

<b>Datenbanksysteme 1</b>		<b>Praktikum Blatt V02: Relationenmodell mit unären Operationen</b>	
Studiengang	B. Sc. Medieninformatik	Bearbeiter	Rakow, Salgert, Hasler, Focken
Semester	SoSe 2023	Datum	01.08.2023

## Musterlösung

### Bearbeitung

Gegeben sei folgendes Schema einer Datenbank des Studierendensekretariats einer amerikanischen Universität (vgl. Elmasri/ Navathe: Fundamentals of Database Systems, 2004).

#### **COURSE ENROLLMENT**

STUDENT (SSN, Name, Major, Bdate)

COURSE (CourseId, Cname, Dept)

ENROLL (SSN, CourseId, Quarter, Grade)

BOOK\_RECOMMENDATION (CourseId, Quarter, Book\_ISBN)

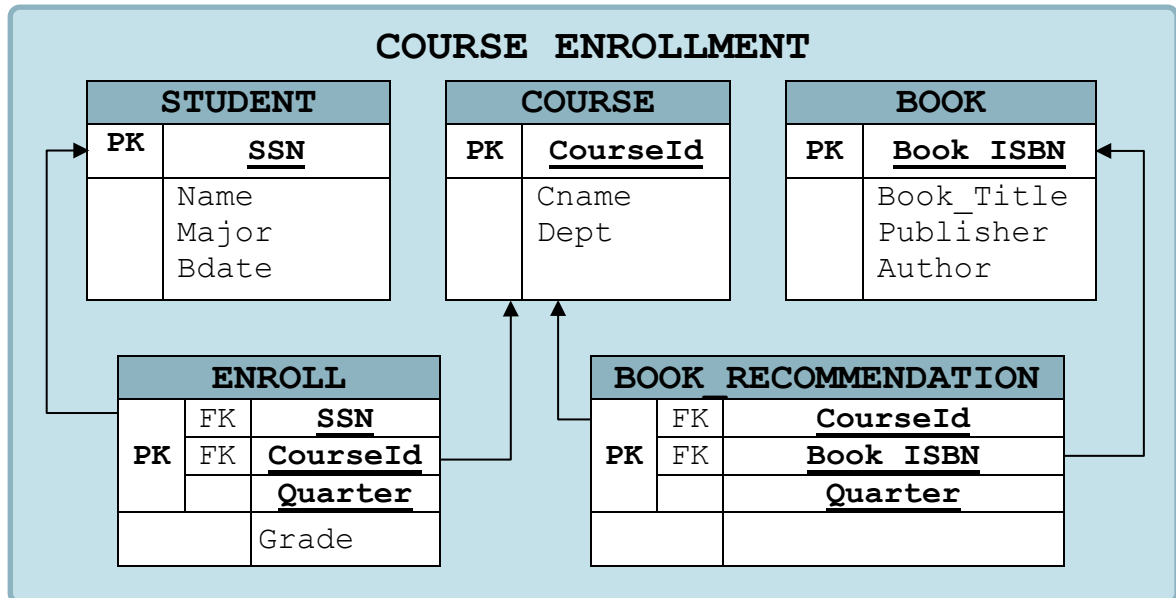
BOOK (Book\_ISBN, Book\_Title, Publisher, Author)

### Aufgabe 1

Diese Aufgabe besteht aus einem Quiz zur Relationenalgebra. Sie können die Erläuterungen zu Ihrer Lösung direkt in Moodle einsehen.

## Aufgabe 2

Zeichnen Sie ein Relationendiagramm für das Schema, in dem die Fremdschlüssel auf die Schlüssel verweisen.



**Pfeilrichtung:** Fremdschlüssel zeigt auf Schlüsselattribut(e) bzw. Relation mit dem Schlüssel

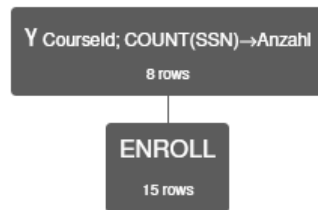
*Ein Attribut kann also Primärschlüssel (PK, engl. primary key), Teil eines Schlüssels und/ oder Fremdschlüssel (FK, engl. foreign key) sein. Andere Notationen bezeichnen nur den Primärschlüssel aus einem Attribut, auch Beziehungspfeile werden nicht verwendet. Diese Darstellung zeigt auch die für den Verbund notwendige Information der Beziehung Fremdschlüssel-Primärschlüssel an, sowohl über die Auszeichnung FK/PK als auch einen Pfeil von Fremdschlüssel zu Primärschlüssel bzw. – aus zeichnerischen Gründen – die Relation des Primärschlüssels.*

### Aufgabe 3

Erstellen Sie ausschließlich mit den **unären Operationen** der Relationenalgebra Ausdrücke für folgende Anfragen.

