

# Bericht Datenbank Praktikum

Julian Sobott (76511), David Sugar (76050), Lukas Mendel (76509)

4. Dezember 2019

# Inhaltsverzeichnis

0.1	Aufgabe 1 . . . . .	2
	0.1.1 a) . . . . .	2
	0.1.2 b) . . . . .	3
	0.1.3 c) . . . . .	3
0.2	Aufgabe 2 . . . . .	5
0.3	Aufgabe 3 . . . . .	5

## 0.1 Aufgabe 1

### 0.1.1 a)

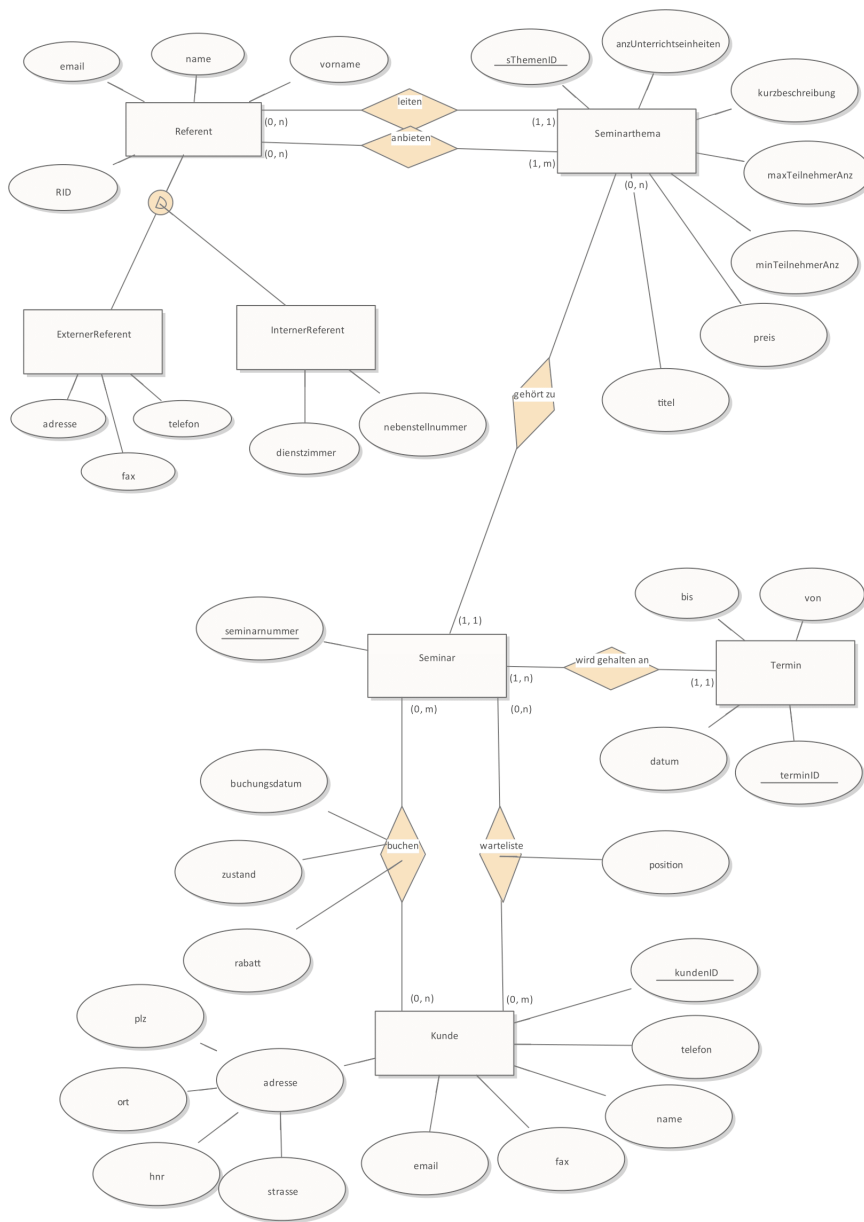


Abbildung 1: ER-Modell Seminarverwaltung

## Entities

```
seminar = ({SEMINARNUMMER:INTEGER})
    termin = ({TERMINID:INTEGER, DATUM:DATE, VON:DATETIME, BIS:DATETIME})
    kunde = ({KUNDENID:INTEGER, TELEFON:VARCHAR, NAME:VARCHAR, FAX:VARCHAR, EMAIL:VARCHAR,
ADRESSE:(PLZ:VARCHR, ORT:VARCHAR, HNR:VARCHAR, STR:VARCHAR)})
    referent = ({RID:INTEGER, email: VARCHAR, name: VARCHAR, vorname: VARCHAR})
    seminarthema = (STHEMAID:INTEGER, ANZUNTERICHTSEINHEITEN:INTEGER, KURZBESCHREI-
BUNG:VARCHAR, MAXTEILNEHMERANZ:INTEGER, MINTEILNEHMERANZ:INTEGER, PREIS:FLOAT,
TITEL:VARCHAR)
    externerReferent = ({RID:INTEGER,adresse: (plz: VARCHAR, ort: VARCHAR, strasse: VARCHAR, hnr:
VARCHAR)}) is_a referent
    internerReferent = ({RID:INTEGER,dienstzimmer: VARCHAR, nebenstellnummer: Integer} is_a referent)
```

