

--AB100-07.sql

--6 Tabelle löschen

--DROP TABLE MYFRIENDS

--DROP TABLE ORT

--Seite 4

--6 Tabelle erstellen (MYFRIENDS)

```
CREATE TABLE MYFRIENDS
  (ID INTEGER PRIMARY KEY
  ,Name TEXT
  ,Vorname TEXT
  ,email TEXT
  ,FKORTSID INTEGER);
```

--6 Tabelle erstellen (ORT)

```
CREATE TABLE ORT
  (ORTSID INTEGER PRIMARY KEY
  ,PLZ TEXT
  ,Ortsname TEXT);
```

--Seite 5

--7 Einfügen von Datenwerten in Tabelle Ort

```
INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('3001', 'Bern');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('3005', 'Bern');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('8001', 'Zürich');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('8002', 'Zürich');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('3280', 'Murten');
```

--7 Weitere Daten einfügen

```
INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('3715', 'Adelboden');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('4053', 'Basel');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('1200', 'Genf');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('4105', 'Biel');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('3922', 'Stalden');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('1400', 'Yverdon');

INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('5600', 'Lenzburg');
```

--Seite 5/6

--8 Kontrolle der Tabellen (SELECT)

--8.1 Alles kontrollieren von einer Tabelle

```
SELECT * FROM ORT;
SELECT * FROM MYFRIENDS;
```

--8.2 Zählt alle Zeilen als "Anzahl Orte" von der Tabelle ORT

```
SELECT COUNT(*) AS "Anzahl Orte" FROM ORT;
```

--8.3 Zeigt die Spalte Ortsname nach dem ABC geordnet an

```
SELECT Ortsname FROM ORT
  ORDER BY Ortsname ASC;
```

--8.3 Zeigt die Spalte Ortsname nach dem ABC geordnet rückwärts

```
SELECT Ortsname FROM ORT
  ORDER BY Ortsname DESC;
```

--8.4 Summe von PLZ berechnen

```
SELECT SUM(PLZ) AS "Summe PLZ" FROM ORT;
```

--8.5 Zeigt PLZ und Ortsnamen an, wenn der Anfangsbuchstaben bei Ortsname kleiner als G ist

```
SELECT PLZ, Ortsname FROM ORT
  WHERE Ortsname < 'G%';
```

--8.6 Listet alle PLZ, Ortsnamen auf welche bei Ortsnamen als Anfangsbuchstaben B haben

```
SELECT PLZ, Ortsname FROM ORT
  WHERE Ortsname LIKE 'B%';
```

--8.7 Zeigt alle PLZ und Ortsnamen an wenn der Anfangsbuchstaben bei Ortsname B oder S ist

```
SELECT PLZ, Ortsname FROM ORT
  WHERE Ortsname LIKE 'B%'
  OR Ortsname LIKE 'S%';
```

--8.8 Zeigt den Ortsnamen mit dem kleinsten Buchstaben an(A,B,C)

```
SELECT MIN(Ortsname) FROM ORT;
```

--8.9 Zeigt den Ortsnamen mit dem grössten Buchstaben an(A,B,C)

```
SELECT MAX(Ortsname) FROM ORT;
```

--8.10 Gibt nichts aus...

```
SELECT PLZ, Ortsname FROM ORT
  WHERE Ortsname LIKE 'B%'
  AND Ortsname LIKE 'S%';
```

--8.11 Zeigt alle Orte an wessen Anfangsbuchstaben nicht B ist

```
SELECT PLZ, Ortsname FROM ORT
  WHERE Ortsname NOT LIKE 'B%';
```

--8.12 Gruppirt den Ortsnamen nach dem Alphabet

```
SELECT PLZ, Ortsname FROM ORT
  GROUP BY Ortsname;
```

--8.13 Zeigt 10 % des Mittelwertes von der Spalte PLZ

```
SELECT AVG(PLZ) * 0.1 AS "10%-Durchschnitt" FROM ORT;
```

--Seite 7

--9 Daten löschen (DELETE FROM)

--Daten einfügen

```
INSERT INTO ORT
  (PLZ, ORTSNAME) VALUES
  ('', 'Müllheim');
```

--9.1 Daten (Zeile) löschen

```
DELETE FROM ORT
  WHERE Ortsname = 'Müllheim';
```

--9.2 Tabellen inhalt löschen?

