

Übungen zur Vorlesung Informatik II (D-BAUG) FS 2017

D. Sidler, F. Friedrich

<http://lec.inf.ethz.ch/baug/informatik2/2017>**Problem set # 11****22.5.2017 – 30.5.2017**

Problem 11.1. ER Modeling

1. People and Traveling

Create an ER model of the following mini-world:

- A person has a name, an age and an email address.
- People like to visit foreign cities.
- Cities have a name and are located in a specific country.
- In order to travel to these cities, people always travel in groups.
- A travel group can have multiple participants, but only one destination.
- A trip to a city requires a start and an end date.

2. Library

Assume you are creating a system to manage a library. Give an ER model with the following properties:

- The library contains one or several books.
- Every book is located at a specific location in a shelf and is identified by the copy number and the ISBN number.
- In addition, a book has a publication year, a title and an author.
- Books are published by publishers.
- A publisher has a name and as a location.
- Each reader needs to provide his/her family name, his/her first name and his/her city of residence in order to register at the library.
- Readers borrow books.
- Upon borrowing a book, a return date is stored.

Submission link: <https://codeboard.ethz.ch/inf2baugex11t01>**Problem 11.2. Relational Model**

1. ER to Relational Model

For each of the ER models from Problem 11.1, create a corresponding relational schema.

2. Relational Algebra

Given the following relational schema:

READER (RDNR, Firstname, Lastname, City, Birthdate)BOOK (ISBN, Title, Author, NumberOfPages, PublicationYear, PublisherName)PUBLISHER (PublisherName, PublisherCity)COPY (ISBN, CopyNumber, Shelf, Position)BORROW (ReaderNr, ISBN, Copy, ReturnDate)

Write the following queries in relational algebra:

- Which are the last names of all the readers living in Zurich?
- Which books (Title, Author) have their publisher located in Zurich?
- Which books (Title, Author) have been borrowed by the reader "John Doe"?

Submission link: <https://codeboard.ethz.ch/inf2baugex11t02>

Problem 11.3. SQL Queries

ETH offers a hosted mySQL Database service for all students, which you can access within the ETH network. Thus, you do not need to install your mySQL on your own machine.

To create a database, log into your NETHZ account at <https://password.ethz.ch> and go to the *My Services* tab. Here you can select *mySQL* (see Figure 1). On the next page you have to set a password for your database. After the database has been created, write down the hostname of the database server (e.g., *mysqlweb1.ethz.ch*) and remember the password you have chosen. The database name corresponds to your NETHZ username.

Using the PhpMyAdmin tool you can administer the database. Log in to <https://phpmyadmin.ethz.ch/> using your NETHZ username and the password defined above and click on *MySQL V4*.



Abbildung 1: NETZ Administration

Test your database

To test your database, we will create a simple 'Person' table in which we will store the first name, last name and email address of a person (Step 1). Next, we will populate (Step 2) and query the database (Step 3).

You will learn SQL in the second database lecture. For now, you can copy the statements provided below.

Step 1: To create a table for storing data about people, we use the 'CREATE TABLE' command.

```
CREATE TABLE person (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50) NOT NULL
)
```

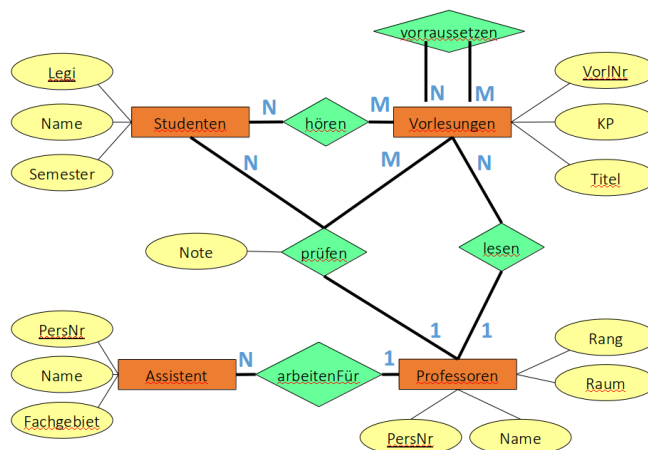
Step 2: Insert values into the newly created table is done via the 'INSERT INTO' command.

```
INSERT INTO person ( id, firstname, lastname, email)
VALUES
(NULL, "Donald", "Duck", "donald.duck@disney.com"),
(NULL, "Mickey", "Mouse", "mickey.mouse@disney.com");
```

Step 3: Query the database to get the email address of "Mickey Mouse".

```
SELECT email
FROM person
WHERE firstname = "Mickey" AND lastname="Mouse";
```

Problem 11.4. SQL Queries



Given the university schema from the lecture, provide the SQL queries to answer the following questions:

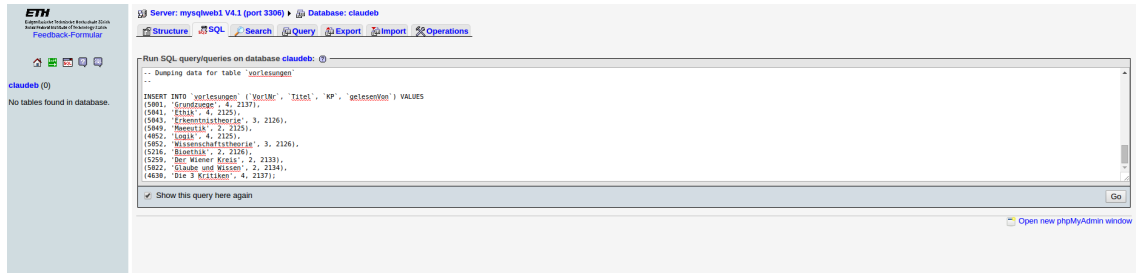
1. Which is the average semester of all students?
2. Which students are enrolled in a higher semester than the average?
3. Return all the names and Id-Numbers of all the professors, assistants and students. In the case of professors and assistants, the Id-Number is the PersNr, and in the case of students, it is their Legi. Sort the result based on the Id-Number in descending order.
4. Provide the names of all students which visit at least one lecture of Sokrates.
5. Provide the name of all professors which give at least one lecture along with the total number of points (KP) of all their courses.
6. Which are the courses which require the lecture 'Grundzuege'? (We only consider direct successors! Bonus question: Can SQL deal with transitivity? If yes, how?)
7. Which student is the most popular? You can assume that students know each other from common courses. The more people know a particular student, the more popular he/she is. Sort the resulting list by the student's popularity.

It is a good idea to try out a lot of examples including the solutions of the above with the real database. To do so, you can install the university schema from the lecture as described below. A detailed description how to set up a database has been provided in exercise 3. We will now populate this database with the university schema.

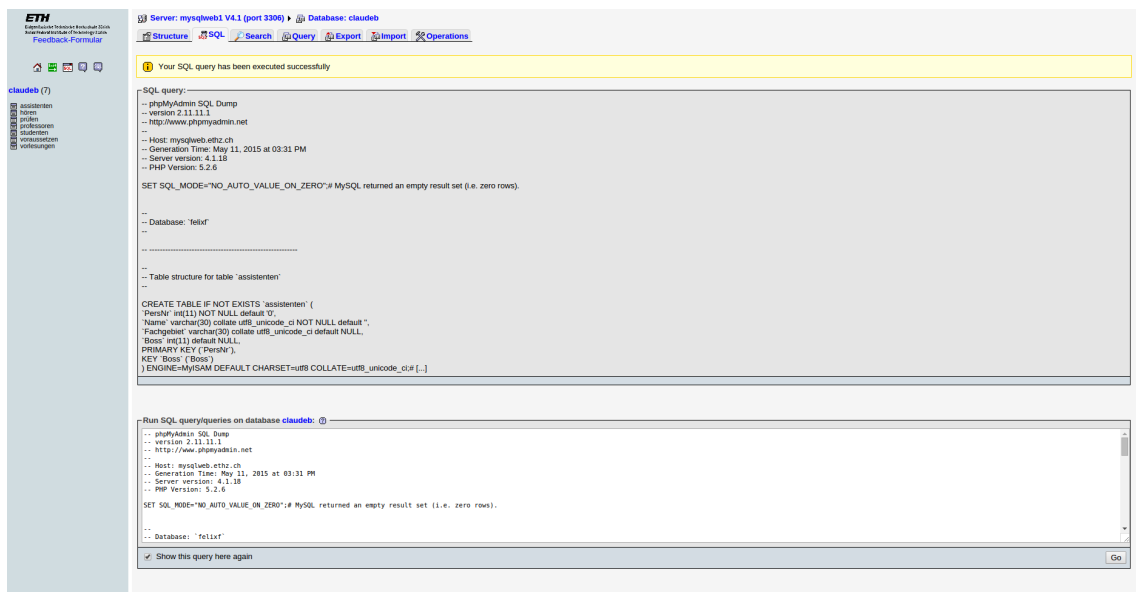
1. Log in to your database at <https://phpmyadmin.ethz.ch/>. You can only access this page from within the ETH network. A VPN connection might be required if you are working remotely.
2. Download the university schema from the course webpage <http://lec.inf.ethz.ch/baug/informatik2/2017/code/Unidb.txt>.
3. In the database administration page, select your database (your NETHZ name) on the left side and then click on the 'SQL' tab. Copy and paste the content of the university schema into the text field and click 'Go'.

Übungen zur Vorlesung Informatik II (D-BAUG), Blatt 11

4



4. The next page will inform you that the query has been executed. On the left side you should now see the individual table names.



Now you can test your solutions for question 1-7 with the real database.

Submission link: <https://codeboard.ethz.ch/inf2baugex11t04>