



Handreichung

(Arbeitsversion vom 25.04.2017,
letzte Änderung am 15.07.2020)

Relationale Datenbanken mit LibreOffice (ab Version 4.1.0.4)

Bernhard-von-Cotta-Gymnasium
Brand-Erbisdorf

Autor: Kaden, Frank
eMail: frank.kaden@t-online.de
FB Informatik

Inhaltsverzeichnis

1. Vorbemerkungen	3
2. Vorstellung der Beispieldatenbank.....	3
3. Anlegen der Datenbank	4
4. Erstellung von Tabellen.....	4
5. Erstellung von Formularen und Unterformularen	6
6. Erstellung von Berichten	8
7. Erstellung von Abfragen mit SQL.....	9
8. Makros und Makrosicherheit	10
9. Startformular erstellen.....	14
10. LibreOffice-Datenbank mit Makro speichern und schließen.....	15
11. Bilder verwalten in LibreOffice	15
12. Webseiten öffnen	15
13. Listenfelder	15
14. Konvertierungen.....	16
15. Quellen.....	17

1. Vorbemerkungen

Diese Handreichung erläutert die Herstellung eines relationalen Datenbanksystems (DBS), welches aus einer Datenbank (DB) und einem Datenbankmanagementsystem besteht (DBMS).

Die Datenbank sollte in einem Ordner abgespeichert werden, auf dem volle Lese- und Schreibrechte existieren. Bei der Nutzung von Makros sollte die Makrosicherheit möglichst tief eingestellt werden und der Makrospeicherort sollte auf den Speicherort der Datenbank zeigen (einstellbar über Extras, Optionen, LibreOffice, Sicherheit,...).

Das vorliegende Script beschreibt in einzelnen Schritten den Datenbankentwurf einer relationalen Datenbank. Weiterhin werden Hinweise zur Implementation der Datenbank in LibreOffice gegeben. Die ersten Schritte beinhalten die Erfassung des Realweltausschnittes und die Modellierung der Datenbank mit Hilfe eines Entity-Relationship-Diagrammes (ERD). Im ERD sind die Entitäten (Tabellen), deren Beziehungen sowie die Attribute erkennbar. Die Primärschlüssel wurden hervorgehoben.

2. Vorstellung der Beispieldatenbank

Datenbank ASTRONOMIE, Dateibezeichnung: *lb_lib_astronomie6.odt*

Der Aufbau der DB erfolgt nach folgendem **ERD**:

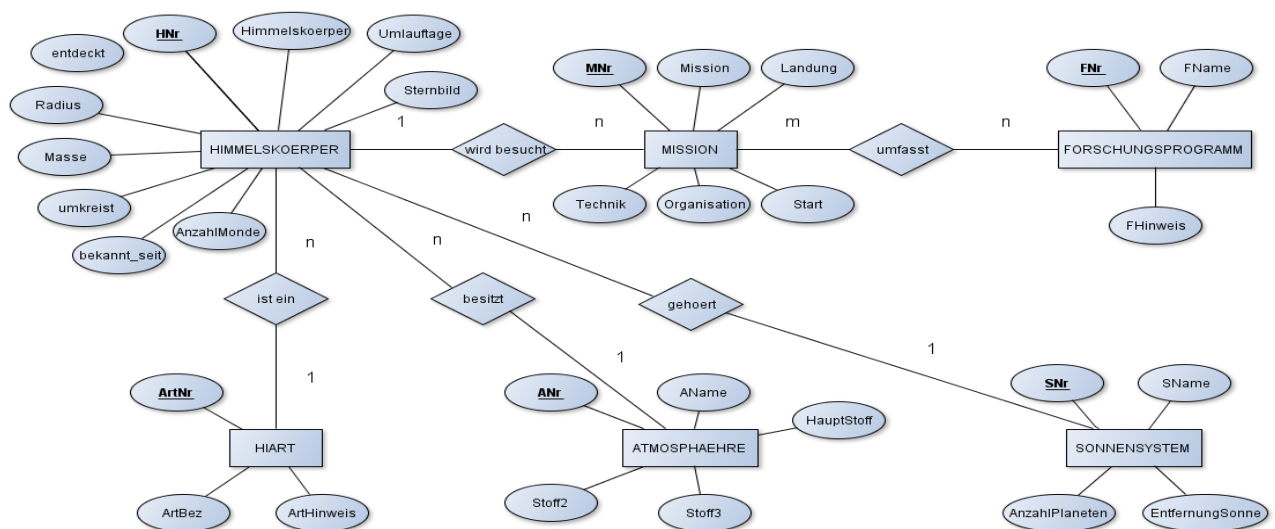


Abbildung 1

CODDsches Relationenmodell:

Modellaufbau: ENTITÄT(Primärschlüssel, Attribute, .. , Fremdschlüssel)

HIMMELSKOERPER (HNr, Himmelskoerper, Umlaufage, entdeckt, Radius, Masse, umkreist, bekannt_seit, AnzahlMonde, Sternbild, ArtNr, ANr, SNr)

MISSION (MNr, Mission, Landung, Technik, Organisation, Start, HNr)

FORSCHUNGSPROGRAMM (FNr, FName, FHinweis)

ATMOSPHAERE (ANr, AName, HauptStoff, Stoff2, Stoff3)

SONNENSYSTEM (SNr, SName, AnzahlPlaneten, EntfernungSonne)

HIART (ArtNr, ArtBez, ArtHinweis)

UMFASST (UNr, MNr, FNr)

3. Anlegen der Datenbank

Wir starten LibreOffice und wählen das Tool Datenbank. Hier nutzen wir den DB-Assistenten. Drei Möglichkeiten stehen zur Auswahl:

- eine neue DB erstellen
- eine vorhandene LibreOffice-DB öffnen
- eine DB aus einem anderen Management-System öffnen, z.B. eine Access-DB

Anschließend wird festgelegt, ob die Datenbank registriert (unter LibreOffice angemeldet) und zum Bearbeiten geöffnet oder eine neue Tabelle eingefügt werden soll.

Damit wird das DBMS geöffnet.