## Relations

```
leiten = (referent X seminarthema)
    anbieten = (referent X seminarthema)
    gehört_zu = (seminarthema X seminar)
    buchen = (seminar x kunke, BUCHUNGSDATUM:DATE, ZUSTAND:VARCHAR, RABATT:FLOAT)
    warteliste = (kunde x seminar, POSITION:INTEGER)
    wird_gehalten_an = (seminar x termin)
```

### 0.1.2 b)

#### Relationen

```
referent = (RID:INTEGER, email: VARCHAR, name: VARCHAR, vorname: VARCHAR)
    ExternerReferent (RID:INTEGER, plz: VARCHAR, ort: VARCHAR, strasse: VARCHAR, hnr: VARCHAR)
    IntererReferent (RID:INTEGER, dienstzimmer: VARCHAR, nebenstellnummer: Integer)
    seminarthema = (STHEMAID:INTEGER, ANZUNTERICHTSEINHEITEN:INTEGER, KURZBESCHREI-
BUNG:VARCHAR, MAXTEILNEHMERANZ:INTEGER, MINTEILNEHMERANZ:INTEGER, PREIS:FLOAT,
TITEL:VARCHAR, LEITER:INTEGER)
    anbieten = (REFERENTID:INTEGER, SEMINARTHEMAID:INTEGER)
    seminar = (SEMINARNUMMER:INTEGER, SEMINARTHEMAID:INTEGER)
    termin = (TERMINID:INTEGER, VON:DATETIME, BIS:DATETIME, DATUM:DATE, SEMINARID:INTEGER)
    kunde = (KUNDENID:INTEGER, TELEFON:VARCHAR, NAME:VARCHAR, FAX:VARCHAR, EMAIL:VARCHAR,
PLZ:VARCHR, ORT:VARCHAR, HNR:VARCHAR, STR:VARCHAR)
    buchen = (KUNDENID:INTEGER, SEMINARNR:INTEGER, BUCHUNGSDATUM:DATE, ZUSTAND:VARCHAR,
RABATT:FLOAT)
    warteliste = (KUNDENID:INTEGER, SEMINARNR:INTEGER, POSTION:INTEGER)
```

#### Referenzen

```
seminarthema|LEITER ⊆ referent|RID
    anbieten|REFERENTID ⊆ referent|RID
    anbieten|SEMINARTHEMAID ⊆ seminarthema|STHEMAID
    seminar|SEMINARTHEMAID ⊆ seminarthema|STHEMAID
    termin|SEMINARID ⊆ seminar|SEMINARNUMMER
    buchen|KUNDENID ⊆ kunde|KUNDENID
    buchen|SEMINARNR ⊆ seminar|SEMINARNR
    warteliste|KUNDENID ⊆ kunde|KUNDENID
    warteliste|SEMINARNR ⊆ seminar|SEMINARNR
```

### 0.1.3 c)

```
CREATE TABLE g8_referent (
    rid serial PRIMARY KEY,
    email VARCHAR(50) ,
    name VARCHAR(50) ,
    vorname VARCHAR(50)
);
```

```

CREATE TABLE g8_seminarthema (
    sthemaid serial PRIMARY KEY,
    anz_ unterrichtseinheiten INTEGER,
    kurzbeschreibung VARCHAR,
    max_ teilnehmeranzahl INTEGER,
    min_ teilnehmeranzahl INTEGER,
    preis FLOAT,
    titel VARCHAR(200),
    leiter INTEGER REFERENCES g8_referent(rid)
);

CREATE TABLE g8_anbieten (
    referenten_id INTEGER REFERENCES g8_referent(rid),
    sthemaid INTEGER REFERENCES g8_seminarthema(sthemaid),
    PRIMARY KEY (referenten_id, sthemaid)
);

CREATE TABLE g8_seminar (
    seminarnummer serial PRIMARY KEY,
    sthemaid INTEGER REFERENCES g8_seminarthema(sthemaid)
);

CREATE TABLE g8_termin (
    terminid serial PRIMARY KEY,
    von TIME,
    bis TIME,
    datum DATE,
    seminarnummer INTEGER REFERENCES g8_seminar(seminarnummer)
);

CREATE TABLE g8_kunde (
    kundenid serial PRIMARY KEY,
    telefon VARCHAR(20),
    name VARCHAR(50),
    fax VARCHAR(20),
    email VARCHAR(50),
    plz VARCHAR(10),
    ort VARCHAR(50),
    hnr VARCHAR(10),
    str VARCHAR(50)
);

CREATE TYPE g8_zustand as ENUM ('offen', 'gebucht', 'berechnet', 'gezahlt', 'storniert');

CREATE TABLE g8_buchen (
    kundenid INTEGER REFERENCES g8_kunde(kundenid),
    seminarnummer INTEGER REFERENCES g8_seminar(seminarnummer),
    datum DATE,
    zustand g8_zustand,
    rabatt FLOAT,
    PRIMARY KEY (kundenid, seminarnummer)
);

CREATE TABLE g8_warteliste (
    kundenid INTEGER REFERENCES g8_kunde(kundenid),
    seminarnummer INTEGER REFERENCES g8_seminar(seminarnummer),
    position INTEGER,
    PRIMARY KEY (kundenid, seminarnummer)
);

