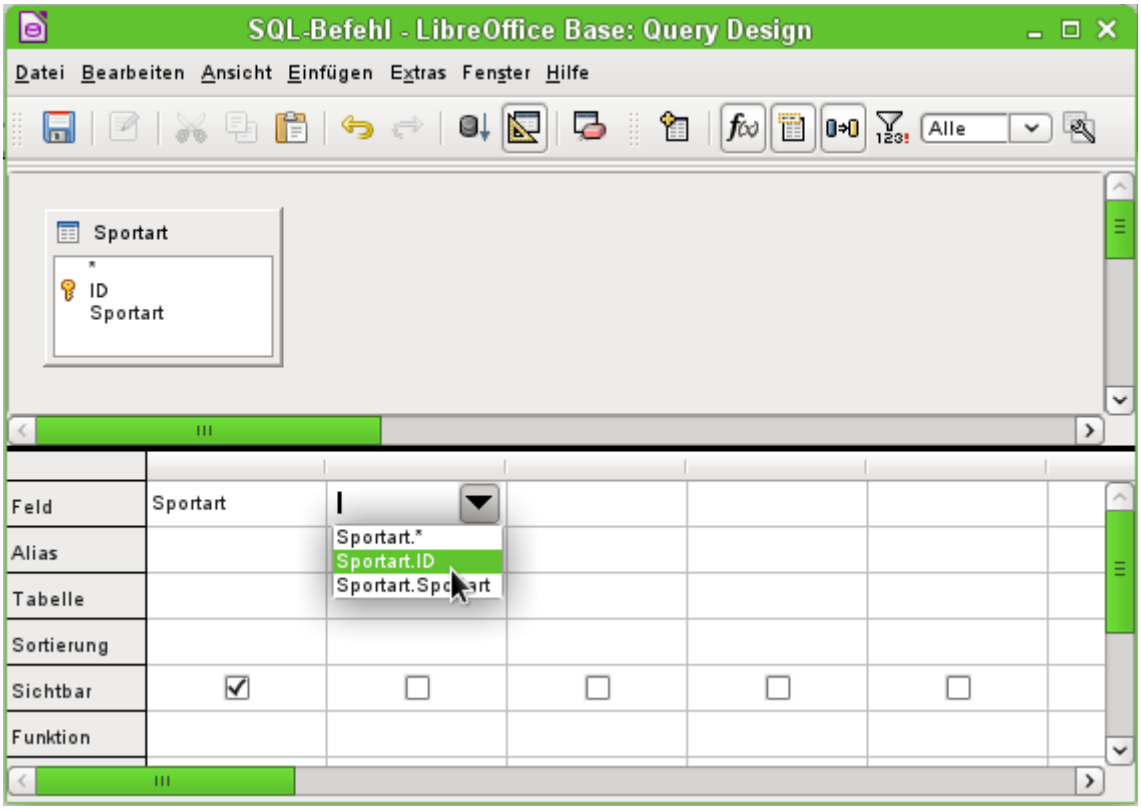
Zuerst wird mit der rechten Maustaste auf den Tabellenkopf des Tabellenkontrollfeldes geklickt. Dort steht «ID_Sportart». Im Kontextmenü wird **Ersetzen durch → Listenfeld** ausgewählt.

Anschließend soll das Listenfeld bearbeitet werden, damit es auch die entsprechenden Daten anzeigen kann. Wieder über das Kontextmenü geht es zu **Spalte...**

Es öffnen sich die Eigenschaften das ausgewählten Listenfeldes. **Daten → Art des Listeninhaltes → SQL** soll gewählt werden. Mit Hilfe von SQL (StructuredQueryLanguage – Standardabfrage-sprache für Datenbanken) soll das Feld seinen Inhalt aus der Tabelle "Sportart" erhalten.

Ist **SQL** ausgewählt, so erscheint in der Zeile **Listeninhalt** rechts ein Button mit 3 Punkten … . Wird dieser Button gedrückt, so öffnet sich der Editor für die Erstellung von Abfragen. Die im Folgenden zu erstellenden Abfrage wird hier nur zusammengestellt und schließlich in dem Listenfeld selbst gespeichert.

Im Dialog «Tabelle oder Abfrage hinzufügen» erfolgt ein Klick auf **Sportart** → **Hinzufügen** → **Schließen**.



Für die erste Spalte wird in dem Abfrage-Editor das Feld "Sportart" ausgewählt. Dieses Feld soll im Listenfeld sichtbar sein.

Die zweite Spalte soll mit dem Feld "ID" bestückt werden. Dieses Feld gibt nachher seinen Wert an die Tabelle weiter, auf der das Unterformular basiert. Es werden also die ausgeschriebenen Worte dargestellt und die passenden Kürzel abgespeichert.

Die Abfrage wird mit dem Speicherbutton in die Eigenschaften des Listenfeldes übertragen. Der Editor wird anschließend geschlossen.

Im Listeninhalt ist jetzt der folgende Text zu lesen:

```
SELECT "Sportart", "ID" FROM "Sportart"
```

Das ist der SQL-Code, der aus der Betätigung des Editors erwachsen ist. Umgangssprachlich ausgedrückt heißt der Code: Wähle von der Tabelle "Sportart" als erstes die "Sportart" und als zweites den dazugehörigen Schlüsselwert aus.

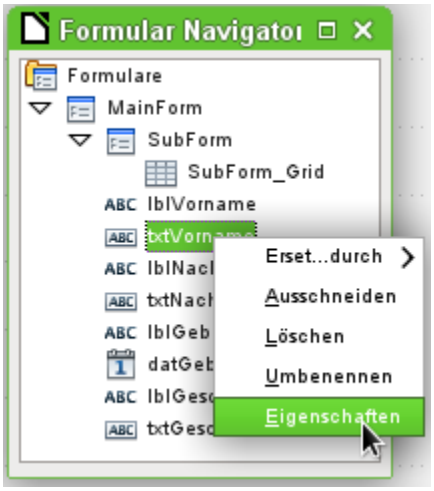
Diese Abfrage stellt das Minimum dar, das ausgewählt werden sollte. Natürlich könnte auch eine Sortierung eingebaut werden. Mit geschickter Wahl der in "ID" abgespeicherten Kürzel ergibt sich aber bereits eine brauchbare Zusammenstellung der Sportarten. Wird nämlich keine Sortierung vorgegeben, so erfolgt die Sortierung immer nach dem Primärschlüsselfeld.

Nun soll noch der Titel geändert werden, der für das Feld "ID_Sportart" eingetragen wurde. Es wird zwar weiterhin "ID_Sportart" abgespeichert. Sichtbar ist aber "Sportart". Deswegen: **Allgemein → Titel → Sportart**.

Soll z.B. das Design bei anderen Feldern angepasst werden, so lassen sich diese am besten über den Formularnavigator ansteuern. Werden Felder direkt angeklickt, so werden nicht nur die Eingabefelder, sondern auch die Beschriftungsfelder markiert. Durch den Assistenten wurden sie zu Gruppen zusammengefasst. Das erfordert dann weitere Aktionen im Kontextmenü der Auswahl.

Der Formularnavigator wird über die Symbolleiste «Formular-Entwurf» am unteren Bildschirmrand gestartet.