Beim Anlegen der DB kann es zu Problemen mit der Laufzeitumgebung (LZU) der virtuellen JAVA-Maschine kommen. Dazu wird wie folgt verfahren:

- im DBMS im Menü *Extras* den Befehl *Optionen* auswählen
- hier im Ordner *LibreOffice* auf *Erweitert* klicken
- Unter *JAVA-Optionen* warten, bis eine LZU gefunden wird und die neueste Version auswählen bzw. es muss erst eine LZU installiert werden
- mit *OK* bestätigen

Sind Makros in der DB vorhanden, müssen weitere Einstellungen einmalig vorgenommen werden. Dazu werden im Kapitel **8. Makros und Makrosicherheit** Hinweise gegeben.

4. Erstellung von Tabellen

Wir erstellen die Tabelle HIMMELSKOERPER. Die Abbildungen zeigen den Tabellenentwurf (links) und die Tabelle selbst mit einigen Datensätzen. Die laut ERD noch fehlenden Attribute können später ergänzt werden (Abb. 2 und 3).

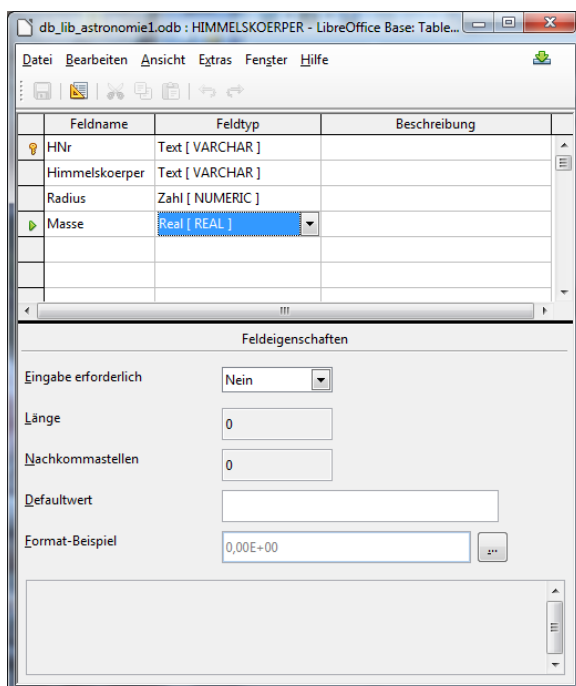


Abbildung 2

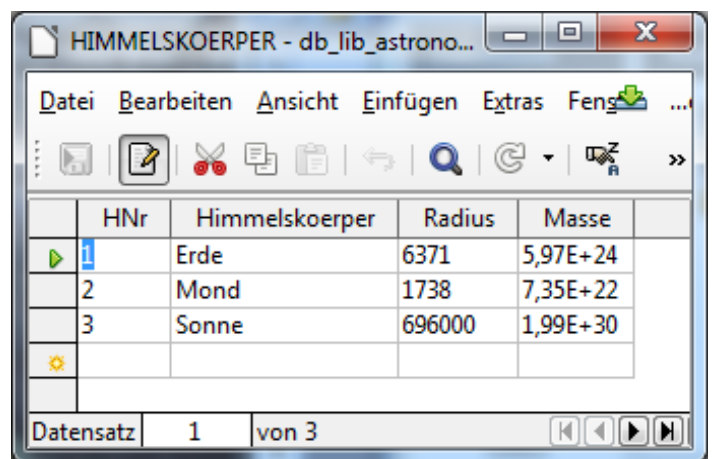


Abbildung 3

Wir wählen im DBMS links unter Datenbank den Button *Tabellen* an und wählen als Aufgabe: *Tabelle in der Entwurfsansicht erstellen*. Bei der Vergabe von Datentypen wurde beim Attribut *Masse* der Real-Typ verwendet, im Format-Beispiel wird eine Eingabemaske für diesen Typ definiert. Übrigens erlauben nicht alle Kommazahlen-Datentypen die Eingabe in dieser Maske, obwohl das Formatbeispiel es zulässt.

Nachdem der Entwurf fertig gestellt ist, wird dieser gespeichert. Vorher fragt das DBMS noch nach einem Primärschlüssel, dieser kann automatisch hinzugefügt werden, dabei wird jedoch ein zusätzliches Primärschlüssel-Attribut erzeugt oder man vergibt **vor** dem Speichern selbst den Primärschlüssel - hier an das Attribut *HNr* - dazu mit der rechten Maustaste auf das graue Feld vor dem Feldnamen klicken und den Befehl Primärschlüssel auswählen. Danach dann speichern und den Tabellennamen - am besten in Großbuchstaben - vergeben.

Soll ein Tabellenentwurf verändert werden, dann kann man im DBMS das Objekt Tabelle mit der rechten Maus anklicken und den Befehl *Bearbeiten* ausführen.

Analog wird nun die Tabelle MISSIONEN erstellt (Abb. 4 und 5).

Zwischen den beiden Tabellen existiert eine 1:n Beziehung, die Tabelle MISSION bekommt deswegen den Fremdschlüssel HNR eingetragen.

