



Vorbereitung

Erstellen und importieren Sie zuerst die Datenbank „mondial“ (das Script „mondial-schema-mysql.sql“ ist zum Erstellen und „mondial-inputs-mysql.sql“ zum Importieren der Daten).

Aufgaben

1. Zeigen Sie alle Einträge der Tabelle „country“.
2. Zeigen Sie von dieser Tabelle nur den Namen und die Hauptstadt („capital“) und benennen Sie die Überschriften mit „Land“ bzw. „Hauptstadt“.
3. Filtern Sie das Resultat, sodass nur diejenigen Länder angezeigt werden, die mehr als 10 Mio. Einwohner haben.
4. Zeigen Sie nur diejenigen Länder an, die weniger als 100'000 Einwohner haben.
5. Zeigen Sie in der „city“-Tabelle den Namen (als „Ort“), das Land (als „Landkurzzeichen“) sowie die Einwohnerzahl („Einwohner“) derjenigen Städte, die höchstens 10'000 Einwohner haben.
6. Zeigen Sie nur die Ländercodes an, die Städte mit mehr als 1 Mio. Einwohner haben.
7. Zeigen Sie die vorherigen Ländercodes sortiert an und jeden Code nur einmal (`⌈ DISTINCT`).
8. Zeigen Sie diejenigen Städte, die höchstens 10'000 Einwohner haben sowie diejenigen, die mind. 8 Mio. Einwohner haben.
9. Zeigen Sie diejenigen Städte, die mind. 100'000 aber höchstens 101'000 Einwohner haben (`⌈ BETWEEN`).
10. Zeigen Sie die Schweizer Städte, deren Einwohnerzahl nicht eingetragen ist.
11. Zeigen Sie die Städte aus Sambia (Code „Z“), deren Einwohnerzahl bekannt ist.
12. Zeigen Sie die Ländernamen derjenigen Länder, die mit „Li“ beginnen.
13. Zeigen Sie die Städte ausserhalb der Schweiz, die mit „Aa“ beginnen.
14. Zeigen Sie die Städte der Schweiz, die mit „el“ enden sowie diejenigen aus Deutschland (Code „D“), die ein „z“ enthalten.
15. Zeigen Sie die Ländereinträge mit den Ländercodes „A“, „CH“, „D“, „FL“ (`⌈ IN`).
16. Zeigen Sie alle Seen („lake“), beginnend beim kleinsten („area“).
17. Zeigen Sie alle Berge („mountain“), beginnend beim höchsten („height“).



Abfragen mit Aggregat-Funktionen:

18. COUNT():
Wieviele amerikanische Städte sind in der Datenbank eingetragen?
19. COUNT():
Wie viele amerikanische Städte haben eine Einwohnerzahl angegeben?
20. SUM():
Berechnen Sie die Gesamt-Einwohnerzahl aller amerikanischen Städte.
21. AVG():
Berechnen Sie die durchschnittliche Einwohnerzahl aller Städte, die in der Datenbank eingetragen sind.
22. AVG():
Berechnen Sie die durchschnittliche Einwohnerzahl aller amerikanischen Städte.
23. MIN(), MAX():
Zeigen Sie die Höhe („height“) des höchsten Berges („mountain“).
24. Zeigen Sie die Fläche („area“) des kleinsten Sees („lake“).

Gruppieren von Aggregat-Funktionen:

25. AVG():
Berechnen Sie die durchschnittliche Einwohnerzahl aller Städte pro Land. Zeigen Sie das Land und die durchschnittliche Einwohnerzahl. Sortieren Sie die Ausgabe absteigend nach dem Durchschnitt und runden Sie den Durchschnitt auf ganze Zahlen.
26. HAVING:
Zeigen Sie bei den durchschnittlichen Einwohnerzahlen aller Städte pro Land nur diejenigen Länder, deren Durchschnitt höher ist als 1'000'000.
27. MIN(), MAX():
Zeigen Sie pro Land die grösste und die kleinste Stadt. Zeigen Sie den Ländercode und die Einwohnerzahl der jeweiligen Städte. Sortieren Sie die Ausgabe aufsteigend nach Ländercode. Wieso können Sie die Namen der Städte nicht anzeigen lassen?



Abfrage über mehrere Tabellen:

28. Zeigen Sie alle Sprachen pro Land an. Für alle Länder soll der Ländername ersichtlich sein. Geben Sie alle Spalten der beiden Tabellen aus. Zeigen Sie auch diejenigen Länder an, bei denen keine Sprache eingetragen ist.
29. Geben Sie bei der vorherigen Abfrage nur die zwei Spalten mit den Namen der Länder und der Sprachen aus und sortieren Sie nach Ländername. Geben Sie den Spalten geeignete Namen.
30. Geben Sie bei der vorherigen Abfrage nur diejenigen Länder aus, die auch mind. eine Sprache eingetragen haben.
31. Zeigen Sie diejenigen Ländernamen an, bei denen die Kindersterblichkeit (Tabelle „population“ / Attribut „infant_mortality“) grösser als 100 ist. Ordnen Sie die Ausgabe so, dass die höchste Sterblichkeit zuoberst ist.
32. Zeigen Sie alle Ländernamen derjenigen Länder an, die „normale“ Mitglieder des Commonwealth sind (□ „is_member“ und „organization“).



