



Exercise for *Database System Concepts for Non-Computer Scientist* im  
WiSe 18/19

Alexander van Renen (renen@in.tum.de)  
<http://db.in.tum.de/teaching/ws1819/DBSandere/?lang=en>

Sheet 07

Exercise 1

Write SQL queries for following tasks on our university database:

- Find all *Students* that know Sokrates from a *Lecture*.
- Find all *Students* that attend at least one *Lecture* together with Fichte.
- Find *Assistants* of *Professors* that taught Carnap – e.g., as a potential advisor for a thesis.
- Determine the names of *Professors* that Theophrastos knows from *Lectures*.
- Which *Lectures* are attended by *Students* in the 1.-4. semester? Print out the title of those *lectures*.

**Solution:**

- Find all *Students* that know Sokrates from a *Lecture*.

```
select distinct s.studNr, s.name
from Students s, attend a, Lectures l, Professors p
where s.studNr = a.studNr
      and a.lectureNr = l.lectureNr
      and l.given_by = p.persNr
      and p.name = 'Sokrates';
```

DISTINCT is necessary if we do not want duplicates in our result. Alternatively, we could also write the query using nested queries. However, by looking at the resulting query it should be clear that the former one should be preferred, as it is easier to read and understand.

```
select distinct s.studNr, s.name
from students s
where s.studNr in
  (select a.studNr
   from attend a
   where a.lectureNr in
     (select l.lectureNr
      from lectures l
      where l.given_by in
        (select p.persNr
         from professors p
         where p.name = 'Sokrates'
        )
     )
  )
```

- (b) Find all *Students* that attend at least one *Lecture* together with Fichte.

```
select distinct other_s.studNr, other_s.name
from Students fichte_s, attend fichte_a, attend
  other_a, Students other_s
where fichte_s.name = 'Fichte'
  and fichte_a.studNr = fichte_s.studNr
  and other_a.lectureNr = fichte_a.lectureNr
  and other_s.studNr = other_a.studNr
  and other_s.studNr <> fichte_s.studnr
```

- (c) Find *Assistants of Professors* that taught Carnap – e.g., as a potential advisor for a thesis.

```
select distinct b.perNr, b.name
from Students s, attend a, Lectures l, Assistants b
where s.name = 'Carnap'
  and a.studNr = s.studNr
  and a.lectureNr = l.lectureNr
  and l.given_by = b.boss
```

The query should return both the name and the persNr, otherwise two assistants with the same name could not be distinguished.

- (d) Determine the names of *Professors* that Theophrastos knows from *Lectures*.

```
select p.persNr, p.name
from Professors p, attend a, Lectures l, Students s
where p.persNr = l.given_by
  and l.lectureNr = a.lectureNr
  and a.studNr = s.studNr
  and s.name = 'Theophrastos';
```

- (e) Which *Lectures* are attended by *Students* in the 1.-4. semester? Print out the title of those *lectures*.

```
select l.title
from Lectures l, attend a, Students s
where l.lectureNr = a.lectureNr
  and a.studNr = s.studNr
  and s.semester between 1 and 4;
```