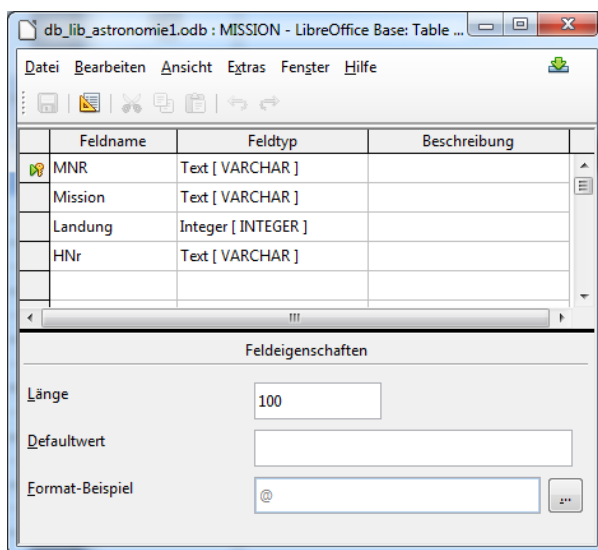


Abbildung 4

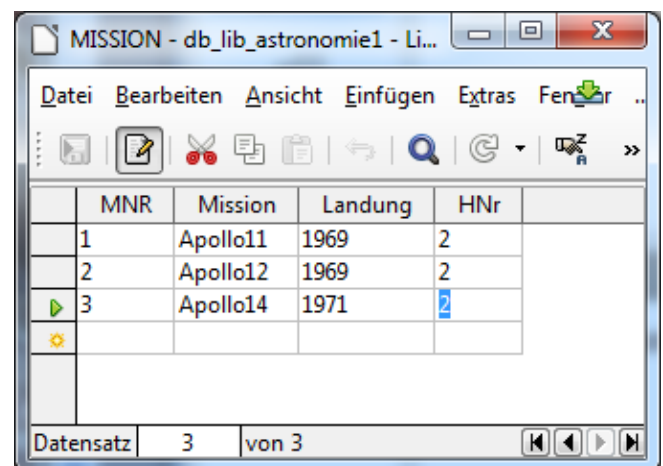


Abbildung 5

Zuletzt werden die Beziehungen zwischen den Tabellen hergestellt. Dafür im Menü *Extras* des DBMS den Befehl *Beziehungen* auswählen. Die gewünschte(n) Tabelle(n) markieren und hinzufügen. Das geht nur nacheinander, man kann die Tabellen nicht gleichzeitig markieren und hinzufügen. Mit der linken Maustaste den Primärschlüssel der ersten Tabelle anfassen und auf den Fremdschlüssel der anderen Tabelle ziehen, die 1:n-Beziehung wird sichtbar gemacht. Bei Doppelklick auf die Verbindungslinie können noch Update-Optionen und Löschoptionen eingestellt werden. Danach das Beziehungsfenster schließen und speichern.

Grundsätzlich gibt es in Datenbanken 1:1, 1:n und m:n- Beziehungen zwischen den Tabellen. Allerdings können nur 1:n-Beziehungen im DBMS implementiert werden.

Es gelten die Regeln:

- 1:1-Beziehungen werden als 1:n-Beziehungen aufgefasst, welche Tabelle dabei den Fremdschlüssel bekommt, ist frei wählbar

- Eine m:n-Beziehung wird in zwei 1:n-Beziehungen zerlegt. Dazu muss eine zusätzliche Tabelle erstellt werden, die sozusagen zwischen den beiden m:n – Tabellen liegt. Sie bekommt einen eigenen Primärschlüssel und nimmt die Primärschlüssel der m:n – Tabellen als Fremdschlüssel auf.

Nun können noch die anderen Tabellen und deren Beziehungen erstellt werden.

5. Erstellung von Formularen und Unterformularen

Zuerst erstellen wir ein einfaches Formular für die Tabelle MISSION. Dazu wählen wir im DBMS links unter Datenbank den Button *Formulare* an. Das Formular erstellen wir am einfachsten mit dem Formular-Assistenten. Wir klicken dazu im DBMS unter Aufgaben auf den Befehl *Formular unter Verwendung des Assistenten erstellen...*. Es öffnet sich das Formular-Entwurfsfenster und das Assistenten-Menü. Hier müssen nun die entsprechenden Schritte abgearbeitet werden.

1. Schritt:

Wir wählen zunächst die Tabelle MISSION aus. Das Formular soll die drei Felder MNr, Mission und Landung anzeigen. Das Fremdschlüsselfeld HNr brauchen wir im Formular nicht. Klicke auf *Weiter*.

2. Schritt:

Hier gibt es nichts zu tun, wir wollen kein Unterformular einfügen. Klicke auf *Weiter*.

5. Schritt:

Wir wählen eine Anordnung der Datenbankfelder aus. Wir wählen hier das dritte Bild - die tabellarische Anordnung - aus. Das hat einen bestimmten Grund - dazu später mehr. Klicke auf *Weiter*.

6. Schritt:

Das Formular soll existierende Daten anzeigen, die Optionsschalter lassen wir vorerst unverändert. Klicke auf *Weiter*.

7. Schritt:

Wir wählen einen geschmackvollen Stil. Klicke auf *Weiter*.

8. Schritt:

Der Formularname kann so bleiben oder man setzt den Buchstaben F davor: F_MISSION. Klicke auf *Fertigstellen*.

Wir schließen vorerst das Formular.

Ein Formular zur Tabelle HIMMELSKOERPER soll ein Unterformular erhalten. Wir erstellen das Formular wieder mit dem Assistenten.

1. Schritt:

Wir wählen die Tabelle HIMMELSKOERPER aus. Das Formular soll die vier Felder HNr, Himmelskoerper, Radius und Masse anzeigen.

Klicke auf *Weiter*.

2. Schritt:

Wir wollen ein Unterformular einfügen. Das Unterformular besteht auf der Beziehung zur Tabelle MISSION.

Klicke auf *Weiter*.

3. Schritt:

Wir wählen die Felder Mission und Landung für das Unterformular aus.

Klicke auf *Weiter*.

4. Schritt: entfällt, wenn im 2. Schritt die Beziehung eindeutig ist, sonst muss im 4. Schritt die Beziehung über die Schlüsselfelder angegeben werden

5. Schritt:

Wir wählen für das übergeordnete Formular die Form Spaltenbeschriftung links, für das Unterformular eine tabellarische Darstellung aus.

Klicke auf *Weiter*.

6. Schritt:

Das Formular soll existierende Daten anzeigen, die Optionsschalter lassen wir vorerst unverändert.

Klicke auf *Weiter*.

7. Schritt:

Wir wählen einen geschmackvollen Stil.

Klicke auf *Weiter*.

8. Schritt:

Der Formularname kann so bleiben oder man setzt den Buchstaben F davor:

F_HIMMELSKOERPER.

Klicke auf *Fertigstellen*.

Wir schließen das Formular.

Beim Testen der Funktionsweise des Formulars F_HIMMELSKOERPER erhalten wir z.B. bei der Anzeige des Datensatzes zum Mond im Unterformular Informationen über die Mondmissionen.

Nach diesem Prinzip werden zu allen weiteren Tabellen Formulare erstellt, je nach Bedarf mit einem Unterformular (F_SONNENSYSTEM, F_FORSCHUNGSPROGRAMM, usw.)

Erweiterung I: Der Tabelle MISSION und dem Formular F_MISSION soll ein weiteres Feld hinzugefügt werden. Zuerst wird die Tabelle MISSION zum Bearbeiten geöffnet und ein weiteres Feld, z.B. *Kosten* hinzugefügt. Danach wird das Formular F_MISSION zum Bearbeiten geöffnet. Hier soll in die tabellarische Form des Formulars die Spalte (das Feld) *Kosten* eingefügt werden. Dazu wählen wir in der unteren Symbolleiste das Icon Feld hinzufügen. Aus dem nun geöffneten Fenster kann das Feld *Kosten* per Drag and Drop in die Tabellenkopfzeile gezogen werden.

