

SQL-Abfragen

Einleitung:

Um die SQL - Abfragen machen Anfänger einen großen Bogen. Zu unrecht, denn die SQL - Abfrage dient in vom Anfänger unterschätzten Maße der Arbeitserleichterung. Die gewonnene Abfrage stellt sich im Mehrzweckbereich dar. Wenn man diese Abfrage ausdrucken will, erreicht man dies über den Menüpunkt Suchen - Abfrageexperte - aktuelle SQL-Abfrage (Fenster unterhalb der Grundmenge öffnen) - Grundmenge. Der Phantasie sind hier kaum Grenzen gesetzt, das Grundschema der SQL-Abfragen ist aber ziemlich gleich. GFAhnen liefert ihnen dazu zahlreiche Musterbeispiele in AF_Daten → AF_Daten und AF_Editor, die sie beliebig verwandeln und unter eigenem Namen speichern können. Weitere Beispiele findet man in der ONLINE-HILFE bei SQL - Beispiele Gruppen 1 – 6. Die SQL-Abfrage ist im Vergleich zum Suchexperten individueller. Für einfache Abfragen bietet sich eher der Suchexperte an. Auch hier besteht die Möglichkeit, ein Suchergebnis in Form einer SQL- Abfrage ohne Ausdruck und ohne Speicherung direkt in den Mehrzweckbereich zu bringen, um dann direkt die Datensätze zu bearbeiten. Näheres finden Sie in einem eigenen Kapitel.

Beispielhafte Anwendungsbereiche:

- a) Auflistung aller aus einem bestimmten Herkunftsort, Geburtsort, Taufort, Sterbeort, Bestattungsort, Ort des Lebensbereiches oder der Eheschließung stammender Personen oder Personen eines bestimmten Familiennamens.
- b) Auflistung aller Personen oder Personen eines bestimmten Familiennamens, die in einem gewählten Zeitraum geboren (getauft, gestorben, begraben) sind oder geheiratet haben
- c) Auflistung aller Personen mit einer bestimmten Signatur oder aller Personen, die keine bestimmte Signatur haben
- c) Fehlersuche:
Alle Datensätze mit Vornamen in VName 2, bei denen ein Komma davor steht (das nicht dahingehört)

Die Einsetzungsmöglichkeiten sind sehr vielfältig, es ist aber unmöglich, alle Eventualitäten hier zu behandeln. Das Grundschema der SQL-Abfragen ist aber ziemlich gleich, so daß ich mich nur auf eine Art Crash-Kurs zum Einstieg und selbständigen Weiterentwickeln beschränken möchte. Weitere als die hier beispielhaften SQL-Abfragen sind zu finden in:

A. Grundkenntnisse:

Die SQL-Abfrage ist eine sog. Select-Anweisung, d.h. ich selektiere einen bestimmten Bereich, den ich dargestellt haben möchte. Hier gibt es eine Eingabehilfe, die entweder eingeschaltet ist bei SQLAbfrage-Datei -Optionen oder die man zuschalten kann.

I. Allgemeine einfache Abfrage:

- a) *Aufgabenstellung: Sie suchen alle Personen, die in „Lage“ geboren sind.*

Die SQL-Abfrage rufen Sie auf mit Menü - Suchen - SQL - Abfragen - allgemein. Es öffnet sich dann das Feld:

```
SELECT /* Felder der Ergebnistabelle */
FROM /* Tabellennamen */
WHERE /* Suchbedingungen */
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
```

Der Cursor steht unter **FROM**. Sie drücken die **Leertaste**, in dem Feld suchen Sie Ihre Datei in Suchen in und wählen **Ahnen.db** und **Öffnen**.

Der Cursor springt dann auf **Select**. Sie drücken die **Leertaste**, in dem sich öffnenden Feld geben Sie die Maus auf PN, Sie erhalten

```
SELECT /* Felder der Ergebnistabelle */
a.PN
```

Sie drücken wieder die **Leertaste**, in dem sich öffnenden Feld geben Sie die Maus auf FName, Sie erhalten

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName
```

sie drücken wieder die **Leertaste**, in dem sich öffnenden Feld geben Sie die Maus auf VName, sie erhalten

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName
```

Sie drücken wieder die **Leertaste**, in dem sich öffnenden Feld geben sie die Maus auf GebDat, sie erhalten

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat
```

Sie drücken wieder die **Leertaste**, in dem sich öffnenden Feld geben sie die Maus auf GebOrt, sie erhalten

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat, a.GebOrt
```

Sie selektieren also hier mit Mausclick rechte Hand - Leertaste linke Hand - Mausclick - Leertaste usw. die Felder, die Sie dargestellt haben möchten. Die Felder, die Sie nicht benötigen, lassen Sie weg.