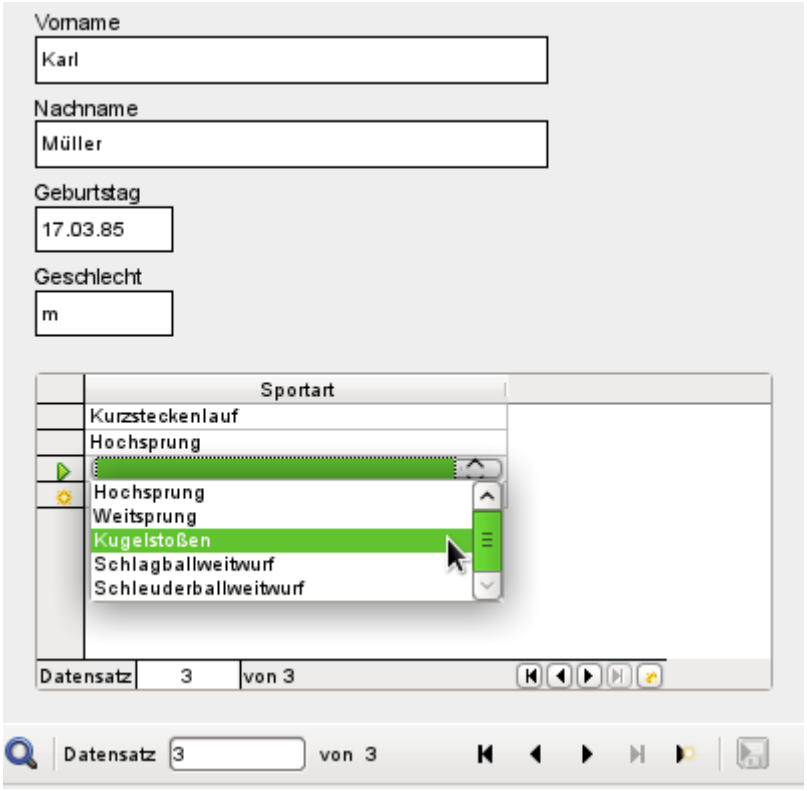
Hinweis	Manchmal kann es vorkommen, dass die Symbolleisten für die Erstellung von Formularen beim Bearbeiten der Formulare nicht korrekt geöffnet werden. Der Formularnavigator ist dann nicht zu finden. Die Symbolleisten lassen sich über **Ansicht → Symbolleisten → Formular-Entwurf** sowie **Formular-Steuerelemente** einblenden. Sollten sie danach auch bei der Dateneingabe sichtbar sein, so muss hier entsprechend wieder die Ansicht ausgeschaltet werden.



Über den Formularnavigator kann jedes Feld einzeln aufgesucht werden. Im Kontextmenü sind dann die Eigenschaften des Feldes aufrufbar. Die Eigenschaften werden nach dem Verlassen der Eigenschaften automatisch gespeichert. Es kann auch bei geöffnetem Eigenschaftsdialog von einem Feld zum anderen gesprungen werden. Hierbei wird auch der jeweilige Zwischenstand gespeichert.

Wenn die Gestaltung nun beendet wurde kann das Formular gespeichert und geschlossen werden. Auch eine Speicherung der Base-Datei ist jetzt wieder sinnvoll.

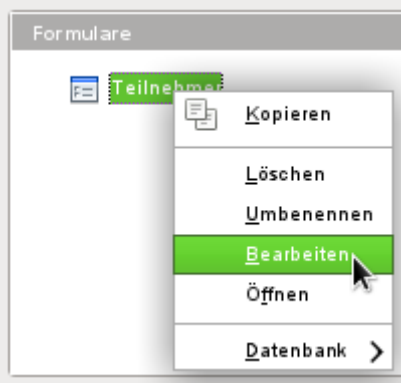
Wird jetzt das Formular zur Eingabe von Daten aufgerufen, dann könnte es ungefähr wie folgt aussehen. In das folgende Formular wurde zum Test ein Datensatz eingetragen. Nach der Eingabe des Geschlechtes wurde erst einmal der Button *Speichern* betätigt. Dann konnten auch Sportarten gewählt werden.



Bei der Nutzung des Formulars fallen einige Unannehmlichkeiten auf:

1. Das Unterformular mit den Sportarten ist nicht direkt erreichbar. Wird mit dem Tabulator navigiert, so springt der Tabulator nach der Eingabe des Geschlechtes direkt zum nächsten Teilnehmerdatensatz.
2. Die Navigationsleiste zeigt, wenn sie im Unterformular steht, die Datensatzzahl des Unterformulars an. Sie sollte besser nur durch das Hauptformular navigieren.
3. Das Geschlecht wird zur Zeit auswendig je nach Benutzervorliebe eingegeben. In der Tabelle ist lediglich die Länge des Feldes auf 1 Zeichen begrenzt worden. Hier muss also eine sicherere Eingabe gewährleistet werden.

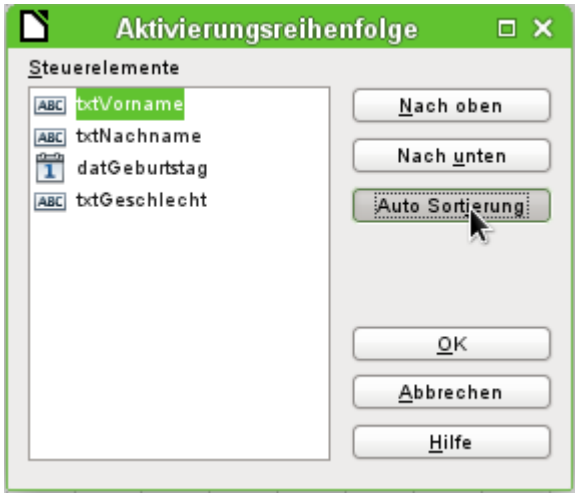
Zur Lösung dieser Probleme muss das Formular nicht für die Dateneingabe, sondern zum Bearbeiten geöffnet werden:



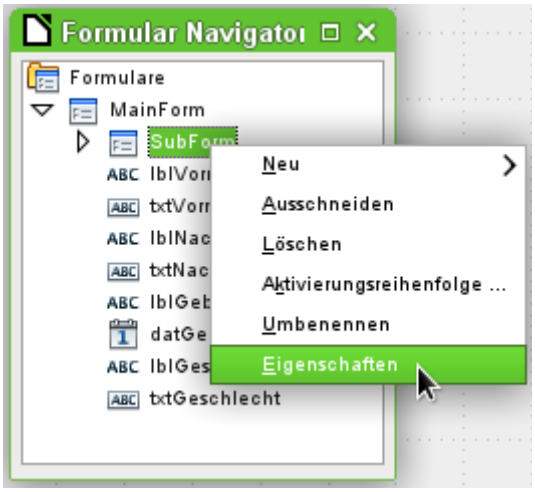
Mit dem Tabulator zum Unterformular

Um mit dem Tabulator nach der Eingabe in das Feld "Geschlecht" nicht gleich weiter zum nächsten Teilnehmer-Datensatz zu rutschen ist einmal die Bearbeitung der Aktivierungsreihenfolge, erreichbar über die Symbolleiste «Formular-Entwurf», notwendig.

Mit der Auswahl *Auto Sortierung* erfolgt nicht nur die Sortierung der angezeigten Steuerelemente, sondern auch die automatische Weiterleitung in das Unterformular. Das ist zwar aus dem Dialog nicht ersichtbar, wird aber im Hintergrund so geregelt.



Navigationsleiste des Hauptformulars auch im Unterformular aktivieren



Der Formularnavigator wird geöffnet. Die Eigenschaften des Unterformulars werden aufgesucht.

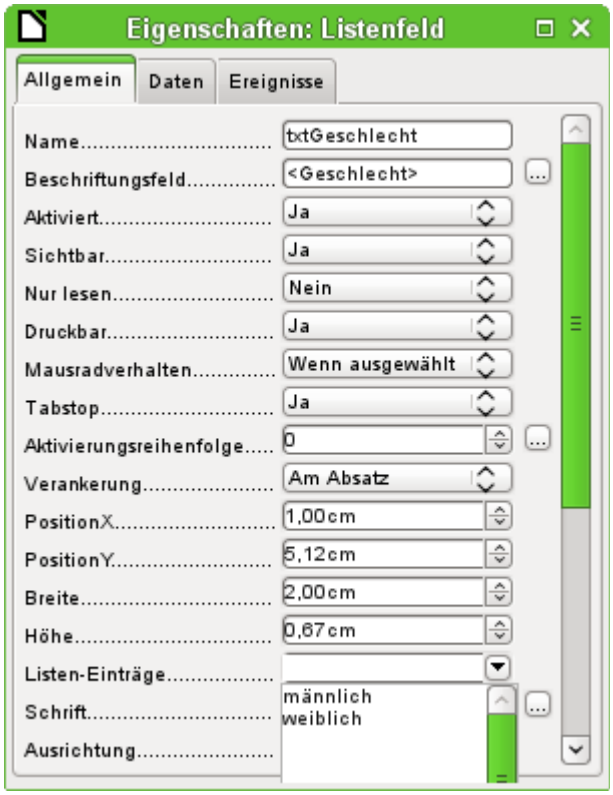
Unter **Daten** → **Symbolleiste Navigation** wird der Wert von «Ja» auf «Übergeordnetes Formular» geändert. Anschließend zeigt die Symbolleiste Navigation immer die Nummer des Datensatzes aus der Tabelle "Teilnehmer" an.

