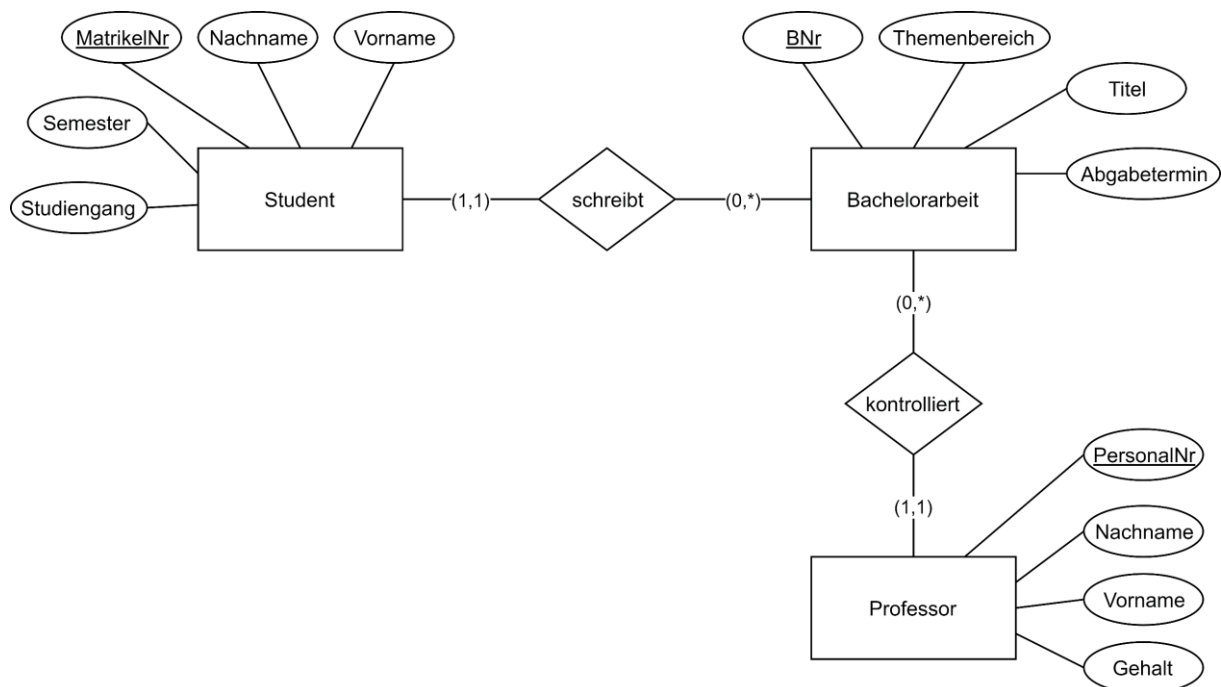


Entity-Relationship-Modell

Sie arbeiten als studentische Hilfskraft im Prüfungsamt Ihrer Fakultät. Eine Ihrer Aufgaben ist die Konzeption eines Datenmodells zur effizienten Abbildung von Studierenden und den von diesen geschriebenen Bachelorarbeiten. Sie schauen sich dafür die aktuelle Studienorganisation an und bekommen die folgenden Informationen:

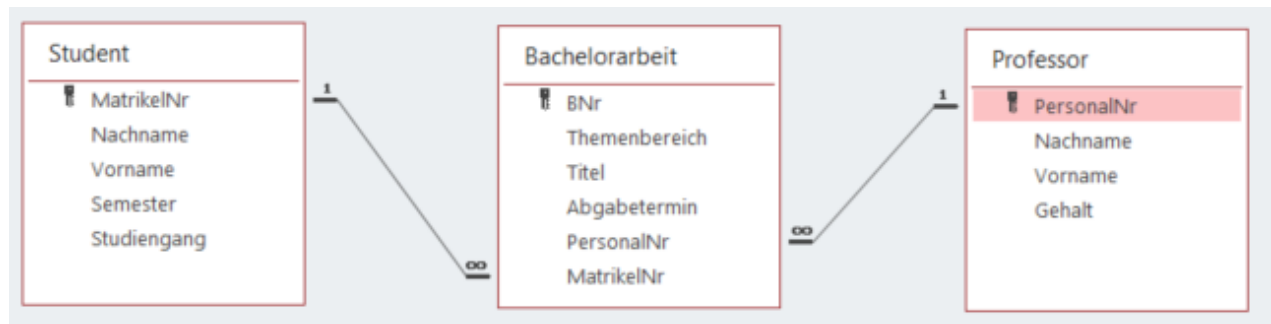
Eine Bachelorarbeit hat einen Themenbereich, einen Titel, einen Abgabetermin und eine eindeutige Nummer. Eine Bachelorarbeit wird von genau einem Studenten geschrieben und von genau einem Professor begutachtet. Studenten haben einen Nachnamen, einen Vornamen, ein bestimmtes Semester, einen Studiengang und eine Matrikelnummer. Professoren haben einen Nachnamen, einen Vornamen, ein Gehalt und eine Personalnummer.