Jetzt bewegen Sie den Cursor unter das Feld **WHERE** und drücken die **Leertaste**. Es müßte jetzt **GebOrt** markiert sein, Mausclick und sie erhalten

```
WHERE /* Suchbedingungen */
(a.GebOrt
```

Leertaste und Sie wählen im dem sich öffnenden blauen Feld **LIKE**

```
WHERE /* Suchbedingungen */
(a.GebOrt LIKE
```

Leertaste und es öffnet sich ein kleines schmales grünes leeres Feld links unten und rechts ist GebOrt markiert.

Links unten schreiben Sie jetzt den Namen des Ortes XY und drücken nun die **Entertaste (!)**

Und Sie erhalten

```
WHERE /* Suchbedingungen */
(a.GebOrt LIKE '%Lage%')
```

also Hochkomma(über #) Prozentzeichen Ort Prozentzeichen Hochkomma

Sie bewegen jetzt den Cursor unterhalb **ORDER BY** . Den markierten GebOrt nehmen Sie nicht, sondern FName

```
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName
```

Dann drücken Sie ausführen und Sie erhalten im Mehrzweckbereich die Tabelle.

Abfrage: allgemein				
PN	FName	VName	GebDat	GebOrt
105	BERG	August "Ludwig"	1861.12.17	Lage
109	BERG	Rudolf Christian Adolf	1907.06.05	Lage
108	BERG	Hermann Ludwig Ernst	1903.09.27	Lage
107	BERG	Marie Elisabeth	1891.11.08	Lage
106	BERG	Helene Marie Wilhelmine	1890.07.18	Lage
110	BERG	Margarete Wilhelmine Charlotte	1899.05.24	Lage
104	BERG	Hermann August	1855.03.29	Lage
103	BERG	Ernst Gustav	1852.02.09	Lage
78	BERG	"Wilhelm "Ernst Heinrich	1842.05.08	Lage
77	BERG	Wilhelmine "Marie" Charlotte	1840.09.04	Lage
46	GROTEBRUNE	Marie Wilhelmine Karoline	1849.11.22	Lage
54	OTTEMEYER	Marie Charlotte Amalie	1873.03.28	Lage

Praktische Verwendung:

1. Die so gewonnene SQL-Tabelle im Mehrzweckbereich können Sie speichern und so immer wieder aufrufen.

Sie gehen so vor: GFAhnen - Menü - Datei - DatenAustausch - Kopieren - anklicken SQL-Abfrageergebnis und als Zieltabelle neu anlegen - Start.

Es öffnet sich dann ein Fenster, in dem Sie das Zielverzeichnis suchen und einen Dateinamen geben. Gespeichert wird als .db (Paradox). Ich kennzeichne die gewonnene Abfrage mit SQLAbfrage_Lage.db. Dann speichern und Sie können diese Tabelle im Mehrzweckbereich über Tabellenansichten/externe Tabellen immer öffnen.

2. Diese SQL - Abfragetabelle können Sie sich ausdrucken oder als html-Ausgabe darstellen lassen über den Abfrageexperten, wie in der allgemeinen Schritt-für-Schrittanleitung (kurz nächster Absatz) beschrieben.

Zur Erinnerung und Ergänzung Abfrageexperte :

GFAhnen - Menü - Suchen - Abfrageexperte . Unter Grundmenge das Kontextmenü öffnen mit dem Pfeil - aktuelle SQL-Abfrage - Grundmenge - AbfrageLage (oder einen anderen Namen eingeben für die Zieltabelle), dann erhalten Sie in blauer Schrift die Menge der nicht sortierten Datensätze - sortieren und Ausgabeform wählen.

Ihre SQL-Abfrage (Unterschied zur Tabelle!) Orte können Sie speichern und immer wieder verwenden. Sie können es nachholen, indem Sie jetzt den Menüpunkt Suchen - SQL - Abfrage ändern wählen. Dann Datei speichern unter - Verzeichnis wählen und eigenen Namen geben.

- b) Diese SQL-Abfrage können Sie nun immer wieder verwenden und auch abwandeln über Abfrage ändern. Wichtig ist, daß Sie immer wieder den Menüpunkt Suchen - SQL - Abfrage ändern aufrufen.