Eingabe eines Kontrollfeldes einschränken

Für die Einschränkung der Eingabe darf das Kontrollfeld kein einfaches Texteingabefeld sein. Im Hauptkapitel Formulare findet sich für die Eingabe des Geschlechtes die Lösung über einen «Gruppierungsrahmen». Hier soll deshalb eine andere Lösung, nämlich noch einmal über ein Listenfeld, vorgestellt werden.

Zuerst wird das Formularfeld für die Eingabe des Geschlechts über den Navigator markiert. Über das mit der rechten Maustaste erreichbare Kontextmenü wird **Ersetzen durch** → **Listenfeld** ausgewählt. Anschließend wird über das Kontextmenü **Eigenschaften** ausgewählt.

Daten → Art des Listeninhalts bleibt auf dem Wert «Werteliste» stehen. In **Daten → Listeninhalt** werden die Kürzel 'm' und 'w' untereinander (erreichbar durch *Shift* + *Eingabetaste*) eingegeben. Diese Kürzel sind die Daten, die an die Tabelle "Teilnehmer" weiter gegeben werden.



In **Allgemein → Listen-Einträge** wird das angegeben, was durch das Listenfeld angezeigt werden soll. Das kann auch 'm' und 'w' sein. Es muss auf jeden Fall die gleiche Reihenfolge haben wie die Einträge unter **Daten**.

Jetzt wird noch unter **Allgemein → Aufklappbar** «Ja» gewählt. Hierzu muss in den allgemeinen Eigenschaften recht weit unten gesucht werden.

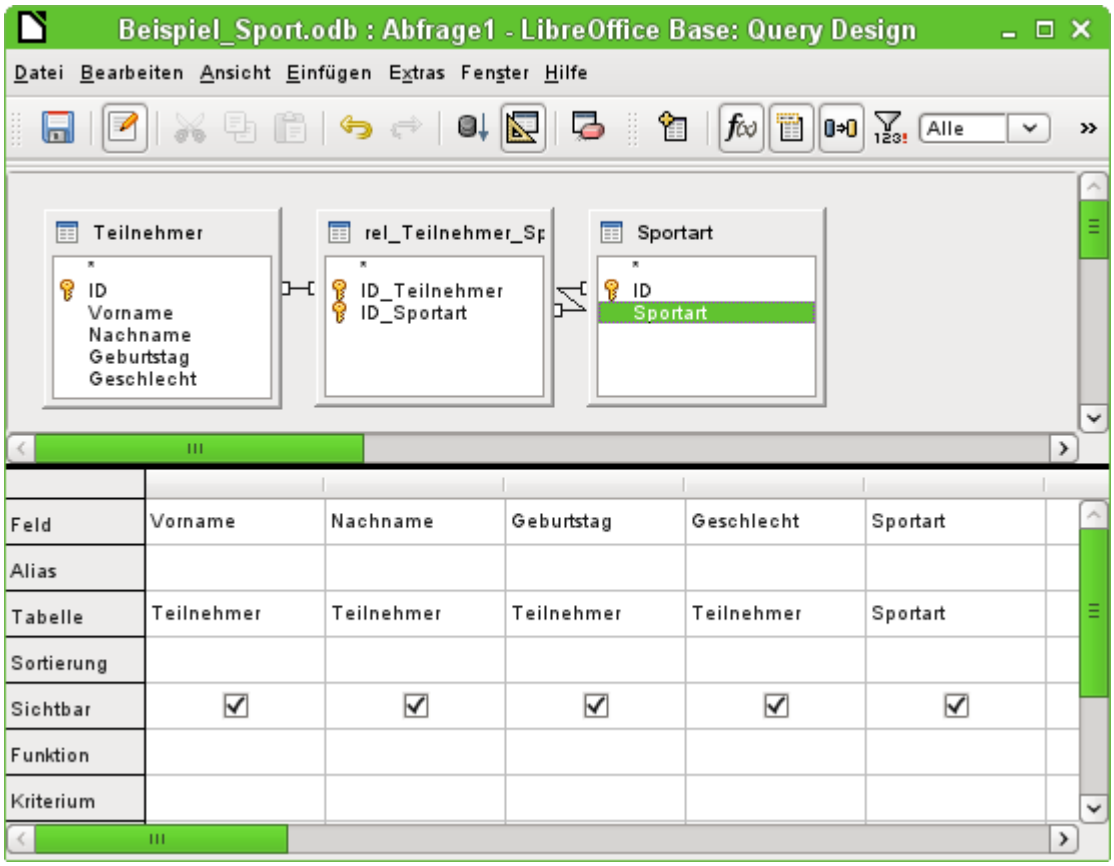
Damit ist das Eingabeformular so weit, dass die auffälligsten Unannehmlichkeiten beseitigt wurden. Das Formular und die Datenbankdatei müssen wieder abgespeichert werden. Die Eingabe von Teilnehmern und Teilnehmerinnen sowie die Zuordnung zu Sportarten kann beginnen.

Damit die folgenden Schritte sinnvoll sind sollten erst einmal einige Datensätze eingegeben werden. Dabei sollte darauf geachtet werden, dass die Teilnehmer und Teilnehmerinnen vom Alter und von der Sportart auch einmal in Konkurrenz zueinander treten können. Sonst sehen die anschließenden Abfragen und Berichte doch recht nichtssagend aus.

Abfrage

In einer Abfrage können z. B. Inhalte verschiedener Tabellen zusammengefasst werden. Mit dem ersten Zugriff sollen erst einmal alle Teilnehmer mit den Sportarten dargestellt werden, die sie belegt haben.

Der Start dazu geht über **Abfragen → Abfrage in der Entwurfsansicht erstellen …** . Es erscheint ein Dialog zur Tabellenauswahl, aus dem wir alle Tabellen für die Abfrage auswählen. Am übersichtlichsten ist es, wenn dabei die Tabelle "rel_Teilnehmer_Sportarten" als zweite Tabelle ausgewählt wird. Dann sind auch gleich die Verbindungen gut zu erkennen.



Die Felder, die in der Abfrage dargestellt werden sollen, können entweder durch einen Doppelklick auf die Feldnamen in den Tabellen oder durch Auswahl in der Zeile «Feld» eingestellt werden. Bei der Auswahl in der Zeile werden die Felder sämtlicher aufgeführter Tabellen aufgelistet. Um die Felder den Tabellen richtig zuordnen zu können werden sie in Abfragen nach "Tabellenname"."Feldname" bezeichnet. Wird statt des Feldnamen ein «*» eingesetzt, so bedeutet dies, dass alle Felder der entsprechenden Tabelle angezeigt werden sollen.

Eine Abfrage sollte grundsätzlich vor dem Abspeichern auch einmal ausgeführt werden um zu sehen, ob sie denn wirklich das erhoffte Ergebnis liefert. Hierzu entweder direkt den oben gezeigten Button anklicken. Es kann auch *F5* gedrückt oder der Weg über **Bearbeiten → Abfrage ausführen** gewählt werden.

Vorname	Nachname	Geburtstag	Geschlecht	Sportart
Karl	Müller	17.03.85	m	Kurzstecken
Karl	Müller	17.03.85	m	Hochsprung
Karl	Müller	17.03.85	m	Kugelstoßer
Mia	Keller	07.04.78	w	Kurzstecken
Mia	Keller	07.04.78	w	Weitsprung
Mia	Keller	07.04.78	w	Kugelstoßer
Dirk	Anders	28.09.87	m	Kurzstecken
Dirk	Anders	28.09.87	m	Langstrecke
Dirk	Anders	28.09.87	m	Weitsprung

Datensatz 1 von 26

Die Abfrage zeigt jetzt alle Kombinationen der Teilnehmer und Sportarten an. Hat ein Teilnehmer mehrere Sportarten belegt, so werden mehrere Datensätze angezeigt. Teilnehmer ohne Sportarten erscheinen nicht.

Um bei einem Wettkampf unterschiedliche Altersgruppen zusammen zu stellen ist der Geburtsjahrgang entscheidend. Es kommt darauf an, wie alt eine Person in dem jeweiligen Jahr wird.

Der Geburtsjahrgang kann mit unterschiedlichen Funktionen ermittelt werden. Die folgende Eingabe zeigt die Variante, die für Personen mit Englischkenntnissen vielleicht auch so erschließbar ist:

Feld	Vorname	Nachname	Geburtstag	Geschlecht	Sportart	YEAR(NOW( )) - YEAR("Geburtstag")
Alias						
Tabelle	Teilnehmer	Teilnehmer	Teilnehmer	Teilnehmer	Sportart	

YEAR(NOW())-YEAR("Geburtstag")

`NOW()` steht für «Jetzt», also den aktuellen Zeitpunkt. Mit **`YEAR(NOW())`** wird also das aktuelle Jahr ausgelesen. **`YEAR("Geburtstag")`** liest das Jahr aus dem Geburtstagsdatum aus. Aus beiden wird hier eine Differenz gebildet, die angibt, wie alt diese Person in dem aktuellen Jahr wird oder bereits geworden ist.

