

# Datenbanken für Techniker

<http://worgtsone.ostermann-rodgau.de> - [mailto: worgtsone @ hush.com](mailto:worgtsone@hush.com)

14. April 2008 – 13. Oktober 2011

## Inhaltsverzeichnis

<b>1</b>	<b>Wer braucht Datenbanken?</b>	<b>2</b>
1.1	Großunternehmen . . . . .	2
1.2	KMU . . . . .	2
1.3	Einzelunternehmen . . . . .	2
1.4	Übung : Richtig oder falsch? Ggf. korrigieren. . . . .	3
<b>2</b>	<b>DBMS und Tabellen</b>	<b>4</b>
2.1	Datentypen . . . . .	4
2.2	M\$ Access . . . . .	4
<b>3</b>	<b>TODO : Aufbau einer Tabelle am Beispiel</b>	<b>4</b>
3.1	Namen für Tabellen und Spalten . . . . .	4
3.2	Übung : Tabelle mp3 . . . . .	4
<b>4</b>	<b>SQL : SELECT, 1 von 3</b>	<b>5</b>
4.1	Übung : Sätze zu DBMS . . . . .	6
4.2	Übung : Finde Heino . . . . .	6
4.3	Übung : Diskutiere Pros und Cons von ID . . . . .	6
<b>5</b>	<b>Normalisierung</b>	<b>7</b>
5.1	Übung : Kunden mit Adressen . . . . .	7
<b>6</b>	<b>SELECT, 2 von 3</b>	<b>8</b>
6.1	Übung : Select an einer Tabelle . . . . .	8
<b>7</b>	<b>Mehrere Tabellen</b>	<b>9</b>
7.1	Beziehungen . . . . .	9
<b>8</b>	<b>SELECT, 3 von 3: Festhalten von Beziehungen</b>	<b>10</b>
8.1	Übung : Vaterlose Kinder (Waisen) . . . . .	10
8.2	Übung : mp3s mit mehreren Genres . . . . .	10
8.3	Übung : Viele Kinder mit vielen Hobbies . . . . .	10
8.4	Übung : Lesen Sie nach . . . . .	10
8.5	Übung : Finden Sie... . . . . .	10
8.6	Übung : Modellieren Sie eine Mediothek (aka Bibliothek) . . . . .	10

**Disclaimer**

Wissen ist zum Teilen da. Ich teile mein Wissen mit Ihnen, lieber Kollege.

Ich bin aber nicht perfekt. Unter [worgtsone@hush.com](mailto:worgtsone@hush.com) nehme ich dankbar Ihre Verbesserungsvorschläge entgegen.

\*

**Legal Blurb:** Alle Informationen in diesem Dokument sind falsch, unvollständig, irreführend, irrelevant und / oder funktionieren einfach nicht.

Wenn Sie es trotzdem benutzen, und es geht dabei etwas kaputt, ist das Ihr Problem, nicht meins.

\*

**Bitte teilen Sie meine Web-Adresse nicht Ihren Schülern mit.**

# 1 Wer braucht Datenbanken?

## 1.1 Großunternehmen

Großunternehmen wie die Telekom, Siemens oder Daimler können sich nicht merken, wieviele Mitarbeiter, Modelle oder Vermögenswerte sie haben.

Sie müssen's aber wissen. Ergo schreiben sie's auf.

Früher auf Papier, heute in Datenbanken. Und schwups! kann jedes Vorstandsmitglied heute nachschauen, wie groß der Umsatz gestern war. Nicht daß das jemand braucht. Aber es geht.

## 1.2 KMU

Kleine und Mittelständische Unternehmen haben die Geschäftsführung und normalerweise 5 Abteilungen:

- Vertrieb (Marketing, Pre- und AfterSales).

Die brauchen Datenbanken, um die Reisekosten und Geschäftsessen nachzuziehen und um zu gucken, welcher Vertreter im letzten Quartal den meisten Umsatz geschrieben hat.

- Lager.