```

```
);
```

```
CREATE TABLE g8_ExternerReferent (  
  
RID int PRIMARY KEY,  
fax VARCHAR(50),  
telefon VARCHAR(20),  
PLZ VARCHAR(15),  
Strasse VARCHAR(50),  
Hnr VARCHAR (20),  
Ort VARCHAR(50),  
  
FOREIGN KEY (RID) REFERENCES g8_referent (RID)  
);
```

```
CREATE TABLE g8_InternerReferent (  
  
RID int PRIMARY KEY,  
Dienstnummer VARCHAR(30),  
nebenstellenummer INTEGER,  
  
FOREIGN KEY (RID) REFERENCES g8_referent (RID)  
  
);
```

## 0.2 Aufgabe 2

```
INSERT INTO g8_referent(email, vorname, name) values  
( 'julian.sobott@wtf.de', 'Julian', 'Sobott'),  
( 'david.sugar@wtf.de', 'David', 'Sugar'),  
( 'lukas.mendel@wtf.de', 'Lukas', 'Mendel'),  
( 'gregor.grambow@wtf.de', 'Gregor', 'Grambow');
```

```
INSERT INTO g8_internerreferent(rid, plz, ort, strasse, hnr) values  
((select rid from g8_referent where name = 'Grambow' and vorname = 'Gregor')  
, '73434', 'Aalen', 'Uni-Str', '111');
```

```
INSERT INTO g8_externerreferent(rid, plz, ort, strasse, hnr) values  
((select rid from g8_referent where name = 'Sugar' and vorname = 'David'), '  
73434', 'Aalen', 'Uni-Str', '111'),  
((select rid from g8_referent where name = 'Sobott' and vorname = 'Julian'),  
'73434', 'Aalen', 'Uni-Str', '111'),  
((select rid from g8_referent where name = 'Mendel' and vorname = 'Lukas'),  
'73434', 'Aalen', 'Uni-Str', '111');
```

```
INSERT INTO g8_seminarthema(anz_unterrichtseinheiten, kurzbeschreibung,  
max_teilnehmeranzahl, min_teilnehmeranzahl, preis, titel, leiter) values  
(10, 'Datenbanken_Grundlagen_erlernen.', 30, 5, 152.50, 'Datenbanken', (  
select rid from g8_referent where name = 'Grambow' and vorname = 'Gregor'  
)),  
(2, 'We_love_RISC', 10, 1, 0.0, 'The_ARM_Architecture', (select rid from  
g8_referent where name = 'Sugar' and vorname = 'David')),  
(3, 'Its_not_a_snake', 15, 3, 43.90, 'Python', (select rid from g8_referent  
where name = 'Julian' and vorname = 'Sobott'));
```

## 0.3 Aufgabe 3

```
SELECT(  
(SELECT COUNT (rid) as AnzahlInterne
```

```

FROM g8_InternerReferent),
(SELECT COUNT (rid) as AnzahlExterne
FROM g8_ExternerReferent),
(SELECT COUNT (rid) as AnzahlGesamt
FROM g8_referent)

);

SELECT seminarnummer, COUNT(seminarnummer)
FROM g8_seminar s JOIN g8_termin t on s.seminarnummer = t.seminarnummer
GROUP BY seminarnummer;

SELECT Seminarummer, COUNT(Seminarummer) as Teilnehmeranzahl
FROM g8_buchen b JOIN g8_seminar s on b.seminarnummer = s.seminarnummer
GROUP BY (Seminarnummer)

SELET Seminarummer, , MAX()

```