Diese und viele andere Funktionen, die mit der eingebauten HSQLDB zusammen arbeiten, sind im Anhang des Handbuches beschrieben.

Vorname	Nachname	Geburtstag	Geschlecht	Sportart	YEAR(NOW( )) - YEAR("Geburtstag")
Karl	Müller	17.03.85	m	Kurzstecken	29
Karl	Müller	17.03.85	m	Hochsprung	29
Karl	Müller	17.03.85	m	Kugelstoßer	29
Mia	Keller	07.04.78	w	Kurzstecken	36
Mia	Keller	07.04.78	w	Weitsprung	36
Mia	Keller	07.04.78	w	Kugelstoßer	36
Dirk	Anders	28.09.87	m	Kurzstecken	27
Dirk	Anders	28.09.87	m	Langstrecke	27
Dirk	Anders	28.09.87	m	Weitsprung	27

Datensatz 1 von 26

Wird die Abfrage ausgeführt, so erscheint die entsprechende Berechnung für das Jahr 2014, da diese Abfrage eben in dem entsprechenden Jahr ausgeführt wurde.

Etwas störend dürfte dabei sein, dass der gesamte Code, den wir als Feld eingegeben haben, eben auch in der Spaltenüberschrift erscheint. Dem wird abgeholfen, indem für den Code ein alternativer Name eingegeben wird, unter dem das Ergebnis erscheinen soll.

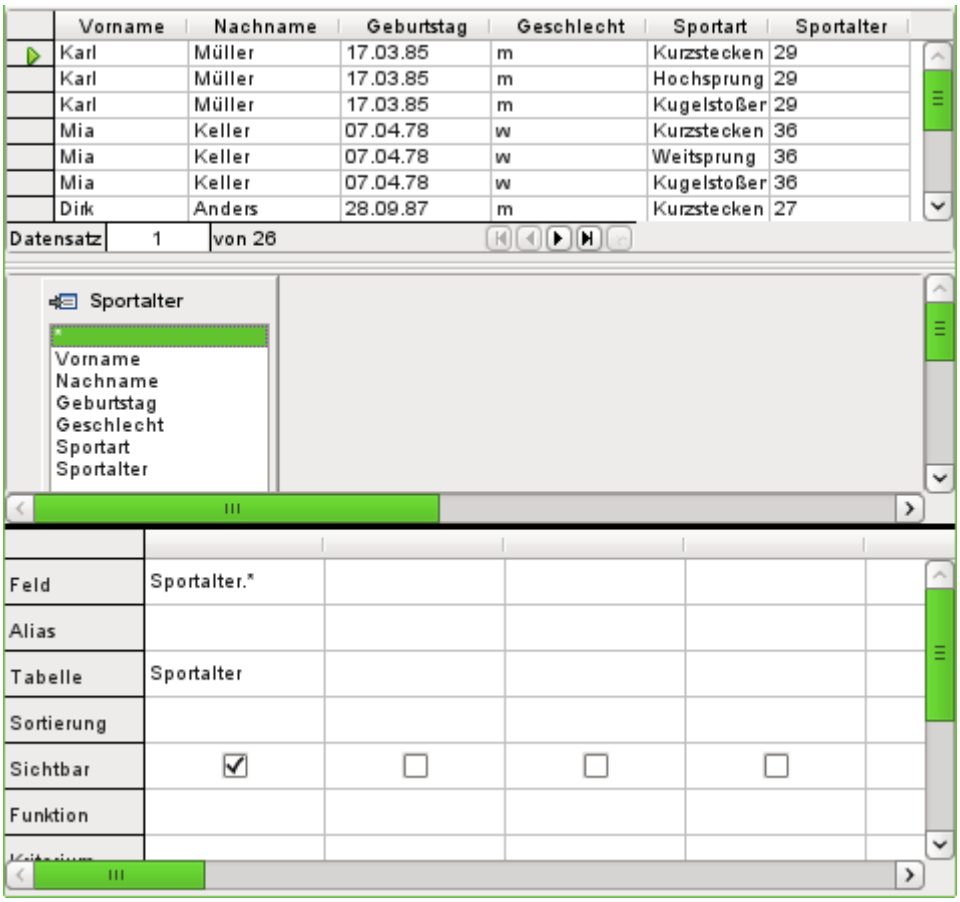
Feld	YEAR(NOW( )) - YEAR("Geburtstag")
Alias	Sportalter
Tabelle	

In der Zeile «Alias» wird die Bezeichnung «Sportalter» eingegeben. Hier sollte nicht «Alter» stehen, weil die Person vielleicht erst im Laufe des Jahres dieses Alter erreicht. Im Kapitel «Aktuelles Alter ermitteln» steht beschreiben, wie das tatsächliche Alter ermittelt werden kann.

Sportart	Sportalter
Kurzstecken	29
Hochsprung	29
Kugelstoßer	29
Kurzstecken	36
Weitsprung	36

Wird die Abfrage erneut ausgeführt, so erscheint in dem Spaltenkopf nicht mehr die Formel sondern die Bezeichnung «Sportalter».

Damit der Code für den Anfang nicht zu verwirrend wird sollte die Abfrage unter dem Namen "Sportalter" abgespeichert werden. Diese Abfrage dient dann als Grundlage für die nächste Abfrage, die das "Sportalter" jetzt einer "Altersklasse" zuordnet.

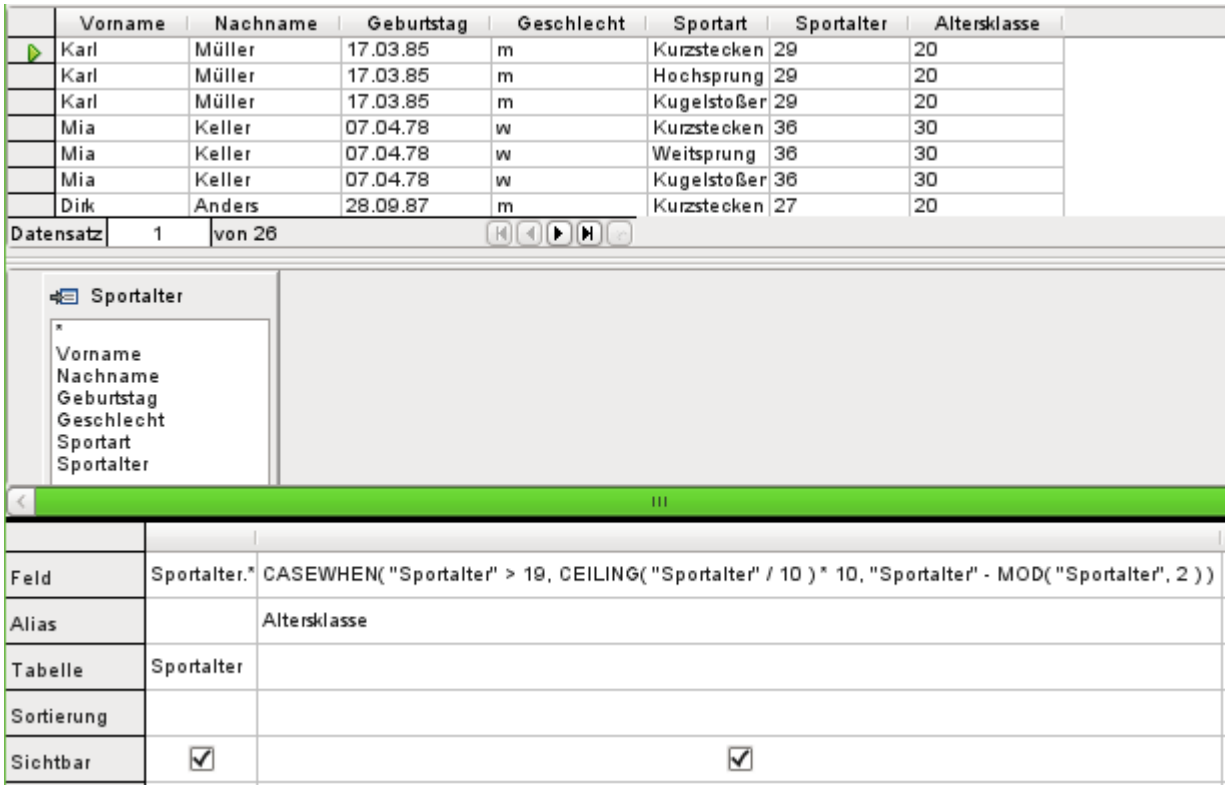


Die Abfrage "Sportalter" wurde als Grundlage für die 2. Abfrage ausgesucht. Vor der Bezeichnung «Sportalter» im Tabellencontainer ist ein anderes Symbol zu sehen als bei den Tabellen der ersten Abfrage. Dieses Symbol zeigt an, dass die Grundlage dieser Abfrage eben eine Abfrage und keine Tabelle ist.