Die brauchen Datenbanken. Wenn eine neue Lieferung kommt, SCANNT der Lagerarbeiter und SCHWUPS weiß das System, wieviele Rohre und Schrauben M6x30 noch da sind. Praktischerweise schaut es nachts nach, welche Bestände unter die Sollgrenze gefallen sind (zB es sind nur noch 5 Schrauben M6x30 da) und bestellt freundlicherweise automatisch 1000 Stück nach.

- Forschung und Entwicklung.

Die brauchen Datenbanken. Sie wollen schauen, wer wann wieviel Geld für welches Projekt ausgegeben hat. Auch die Reisekosten machen sich, einmal eingegeben, fast von selbst.

- FiBu (Finanzbuchhaltung).

Zahlen, Zahlen, nix als Zahlen. Eine Domäne für Datenbanken. Hier sind die richtigen Benutzerrechte (Wer darf User anlegen, Gehälter und Kontonummern ändern, Tabellen löschen?) besonders wichtig.

- Produktion.

Die brauchen Datenbanken. Sie wollen jeder Maschine vorbeugende Wartung geben, jedem Werkzeug vorbeugendes Nachschleifen, wollen die Stillstände der Deckel- mit denen der Nokaku-Maschinen vergleichen usw.

## 1.3 Einzelunternehmen

Bäcker, Zimmermann und Maurer freuen sich über vereinfachte Angebotserstellung, Lagerhaltung incl. Bestellung, Abrechnung und Rechnungsverfolgung (Wer hat schon bezahlt? Wer muß noch?).

## 1.4 Übung : Richtig oder falsch? Ggf. korrigieren.

1. Datenbanken sind für zeitgenössische Unternehmen überlebenswichtig.
2. Mehrere Datenbanken sitzen in einem DBMS.
3. Die FiBu darf ihr eigenes DBMS haben, das hinter Firewall und Proxy geschützt lebt, vieles liest, wenig verrät und keinen Außenstehenden schreiben läßt.
4. Die Datenbanken im DBMS haben Namen wie zB research, marketing, (.....) (.....) (.....)
5. Das DBMS sitzt auf einem Rechner mit Netzwerkanschluß.
6. Es beantwortet die Anfragen von wißbegierigen Rechnern und prüft anschließend, ob die das wirklich fragen oder sogar ändern dürfen.
7. Es erlaubt jedem, Benutzer, Datenbanken Tabellen anzulegen, zu ändern oder zu löschen.
8. Das DBMS sitzt auf einem zuverlässigen Athlon-200 mit 32MB und 4-GB-Platte.
9. Es steht neben der Heizung in einem nicht abschließbaren, weder erdbeben- noch hochwassergeschützten Raum.
10. Rechnerdiebe benehmen sich wie die Vandalen. Anschließend sind die Daten kaputt.
11. Datendiebe benehmen sich. Anschließend tauchen die Daten an Stellen auf, wo sie nicht hingehören.
12. Der Rechnerraum ist vom Empfang und vom Lager aus leicht zu erreichen.
13. Falls die Festplatte mit den Kunden, Produktionsplänen und Blaupausen geklaut wird und in China wieder auftaucht, kaufen wir eine neue.
14. Die Leute meißen ihre Anfragen in Sandsteinplatten. Das System kratzt seine Antworten in Wachstäfelchen. Das kann ganz schön lange dauern.
15. Bei Datenbanken geht Zuverlässigkeit vor Schnelligkeit.
16. Man kann Rechner mit dreifach-Backup, drei USVs und extrastabilem RAM kaufen, aber für Datenbanken und Fileserver lohnt sich das einfach nicht.

## 2 DBMS und Tabellen

Datenbanken sitzen in DBMS (Datenbank-Management-Systemen).

DBMS sitzen auf zuverlässigen Rechnern, sog. Servern.

Eine Datenbank enthält Tabellen mit Namen.

Die Tabellen enthalten Spalten und Zeilen.

Die Spalten geben die Struktur der Datensätze vor. Es gibt dabei nur wenig erlaubte Datentypen.

Die Zeilen sind die Datensätze. Was in einer Zeile steht, gehört zusammen.

### 2.1 Datentypen

TINYINT : von -127 bis 128.

INT : von -2147483648 bis 2147483647 (in access : Zahl).