Lösungen

1.

```
SELECT  *
FROM    country;
```
2.

```
SELECT  name      AS Land,
        capital   AS Hauptstadt
FROM    country;
```
3.

```
SELECT  name      AS Land,
        capital   AS Hauptstadt
FROM    country
WHERE   population > 10000000;
```
4.

```
SELECT  name      AS Land,
        capital   AS Hauptstadt
FROM    country
WHERE   population < 100000;
```
5.

```
SELECT  name      AS Ort,
        country   AS Landkurzzeichen,
        population AS Einwohner
FROM    city
WHERE   population <= 10000;
```
6.

```
SELECT  country
FROM    city
WHERE   population > 5000000;
```
7.

```
SELECT  DISTINCT country
FROM    city
WHERE   population > 5000000
ORDER BY country;
```
8.

```
SELECT  name      AS Ort,
        country   AS Landkurzzeichen,
        population AS Einwohner
FROM    city
WHERE   population <= 3000
OR      population >= 8000000;
```
9.

```
SELECT  name      AS Ort,
        country   AS Landkurzzeichen,
        population AS Einwohner
FROM    city
WHERE   population BETWEEN 100000 AND 101000;
```
10.

```
SELECT  name      AS Ort,
        country   AS Landkurzzeichen,
        population AS Einwohner
FROM    city
```



```
WHERE country = 'CH'
AND population IS NULL;
```

```
11. SELECT name          AS Ort,
           country       AS Landkurzzeichen,
           population    AS Einwohner
FROM city
WHERE country = 'Z'
AND population IS NOT NULL;
```

```
12. SELECT name          AS Land
FROM country
WHERE name LIKE 'li%';
```

```
13. SELECT name          AS Ort,
           country       AS Landkurzzeichen,
           population    AS Einwohner
FROM city
WHERE country <> 'CH'
AND name LIKE 'aa%';
```

```
14. SELECT name          AS Ort,
           country       AS Landkurzzeichen,
           population    AS Einwohner
FROM city
WHERE (country = 'CH'
AND name LIKE '%el')
OR (country = 'D'
AND name LIKE '%z%');
```

```
15. SELECT *
FROM country
WHERE code IN ('CH', 'A', 'D', 'FL');
```

```
16. SELECT *
FROM lake
ORDER BY area;
```

```
17. SELECT *
FROM mountain
ORDER BY height DESC;
```



Abfragen mit Aggregat-Funktionen:

- ```
18. SELECT COUNT(*)
 FROM city
 WHERE country = 'USA';

19. SELECT COUNT(population)
 FROM city
 WHERE country = 'USA';

20. SELECT SUM(population)
 FROM city
 WHERE country = 'USA';

21. SELECT AVG(population)
 FROM city;

22. SELECT AVG(population)
 FROM city
 WHERE country = 'USA';

23. SELECT MAX(height)
 FROM mountain;

24. SELECT MIN(area)
 FROM lake;
```

**Gruppieren von Aggregat-Funktionen:**

- ```
25. SELECT  country                AS Laendercode,
           ROUND(AVG(population)) AS Einwohnerdurchschnitt
    FROM    city
    GROUP BY country
    ORDER BY AVG(population) DESC;

26. SELECT  country                AS Laendercode,
           ROUND(AVG(population)) AS Einwohnerdurchschnitt
    FROM    city
    GROUP BY country
    HAVING  AVG(population) > 1000000
    ORDER BY AVG(population) DESC;

27. SELECT  country                AS Laendercode,
           MIN(population)         AS MinEinwohner,
           MAX(population)         AS MaxEinwohner
    FROM    city
    GROUP BY country
    ORDER BY Laendercode;
```

Städtenamen können nicht angezeigt werden, weil sie nicht mit der Aggregat-Funktion verknüpft werden können, d.h. es würde eine beliebige Stadt angezeigt (normalerweise die erste im Alphabet).



Abfrage über mehrere Tabellen:

```
28. SELECT  *
      FROM    country
            LEFT JOIN language ON country.code = language.country
```

```
29. SELECT  c.name      AS Land,
            l.name      AS Sprache
      FROM    country AS c
            LEFT JOIN language AS l ON c.code = l.country
      ORDER BY c.name
```

```
30. SELECT  c.name      AS Land,
            l.name      AS Sprache
      FROM    country AS c
            INNER JOIN language AS l ON c.code = l.country
      ORDER BY c.name
```

```
31. SELECT  c.name      AS Land,
            p.infant_mortality AS Kindersterblichkeit
      FROM    country AS c
            LEFT JOIN population AS p ON c.code = p.country
      WHERE   p.infant_mortality > 100
      ORDER BY p.infant_mortality DESC
```

32. Mögliche Entwicklung des Statements:

```
SELECT  *
FROM    organization
WHERE   name = 'commonwealth'
```

```
SELECT  *
FROM    organization AS o
      LEFT JOIN is_member m ON m.organization = o.abbreviation
WHERE   name = 'commonwealth'
```

```
SELECT  *
FROM    organization AS o
      LEFT JOIN is_member im ON im.organization = o.abbreviation
      LEFT JOIN country c   ON c.code = im.country
WHERE   o.name = 'commonwealth'
```

```
SELECT  c.name AS 'Ländername'
FROM    organization AS o
      LEFT JOIN is_member AS im ON im.organization = o.abbreviation
      LEFT JOIN country AS c   ON c.code = im.country
WHERE   o.name = 'commonwealth'
AND     im.type = 'member'
ORDER BY c.name
```