Hinweis I: Attribut und Attributwert treten standardmäßig gruppiert auf. Soll z.B. das Feld für den Attributwert vergrößert werden, muss die Gruppe markiert werden, dann mit rechter Maus den Befehl *Gruppieren* und dort den Befehl *Betreten* wählen. Jetzt kann das Attributwertfeld einzeln markiert und dessen Größe verändert werden.

Hinweis II: Fehlende Einheiten sollten im Formular als Beschriftungsfeld ergänzt werden. Dazu ein Beschriftungsfeld in das Formular an die gewünschte Stelle ziehen, rechte Maus, Befehl *Steuerelement...* und hier die Eigenschaft *Titel* verändern.

6. Erstellung von Berichten

Zunächst wollen wir Berichte mit Hilfe des Berichtsassistenten erstellen. Dazu wählen wir das Datenbankobjekt Bericht aus und unter Aufgaben klicken wir auf den Assistenten.

Beispiel 1: Einfacher und nicht gruppierter Bericht zur Tabelle HIMMELSKOERPER:

Die einzelnen Schritte sind sehr gut selbsterklärend. Deshalb werden hier nicht die Schritte einzeln erklärt, sondern nur Schwerpunkte der Berichtserstellung beschrieben.

Wir wählen für den Bericht die Himmelskörper, deren Radien und Massen aus. Den Gruppierungsschritt lassen wir aus und sortieren nach Himmelskörper. Zuletzt wählen wir die Variante Dynamischer Bericht. Damit wird nach einer Änderung der Daten der Bericht automatisch angepasst (Abb. 6). Allerdings hat sich die vorliegende LibreOffice-Version nicht besonders viel Mühe gegeben, der Bericht sieht etwas umständlich aus - jeder Datensatz hat einen eigenen Spaltenkopf. Dies müssen wir (leider) per Hand in der Berichtsdefinition ändern.

Himmelskoerper	Radius	Masse
47 Ursae Majoris	904800	1,99E+030
Himmelskoerper	Radius	Masse
47 Ursae Majorisb	Radius	5,99E+027
Himmelskoerper	Radius	Masse
47 Ursae Majorisc	Radius	1,50E+027
Himmelskoerper	Radius	Masse
Adraatea	8	1,90E+016
Himmelskoerper	Radius	Masse
Alpha Centauri B	598560	1,85E+030
Himmelskoerper	Radius	Masse
Alpha Centauri Bb	Radius	6,57E+024
Himmelskoerper	Radius	Masse
Amalthea	84	2,10E+018
Himmelskoerper	Radius	Masse
Ananke	14	3,00E+016
Himmelskoerper	Radius	Masse
Ariel	579	1,35E+021
Himmelskoerper	Radius	Masse
Carne	23	1,30E+017
Himmelskoerper	Radius	Masse
Ceres	480	9,35E+020

Abbildung 6

Wir öffnen den Bericht zum Bearbeiten. Jetzt muss in der Berichtsdefinition die Kopfzeile aus dem Berichtskopf per Hand in den Seitenkopf verschoben werden. Dazu die Textobjekte einfach per Drag and Drop nach oben verschieben und die Objektbreiten etwas verkleinern, sonst kann es Schwierigkeiten bei der waagerechten Ausrichtung geben. Zusätzlich wurden noch Datum, Uhrzeit, Maßeinheiten und eine waagerechte Linie ergänzt. Jetzt kann man den Bericht ausdrucken bzw. als PDF- oder JPG-Datei exportieren (Abb. 7).

29.12.13
19.07

Himmelskoerper	Radius in km	Masse in kg
47 Ursae Majoris	904800	1,99E+030
47 Ursae Majorisb		5,99E+027
47 Ursae Majorisc		1,50E+027
Adrastea	8	1,90E+016
Alpha Centauri B	598560	1,85E+030
Alpha Centauri Bb		6,57E+024
Amalthea	84	2,10E+018
Ananke	14	3,00E+016
Ariel	579	1,35E+021
Carne	23	1,30E+017
Ceres	480	9,35E+020
Deimos	6	1,80E+015
Desphina	75	2,10E+018
Dione	562	1,10E+021
Elara	39	8,70E+017
Enceladus	252	1,08E+020
Epsilon Eridani	584640	1,69E+030
Epsilon Eridani b		2,93E+027

Abbildung 7

Beispiel 2: Gruppierter Bericht mit Zähl-Statistik zur Tabelle HIMMELSKOERPER:

Wir wählen für den Bericht die Attribute *Himmelskörper*, *umkreist* und *AnzahlMonde* aus. Wir nutzen wieder den Assistenten und gruppieren diesmal nach *umkreist*, um die Planeten und Monde den Sonnen und Planeten zu zuordnen. Wir ergänzen einen Gruppenfuß, um statistische Daten zu erhalten - Menü Ansicht / Sorting and Grouping, den Gruppenfuß setzen wir auf *Vorhanden*. Dort hinein legen wir ein Textfeld. In den Eigenschaften wählen wir im Register Daten den Datenfeldtyp **Zähler**. Damit wird die Anzahl der Gruppeneinträge, also die Anzahl der Begleiter gezählt. Mit dem Datenfeldtyp **Funktion** kann man außerdem Summe, Minimum und Maximum eines Zahlenfeldtyps berechnen lassen (Durchschnitt fehlt in dieser Office-Version leider) (Abb. 8).

Himmelskörper und ihre Begleiter

wer umkreist 47 Ursae Majoris

Himmelskoerper
47 Ursae Majorisb
47 Ursae Majorisc

AnzahlMonde

Anzahl der Begleiter 2

wer umkreist Alpha Centauri B

Himmelskoerper
Alpha Centauri Bb

AnzahlMonde

Anzahl der Begleiter 1

wer umkreist Epsilon Eridani

Himmelskoerper
Epsilon Eridani b

AnzahlMonde

Anzahl der Begleiter 1

wer umkreist Erde

Himmelskoerper
Erdmond

AnzahlMonde

Anzahl der Begleiter 1

Abbildung 8

7. Erstellung von Abfragen mit SQL

SQL-Abfragen lassen sich mit LibreOffice sehr einfach erstellen. Dazu wählen wir im DBMS unter Datenbank das Objekt *Abfragen* und unter Aufgaben *Abfrage in SQL-Ansicht erstellen*. Es öffnet sich das Query-Design-Fenster. Hier wird die Abfrage in SQL direkt eingegeben.