--DELETE FROM ORT;

--Seite 7/8

--10 Daten aktualisieren (UPDATE)

--Daten einfügen

```
INSERT INTO MYFRIENDS
  (Name, Vorname, email) VALUES
  ('Maurer', 'Ralph',
  'ralph.maurer@gibb.ch');

INSERT INTO MYFRIENDS
  (Name, Vorname, email) VALUES
  ('Frei', 'Beat',
  'beat.frei@gibb.ch');

INSERT INTO MYFRIENDS
  (Name, Vorname, email) VALUES
  ('Glarner', 'Reto',
  'reto.glarner@gibb.ch');
```

INSERT INTO MYFRIENDS

```
  (Name, Vorname, email) VALUES
  ('Schluep', 'miriam',
  'miriam.schluep@gibb.ch');
```

INSERT INTO MYFRIENDS

```
  (Name, Vorname, email) VALUES
  ('Yilmaz', 'Günel',
```

```
  'guenel.yilmaz@gibb.ch');
```

INSERT INTO MYFRIENDS

```
  (Name, Vorname, email) VALUES
  ('Bielawski', 'barbara',
  'barbara.bielawski@gibb.ch');
```

--10 Schnell prüfen

```
SELECT * FROM MYFRIENDS;
```

--10 Update Statement um die Vornamen zu korrigieren

```
UPDATE MYFRIENDS
  SET Vorname = 'Miriam'
  WHERE Vorname = 'miriam';

UPDATE MYFRIENDS
  SET Vorname = 'Barbara'
  WHERE Vorname = 'barbara'
```

--10.1

```
UPDATE MYFRIENDS
  SET FKORTSID= (SELECT ORTSID FROM ORT WHERE
  Ortsname= 'Murten')
  WHERE Name = 'Maurer';

UPDATE MYFRIENDS
  SET FKORTSID= (SELECT ORTSID FROM ORT WHERE
  Ortsname= 'Bern' AND PLZ= '3005')
  WHERE Name = 'Frei';

UPDATE MYFRIENDS
  SET FKORTSID= (SELECT ORTSID FROM ORT WHERE
  Ortsname= 'Adelboden')
  WHERE Name = 'Glarner';

UPDATE MYFRIENDS
  SET FKORTSID= (SELECT ORTSID FROM ORT WHERE
  Ortsname= 'Basel')
  WHERE Name = 'Schluep';

UPDATE MYFRIENDS
  SET FKORTSID= (SELECT ORTSID FROM ORT WHERE
  Ortsname= 'Zürich' AND PLZ= '8001')
  WHERE Name = 'Yilmaz';

UPDATE MYFRIENDS
  SET FKORTSID= (SELECT ORTSID FROM ORT WHERE
  Ortsname= 'Lenzburg')
  WHERE Name = 'Bielawski';
```

--Letzte Kontrolle

```
SELECT * FROM ORT;
SELECT * FROM MYFRIENDS;
```

--AB100-08.sql

--Tabelle löschen

--DROP TABLE MYFRIENDS;

--DROP TABLE ORT;

--Seite 1

--Freunde in Myfriends hinzufügen

INSERT INTO MYFRIENDS (Name, Vorname, email)
VALUES

('Duddle', 'Hugo', 'HugoD@gmail.com');

INSERT INTO MYFRIENDS (Name, Vorname, email)
VALUES

('Osipov', 'Valerie', 'vosi@mail.ru');

INSERT INTO MYFRIENDS (Name, Vorname, email)
VALUES

('Daniels', 'Jack', 'nogo@whiskey.com');

INSERT INTO MYFRIENDS (Name, Vorname, email)
VALUES

('Khortyza', 'vodki', 'go@vodka.com');

--1. Theorie Aufgabe mit SELECT kontrollieren

SELECT * FROM ORT;

SELECT * FROM MYFRIENDS;

--JOIN Typen

--Seite 2/3

--JOIN (verbindet diejenige, bei denen fkortsid und
ortsid gleich sind)

SELECT * FROM MYFRIENDS AS f

JOIN ORT AS o ON

f.fkortsid=o.ortsid;

--LEFT JOIN (links alles und rechts nur was mit dem
fkortsid übereinstimmt)

SELECT * FROM MYFRIENDS AS f

LEFT JOIN ORT AS o ON

f.fkortsid=o.ortsid;

--2. Theorie Aufgabe LEFT JOIN (Vorname, Name,
email von MIFRIENDS wenn ortsid == Null)

SELECT Vorname, Name, email FROM MYFRIENDS AS f

LEFT JOIN ORT AS o ON f.fkortsid=o.ortsid

WHERE o.ortsid IS NULL;

--Seite 4/5

--RIGHT JOIN (NICHT AUSFÜHRBAR! wegen sqlite3)

SELECT * FROM MYFRIENDS AS f

RIGHT JOIN ORT AS o ON

f.fkortsid=o.ortsid;

--3. Theorie Aufgabe RIGHT JOIN (NICHT
AUSFÜHRBAR!)