Löschen Sie den Ort Lage innerhalb der Prozentzeichens und wählen Sie einen anderen Ort, dann ausführen. Speichern unter....

```
WHERE /* Suchbedingungen */
(a.GebOrt LIKE '%Detmold%')
```

c) **Abfrage ändern:** Suchen Sie nach FName Berg

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat, a.GebOrt
FROM /* Tabellennamen */
Ahnen.db a
WHERE /* Suchbedingungen */
```

```
(a.FName LIKE '%BERG%')
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName
```

Hier habe ich GebOrt durch FName per Hand eingefügt,
Lage ersetzt durch BERG
Ausführen

Sie können die Abfrage auch nach GebDat ändern (alle möglichen Abfragen sehen Sie in SELECT).

```
WHERE /* Suchbedingungen */
(a.GebDat LIKE '%1850%')
```

und über einen Klick auf die PN-Nummer erhalten Sie sofort jeden einzelnen Datensatz.
Speichern unter....

II. Erweiterte Abfrage:

1. Aufgabenstellung: *Sie suchen alle Personen, die in Detmold **oder** Barntrup geboren **oder** getauft sind.*

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat, a.GebOrt, a.TaufOrt
FROM /* Tabellennamen */
Ahnen.db a
WHERE /* Suchbedingungen */
(a.GebOrt LIKE '%Detmold%') OR (a.TaufOrt LIKE '%Detmold%') OR (a.GebOrt
LIKE '%Barntrup%') OR (a.TaufOrt LIKE '%Barntrup%')
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName
```

Vorgehensweise:

Menü - Suchen - SQL - allgemein
Cursor steht unter FROM
Ich benötige die Ahnen.db, also
Leertaste - Verzeichnis Datei - Ahnen.db

Cursor springt unter SELECT

Linke Hand, Rechte Hand: Leertaste PN - Leertaste - FName - Leertaste - VName -
Leertaste - GebDat - Leertaste - GebOrt - Leertaste - TaufOrt

Cursor unter WHERE

Leertaste - GebOrt - Leertaste - LIKE - Leertaste - (kleines grünes Feld) Detmold -
enter - **Leertaste** - OR - Leertaste - TaufOrt - Leertaste - LIKE - Leertaste -
Detmold - **enter** - **Leertaste** - OR - Leertaste - GebOrt - Leertaste - LIKE -
Leertaste - Barntrup - **enter** - **Leertaste** - OR - Leertaste - TaufOrt - Leertaste -
LIKE - Leertaste - Barntrup

Cursor unter ORDER BY

Leertaste FName

Ausführen

Schriftlich über Abfrageexperte - unter Grundmenge aktuelle SQL-Abfrage -
Grundmenge - Tabelle sortieren usw.

Diese Abfrage abspeichern:

Menü - Suchen - SQL - Abfrage ändern - Datei - speichern unter

2. Abwandlung: *Sowohl als auch*

Sie möchten alle Personen, die **sowohl** in Detmold **als auch** in Barntrup geboren **oder** getauft sind.

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat, a.GebOrt, a.TaufOrt
FROM /* Tabellennamen */
Ahnen.db a
WHERE /* Suchbedingungen */
(a.GebOrt LIKE '%Detmold%') OR (a.TaufOrt LIKE '%Detmold%') AND
(a.GebOrt LIKE '%Barntrup%') OR (a.TaufOrt LIKE '%Barntrup%')
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName
```

3. Abwandlung: *Sie möchten alle Personen, die Berg heißen und in Detmold geboren oder getauft sind.*

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat, a.GebOrt, a.TaufOrt
FROM /* Tabellennamen */
Ahnen.db a
/* Suchbedingungen */
WHERE (a.FName LIKE '%Berg %') AND (a.GebOrt LIKE '%Detmold%') OR
(a.TaufOrt LIKE '%Detmold%')
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName
```

4. Abwandlung, *Sie möchten alle Personen, die Berg heißen, in Detmold geboren oder getauft sind und sortieren nach Name und Geburtsdatum*

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat, a.GebOrt, a.TaufOrt
FROM /* Tabellennamen */
Ahnen.db a
WHERE /* Suchbedingungen */
(a.FName LIKE '%Berg %') AND (a.GebOrt LIKE '%Detmold%') OR (a.TaufOrt
LIKE '%Detmold%')
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName , a. GebOrt
```

Diese Abfragen können Sie beliebig abwandeln