Durch Doppelklick auf «*» oder Auswahl von «Sportalter.*» werden alle Felder ausgewählt. Die gesamte Abfrage von vorher wird also übernommen. Nur die Formel ist nicht mehr zu sehen.

Auf das Feld "Sportalter" soll nun zugegriffen werden. Das Sportalter bestimmt ja, in welcher Altersklasse die jeweiligen Personen starten.

Damit die Ermittlung nicht allzu kompliziert wird soll hier folgende Zuordnung nachvollzogen werden: Unter 20 Jahren werden die Teilnehmer immer in zwei Jahrgängen, beginnend mit 0, zusammen gefasst. Ab 20 Jahren werden Gruppen für jeweils 10 Jahre gebildet, also von 20-30 Jahren usw.



Solche Formeln gehören eigentlich längst nicht mehr zum Einstiegs-Repertoire. Eine einfachere Zuteilung der Altersklassen wäre sogar mit Hilfe des folgenden Berichtes möglich. Für differenziertere Zuteilung sei erneut auf den Anhang des Handbuches verwiesen.

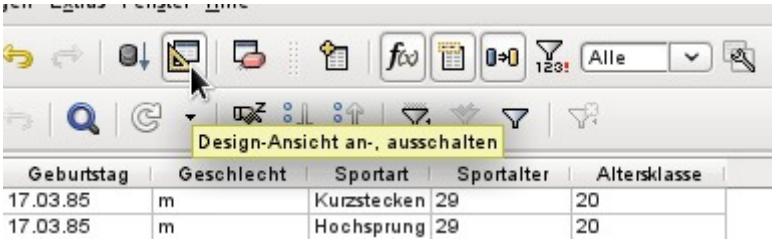
```
CASEWHEN( "Sportalter" > 19, CEILING( "Sportalter" / 10 ) * 10, "Sportalter" -
MOD( "Sportalter", 2 ) )
```

CASEWHEN(Überprüfung, korrekt: diesen Wert, nicht korrekt: den anderen Wert).
In die deutsche Sprache übertragen bedeutet das so viel wie:
Für den Fall, dass das Sportalter über 19 Jahren ist, dividiere das Sportalter durch 10 und gib nur den Ganzzahlwert wieder. Diesen Ganzzahlwert multipliziere wieder mit 10.
Ist das Sportalter unter 19 Jahren, so nehme den Wert des Sportalters und subtrahiere davon den Rest, der sich ergibt, wenn das Sportalter durch 2 dividiert wird.

Alle Teilnehmer über 19 Jahren werden damit einer Gruppe für das jeweilige Jahrzehnt zugewiesen. Alle Teilnehmer darunter werden einer Gruppe zugewiesen, die der jeweiligen niedrigeren geraden Zahl entspricht.

Dieser Formel wurde wieder ein Alias zugewiesen, in diesem Falle der Alias "Altersklasse".

Die Abfrage kann jetzt unter dem Namen "Meldung" gespeichert werden.



Das Ausschalten der Design-Ansicht ist hier eigentlich nicht notwendig, da alle Eingaben in der Design-Ansicht ohne größere Probleme möglich sind. Den Code von Abfragen einmal genauer anzusehen kann allerdings nie schaden. Schließlich gibt es auch SQL-Formulierungen, die nur schlecht in die Design-Ansicht passen oder die dort gar nicht möglich sind. Dafür wird dann die direkte Eingabe im SQL-Code genutzt.

Im Code sind Felder und Tabellen in doppelte Anführungszeichen gesetzt und in einem ockerfarbigen Farbton dargestellt. Begriffe des SQL-Codes sind in blau gehalten, Funktionen der Datenbank in grün.

```
SELECT
    "Sportalter".*,
    CASEWHEN( "Sportalter" > 19, CEILING( "Sportalter" / 10 ) * 10, "Sportalter" -
        MOD( "Sportalter", 2 ) ) AS "Altersklasse"
FROM "Sportalter"
```

Der Teil mit der Formel wurde ja schon erwähnt. Hier jetzt der Rahmen.

```
SELECT "Sportalter".*, ...  AS "Altersklasse" FROM "Sportalter"
```

Wähle aus der Tabelle "Sportalter" alle Datensätze aus und zusätzlich das, was durch die Formel bestimmt wird. Das, was durch die Formel bestimmt wird, bezeichne als "Altersklasse".

Der Code unterscheidet nicht zwischen Tabellen und Abfragen als Grundlage der Daten. Er funktioniert deshalb nur in der grafischen Benutzeroberfläche von Base. Die Benennung von Abfragen muss dafür immer anders sein als die Benennung von Tabellen.

Sobald in einer Abfrage bei ausgeschalteter Design-Ansicht der Button *SQL* (für **SQL-Kommando direkt ausführen**) gedrückt wird, wird die Datenbank mit einer Fehlermeldung reagieren. Die eingebettete HSQLDB kennt die Abfrage "Sportalter" nicht, auf die sich die aktuellen Abfrage "Meldung" bezieht.

Ausdruck im Bericht

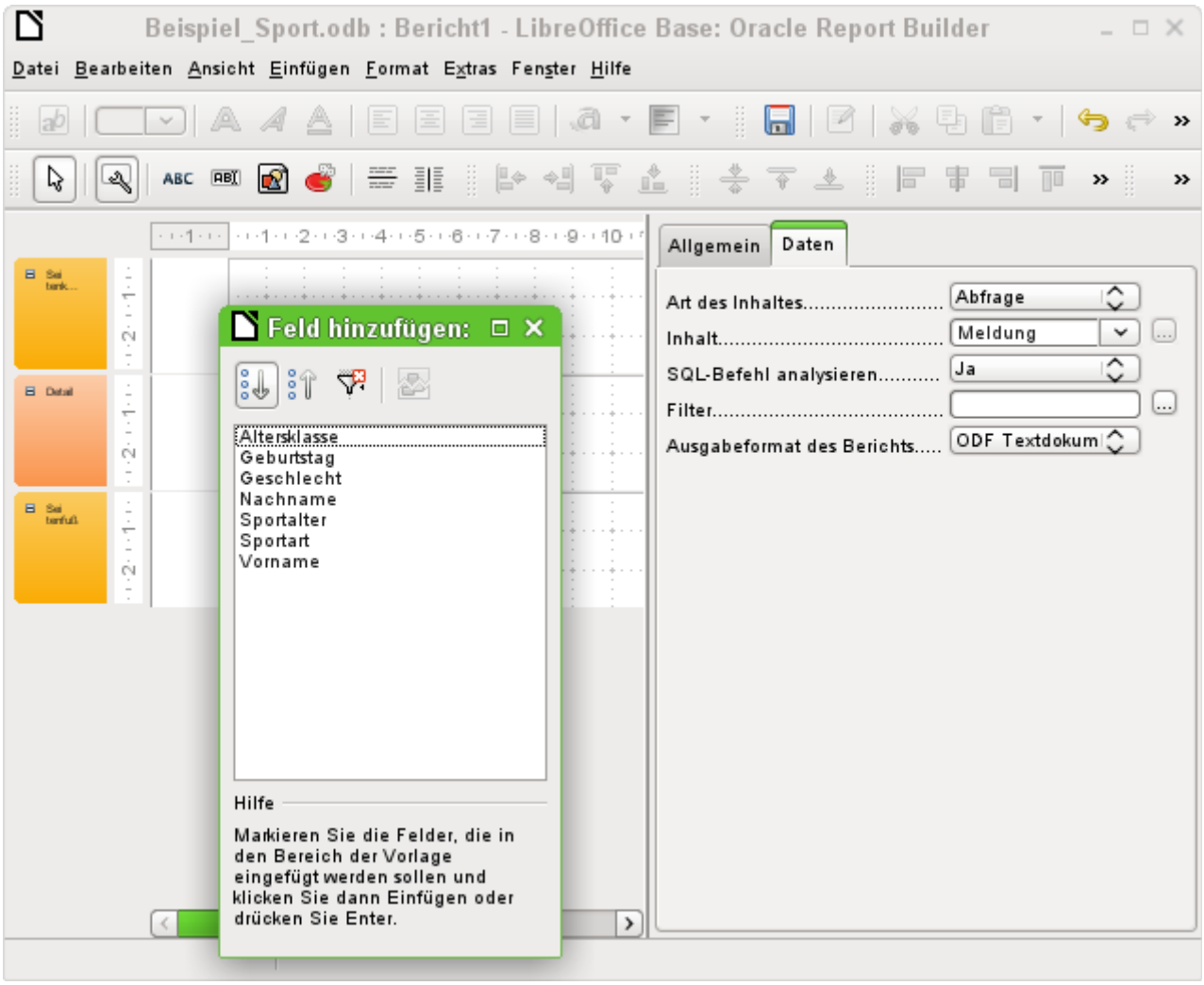
Über **Berichte → Bericht in der Entwurfsansicht erstellen …** wird ein Bericht erstellt, der eine übersichtliche Teilnehmerliste, sortiert nach Sportart, Geschlecht und Altersklasse darstellen soll.

