Datenbanksysteme I
6. Übungsserie

Aufgabe 1:

(a)

```
update Professoren
set Rang = 'C4'
where Rang = 'C3'
```

(b)

```
delete from Assistenten
where Fachgebiet = 'Planetenbewegung'
```

Aufgabe 2:

(a)

```
create table User (
        id              varchar(15) primary key,
        name    varchar(50)
);
create table Tweet (
        id              bigint primary key,
        user_id varchar(15),
        foreign key (user_id)
        references User(id),
        text    varchar(280) not null,
        date    timestamp not null
)
create table follows (
        follower_id     varchar(15) not null,
        follows_id      varchar(15) not null,
        primary key (follower_id, follows_id),
        foreign key (follower_id)
        references User(id),
        foreign key (follows_id)
        references User(id)
)
create table likes (
        user_id         varchar(15) not null,
        tweet_id        bigint not null,
        date            timestamp,
        primary key (user_id, tweed_id),
        foreign key (user_id)
        references User(id),
        foreign key (tweet_id)
        references Tweet(id)
)
```

(b)

```
create table User (
        id              varchar(15) primary key,
```

```sql
        name      varchar(50)
);
create table Tweet (
        id              bigint primary key,
        user_id varchar(15) references User(id) on delete set null,
        text    varchar(280) not null,
        date    timestamp not null
);
create table follows (
        follower_id     varchar(15) references User(id) on delete cascade,
        follows_id      varchar(15) references User(id) on delete cascade,
        primary key (follower_id, follows_id)
);
create table likes (
        user_id         varchar(15) references User(id) on delete cascade,
        tweet_id        bigint references Tweet(id) on delete cascade,
        date            timestamp,
        primary key (user_id, tweed_id)
);
```

(c)
```sql
create table User (
        id              varchar(15) primary key,
        name    varchar(50)
);
create table Tweet (
        id              bigint primary key,
        user_id varchar(15) references User(id) on delete set null,
        text    varchar(280) not null,
        date    timestamp not null,
        check (not user_id is null or text = 'removed')
);
create table follows (
        follower_id     varchar(15) references User(id) on delete cascade,
        follows_id      varchar(15) references User(id) on delete cascade,
        primary key (follower_id, follows_id)
);
create table likes (
        user_id         varchar(15) references User(id) on delete cascade,
        tweet_id        bigint references Tweet(id) on delete cascade,
        date            timestamp,
        primary key (user_id, tweed_id),
        check (
        tweet_id in (
                select Tweet.id
                from Tweet
                where likes.tweet_id = Tweet.id
                and likes.date > Tweet.date
                )
        )
);
```

Aufgabe 3:

```
with factor as (values(1),(2),(3),(4),(5),(6),(7),(8),(9),(10))
select *, factor1 * factor2 as product
from factor as f1(factor1), factor as f2(factor2)
order by factor1, factor2
```