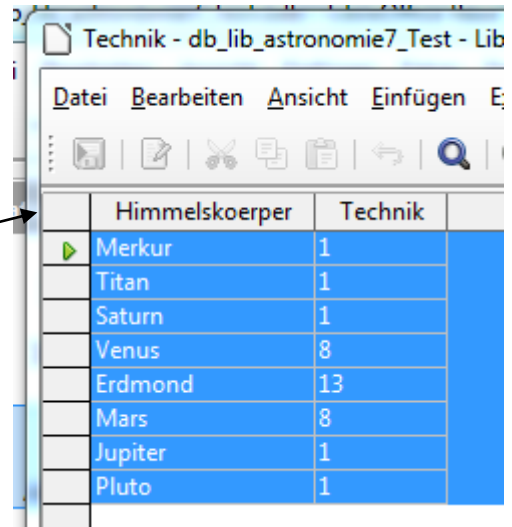
Beispiel 1: Gesucht sind alle Himmelskörper, deren Radius kleiner als 1000 km ist.

**SELECT Himmelskoerper, Radius
FROM HIMMELSKOERPER
WHERE Radius < 1000;**

Die Abfrage wird mit dem Abfrage-Ausführungssymbol oder [F5]->Taste ausgeführt. Anschließend sollte die Abfrage unter einem sinnvollen Namen gespeichert werden.

Die Ergebnistabelle kann als PDF-, JPG- oder HTML-Datei exportiert werden. Dazu wird zunächst ein neues CALC-Dokument geöffnet. Wir klicken jetzt im DBMS mit der rechten Maustaste auf den Namen der Abfrage-Tabelle und wählen *Kopieren*. Im CALC-Fenster klicken wir auf die Zelle A1 und dann auf das Menü *Bearbeiten* und den Befehl *Einfügen*.

Natürlich kann die Abfrage-Tabelle auch direkt in ein Text-Dokument kopiert werden. Dazu wird die gesamte Abfrage-Tabelle durch Klicken auf den grauen Button oben links in der Tabellenecke (siehe Abb. 9) markiert, anschließend mit STRG+C kopiert und mit STRG+V in das Text-Dokument eingefügt.



Himmelskoerper	Technik
Merkur	1
Titan	1
Saturn	1
Venus	8
Erdmond	13
Mars	8
Jupiter	1
Pluto	1

Abbildung 9

8. Makros und Makrosicherheit

Beispiel 1: Erstellung eines Buttons zum Öffnen eines Formulars mit dem Steuer-Elementenamen „F_HIMMELSKOERPER“. Dabei ist es beliebig, ob es sich um ein Formular, welches aus einer Tabelle oder einer Abfrage entstanden ist, handelt.

1. Form im Bearbeitungsmodus öffnen
2. OK-Schaltfläche auf das Formular ziehen, mit rechter Maus anklicken, *Steuerelement...*-Befehl anklicken, Register *Allgemein* betrachten und Schaltflächen-Eigenschaft *Titel* anpassen, vorerst wieder schließen
3. im Form-Menü Extras / Makros / Makros verwalten / LibreOfficeBasic... anwählen
4. bei erstmaliger Verwendung den Unterordner Module1 im Ordner der LibreOffice-Datenbank unter Standard erzeugen, sonst diesen Ordner öffnen (Abb. 10)
5. Schalter Bearbeiten anklicken und Makro erstellen (siehe Abb. 10)
6. Makro speichern und Fenster schließen
7. die zu bearbeitende Schaltfläche wieder mit rechter Maus anklicken, wieder den *Steuerelement...*-Befehl anklicken und das Register *Ereignisse* öffnen
8. Beim Ereignis *Maustaste gedrückt* den [...] -Schalter bedienen (Abb. 11), ein weiteres Fenster öffnet sich und das Makro über den Schalter Makro... zuweisen (Abb. 12), indem im Makro-Selektor (Abb. 13) das entsprechende Makro ausgewählt wird
9. alle Fenster wieder mit OK bzw. dem X-Schalter schließen