Wird der Editor gestartet, so ist zuerst der Dialog «Feld hinzufügen» im Vordergrund. Für diesen Dialog wird die von der alphabetischen Sortierung her erste Tabelle als Basis genommen. Es muss für den Bericht aber die Abfrage als Datenquelle herausgesucht werden.

Im Berichtsfenster befindet sich auf der rechten Seite die Übersicht über die Eigenschaften des gerade aktiven Objektes. Sollte die Leiste nicht sichtbar sein, so muss sie über **Ansicht → Properties** ausgewählt werden.

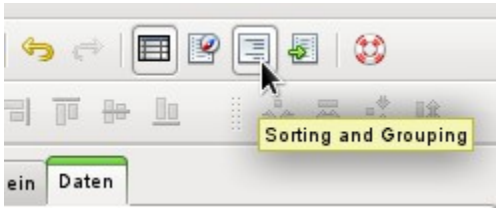
Hinweis	Leider sind beim Report-Builder bereits auf der grafischen Benutzeroberfläche in den neueren Ausgaben von LibreOffice einige Elemente nicht mehr übersetzt. An dieser Stelle ist wohl zu viel an Code bereinigt worden. Formeln, wie weiter unten erwähnt, waren im Report-Builder allerdings nie übersetzt sondern immer in der englischsprachigen Originalfassung zu erstellen.

Aus den Eigenschaften wird im Reiter Daten → Art des Inhaltes → Abfrage ausgewählt. Als Inhalt wird dann die Abfrage "Meldung" gewählt, die als letzte zusammenfassende Abfrage erstellt wurde.

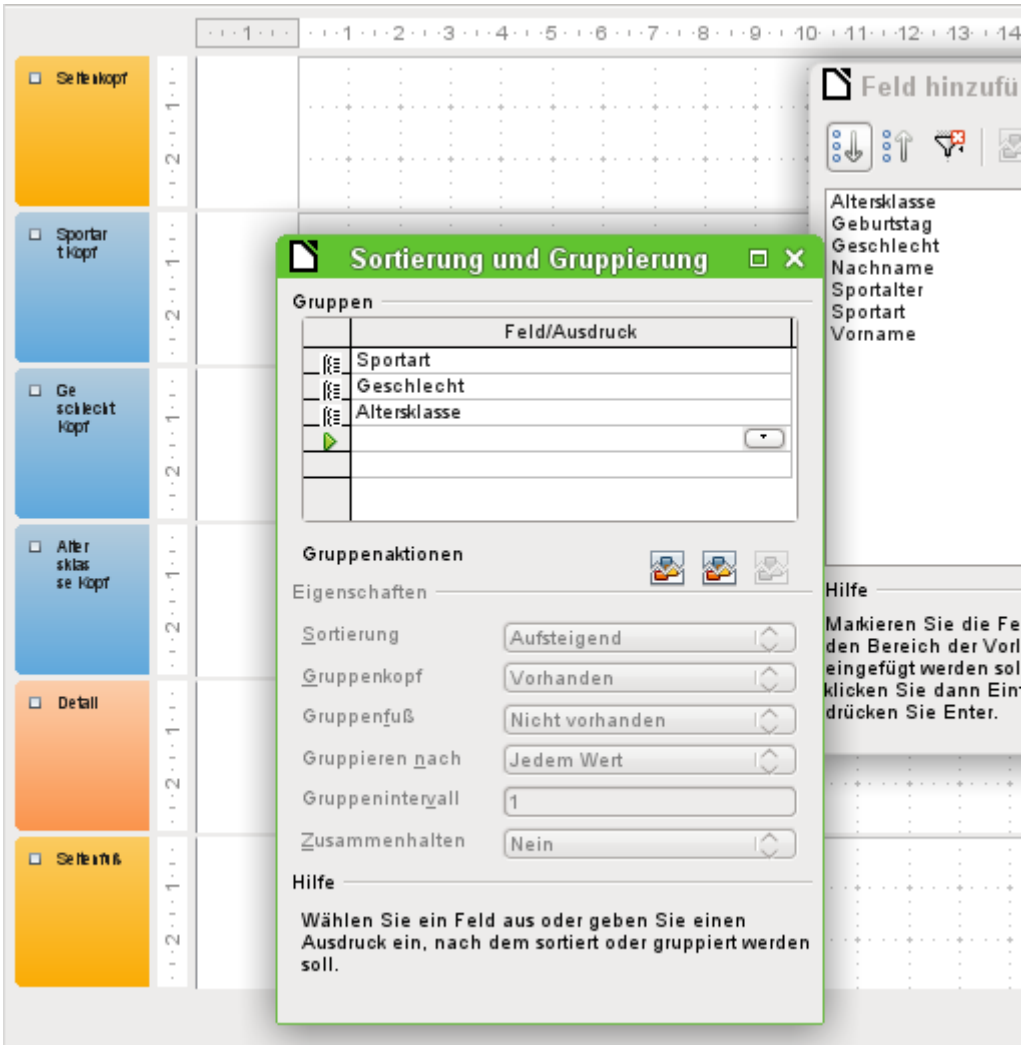


Der Report-Builder zeigt jetzt sämtliche Felder der ausgesuchten Abfrage an.

Im nächsten Schritt wird der Sortier-Dialog ausgewählt:



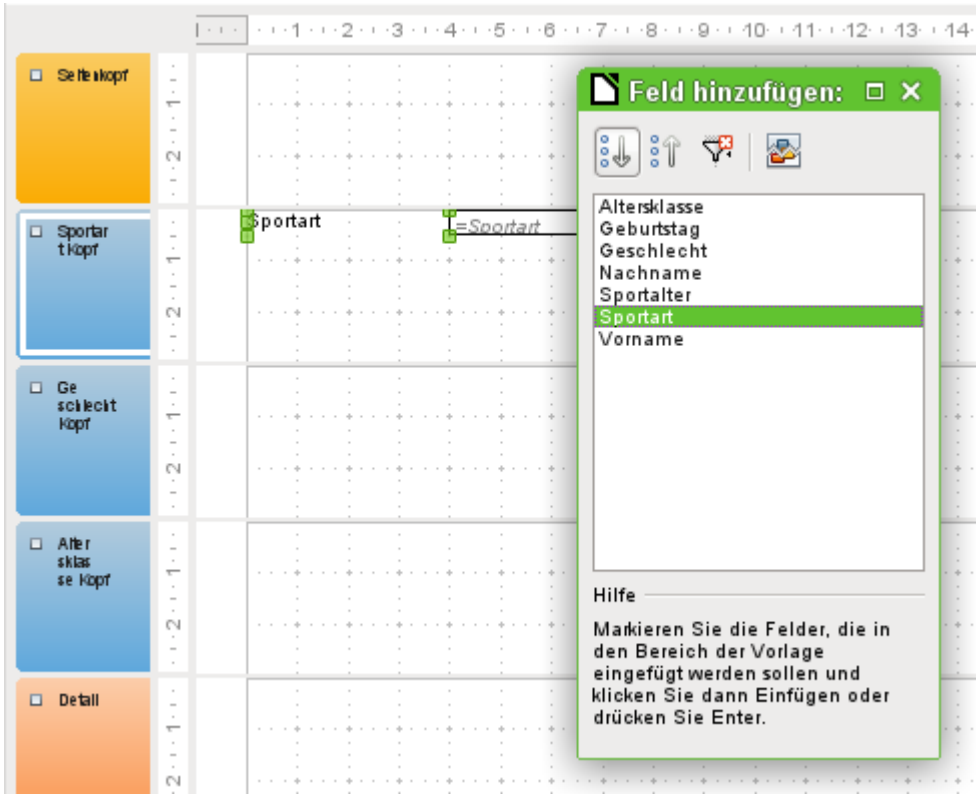
Die Sortierung kann auch über **Ansicht** → **Sorting and Grouping** angezeigt werden.