```
SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.GebDat, a.SterOrt, a.BestOrt
FROM /* Tabellennamen */
Ahnen.db a
WHERE /* Suchbedingungen */
(a.FName LIKE '%Berg %') AND (a.SterOrt LIKE '%Detmold%') OR (a.BestOrt
LIKE '%Detmold%')
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName , a. SterOrt
```

Sie können mit anderen Worten innerhalb der selektierten Felder kombinieren wie sie wollen, immer mit LIKE, AND und OR.

5. Abwandlung: *Sie können auch NOT LIKE verwenden,*
Bsp. OR (a.TaufOrt NOT LIKE '%Barntrup%'), das heißt Sie suchen Personen, die überall aber nicht in Barntrup getauft worden sind.

6. Abwandlung: *Sie suchen alle Personen mit Familienname Berg und Sterbort in Detmold und Sterbedatum 1850 (zu Übungszwecken ☺)*

```

SELECT /* Felder der Ergebnistabelle */
a.PN, a.FName, a.VName, a.SterDat, a.SterOrt
FROM /* Tabellennamen */
Ahnen.db a
WHERE /* Suchbedingungen */
(a.FName LIKE '%Berg %') AND (a.SterDat LIKE '%1850%') AND (a.SterOrt
LIKE '%Detmold%')
ORDER BY /* Felder, nach denen das Ergebnis sortiert wird */
a.FName , a. SterOrt

```

Ich denke, das Prinzip dieser allgemeinen Abfrage dürfte hiermit klar geworden sein. Die Kombinationsmöglichkeiten ergeben sich aus den gewählten SELECT - Feldern, mit OR, AND, AND NOT oder OR NOT können Sie in Where ihre Suchbedingungen eingeben.

Die Sortierfelder können Sie auch in der Reihenfolge bestimmen .

Sie können aber sie müssen nicht alle Selectfelder in WHERE benutzen, aber Sie können nur die SELECT-Felder benutzen. Sie können alle im Kontextmenü gezeigten SELECT-Felder eingeben, wie Sie möchten und entsprechend in WHERE abfragen.

III. Vorformulierte SQL - Abfragen

Im Menü *Suchen - SQL - Abfragen* findet man vorformulierte SQL - Abfragen Personen, Ehen, Lebensphasen, Orte und Medien. Ehen, Lebensphasen und Orte erleichtern die Arbeit, die Erleichterung bei Personen ist Geschmacksache - persönlich arbeite ich hier lieber mit Abfrage allgemein. SQL- Abfrage. Medien kann ich nicht erklären und lasse diesen Punkt unberücksichtigt.

Diesen vorformulierten Abfragen ist, mit Ausnahme Abfrage Medien, gemeinsam, daß Sie die für Anfänger so wichtige *Eingabehilfe* haben, *die sie hier aber selbst aktivieren müssen*. Dies machen Sie mit *SQL-Editor Abfragen - Optionen*

A. SQL - Abfrage Ehen.db

Der Befehl hierzu lautet übersetzt: Selektiere Personennummer Mann, Ehenummer Mann, Personennummer Frau, Ehenummer Frau, Art der Ehe, Datum, Ort , Text von der Ehen.db

Mit der von Ihnen aktivierten Eingabehilfe füllen Sie aus WHERE und ORDER BY.

```

SELECT
a.PNM, a.ENrM, a.PNF, a.ENrF , a.A, a.Datum, a.Ort, a.Text
FROM
Ehen a
WHERE
ORDER BY

```

Bewegen Sie den Mauszeiger unter WHERE und öffnen Sie mit der *Leertaste* das Kontextmenü. Darin stehen Ihnen die Optionen, nach denen sie Suchen können, zur Verfügung. Dann geben Sie das ein, wonach Sie abfragen wollen, z.B. Ort

Dann wieder *Leertaste* - Kontextmenü blau -

LIKE - grünes Feld Ort eingeben z.B. Lage (Erweiterbar mit OR oder AND oder OR NOT oder AND NOT)

Entertaste (!)

Cursor unter Order BY (Sortierklausel)

Leertaste

PNM Datum

Ausführen.

Hinweis: Da hier keine PN Nummer, sondern PNM und PNF

B. SQL - Abfrage Lebensphasen.db

Die obigen Ausführungen zur Ehen.db gelten entsprechend für die SQL-Abfrage in der Lebensphasen.db.

```
SELECT
  a.PN, a.Art, a.Folge, a.Vorzug, a.vonDatum, a.V, a.bisDatum,
  a.B, a.LebensPh, a.Ort, a.Zusatz
FROM
  LebPhasen.db a
WHERE
ORDER BY
```

Bei dieser Tabelle haben Sie PN -Nummern, d.h. Sie können die Datensätze nach und nach im Hauptfeld - Mehrzweckbereich ansehen und gegebenenfalls berichtigen. *Zurück zur Tabelle kommen Sie mit F11.*