SELECT Vorname, Name, email FROM MYFRIENDS AS f

RIGHT JOIN ORT AS o ON

f.fkortsid=o.ortsid;

--Konversion RIGHT JOIN (NICHT AUSFÜHRBAR!)

SELECT * FROM MYFRIENDS AS f

RIGHT JOIN ORT AS o ON

f.fkortsid=o.ortsid;

--Konversion LEFT JOIN(äquivalent)

SELECT ID, Name, Vorname, email, fkortsid, ortsid, plz,
ortsname FROM ORT AS o

LEFT JOIN MYFRIENDS AS f ON

f.fkortsid=o.ortsid;

--Testen was mit Stern

SELECT * FROM ORT AS o

LEFT JOIN MYFRIENDS AS f ON

f.fkortsid=o.ortsid;

--4. Theorie Aufgabe (fkortsid == NULL wird PLZ und
Ortsname angezeigt)

SELECT plz,ortsname FROM ORT AS o

LEFT JOIN MYFRIENDS AS f ON

f.fkortsid=o.ortsid

WHERE f.fkortsid IS NULL;

--5. Theorie Aufgabe

SELECT plz, ortsname FROM MYFRIENDS, ORT;

SELECT name,plz, ortsname FROM MYFRIENDS, ORT;

--Seite 6

--Beispiel 1

SELECT Ortsname FROM MYFRIENDS AS F

INNER JOIN ORT AS O ON

F.FKORTSID=O.ORTSID

WHERE NAME =

'Maurer';

--1.Aufgabe = Anzahl 6 mit B oder Z

SELECT Ortsname FROM ORT

WHERE Ortsname LIKE 'B%'

OR Ortsname LIKE 'Z%';

--2.Aufgabe a/b sortieren nach Alphabet

SELECT * FROM MYFRIENDS AS F

LEFT JOIN ORT AS O ON F.fkortsid=O.ortsid

WHERE O.ortsid GROUP BY

Ortsname;

--3.Aufgabe Zeigt alle an, die bei fkortsid NULL haben

SELECT Vorname, Name, email FROM MYFRIENDS AS F

LEFT JOIN ORT AS O ON F.fkortsid=O.ortsid

WHERE F.fkortsid IS Null

GROUP BY Name;

--4.Aufgabe Name aktualisieren von vodki in Vodka

UPDATE MYFRIENDS

SET Vorname = 'Vodka'

WHERE Vorname = 'vodki';

--5.Aufgabe Neuen Freund hinzufügen mit Sub Select

INSERT INTO MYFRIENDS (Name, Vorname, email,

FKORTSID) VALUES

('Gsponer', 'Charly', 'cg@vs.ch',(SELECT

ORTSID FROM ORT WHERE Ortsname ='Stalden'));

--Seite 7

--6.Aufgabe nimmt id,vorname,name,ortsname dort
wo Ortsname Bern,Zürich,Genf oder FKORTSID Null ist

SELECT ID, Vorname, Name, Ortsname FROM

MYFRIENDS AS F

LEFT JOIN ORT AS O ON F.fkortsid=O.ortsid

WHERE Ortsname IS 'Genf' OR

Ortsname IS 'Bern' OR Ortsname IS 'Zürich' OR

FKORTSID IS NULL ;

--7.Aufgabe zeigt die 1. sechs Ortsnamen nach dem

Alphabet gelistet an

SELECT Ortsname FROM ORT

GROUP BY Ortsname LIMIT 6;

--8.Aufgabe a Zeigt die Zeilen an, welche

übereinstimmen

SELECT * FROM MYFRIENDS AS F

LEFT JOIN ORT AS O ON F.fkortsid=O.ortsid;

--8. Aufgabe b Zählt die Anzahl Zeilen, welche

übereinstimmen

SELECT COUNT(*) FROM MYFRIENDS AS F

LEFT JOIN ORT AS O ON F.fkortsid=O.ortsid;

--9.Aufgabe Zeigt ID, Name, email, PLZ, Ortsname von

Myfriends und Ort an nach dem Ortsnamen geordnet

SELECT ID, Name, email, PLZ, Ortsname FROM

MYFRIENDS AS f

LEFT JOIN ORT AS o ON f.fkortsid = o.ortsid

GROUP BY Ortsname;

--10.Aufgabe Ortsnamen geordnet nach Ortsnamen

ausgeben wie kann man doppelte vermeiden ohne

GROUP? Mit ORDER geht es nicht

SELECT Ortsname FROM ORT

GROUP BY Ortsname;

--11.Aufgabe Aktualisiert die FKORTSID von Hugo

Duddle zu 10(Yverdon)

UPDATE MYFRIENDS

SET FKORTSID= (SELECT ORTSID FROM ORT

WHERE Ortsname IS 'Yverdon')

WHERE Name IS 'Duddle';