Im Sortierungs- und Gruppierungs-Dialog werden nacheinander die Felder "Sportart", "Geschlecht" und "Altersklasse" ausgewählt.

Auf der linken Seite des Berichtes erscheint bei jeder Sortierungsauswahl ein Gruppenkopf für die Sortierung. Die Eigenschaften der Sortierung und Gruppierung werden einfach erst einmal so gelassen wie im Standard vorgesehen.

Der Dialog «Sortierung und Gruppierung» kann nun geschlossen werden.

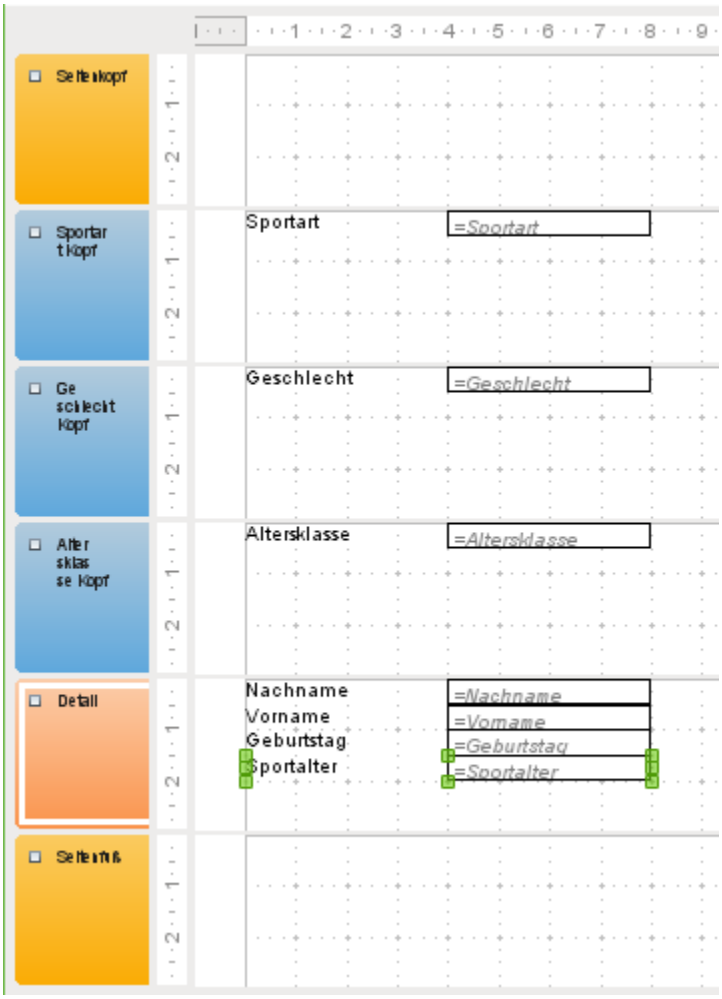


Der Gruppenkopf «Sportart Kopf» wird ausgewählt. Dies ist an der weißen Umrandung des Kopfes sichtbar. Anschließend wird im Dialog «Feld hinzufügen:» mit einem Mausklick auf das Feld «Sportart» ein Beschriftungsfeld und ein Textfeld eingefügt, das den Inhalt des Feldes «Sportart» aufnehmen soll.

Mit dem gleichen Verfahren werden die Felder «Geschlecht» und «Alter» den passenden Gruppenköpfen zugewiesen.

Sämtliche verbleibenden Felder werden in den Bereich «Detail» eingefügt.

Der Berichtsentwurf dürfte jetzt wie folgt aussehen:



Anschließend sollte der Bericht erst einmal abgespeichert werden. Als Name würde sich so etwas wie «Teilnehmerliste» anbieten.

Auch die Datenbankdatei selbst sollte abgespeichert werden, da sonst die Speicherung des Berichtes erst einmal nur temporär vorgenommen wird.

Hinweis	Gerade beim Gestalten eines Berichtes mit dem Report-Builder fallen oft Instabilitäten des Programms auf. Ist ein wichtiger Schritt gelungen so sollte auf jeden Fall sowohl der Bericht als auch die Datenbankdatei gespeichert werden. Die spätere Ausführung eines Berichtes ist von diesen Instabilitäten zum Glück nicht betroffen.

Wird dieser Bericht mit entsprechenden Daten ausgeführt, dann gibt sich in etwa das folgende Bild:

Der Beginn des Berichtes zeigt hier zwei männliche Teilnehmer an, die Hochsprung machen wollen und der Altersklasse 20 angehören.

Bei der Ausführung des Berichtes fallen einige Designmängel auf:

1. Die Abstände zwischen den Gruppenköpfen und dem Inhalt des Detailbereichs ist viel zu groß.
2. Das Geschlecht wird mit «m» bzw. «w» angegeben. Besser wäre eine Bezeichnung wie «Herren» und «Damen».
3. Bei der Altersklasse steht, vermutlich wegen der Division in der Abfrage, eine Zahl mit einer Nachkommastelle, wobei als Dezimaltrenner ein Punkt gesetzt wird.
4. Die Bezeichner für die dargestellten Felder aus dem Bereich «Detail» («Nachname», «Vorname» usw.) sollten besser gemeinsam für alle Inhalte des Bereiches als Tabellenköpfe über den Inhalten stehen.

Abstände zwischen den Berichtsbereichen einstellen

Die Abstände zwischen den Bereichen können direkt an den Tabellenköpfen durch Schieben mit der Maus verringert werde, Es ist auch möglich, einen Bereich zu markieren und in den Eigenschaften des Gruppenkopfes die Höhe über eine Zahleneingabe zu regeln. Dafür darf kein Feld, sondern nur der Gruppenkopf markiert sein.

Es ist nicht möglich, einen Bereich kleiner zu wählen als die darin existierenden Beschriftungs- und Textfelder sind.

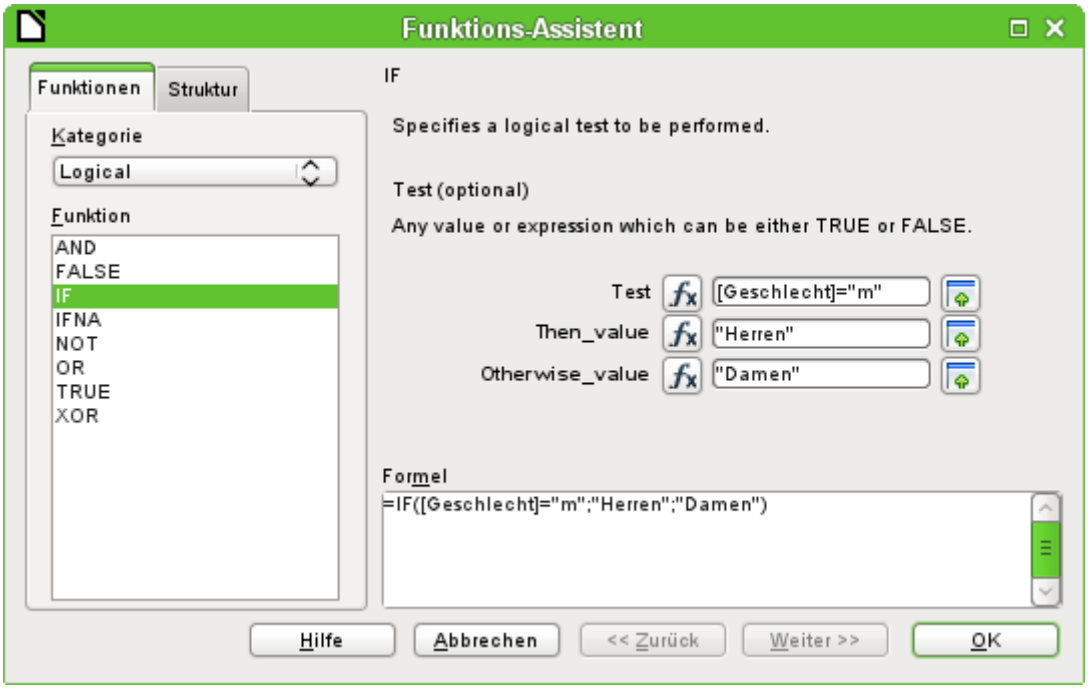
Der Bereich «Altersklasse sollte dabei so groß gewählt werden, dass er noch die Beschriftungsfelder aus dem Bereich «Detail» aufnehmen kann.

