

# MySQL: HAVING

# Noch einmal der Merksatz:

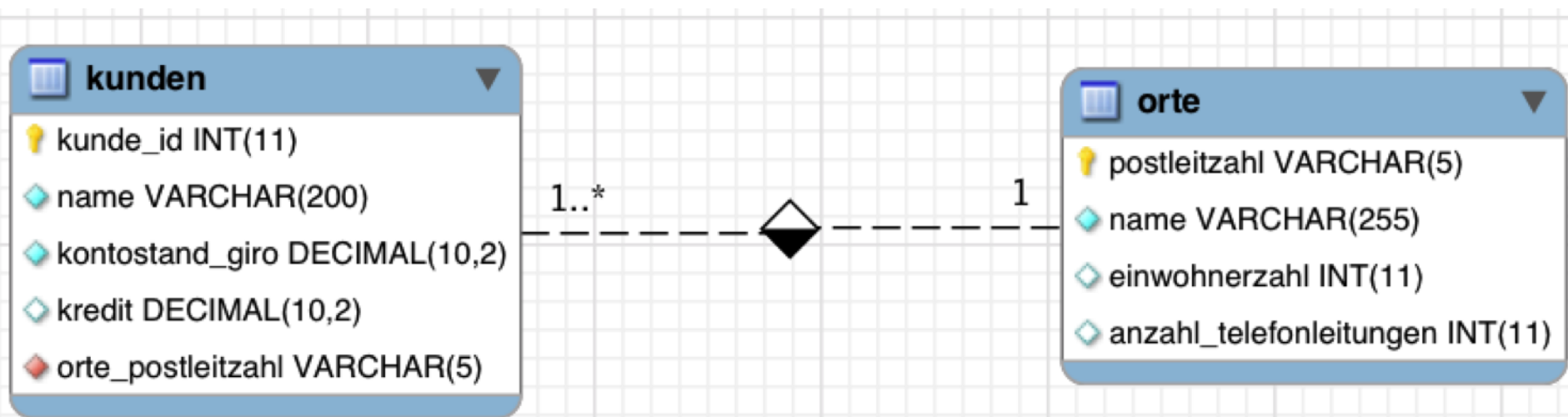
Warum geht Herbert oft laufen?

```
SELECT ...  
FROM ...  
WHERE ...  
GROUP BY ...  
HAVING ...  
ORDER BY ...  
LIMIT
```

# Beispieldatenbank "Kunden"

kunden (kunde\_id, name, ↑ort\_postleitzahl,  
kontostand\_giro, kredit)

orte (postleitzahl, name, einwohnerzahl,  
anzahl\_telefonleitungen)



# Beispieldatenbank "Kunden"

kunden (kunde\_id, name, ↑ort\_postleitzahl, kontostand\_giro, kredit)

orte (postleitzahl, name, einwohnerzahl, anzahl\_telefonleitungen)

postleitzahl	name	einwohnerzahl	anzahl_telefonleitungen
80995	München	1000000	385
79312	Emmendingen	40000	12
79111	Freiburg	280000	195
20095	Hamburg	2000000	1004

	kunde_id	name	ort_postleitzahl	kontostand_giro	kredit
▶ 1	1	John	79111	182.00	-430320.22
2	2	Herbert	79312	10291.32	-10000.00
3	3	Sabina	79312	-253.21	-3205.32
4	4	Mary	79111	-832.01	NULL
5	5	Heinrich	79111	15302.85	0.00
6	6	Usal	80995	23012.21	NULL
7	7	Johannes	80995	159.31	0.00
8	8	Carla	79312	503.06	-15302.68
9	9	Ludowika	79111	25201.07	-82213.99
10	10	Niemand	99999	-5021.30	-3024.21

für Copy-Paste

**www.informatikzentrale.de**

# Wiederholung: GROUP BY

```
SELECT
    orte.name, COUNT(*) AS anzahlKundenProOrt
FROM
    kunden, orte
WHERE
    orte.postleitzahl = kunden.ort_postleitzahl
GROUP BY
    orte.postleitzahl;
```

name	anzahlKundenProOrt
Emmendingen	3
Freiburg	4
München	2

# Eingrenzung einer Ergebnismenge

```
SELECT
    orte.name, COUNT(*) AS anzahlKundenProOrt
FROM
    kunden, orte
WHERE
    orte.postleitzahl = kunden.ort_postleitzahl
GROUP BY
    orte.postleitzahl;
```

name	anzahlKundenProOrt
Emmendingen	3
Freiburg	4
München	2

**Wir wollen nur Orte angezeigt bekommen,  
in denen mehr als 2 Kunden wohnen –  
also unsere schon gruppierte Ergebnismenge eingrenzen.**

# Eingrenzung einer Ergebnismenge

**Achtung: Fehler!**

```
SELECT
    orte.name, COUNT(*) AS anzahlKundenProOrt
FROM
    kunden, orte
WHERE
    orte.postleitzahl
AND
    anzahlKundenProOrt > 2 -- FALSCH!
GROUP BY
    orte.postleitzahl;
```

**Berechnetes oder aggregiertes Alias kann nicht in WHERE-Klausel verwendet werden!**

**Denn: WHERE-Klausel kann immer nur eine einzige Zeile überprüfen!**

name	anzahlKundenProOrt
Emmendingen	3
Freiburg	4
München	2

**Wir wollen nur Orte angezeigt bekommen, in denen mehr als 2 Kunden wohnen – also unsere schon gruppierte Ergebnismenge eingrenzen.**



# Eingrenzung einer Ergebnismenge

SELECT

2

orte.name, COUNT(\*) AS anzahlKundenProOrt

FROM

kunden, orte

WHERE

1

orte.postleitzahl

AND

anzahlKundenProOrt > 2 -- FALSCH!