Zeichnen Sie auf Basis dieser Informationen ein Entity- Relationship- Modell (auf Papier oder www.draw.io).



Hinweise: Das Sternchen („*“) steht für „viele“ und wird teilweise auch mit „n“, „m“ oder in Access mit „∞“ dargestellt. Inhaltlich gibt es hierbei keinen Unterschied. Diese Lösung nutzt die Min-Max-Notation. Häufig wird auch die „einfache“ Notation verwendet, wobei das „Min“ in der jeweiligen Angabe fehlt (siehe untenstehende Darstellung in Access). Weiterhin sollte beachtet werden, dass die Min-Max-Notation teilweise auch umgedreht wird, also z.B. das „(1,1)“ und „(0,*)“ in der Beziehung zwischen Student und Bachelorarbeit die Positionen tauschen (wobei das aber nur eine andere grafische Darstellung ist und nichts an der inhaltlichen Aussage ändert).

Legen Sie die Datenbank in Access an. Erstellen Sie hierfür die Tabellen, modellieren Sie die jeweiligen Felder und Felddatentypen, hinterlegen Sie geeignete Schlüssel und erstellen Sie die entsprechenden Beziehungen.



SQL-Aufgaben Lösung (Hinweis: Für die Klausur müssen Sie SQL-Statements nur lesen und interpretieren, aber nicht selbständig schreiben können; vgl. Altklausuraufgabe SoSe 2017. Die SQL-Zusatzaufgaben aus Aufgabe 2.20 sind nicht klausurrelevant.)

Ausgabe sämtlicher Spalten der Tabelle Kunde.

```
SELECT *
FROM KUNDE;
```

Ausgabe von Vorname, Nachname und Telefonnummer sämtlicher Kunden.

```
SELECT Name, Vorname, TelNR
FROM KUNDE;
```

Ausgabe aller verschiedenen Autotypen.

```
SELECT DISTINCT Bezeichnung
FROM PKW;
```

Zeige die PKW auf, die mehr als 100 €/Tag kosten.

```
SELECT *
FROM PKW
WHERE Preis_pro_Tag > 100;
```

Liste aller Autovermietungen, bei denen die Schlüsselnummer 206 ist und die Anzahl_Tage_gemietet mindestens 3 ist.

```
SELECT *
FROM VERMIETUNG
WHERE SchluesseINR = 206 AND Anzahl_Tage_gemietet >= 3;
```

Ausgabe aller PKW, die zwischen 70 €/Tag und 170 €/Tag kosten.

```
SELECT *  
FROM PKW  
WHERE Preis_pro_Tag BETWEEN 70 AND 170;
```

Ausgabe der Kunden, deren Kundennummer zwischen 103 und 108 liegt, die aber nicht in Marburg wohnen.

```
SELECT *  
FROM KUNDE  
WHERE KdNR BETWEEN 103 AND 108 AND NOT Ort='Marburg';
```

Ausgabe aller Kunden, deren Name (Nachname) mit „S“ beginnt.

```
SELECT *  
FROM KUNDE  
WHERE Name LIKE 'S*';
```

Liste aller Kunden, deren Telefonnummer mit 06421 beginnt und deren Nachname 6 Buchstaben hat.

```
SELECT *  
FROM KUNDE  
WHERE TeINR LIKE '06421*' AND Name LIKE '??????';
```

Ausgabe von Nachname, Vorname und PLZ der Kunden, die die Schlüsselnummer 206 schon einmal gebucht haben.

```
SELECT Name, Vorname, PLZ  
FROM KUNDE, VERMIETUNG  
WHERE KUNDE.KdNR=VERMIETUNG.KdNR AND SchluesseINR=206;
```

Ausgabe von Auftragsnummer, Name, Vermietungstag und Vermietungsdauer derjenigen Kunden, die einen VW Golf gemietet haben.

```
SELECT AuftragsNR, Name, Datum, Anzahl_Tage_gemietet  
FROM KUNDE, VERMIETUNG, PKW  
WHERE PKW.SchlNR=VERMIETUNG.SchluesseINR AND KUNDE.KdNR=VERMIETUNG.KdNR  
AND Bezeichnung='VW Golf';
```