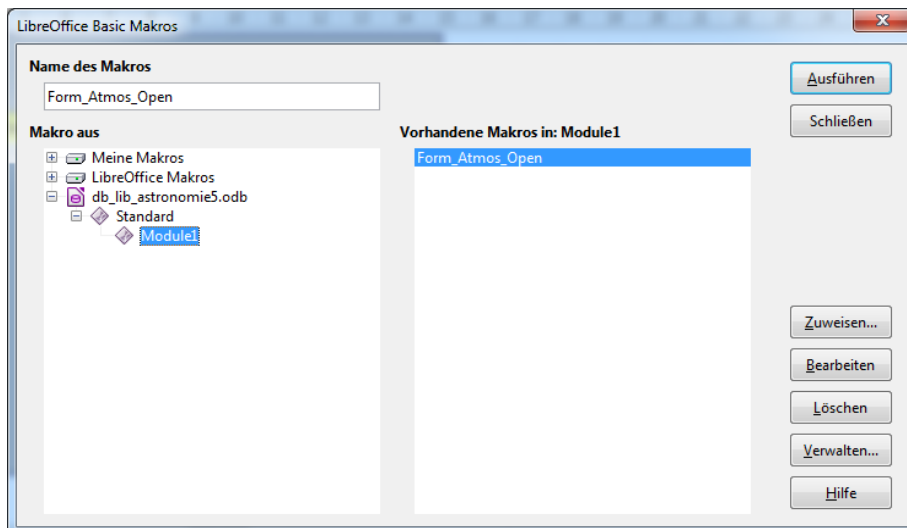


Abbildung 10

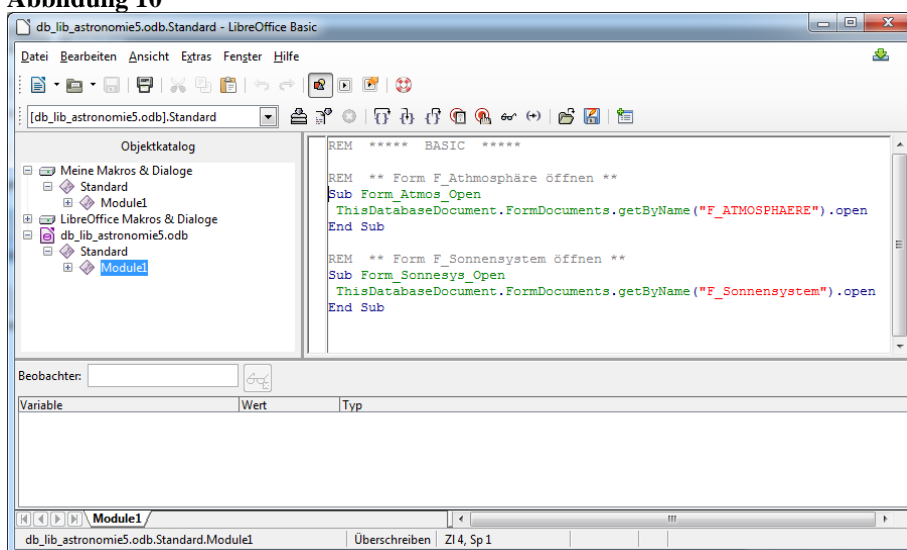


Abbildung 11

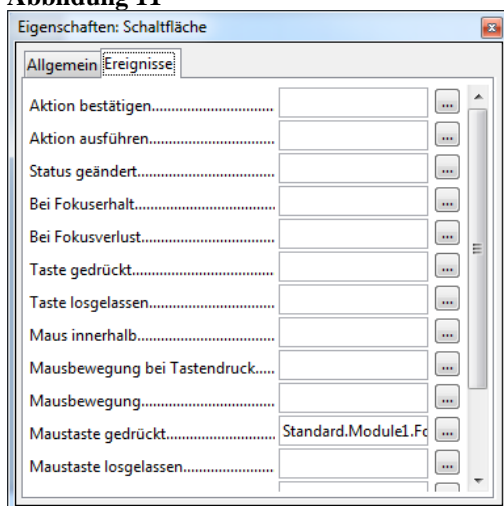


Abbildung 12

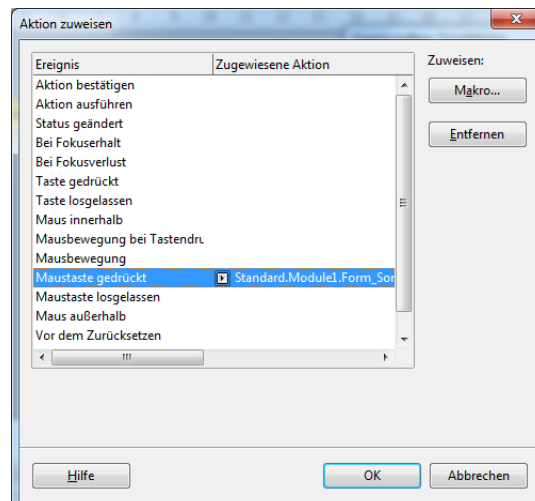


Abbildung 13

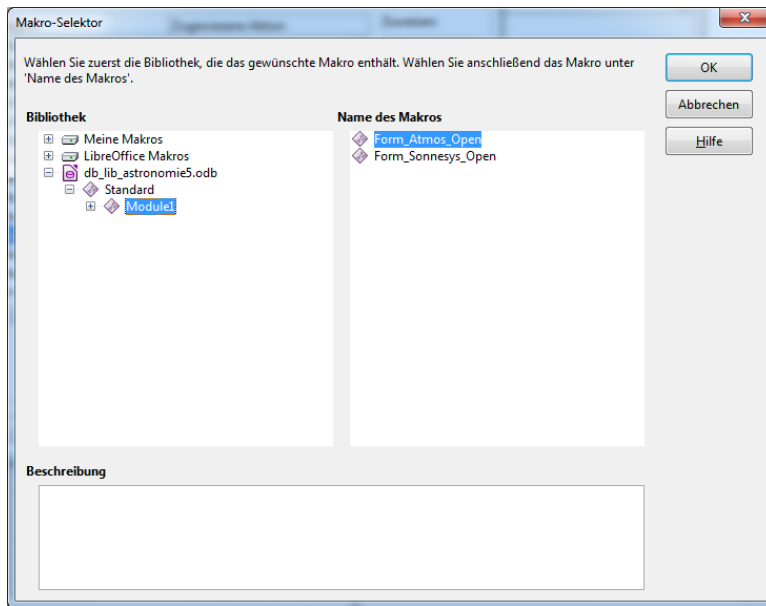


Abbildung 14

Das Makro zum Öffnen des Formulars sieht dann wie folgt aus (siehe auch Abb. 10)

```
REM ** Form F_Himmelskörper öffnen **
Sub Form_Himmelskoerper_Open
  ThisDatabaseDocument.FormDocuments.getByName("F_HIMMELSKOERPER").open
End Sub
```

bzw. in verbesserter Variante, wobei die beiden Objekte oFrame und oWin dazu dienen, die Formulargröße (speziell die Fenstergröße) einzustellen. Hier wird das Fenster an der linken oberen Ecke des Bildschirms mit einer Breite von 600 Punkten und einer Höhe von 400 Punkten dargestellt. Die letzte Ziffer weist darauf hin, dass alle Punkte angegeben wurden. Sie wird als **'Flag'** bezeichnet.

```
REM ** Form F_Himmelskoerper öffnen **
Sub Form_Himmelskoerper_Open
  DIM oFrame AS OBJECT
  DIM oWin AS OBJECT
  ThisDatabaseDocument.FormDocuments.getByName("F_HIMMELSKOERPER").open
  oFrame = StarDesktop.getCurrentFrame()
  oWin = oFrame.getContainerWindow()
  oWin.setPosSize(0,0,600,400,15)
End Sub
```

Hinweise: Werden Makros nicht ausgeführt, sollten nachfolgende Einstellungen überprüft werden:

- Menü Extras / Optionen / LibreOffice / Sicherheit / Makrosicherheit
 - Sicherheitslevel: Niedrig
 - Vertrauenswürdige Quellen: hier den Ordner angeben, auf dem die Makros gespeichert werden (in der Regel der Ordner der DB selbst)
- das Makro kann Syntaxfehler enthalten, insbesondere muss immer auf spezifische Steuerelement-Namen geachtet werden - hier "F_HIMMELSKOERPER" (Groß- und Kleinschreibung beachten)

Diese Einstellungen sind am PC nur einmal auszuführen. Danach sollte LibreOffice auf alle Fälle beendet und neu gestartet werden.