DATETIME, TIMESTAMP : für Zeitpunkte und Zeitdifferenzen. In Windos ist Zeitpunkt 0 der 1.1.1901, 00:00 Uhr. In Apple mit Windos-Programen ist Zeitpunkt 0 der 1.1.1904, 00:00 Uhr. In Linux ist Zeitpunkt 0 der 1.1.1970, 00:00 Uhr. Der Start der Epoche. Der Un\*x-Epoche.

String : für Zeichenketten. Mit allen Zeichen, Ziffern und Satzzeichen außer ' und ".

### 2.2 M\$ Access

Access kann nur einen Subset von SQL und kennt keine Benutzer-Einschränkungen. Zum Spielen in der Schule ist es genau richtig.

## 3 TODO : Aufbau einer Tabelle am Beispiel

### 3.1 Namen für Tabellen und Spalten

**Benutzen Sie zum Basteln nur Klein- und Großbuchstaben sowie Ziffern. KEINE LEERZEICHEN. KEINE UMLAUTE. NIEMALS.**

### 3.2 Übung : Tabelle mp3

Sie wollen eine Tabelle kreieren, um Ihre MP3s zu verwalten. Jede MP3 hat einen Titel, einen Dateinamen, einen Interpreten, ein Baujahr und ein Genre.

Schlagen Sie eine Struktur vor, zB: id AutoWert PrimaryKey, Titel varchar, etc.

## 4 SQL : SELECT, 1 von 3

SQL ("Structured Query Language") ist eine Sprache, mit der sowohl Menschen als auch Maschinen Datenbanken abfragen<sup>1</sup> können.

So fragen Sie Ihre Datenbank nach Stücken von Heino mit folgendem SQL:

```
SELECT * FROM mp3s ...
... WHERE interpret="Heino" ;           // findet Gleichheit
WHERE interpret LIKE "Heino"           // findet auch Gleichheit

WHERE interpret LIKE "*e*"             // findet IRGENDWASeIRGENDWAS
WHERE interpret LIKE "e*"              // findet e oder E am Anfang
WHERE interpret LIKE "*e"              // findet e oder E am Ende

WHERE interpret LIKE "*o*" AND interpret LIKE "*e*" // rate mal
WHERE interpret LIKE "*o*" OR interpret LIKE "*e*" // rate mal
```

---

<sup>1</sup>und kreieren und Benutzer verwalten etc. – aber das machen wir nicht

## 4.1 Übung : Sätze zu DBMS

Korrigieren und/oder vervollständigen Sie die folgenden Sätze.

- Eine Abfrage greift auf Datensätze in Tabellen in Datenbanken in einem DBMS auf einem Server zu.
- DBMS heißt (.....)
- Programme, die stets zuverlässig Abfragen bearbeiten müssen, heißen (.....) Programme. Man läßt sie auf (.....) laufen, zu denen normale User keinen Zutritt haben. Diese werden daher auch (.....) genannt.
- Um eine Verbindung im ftp (.....) aufzubauen, brauche ich (lokal) einen (.....-.....) und (remote = entfernt) einen (.....-.....)
- FBI und CIA speichern ihre Daten vermutlich auf PC-Festplatten.

## 4.2 Übung : Finde Heino

Finden Sie in SQL

- alle mp3s von heino,
- alle mp3s von anneliese rothenberger,
- alle mp3s kürzer als 30 Sekunden,
- alle, wo der Interpret mit h anfängt und der Titel mit h aufhört.

## 4.3 Übung : Diskutiere Pros und Cons von ID

These 1: Man soll jedem Datensatz eine eindeutige ID vom Typ INT geben.

These 2: Gegenstände, zB in Webshops, speichert man mit id, Bezeichnung, Beschreibung, wasManSonstSoBraucht.

Diskutieren Sie die Thesen anhand des Beispiels "Jugendstilschrank, ca 1910".

## 5 Normalisierung

1. Wenn Sie überlegen, etwas als komma-getrennte Strings zu speichern, machen Sie zwei getrennte Datenfelder daraus.
2. Wenn andere Datensätze auf dieselben Daten zugreifen sollen, machen Sie eine Extra-Tabelle davon.

