

# Datenbanksysteme II

## SS 2017 – Übungsblatt 5

### Teillösung

Universität Leipzig, Institut für Informatik

Abteilung Datenbanken

Prof. Dr. E. Rahm,

V. Christen, M. Franke

# Aufgabe 1 – SQL:2003 Tabellendefinition, Anfragen

---

```
CREATE TYPE PersonT AS  
(Name VARCHAR (40),  
Fak REF (FakultaetT))  
NOT FINAL;
```

```
CREATE TYPE StudentT UNDER PersonT AS NOT FINAL;  
(MatNr INT,  
Hauptfach VARCHAR (40),  
Nebenfach VARCHAR (40),  
Bachelor BOOLEAN)  
NOT FINAL;
```

```
CREATE TYPE ProfT UNDER PersonT AS  
(Buero BueroT,  
Besoldung CHAR(2),  
DRTitel VARCHAR(20),  
ForschGebiete VARCHAR(20) ARRAY[5])  
NOT FINAL;
```

```
CREATE TYPE BueroT AS  
(Gebaeude VARCHAR(40),  
Stockwerk INTEGER,  
Nummer INTEGER,  
Telefon INTEGER)
```

```
CREATE TYPE FakultaetT AS  
(FName VARCHAR (40),  
Dekan REF (ProfT),  
Professoren REF(ProfT) ARRAY[50],  
Studenten REF(StudentT)  
ARRAY[3000])  
NOT FINAL;
```

## Aufgabe 1a – Tabellendefinition

---

**CREATE TABLE** Student-Table **OF** StudentT  
( **REF IS** oid **SYSTEM GENERATED**,  
    **PRIMARY KEY** Name,  
    Fak **WITH OPTIONS SCOPE** Fakultaet-Table)

**CREATE TABLE** Prof-Table **OF** ProfT  
( **REF IS** oid **SYSTEM GENERATED**,  
    **PRIMARY KEY** Name,  
    Fak **WITH OPTIONS SCOPE** Fakultaet-Table)

**CREATE TABLE** Fakultaet-Table **OF** FakultaetT  
( **REF IS** oid **SYSTEM GENERATED**,  
    **PRIMARY KEY** Fname,  
    Dekan           **WITH OPTIONS SCOPE** Prof-Table,  
    Professoren   **WITH OPTIONS SCOPE** Prof-Table,  
    Studenten     **WITH OPTIONS SCOPE** Student-Table)

# Aufgabe 1b – Methode

---

- Methode: Spezialfall einer Funktion; an strukturierten Typ gebunden
- Deklaration und Implementierung strikt getrennt
- Methoden-Typen: INSTANCE, CONSTRUCTOR, STATIC
- Aufruf:

INSTANCE:            <Objektwert>.<Methodenname>[(Parameter)]

CONSTRUCTOR:    NEW <Typname>( <Parameter> )

STATIC:            <Typname>::<Methodenname>[(Parameter)]

## Deklaration:

**CREATE TYPE** FakultaetT **AS** ( ...

**INSTANCE METHOD** countStudents (bac **BOOLEAN**) **RETURNS INT**

)

## Definition:

**CREATE METHOD** countStudents (bac **BOOLEAN**) **RETURNS INT FOR** FakultaetT

**BEGIN**

**RETURN SELECT** count(\*) **FROM UNNEST** (Studenten) **AS S**

**WHERE S-> Bachelor = bac;**

**END;**

# Aufgabe 1c – Konstruktor

---

Deklaration:

```
CONSTRUCTOR METHOD StudentT(p-name  VARCHAR (40),  
                               f-name  VARCHAR(40),  
                               mat-nr  INTEGER,  
                               hauptfach VARCHAR (40),  
                               nebenfach VARCHAR (40),  
                               bachelor BOOLEAN) RETURNS StudentT
```

Definition:

```
CREATE CONSTRUCTOR METHOD ...
```

```
BEGIN
```

```
  DECLARE  fk REF(FakultaetT);
```

```
  DECLARE s StudentT;
```

```
  SET s = NEW StudentT();
```

```
  SET s.Name = p-name;
```

```
  SET s.MatNr = mat-nr;
```

```
  SET s.Hauptfach = hauptfach;
```

```
  SET s.Nebenfach = nebenfach;
```

```
  SET s.Bachelor = bachelor;
```

```
  SELECT f.oid INTO fk  
  FROM  Fakultaet-Table f  
  WHERE f.Fname = f-name;
```

```
  SET      s.Fak = fk;
```

```
  UPDATE Fakultaet-Table
```

```
  SET Studenten =
```

```
      Studenten || ARRAY[s.oid]
```

```
  WHERE Fname = f-name;
```

```
  RETURN s;
```

```
END;
```

# Aufgabe 3a – DTD

---

<?xml version="1.0" encoding="UTF-8"?>

<!ELEMENT *bib* (biblioentry\*)>

<!ELEMENT *biblioentry* (authorgroup,title,subtitle?,  
publisher?, pubdate,  
pagenums?)>

<!ATTLIST *biblioentry*

*id* ID #REQUIRED>

<!ELEMENT *authorgroup* (author+,othercredit\*)>

<!ELEMENT *title* (#PCDATA)>

<!ELEMENT *subtitle* (#PCDATA)>

<!ELEMENT *publisher* (publishername,address)>

<!ELEMENT *pubdate* (#PCDATA)>

<!ELEMENT *pagenums* EMPTY>

<!ATTLIST *pagenums*

*start* NMTOKEN #REQUIRED

*end* NMTOKEN #REQUIRED>

<!ELEMENT *author* ( surname,  
firstname+)>

<!ELEMENT *othercredit*( surname,  
firstname+,  
address)>

<!ELEMENT *publishername* (#PCDATA)>

<!ELEMENT *address* ( email?, city?)>

<!ELEMENT *email* ( #PCDATA)>

<!ELEMENT *city* ( #PCDATA)>

<!ELEMENT *surname* ( #PCDATA)>

<!ELEMENT *firstname* ( #PCDATA)>

# Aufgabe 3b – Instanz

---

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

```
<!DOCTYPE bib SYSTEM "bib.dtd">
```

```
<bib xmlns:xsi=  
  "http://www.w3.org/2001/XMLSchema-instance"  
  xmlns="http://dbs.uni-leipzig.de/ns/bib"  
  xsi:schemaLocation=  
    "http://dbs.uni-leipzig.de/ns/bib  
    bib.xsd"  
>
```

```
<biblioentry id="BeBoRa11">
```

```
<authorgroup>
```

```
<author>
```

```
<surname>Zohra</surname>
```

```
<firstname>Bellahsene</firstname>
```

```
</author>
```

```
<author>
```

```
<surname>Angela</surname>
```

```
<firstname>Bonifati</firstname>
```

```
</author>
```

```
<author>
```

```
<surname>Erhard</surname>
```

```
<firstname>Rahm</firstname>
```

```
</author>
```

```
</authorgroup>
```

```
<title>Schema Matching and Mapping</title>
```

```
<publisher>
```

```
<publishername>Springer Verlag</publishername>
```

```
<address>
```

```
<city>Heidelberg</city>
```

```
</address>
```

```
</publisher>
```

```
<pubdate>2011</pubdate>
```

```
</biblioentry>
```

```
</bib>
```