

Informatik: Datenbanken

Aggregationsfunktionen, Gruppierung, Verzweigungen

Inhalte

- ▶ Wiederholung: Structured Query Language (SQL)

Inhalte

- ▶ Wiederholung: SQL
- ▶ Aggregatsfunktionen: COUNT, MAX, MIN, SUM, AVG, LENGTH

Inhalte

- ▶ Wiederholung: `SQL`
- ▶ Aggregatsfunktionen: `COUNT`, `MAX`, `MIN`, `SUM`, `AVG`, `LENGTH`
- ▶ Gruppierung: `GROUP BY`

Inhalte

- ▶ Wiederholung: `SQL`
- ▶ Aggregatsfunktionen: `COUNT`, `MAX`, `MIN`, `SUM`, `AVG`, `LENGTH`
- ▶ Gruppierung: `GROUP BY`
- ▶ Erste n Zeilen: `LIMIT`

Inhalte

- ▶ Wiederholung: `SQL`
- ▶ Aggregatsfunktionen: `COUNT`, `MAX`, `MIN`, `SUM`, `AVG`, `LENGTH`
- ▶ Gruppierung: `GROUP BY`
- ▶ Erste n Zeilen: `LIMIT`
- ▶ Verzweigungen: `CASE WHEN`

Einfache Selektion: **SELECT**

Aufgabe 1.1

Geben Sie die Namen und Wohnorte aller Mitglieder aus.

Lösungsvorschlag zu Aufgabe 1.1

```
SELECT username, city  
FROM users
```

Selektion mit Filter: WHERE

Aufgabe 1.2

Geben Sie alle weiblichen Mitglieder aus, die zwischen 150 und 155 gross sind.

✓ Lösungsvorschlag zu Aufgabe 1.2

```
SELECT name, gender, centimeters
FROM users
WHERE gender="female"
AND centimeters between 150 and 155
```

Eindeutige Selektion: `SELECT DISTINCT`

Aufgabe 1.3

Geben Sie jeden Wohnort nur einmal aus.

✓ Lösungsvorschlag zu Aufgabe 1.3

```
SELECT DISTINCT city  
FROM users
```

Ungefähre Matches: LIKE

Aufgabe 1.4

Wählen Sie alle Personen mit dem Namen Naomi aus, die nicht aus Berlin kommen.

Lösungsvorschlag zu Aufgabe 1.4

```
SELECT name, city
FROM users
WHERE name LIKE "Naomi%"
AND city <> "Berlin"
```

Sortieren: ORDER BY

Aufgabe 1.5

Geben Sie alle Benutzernamen in sortierter Reihenfolge aus (a → z).

Lösungsvorschlag zu Aufgabe 1.5

```
SELECT username  
FROM users  
ORDER BY username ASC
```

Aggregatsfunktionen: COUNT, MAX, MIN, SUM, AVG, LENGTH

Aufgabe 1.6

Geben Sie die Anzahl registrierten Mitglieder in Berlin aus. Die resultierende Spalte soll „Registrierte Mitglieder in Berlin“ heissen.

Lösungsvorschlag zu Aufgabe 1.6

```
SELECT COUNT(*) AS "Registrierte  
    Mitglieder in Berlin"  
FROM users  
WHERE city="Berlin"
```

Gruppieren: GROUP BY

Aufgabe 1.7

Geben Sie die durchschnittliche Körpergrösse aller Mitglieder in jeder Stadt aus. Zeigen Sie nur die Städte, in denen die Menschen im Durchschnitt zwischen 150 und 155 gross sind.

Lösungsvorschlag zu Aufgabe 1.7

```
SELECT city, AVG(centimeters) AS "  
    Durchschnittliche Körpergrösse"  
FROM users  
GROUP BY city  
HAVING `Durchschnittliche Körpergrösse`  
    BETWEEN 150 AND 155
```

Erste n Zeilen: LIMIT

Aufgabe 1.8

Zeigen Sie die Namen und Körpergröße der 5 grössten Mitglieder an.

Lösungsvorschlag zu Aufgabe 1.8

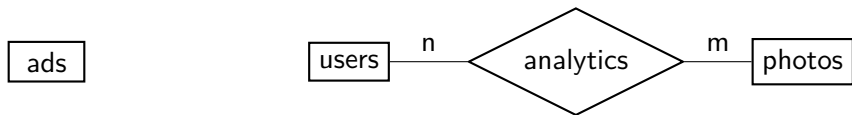
```
SELECT name, centimeters  
FROM users  
ORDER BY centimeters DESC  
LIMIT 5
```

Verzweigungen: CASE WHEN

```
SELECT name, centimeters,  
CASE  
    WHEN centimeters > 179 THEN 'Gross'  
    WHEN centimeters < 179 THEN 'Klein'  
    ELSE 'Genau im Durchschnitt'  
END AS Durchschnittlich  
FROM users  
WHERE gender="male"
```

Werbungs-Teil von InstaHub

Entity-Relationship Model (ERM)




Attribute von der Tabelle analytics

Was wird alles erfasst beim Browsen?

Feld	Beschreibung
id	Primärschlüssel, fortlaufende Nummer
ip	Die ersten drei Blöcke der IPv4-Adresse
device	Desktop, mobile, tablet oder bot
brand_family	Z.B. Apple oder Samsung
brand_model	Z.B. GALAXY S5
browser_family	Z.B. Firefox, Chrome, Safari
browser_version	Die Versionsnummer des Browsers
platform_family	Z.B. Windows, Mac, Linux, iOS, Android
platform_version	Die Versionsnummer der Plattform
user_id	Benutzer der sich das Foto angesehen hat
photo_id	Angesehenes Foto
created_at	Zeitpunkt, als das Foto sich angesehen wurde
updated_at	I.d.R. = created_at

Personalisierte Werbung

 Beispiel auf [InstaHub](#)

Auftrag

Aggregatsfunktionen

Moodle



2.1 - 2.3



Challenge: Aufgabe 2.4

Skript



Challenge: Aufgabe 1.31 (GROUP BY)



Lesen Sie 1.2.11



Challenge: Aufgabe, 1.32, 1.33



Challenge: Aufgaben 1.34, 1.35