

### Tabellen erstellen

```
CREATE TABLE table_name
(
id data_type(size) NOT NULL,
column_name2 data_type(size),
column_name3 data_type(size),
....
PRIMARY KEY (id)
);
```

### Daten einfügen

```
INSERT INTO table_name
(column1,column2,column3,...)
VALUES (value1,value2,value3,...);
UPDATE table_name
SET column1=value1,column2=value2,...
WHERE some_column=some_value;
DELETE FROM table_name
WHERE some_column=some_value;
```

### Struktur einer Abfrage

```
SELECT [Überschriften/Spaltennamen,
Anweisungen]
FROM [Tabellennamen]
WHERE [Bedingungen, Gruppierung,
Sortierung]
```

### Beispiel Abfrage

```
SELECT einwohner/1000 AS "Einwohner"
FROM cia
WHERE region="Asien"
```

Einwohnerzahl der asiatischen Länder in der Tabelle cia in Tausenden angeben

### logische Operatoren

NOT	logische Negation (nicht)
OR	logisches ODER
AND	logisches UND
IS NULL	kein Wert eingetragen

Vor und nach einem logischen Operator muss eine vollständige Bedingung stehen.

### Vergleichsoperatoren

=	gleich
<>	nicht gleich
>	grösser als
<	kleiner als
>=	grösser oder gleich
<=	kleiner oder gleich

### Textbedingungen

%	beliebige Anzahl Zeichen
_	genau ein Zeichen

Diese Bedingungen können nur mit **LIKE** benutzt werden.

### Beziehungen

Die Verbindungen müssen explizit angegeben werden

```
tableID = fk_tableID
```

fk\_ bezeichnet den Fremdschlüssel

### Ausgabe aufteilen/beschränken

```
SELECT []
FROM []
WHERE []
LIMIT Startzeile, Anzahl
```

Limit 2,3 zeigt die 3 Elemente nach der Zeile 2

### Gruppenfunktionen

COUNT(expr)	Zeilen zählen
AVG(expr)	Durchschnitt
MIN(expr)	kleinster Wert
MAX(expr)	grösster Wert
SUM(expr)	Summe von Werten

Für *expr* gibt man die Spaltenüberschrift an

### Beispiel Abfrage II

```
SELECT count(*)
FROM mitarbeiter
WHERE abt = "Verkauf"
```

Anzahl der Mitarbeiter in der Abteilung "Verkauf"

### Gruppieren

```
SELECT []
FROM []
WHERE []
GROUP BY spalten_name
```

### Sortieren

```
SELECT []
FROM []
WHERE []
ORDER BY spalten_name
```