GROUP BY

orte.postleitzahl

**Achtung: Fehler!**

**Berechnetes oder aggregiertes Alias kann nicht in WHERE-Klausel verwendet werden!**

**WHERE-Klausel kann immer nur eine einzige Zeile überprüfen!**

**Das könnte man sich so merken:**

Zuerst wird das Ergebnis mit WHERE eingeschränkt (1),  
DANN erst wird aggregiert (z.B. mit COUNT(...) gezählt) (2).

Diese Zahl kann also nicht in der WHERE-Klausel verwendet werden.

**Wir wollen**

**in denen mehr als 2 Kunden wohnen –  
also unsere schon gruppierte Ergebnismenge eingrenzen.**

**Berechnetes oder  
aggregiertes Alias kann nicht  
in WHERE-Klausel verwendet  
werden!**

**Denn:  
WHERE-Klausel kann immer nur  
eine einzige Zeile überprüfen!**

# Eingrenzung einer Ergebnismenge mit HAVING

```
SELECT
```

```
    orte.name, COUNT(*) AS anzahlKundenProOrt
```

```
FROM
```

```
    kunden, orte
```

```
WHERE
```

```
    orte.postleitzahl = kunden.ort_postleitzahl
```

```
GROUP BY
```

```
    orte.postleitzahl
```

```
HAVING
```

```
    anzahlKundenProOrt > 2;
```

	name	anzahlKundenProOrt
►	Freiburg	4
	Emmendingen	3

**Wir wollen nur Orte angezeigt bekommen,  
in denen mehr als 2 Kunden wohnen –  
also unsere schon gruppierte Ergebnismenge eingrenzen.**

# Eingrenzung einer Ergebnismenge mit HAVING

```
SELECT
    orte.name, SUM(kredit)
    AS `Kredite in den einzelnen Orten`
FROM
    kunden, orte
WHERE
    orte.postleitzahl = kunden.ort_postleitzahl
GROUP BY
    kunden.ort_postleitzahl;
```

	name	Kredite in den einzelnen Orten
►	Freiburg	-512534.21
	Emmendingen	-28508.00
	München	0.00

**Aufgabe: Zeige nur Orte, in denen  
mehr als 500.000 Euro Kredite vergeben wurden.**

# Eingrenzung einer Ergebnismenge mit HAVING

**Achtung: Fehler!**

```
SELECT
    orte.name, SUM(kredit)
    AS `Kredite in den einzelnen Orten`
FROM
    kunden, orte
WHERE
    orte.postleitzahl = kunden.ort_postleitzahl
    AND
    SUM(kredit) < -500.000 -- FALSCH!
GROUP BY
    kunden.ort_postleitzahl
```

name	Kredite in den einzelnen Orten
Freiburg	-512534.21
Emmendingen	-28508.00
München	0.00

**Aufgabe: Zeige nur Orte, in denen  
mehr als 500.000 Euro Kredite vergeben wurden.**

# Eingrenzung einer Ergebnismenge mit HAVING

**Achtung: Fehler!**

```
SELECT
    orte.name, SUM(kredit)
    AS `Kredite in den einzelnen Orten`
FROM
    kunden, orte
WHERE
    orte.postleitzahl
    AND SUM(kredit) ist ein Aggregat (= Zusammenfassung) -- tzahl
    -- deshalb HAVING verwenden
    SUM(kredit) < -500000 -- FALSCH!
GROUP BY
    kunden.ort_postleitzahl;
```

**Aufgabe: Zeige nur Orte, in denen  
mehr als 500.000 Euro Kredite vergeben wurden.**

# Eingrenzung einer Ergebnismenge mit HAVING

```
SELECT
    orte.name, SUM(kredit)
    AS `Kredite in den einzelnen Orten`
FROM
    kunden, orte
WHERE
    orte.postleitzahl = kunden.ort_postleitzahl
GROUP BY
    kunden.ort_postleitzahl
HAVING
    SUM(kredit) < -500000
```

	name	Kredite in den einzelnen Orten
►	Freiburg	-512534.21

Aufgabe: Zeige nur Orte, in denen  
mehr als 500.000 Euro Kredite vergeben wurden.

# HAVING

## Beispiel bei einfacher Addition

```
SELECT
    k.name AS n, o.name AS ortname,
    (kontostand_giro + kredit) AS bilanz
FROM
    kunden as k, orte as o
WHERE
    k.ort_postleitzahl = o.postleitzahl
HAVING
    bilanz < 0
```

n	ortname	bilanz
Sabina	Emmendingen	-3458.53
Carla	Emmendingen	-14799.62
John	Freiburg	-430138.22
Ludowika	Freiburg	-57012.92



# HAVING

**Achtung: Fehler!**

## Beispiel bei einfacher Addition

```
SELECT
    k.name AS n, o.name AS ortname,
    (kontostand_giro + kredit) AS bilanz
FROM
    kunden as k, orte as o
-- WHERE
--     k.ort_postleitzahl = o.postleitzahl
-- AND
--     bilanz < 0
```

Achtung, Fehler:

berechnetes Alias kann nicht in WHERE-Bedingung verwendet werden!

deshalb

...

HAVING

bilanz < 0