

# Blatt 11 - Lars Groeber - Gruppe 7

## Aufgabe 1

---

a)

- Es liegt kein Fehler vor, Ergebnis ist 0.
- Es liegt ein Fehler vor, die Spalte "name" ist nicht eindeutig.
- Es liegt kein Fehler vor, Ergebnis ist ein Eintrag (id = 3, sum(promineny) = 12025)

b)

```
select id as ID, name as Name, hoehe_ueber_nn - prominenz as Differenz
from berg where prominenz < hoehe_ueber_nn;
```

```
select b.id, b.name, bs.jahr from berg b
  join bestiegen bs on b.id = bs.id_berg
  join bergsteiger br on bs.id_bergsteiger = br.id
where br.name = "Gerlinde" and br.vorname = "Kaltenbrunner"
```

```
select b.id, b.name from berg b
  join bestiegen bs on b.id = bs.id_berg
group by b.id
having count(b.id) > 1
order by b.hoehe_ueber_nn desc
```

## Aufgabe 2

---

```
DROP DATABASE IF EXISTS Lars_Groeber;
CREATE DATABASE Lars_Groeber COLLATE "utf8_general_ci";
USE Lars_Groeber;

# a)
# column names are too long...
CREATE TABLE gesundheit
(
  Codes INT NOT NULL PRIMARY KEY,
  Stadtteil VARCHAR(255) NOT NULL,
  a INT NOT NULL,
  b INT NOT NULL,
  c INT NOT NULL,
  d INT NOT NULL,
  e INT NOT NULL,
  f INT NOT NULL
);

# b)
SET GLOBAL local_infile = 'ON';
LOAD DATA LOCAL INFILE
'/Users/lars/Downloads/gesundheit_einfach.csv' INTO TABLE Lars_Groeber.gesundheit
CHARACTER
SET 'latin2'
FIELDS TERMINATED BY ';'
ENCLOSED BY '"' LINES TERMINATED BY '\n' IGNORE 1 LINES;

# c)
CREATE VIEW revier_telefonliste
AS
```

```

SELECT
    name,
    telefon
FROM polizeirevier;

CREATE VIEW strassen_polizei
AS
    SELECT
        p.nr AS polizei_nr,
        p.name AS polizei_name,
        p.telefon,
        p.flaeche,
        p.einwohner,
        p.pendler,
        s.nr AS strasse_nr,
        s.name AS strasse_name,
        s.auftrennungstyp,
        s.von,
        s.bis,
        s.stadtteil,
        s.postleitzahl
    FROM polizeirevier p
        JOIN strassenverzeichnis s ON s.polizeirevier = p.nr;

CREATE VIEW strassen_aerzte
AS
    SELECT
        s.nr AS strasse_nr,
        s.name AS strasse_name,
        s.auftrennungstyp,
        s.von,
        s.bis,
        s.postleitzahl,
        g.*
    FROM strassenverzeichnis s
        JOIN gesundheit g on g.Stadtteil = s.stadtteil;

# d)
SELECT DISTINCT telefon
from strassen_polizei
where strasse_name = "Berger Straße";

SELECT strasse_name, Stadtteil, max(f)
from strassen_aerzte
group by Stadtteil, strasse_name
having max(f) = (SELECT max(f)
from strassen_aerzte);

```

## Aufgabe 3

---

**a)**

Nicht rechts: {A, B, E}

ABE = {A, B, E, C, D}

Schlüssel: {ABE}

**b)**

Nicht rechts: {C, B}

CB = {C, B, A, D, E}

Schlüssel: {CB}

**c)**

Nicht rechts: {}

$D = \{D, A, B, C, E\}$   $AB = \{A, B, D, E, C\}$   $CE = \{C, E, A, B, D\}$

Schlüssel: {D, AB, CE}

**d)**

Nicht rechts: {A}

$A = \{A\}$   $AB = \{A, B, C, E, D\}$   $AD = \{A, D, B, C, E\}$   $AE = \{A, E, D, B, C\}$

Schlüssel: {AB, AD, AE}

**e)**

Nicht rechts: {A}

$A = \{A\}$   $AB = \{A, B\}$   $AC = \{A, C\}$   $AD = \{A, D, B, C, E\}$   $AE = \{A, E, D, B, C\}$

Schlüssel: {AD, AE}