Außerdem muss bei der Auswahl der Abstände berücksichtigt werden, dass eine folgende Gruppe nicht zu nah unter der vorhergehenden Gruppe erscheint. Die Beschriftungs- und Textfelder benötigen gegebenenfalls auch einen Abstand nach oben. Wird dieser Abstand nach oben nicht gewünscht, so kann damit nachgeholfen werden, dass zusätzlich zu den Gruppenköpfen auch Gruppenfüße angezeigt werden. Eine solche Einstellung ist über **Ansicht → Sorting and Grouping** für jede Gruppe möglich.

Beeinflussung eines Textfeldinhaltes durch eine Formel

Die Bezeichnung des Geschlechtes aus der Tabelle reicht für die Ansprüchen einer Teilnehmerliste nicht aus. Es könnte eine entsprechende Umbenennung bereits in der Abfrage erfolgen. Da die Abfrage aber bereits erstellt wurde werden jetzt die Möglichkeiten des Berichtes genutzt.

Das Textfeld «=Geschlecht» wird markiert. In den Eigenschaften auf der rechten Seite des Report-Builders wird der Reiter **Daten** → **Datenfeld** aufgesucht. Dort wird der Button mit den 3 Punkten … betätigt. Es erscheint der Funktions-Assistent. Alle Funktionen sind hier allerdings nicht übersetzt sondern in englischer Sprache erreichbar.



In der Kategorie «Logical» steckt die Funktion «IF». Die Vorgehensweise bei dieser Funktion ist gleich der Funktion «WENN» aus LO-Calc.

Ein Testwert wird eingegeben. Dabei ist das Feld der Abfrage, aus der die Daten ausgelesen werden, in eckige Klammern zu setzen. Texte müssen in doppelte Anführungszeichen gesetzt werden. Wenn im Feld "Geschlecht" der Wert «m» steht, dann soll in dem Feld des Berichtes «Herren» angezeigt werden. Steht dort kein «m», dann wird «Damen» angezeigt.

Die Eingabe wird mit OK bestätigt.

Damit ist die Formulierung des Feldes entsprechend umgestellt.

Umstellung der Formatierung eines Textfeldes

Das Feld, das den Inhalt der Datenbank aufnehmen soll, wird zwar in Berichten als «Textfeld» bezeichnet. Es kann aber genau so formatiert werden wie Felder in Tabellen im Writer oder in Calc.

Das Feld «=Altersklasse» wird markiert. Bei den Eigenschaften am rechten Bildschirmrand wird **Allgemein** → **Formatierung** aufgesucht. Die Formatierung steht dort auf «Text». Der Button mit den 3 Punkten (…) wird gedrückt. Es öffnet sich der Formatierungs-Dialog, der auch aus Calc, dem Writer oder bei Formularerstellungen bekannt ist. Hier wird **Kategorie** → **Zahl** ausgewählt und mit OK bestätigt.

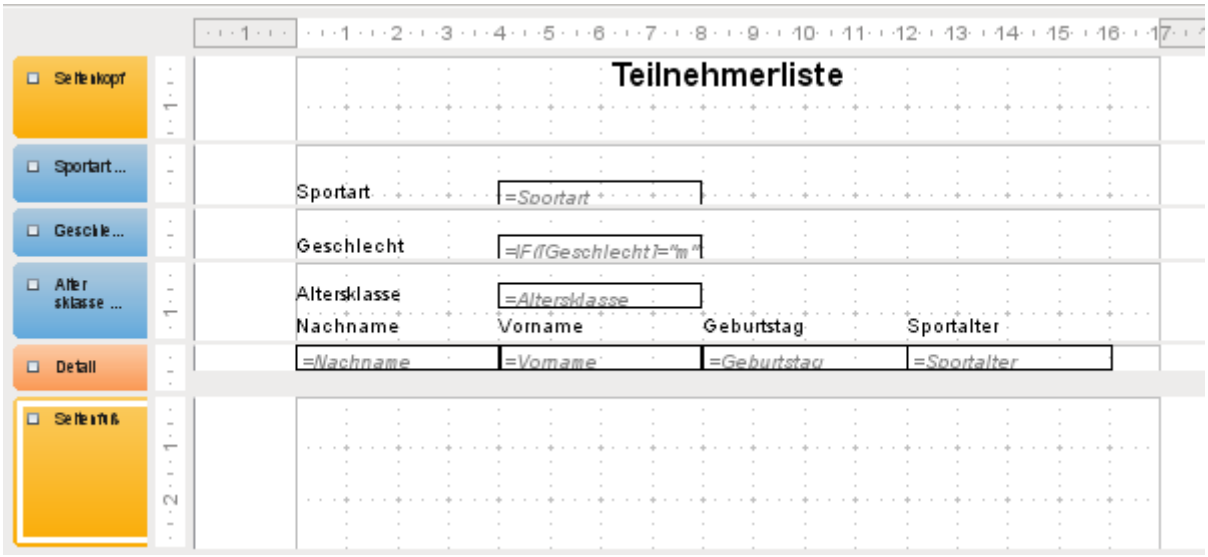
Die Formatierung der Anzeige der ist nun von Text auf Zahl umgestellt.

Verschieben von Feldern im Report-Builder

Felder können beim Report-Builder auch über die Grenzen eines Bereiches hinweg verschoben werden. Im Zielbereich muss allerdings genügend Platz für das Feld vorhanden sein. Es kann nämlich nicht ein Feld über einem anderen liegen.

Die Positionierung mit der Maus ist je nach System sogar so ungenau, dass am besten reichlich Platz vorhanden sein sollte. Sind erst einmal die Felder in einem Bereich wie hier dem Gruppenkopf «Altersklasse», so lassen sie sich auch mit den Pfeilfeldern und mit Tastatureingaben genau positionieren.

Zur Positionierung mit der Tastatur ist in den Eigenschaften **Allgemein → Position X** und **Allgemein → Position Y** auf zu suchen.



Hier wurde noch ein Beschriftungsfeld für die Überschrift hinzugefügt. Die Eingabe des Textes für das Beschriftungsfeld erfolgt in den Eigenschaften des Feldes.

Der Seitenfuß wurde in den Eigenschaften auf **Sichtbar → Nein** eingestellt. Er nimmt sonst am unteren Rand des Dokumentes zu viel Platz weg. Hierbei sollte bedacht werden, dass die verfügbare Höhe bereits durch die Größe der Seitenränder zusätzlich verringert wird. Standardmäßig sind alle Seitenränder bei einer DIN-A4-Seite auf 2 cm eingestellt.

Weitere Formatierungsmöglichkeiten für den Bericht werden in dem gesonderten Kapitel «Berichte mit dem Report_Designer» beschreiben.

Erweiterungen der Beispieldatenbank

Das vorgestellte Beispiel ist natürlich nur der erste Schritt für eine Datenbank im Sportbereich. Zuerst einmal sollte auf jeden Fall ein Feld ergänzt werden, das die jeweiligen Leistungen aufnimmt. Solch ein Feld wäre gut in der Tabelle "rel_Teilnehmer_Sportart" unter zu bringen.

Vielleicht käme noch die Vereinszugehörigkeit hinzu.

Dann muss natürlich, wie bei allen Wettbewerben, ermittelt werden, wer denn nun in der jeweiligen Altersklasse auf welcher Platzierung gelandet ist. Sortierung ist hier gefragt, die vielleicht wieder als Bericht in einer Ergebnisliste landet.

Schön wäre dann, wenn jeder Teilnehmer (wieder über einen Bericht) eine schön gestaltete Urkunde mit den persönlichen Leistungen und der Platzierung erhält.

Solche Erweiterungen sind ohne weiteres möglich. Das sollen die kommenden Kapitel des Handbuches zeigen.