### 5.1 Übung : Kunden mit Adressen

Zu wieviel Tabellen kommen Sie, wenn Sie die obigen Sätze auf Kunden mit Adressen anwenden?

Wie vernünftig ist das?

Wie genau müssen die Aufgaben der Daba bekannt sein, BEVOR Sie ans Erstellen der Tabellen gehen können?

## 6 SELECT, 2 von 3

```
SELECT * FROM table;           // gibt alle Zeilen und alle Spalten
```

```
SELECT col03, col01 FROM table; // gibt nur die dritte und erste Spalte
```

```
SELECT * FROM table
  WHERE c1 = 'Meyer'           // gibt alle Zeilen, wo Meyer in Spalte c1 steht,
  ORDER BY c2 DESC;           // absteigend nach col2 sortiert.
```

```
SELECT * FROM table           // gibt alle Zeilen, wo c1 a ist
  WHERE c1='a'
  AND (c2='b' OR c3='c');      // und außerdem entweder c2=b oder c3 = c.
```

Rechenzeichen (+-\*/) sind bei Zahlen zulässig. count(), SUM(), AVG() auch.

### 6.1 Übung : Select an einer Tabelle

Saugen Sie <http://worgtsone.scienceontheweb.net/worgtsone/table01.csv>, und finden Sie

1. die größte und kleinste Zahl jeder Spalte
2. die größte Zeilensumme
3. das größte und kleinste Zeilenprodukt
4. die Anzahl der Zeilen, wo  $col01 > col02 > col03 > col04$

## 7 Mehrere Tabellen

### 7.1 Beziehungen

Sachen, die in einer Datenbank verwaltet werden sollen, können 3 Sorten Beziehungen haben:

1. 1:1-Beziehung:

Jedes A hat genau ein B, Jedes B hat genau ein A.

Gehört in eine Zeile.

2. 1:n-Beziehung:

Jedes A hat genau ein B, Jedes B hat 0 bis viele A.

Gehört in zwei Tabellen. Die eine Tabelle (Vater) braucht einen eindeutigen Identifizierer (id: Primärschlüssel, primary key). Die andere (Kind) greift auf diesen Identifizierer zu (Fremdschlüssel, foreign key).

Beispiel: mp3s mit mehreren Genres.

3. n:m-Beziehung:

Jedes A hat 0 bis viele B, Jedes B hat 0 bis viele A.

Gehört in drei Tabellen. Wird in 2 1:n-Beziehungen aufgelöst.

Beispiel : Kinder und Hobbies.

## 8 SELECT, 3 von 3: Festhalten von Beziehungen

```
SELECT ...  
  WHERE vater.pk = kind.fk;
```

pk = primary key, fk = (rate mal)...

### 8.1 Übung : Vaterlose Kinder (Waisen)

Was bedeutet das, wenn...

- pk gar nicht auftaucht
- pk mehrfach auftaucht
- wenn Sie in der Vater-Tabelle die Zeile mit dem pk löschen ?

### 8.2 Übung : mp3s mit mehreren Genres

Manchmal kann man ein Musikstück nicht in nur ein Genre einpassen.  
Entwerfen Sie die Datenbank.

### 8.3 Übung : Viele Kinder mit vielen Hobbies

Ein Kind kann 0..viele Hobbies haben.  
Mehrere Kinder können dasselbe Hobby haben.  
Entwerfen Sie die Datenbank.

### 8.4 Übung : Lesen Sie nach

DISTINCT und COUNT.

### 8.5 Übung : Finden Sie...

1. alle mp3s mit Genre RuB;
2. alle mp3s von SteveWonder;
3. alle mp3s mit genau 2 genres;
4. alle Kinder mit Hobby Bungee;
5. alle Hobbies von Klaus;
6. alle Kinder ohne Hobby;

### 8.6 Übung : Modellieren Sie eine Mediothek (aka Bibliothek)

Nichts (keine Namen, keine Adressen) ändert sich.  
Ein Medium ist zurückgegeben, wenn ein Rückgabedatum eingetragen ist.  
Nennen Sie letale Fehler dieses vereinfachenden Modells.