1. Wie viele Studierende befinden sich jeweils in den Kursen?

```
γ CourseId; count(SSN) -> Anzahl (ENROLL)
```



```
γ CourseId; COUNT(SSN) -> Anzahl ( ENROLL )
```

ENROLL.CourseId	Anzahl
1	2
2	2
3	1
5	1
9	6
6	1
7	1
8	1

1

---

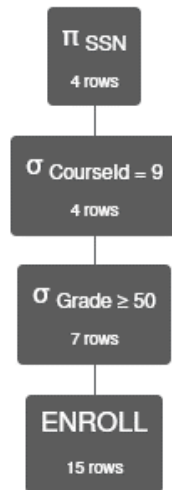
<sup>1</sup> Die Formeln sind in RelAX direkt verwendbar. RelAX zeigt das Ergebnis als Operatorbaum und Ergebnistabelle. RelAX wurde am Institut Databases and Information Systems (DBIS) der Universität Innsbruck entwickelt, s. Literaturverzeichnis im Lernportal.

2. Geben Sie die SSNs der Studierenden aus, die Webprogrammierung (CourseId = 9) bestanden haben.

```

π SSN (
  σ CourseId = 9 (
    σ Grade >= 50 (
      ENROLL
    )
  )
)

```



```

π SSN ( σ CourseId = 9 ( σ Grade ≥ 50 ( ENROLL ) ) )

```

ENROLL.SSN
912393
930941
920148
928521

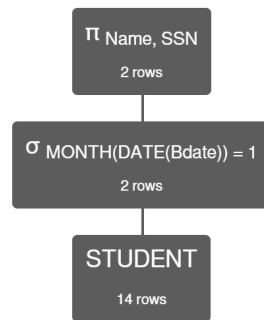
*Hierbei gehen wir von der Annahme aus, dass – wie bei uns – die Grenze zum Bestehen bei mindestens 50 (Punkte) liegt. Wenn wir Grade mit der Projektion ebenfalls ausgeben, können wir die (hoffentlich) korrekte Ausgabe überprüfen.*

3. Geben Sie eine Namensliste der Studierenden aus, die im Januar Geburtstag haben. Sorgen Sie für eine eindeutige Ausgabe der Namen.

```

π Name, SSN (
  σ month(date(Bdate))=01 (
    STUDENT
  )
)

```



```

π Name, SSN ( σ MONTH(DATE(Bdate)) = 1 ( STUDENT ) )
Ausführungszeit: 0 ms

```

STUDENT.Name	STUDENT.SSN
'Anna'	927397
'Anna'	930941

*Nur die Ausgabe des Schlüssels sorgt für die Eindeutigkeit der Namen.*

*Um den Monat aus dem `string` für das Geburtsdatum zu erhalten, können wir eine Funktion `month()` einsetzen. Diese setzt allerdings den Datentyp `date` voraus, so dass wir noch auf diesen Datentyp casten.*

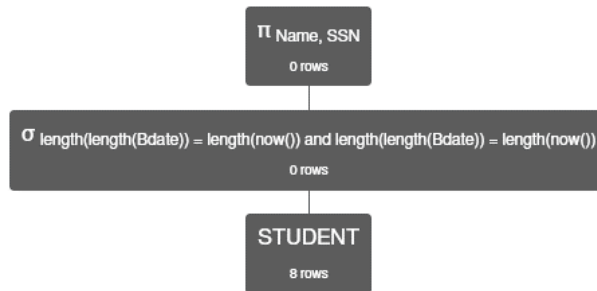
*Sogenannte **skalare Funktionen** obiger Art werden auf den Attributwert jedes auszugebenden Tupels angewendet. Sie dürfen nicht mit den Aggregierungsfunktionen wie in Aufgabe 1 verwechselt werden, die bei einer Gruppierung auf alle Attributwerte jeweils der Gruppen bzw. der ganzen Relation angewendet werden.*

4. Ändern Sie den erstellten Ausdruck so ab, dass eine Liste aller Studierenden angezeigt wird, die heute Geburtstag haben. Sorgen Sie auch hier für eine eindeutige Ausgabe der Namen.

```

π Name, SSN (
  σ month(date(Bdate))=month(now()) ∧
    day(date(Bdate))=day(now()) (
    STUDENT
  )
)

```



```

π Name, SSN ( σ length(length(Bdate)) = length(now()) and length(length(Bdate)) = length(now()) ( STUDENT ) )

```

STUDENT.Name	STUDENT.SSN
--------------	-------------

Die Funktion `now()` liefert uns das aktuelle Datum, denn die Anfrage soll ja auch morgen für heute die korrekte Antwort liefern. Und übermorgen usw.

Je nach Datum der Ausführung der Anfrage kann sich die Ergebnismenge unterscheiden.