Beispiel 2: Makro für einen Button zum Schließen eines Formulars

Die Steuerelement-Namen (im Folgenden z.B. „FORMNAME“ sind in den Makros noch an die konkreten Namen in der Datenbank anzupassen.

```
REM ** Form FORMNAME schließen **
Sub Form_FORMNAME_Close
  ThisDatabaseDocument.FormDocuments.getByName("FORMNAME").close
End Sub
```

Beispiel 3: Makro für einen Button zum Öffnen eines Berichtes

```
REM ** Bericht BERICHTSNAME öffnen **
Sub Ber_BERICHTSNAME_Open
  ThisDatabaseDocument.ReportDocuments.getByName("BERICHTSNAME").open
End Sub
```

Beispiel 4: Makro für einen Button zum Starten eines Windows-Programmes, welches auf einem festen Windowsordner liegt (hier als Beispiel Starten des Windows-Rechners):

```
REM ** Windows-Rechner starten ** mit absoluter Pfadangabe **
SUB Windows_Rechner_starten
  shell("C:/Windows/System32/calc.exe",10,"") ' Slash oder Backslash möglich!!
End Sub
```

Der Shell-Befehl enthält vier Parameter:

Shell(Pathname, Windowstyle, Param, bSync)

Pathname legt Pfad und Name des aufzurufenden Programms fest

Windowstyle definiert, in welchem Fenster das Programm gestartet wird. Möglich Werte:

- 0 – Programm erhält Fokus und wird in einem versteckten Fenster gestartet.
- 1 – Programm erhält Fokus und wird in einem Fenster in Normalgröße gestartet.
- 2 – Programm erhält Fokus und wird in einem minimierten Fenster gestartet.
- 3 – Programm erhält Fokus und wird in einem maximierten Fenster gestartet.
- 4 – Programm wird in einem Fenster in Normalgröße gestartet, jedoch ohne Fokus
- 6 – Programm wird in einem minimierten Fenster gestartet, jedoch ohne Fokus
- 10 – Das Programm wird im Vollbild-Modus gestartet.

Param gestattet es, Kommandozeilenparameter an das zu startende Programm zu übergeben. Dieser Parameter ist optional.

bSync legt fest, wie sich das Makro nach dem Starten der Shell verhält.

Der Standardwert ist false. Dann arbeitet das Makro direkt weiter. Wird der Parameter auf true gesetzt wartet das Makro bis der Shellaufruf beendet worden ist.

Beispiel 5: Makro für einen Button zum Starten eines Windows-Programmes, welches auf einem Unterordner des Ordners liegt, in dem sich die Datenbank befindet. Dafür werden relative Pfadangaben verwendet, denn damit können bei einer Kopie der Datenbank auch alle Unterordner mit kopiert werden (die kursiven Textteile im Makro sind Kommentazeilen):

```

REM ** Programm XYZ im Unterordner Programme starten **
SUB Programm_XYZ_starten
  oDB = ThisComponent.Parent 'Der Zugriff auf die URL ist nicht vom Formular aus
  direkt möglich. Es muss auf den darüberliegenden Frame der Datenbank Bezug
  genommen werden.
  stDir = Left(oDB.Location,Len(oDB.Location)-Len(oDB.Title)) 'Der Titel der
  Datenbank wird von der URL abgetrennt.
  shell(stDir & "/Programme/programmxyz.exe",1,"") 'immer Slash statt Backslash
  verwenden!!
End Sub

```

Beispiel 6: Makro für einen Button zum Öffnen eines PDF-Dokumentes, welches auf einem Unterordner des Ordners liegt, in dem sich die Datenbank befindet. Dafür werden wie in Beispiel 5 relative Pfadangaben verwendet:

```

REM ** PDF-Dokument im Unterordner Texte öffnen **
sub PDF_Dokument_oeffnen
  oDB = ThisComponent.Parent 'Der Zugriff auf die URL ist nicht vom Formular aus
  direkt möglich. Es muss auf den darüberliegenden Frame der Datenbank Bezug
  genommen werden.
  stDir = Left(oDB.Location,Len(oDB.Location)-Len(oDB.Title)) 'Der Titel der
  Datenbank wird von der URL abgetrennt.
  oSer = createUNOService("com.sun.star.system.SystemShellExecute")
  oSer.execute(stDir & "/Texte/PDF_Dokument.pdf",,0)
end sub

```

9. Startformular erstellen

Beim Öffnen der Datenbank soll ein Startformular mit dem Namen **StartForm** automatisch geöffnet werden. Das Startformular dient als Menü und Hauptnavigation der Datenbank. Es enthält Buttons zu den Formularen, Tabellen, Berichten und Abfragen. Die Buttons werden über Makros gesteuert. Dazu wie folgt vorgehen:

1) Neues Formular in Entwurfsansicht erstellen und gestalten

- Hintergrundfarbe: Menü Format, Seite..., Hintergrund
- Überschriften: Toolbox - Beschriftungsfeld (Verankerung an der Seite), Eigenschaft *Steuerelement* (Tittel), Schrift,...
- Buttons für Formulare, Berichte, Abfragen
- Bilder

2) Zum Öffnen des Startformulars wird folgendes Makro benötigt:

```

REM ** Form StartForm öffnen **
Sub OnOpenDocument()
  Dim oController
  oController = ThisDatabaseDocument.CurrentController
  If not (oController.isConnected()) Then oController.connect() 'Verbindung zur
  DB herstellen, wenn nötig
  oController.loadComponent(com.sun.star.sdb.application.DatabaseObject.FORM, -
  "StartForm", FALSE) 'Parameter: ObjectType, ObjectName, ForEditing
End Sub

```

3) Menü Extras / Anpassen / Ereignisse:

Ereignis = Dokument öffnen

zugewiesene Aktion: Zuweisen: Makro: (obiges Makro zuweisen)

Speichern [1]

4) Ausschalten von Formularanzeigen

Im Bearbeitungsmodus können "überflüssige" Formularanzeigen ausgeschaltet werden. Damit sieht das Startformular dann sehr "aufgeräumt" aus:

- Menü Ansicht: Statusleiste aus, Lineale aus
- Menü Ansicht - Symbolleisten: Formular-Steuerelemente, Formular-Navigation, Formatierungen, Formular-Entwurf

10. LibreOffice-Datenbank mit Makro speichern und schließen

Die Beendigung der Arbeit mit der Datenbank umfasst immer die Schritte:

- Formulare schließen
- Datenbank-Manger speichern und schließen

Dieser Vorgang lässt sich aus dem Startformular mit einem Makro ausführen. Damit werden automatisch alle geöffneten Formulare geschlossen. Gleichzeitig werden alle Änderungen im Datenbankmanagement gespeichert und das Datenbanksystem geschlossen. Das Makro sieht wie folgt aus:

```
REM ** Datenbank schließen
SUB speichern_und_beenden(event as object)
    oController = ThisDatabaseDocument.CurrentController
    oController.closeSubComponents
    oController.ActiveConnection.close()
    oDoc = ThisDatabaseDocument
    oDoc.store()
    oDoc.close(True)
END SUB
```

11. Bilder verwalten in LibreOffice

Wir erstellen zuerst eine Tabelle BILDER mit den Feldern BildNr, Bildtitel, Bildpfad und Bildbeschreibung. Als Datentyp für Bildpfad verwenden wir BILD[LONGVARBINARY]. Danach erstellen wir ein Formular, am besten enthält es alle 4 Felder der BILDER-Tabelle. Die Bilder speichern wir vorher alle in einem Unterordner (z.B. DB_Bilder) ab. Das Formular wird jetzt geöffnet. Wir klicken mit der rechten Maustaste auf das Ausgabefeld von Bildpfad und führen den Befehl *Bild einfügen von...* aus. Das Bild kann jetzt im Dateiverzeichnis unter DB_Bilder gesucht und geöffnet werden. Damit wird das Bild im Anzeigefeld des Formulars auch ausgegeben.

12. Webseiten öffnen

Das Öffnen von Webseiten aus einem Formular heraus ist ohne Makro möglich. Dazu wird zunächst eine Schaltfläche auf dem Formular erstellt. In den Eigenschaften der Schaltfläche wird im Reiter Allgemein bei der Eigenschaft „Aktion“ der Wert „Dokument/Webseite öffnen“ ausgewählt. Jetzt kann in der Eigenschaft „URL“ die Webadresse eingetragen werden.

13. Listenfelder

Beispiel 1: Im Formular F_SONNENSYSYSTEM wollen wir die Navigation innerhalb der Datensätze erleichtern. Mittels eines Listenfeldes soll ein bestimmter Datensatz direkt

aufzurufen sein. Damit kann man sich das Durchblättern durch die einzelnen Datensätze ersparen. Dieses Vorgehen entspricht letztlich auch einer Suchanfrage an die Tabelle zu einem speziellen Datensatz. Im Hintergrund dazu werkelt ein SQL-Befehl und der Suchauftrag muss über ein Mausklick-Ereignis ausgelöst werden, das wiederum ein Makro übernimmt.

Wir erstellen auf dem Formular das Steuerelement *Listenfeld*. Der Assistent wird in diesem Fall wie folgt abgearbeitet:

1. Auswahl der Tabelle für das Listenfeld - hier Tab. SONNENSYSTEM
2. Auswahl des Feldes, dessen Daten in dem Listenfeld stehen sollen - hier SName
3. Herstellung des Zusammenhangs Feld aus der Wertetabelle= Feld aus der Listentabelle
4. Fertigstellen
5. mit rechter Maus auf das erstellte Listenfeld klicken - Befehl *Steuerelement...* auswählen
6. im Listenfeld-Eigenschaften-Register *Allgemein* beim Namen **ListBox1** eintragen
7. im Register *Ereignisse* beim Ereignis *Status_geändert* das Makro **SelectListBox1** zuweisen

Das Makro SelectListBox muss natürlich vorher erstellt werden. Es lautet wie folgt:

```
REM ** Listbox_auswerten **
Sub SelectListBox1
Dim oForm, oListbox As Object
Dim LB_Item as integer
    oForm = ThisComponent.drawpage.forms.getbyindex(0)
    oListbox = oForm.getByName("ListBox1")
    LB_Item = oListbox.SelectedItems(0)
    oForm.absolute(LB_Item+1)
End Sub
```

Beispiel 2: Im Formular F_MISSION existiert der Fremdschlüssel HNr, der auf den Himmelskörper zeigt, zu welchem die jeweilige Mission stattfand. Schöner wäre es, wenn beim Aufruf einer Mission der entsprechende Himmelskörper angezeigt würde. Auch hier nutzen wir die Möglichkeiten des Listenfeldes. Dieses Mal benötigen wir dafür kein Makro, sondern können die Funktionalität des Listenfeldes direkt nutzen. Vorgehensweise:

1. Listenfeld im Formular F_MISSION erstellen und Assistent abarbeiten
2. Tab. wählen, deren Daten die Grundlage für den Listeninhalt bilden - hier Tab. HIMMELSKOERPER
3. Feld Himmelskoerper als Anzeigefeld auswählen
4. Feld aus der Wertetabelle HNr in Beziehung zum Feld in der Listentabelle HNr setzen
5. Fertigstellen und speichern

14. Konvertierungen

Typische Konvertierungen sind:

1. Tabellen übertragen von MS ACCESS nach LibreOffice
2. Tabellen übertragen von LibreOffice nach MS ACCESS

Zu 1.:

In MS Access die gewünschte Tabelle mit rechter Maus anklicken, Befehl *Kopieren*.

In LibreOffice im Bereich Tabellen rechte Maus klicken und Befehl *Einfügen* ausführen, dabei dem Assistenten folgen und für alle Attribute die Datentypen überprüfen und evtl ändern. Der Primärschlüssel sollte analog zur MS Access-DB gesetzt werden.

Zu 2.:

In LibreOffice die Tabelle auswählen und kopieren und in MS Excel einfügen, als Excel-Datei speichern. In MS Access im DBMS im Bereich Tabellen rechte Maus klicken und Befehl Importieren ausführen. Die erstellte Excel Datei aufsuchen und dem Assistenten folgen, kleine Nachbesserungen sind evtl. notwendig, der Primärschlüssel muss noch gesetzt werden.

15. Quellen

[1] <http://www.mail-archive.com/users@de.openoffice.org/msg61394.html>, 19.12.2013, 18:25