

SQL

Einfache Abfragen

EINFACHE ABFRAGEN

Aufbau der **select**-Anweisung

select	<i>was</i>
from	<i>woher</i>
[where	<i>Bedingung]</i>
[group by	<i>Gruppierung]</i>
[having	<i>Gruppenbedingung]</i>
[order by	<i>Sortierung]</i>
[limit	<i>anzahl]</i>

EINFACHE ABFRAGEN

```
select    name  
from      kunde;
```

```
select    name, vorname  
from      kunde;
```

```
select    *  
from      kunde;
```

ABFRAGEN MIT FILTER

```
select    name  
from      kunde  
where     ort = 'Basel';
```

```
select    name, vorname  
from      kunde  
where     ort = 'Basel'  
and       anrede = 'Herr';
```


FILTER (WHERE-KLAUSEL)

- ▶ **=** gleich
- ▶ **<** kleiner
- ▶ **>** grösser
- ▶ **<>** ungleich
- ▶ **<=** kleiner oder gleich
- ▶ **>=** grösser oder gleich
- ▶ logische Verbindung
 - AND** und beide Bedingungen gleichzeitig
 - OR** oder mindestens eine Bedingung
 - NOT** nicht Negation
- ▶ Mustervergleich bei Zeichenketten mit LIKE
 - %** beliebige Zeichenfolge name like 'Mei**%**'
 - _** genau ein Zeichen name like 'M_**_**er'
- ▶ Test auf "leer"
 - is null**
 - is not null**

etwas Boolesche
Algebra einbringen

WEITERE FILTERBEDINGUNGEN

- ▶ where temperatur **>=** 5 and temperatur **<=** 6
- ▶ alternative: **between**
where temperatur **between** 5 and 6
- ▶ where ort = 'Basel' or ort = 'Bern' or ort = 'Lugano'
- ▶ alternative: **in**
where ort **in** ('Basel', 'Bern', 'Lugano')

SPALTEN UMBENENNEN

- ▶ In einer Tabelle sind die Spaltenbezeichnungen oftmals "nicht schön".

Mit **as** können "schönere" Spaltenüberschriften für die Ausgabe gesetzt werden.

- ▶

```
select nname as NachName,  
       vname as Vorname  
from Kunde;
```

DATENTYPEN

- ▶ Jede Spalte hat einen definierten Datentyp
- ▶ numerische Typen
 - ⇒ int
ganzzahlige Werte
 - ⇒ float, decimal
Werte mit Kommastellen (mit Punkt geschrieben)
- ▶ Zeichen und Zeichenketten (Strings)
 - ⇒ char, varchar, text
- ▶ Datum und Zeit
 - ⇒ date, time, timestamp

REGELN

- ▶ numerische Werte immer mit Punkt, wenn Kommastellen verwendet werden:
13 27.43 -8 .75
- ▶ Zeichenketten immer mit einfachen geraden Hochkommas
'Meier' 'Bern'
- ▶ Datum immer als Zeichenkette im Format YYYY-MM-DD angeben
'2014-01-30' für 30. Januar 2014
'17:25' nur Zeit
'2014-05-29 16:30' Datum und Zeit
- ▶ Ausgabe von Datum
date_format(geburtsdatum, '%d. %m %Y')
%d Tag, %m Monat numerisch, %M Monat als Wort, %y, %Y Jahr 2/4-stellig

ÜBUNGEN

- ▶ 1, einfache Abfragen
ca 20 Min
- ▶ 2, einfache Abfragen mit Datenselektion
ca 30 Min

SORTIEREN

Sortieren nach Spalten

```
select name, vorname  
from person  
order by name;
```

Auf- oder Absteigend

```
order by attr [asc | desc]
```

SORTIERTE ABFRAGE

```
select    name  
from      kunde  
order by  name;
```

```
select    name, vorname  
from      kunde  
order by  name,  
          vorname desc;
```

desc descending / absteigend

asc ascending / aufsteigend (**Default**)

DUPLIKATE VERMEIDEN

```
select name  
from person;
```

... findet 5 Meier

```
select distinct(name)  
from person;
```

zeigt nur noch einmal "Meier"

ÜBUNG

- ▶ 3, Sortieren
ca 20 Min