

## Übung 2 – Musterlösung

Aufgabe 1: Nehmen Sie das Konzeptuelle Modell aus der zweiten Übung (Aufgabe 1- Storchenspiel) und machen Sie daraus einen Implementationsentwurf. Überführen sie im ersten Schritt die Entitäten in Tabellen und im zweiten dann die Beziehungen. Wenn alle Tabellen vorliegen, überprüfen Sie, ob Sie ihren Implementationsentwurf nach den in der Vorlesung präsentierten Regeln vereinfachen können. Benutzen Sie als Datentypen integer für natürliche Zahlen, text für Zeichenketten und Date für Datum

Storch: {[StorchID:INT, Geschlecht: Text, StorchenArt:Text]}

Storchenart: {[ArtnameL:Text, ArtnameD:Text, Storchengattung:Text]}

Storchengattung: {[GatnameL:Text, GatnameD:Text]}

Nest: {[NestID:INT, PLZ:Text]}

wirdGesichtet: {[NestID:INT, StorchID:INT, Datum:Date]}

hatFamilie: {[Kind:StorchID, Eltern:StorchID]}

Aufgabe 2: Erstellen Sie eine Datenbank und erzeugen Sie die Tabellen Storch, Storchenart, Nest und wirdGesichtet. Setzen Sie für jede Tabelle den Primärschlüssel. Fügen Sie pro Tabelle mindestens einen Datensatz ein.

```
CREATE TABLE storchenart(  
  artnameL VARCHAR(39) PRIMARY KEY,  
  artnameD VARCHAR(39));
```

```
CREATE TABLE storch(  
  storchid INT PRIMARY KEY,  
  geschlecht CHAR(1),  
  storchenart VARCHAR(39));
```

```
CREATE TABLE nest(  
  nestid int PRIMARY KEY,  
  plz CHAR(5));
```

```
CREATE TABLE gesichtet(  
  nest INT,  
  storch INT,  
  tag date,  
  primary key(nest,storch,tag));
```

Aufgabe 3: Löschen Sie alle erzeugten Tabellen der Datenbank.

```
drop table storchenart;  
drop table storch;  
drop table nest;  
drop table gesichtet;
```

Aufgabe 4: Erzeugen Sie eine Datenbank mit dem Universitätsschema. Script können Sie sich von moodle herunterladen.

Aufgabe 5: Versuchen Sie die folgenden Abfragen zu formulieren – erst als Anfrage in Relationaler Algebra, dann als SQL-Anfrage:

- a Suchen Sie nach den Namen aller Studenten.

$$\Pi_{\text{Name}}(\text{Studenten})$$

select Name from Studenten;

- b Suchen Sie die Vorlesungsnummer der Vorlesung Ethik.

$$\Pi_{\text{VorlNr}}(\sigma_{\text{titel}='Ethik'}(\text{Vorlesungen}))$$

select VorlNr from Vorlesungen where Titel='Ethik';

- c Suchen Sie alle Studenten, die mehr als 10 Semester oder weniger als 6 Semester studiert haben.

$$\Pi_{\text{Name}}(\sigma_{\text{Semester}>10 \text{ or Semester}<6}(\text{Studenten}))$$

select Name from Studenten where Semester>10 or Semester<6;

- d Suchen Sie die Nummer aller Vorlesungen, die Herr Carnap gehört hat.

$$\Pi_{\text{VorlNr}}(\sigma_{\text{ hoeren.MatrNr=Studenten.MatrNr and Name='Carnap' }}(\text{ hoeren x Studenten}))$$

select VorlNr from Studenten s, hoeren h where s.VorlNr=h.VorlNr and s.Name='Carnap';

- e Suchen Sie die Namen der Studenten, die alle mit 1 geprüft worden sind.

$$\Pi_{\text{Name}}(\sigma_{\text{ Studenten.MatrNr=pruefen.MatrNr and note=1 }}(\text{ Studenten x pruefen}))$$

select Name from Studenten s, pruefen p where s.MatrNr=p.MatrNr and Note=1;

- f Suchen Sie alle Assistenten, die für Sokrates arbeiten.

$$\Pi_{\text{Name}}(\sigma_{\text{ Professoren.PersNr=boss and Professoren.Name='Sokrates' }}(\text{ Assistenten x Professoren}))$$

select a.Name from Assistenten a, Professoren p where a.Boss=p.PersNr and p.Name='Sokrates';

- g Suchen Sie alle Studenten, die die Vorlesung ‚Grundzüge‘ gehört haben.

$$\Pi_{\text{Name}}(\sigma_{\text{ Studenten.MatrNr= hoeren.MatrNr and Vorlesungen.VorlNr= hoeren.VorlNr and titel='Grundzuege' }}(\text{ Vorlesungen x hoeren x Studenten s}))$$

select s.Name from Studenten s, Vorlesungen v, hoeren h where s.MatrNr=h.MatrNr and h.VorlNr=v.VorlNr and v.Titel='Grundzuege';

- h Suchen Sie den Titel der Vorlesungen in der sich Herr Carnap hat prüfen lassen und lassen Sie sich zusätzlich noch die Note anzeigen.

$$\Pi_{\text{titel}}(\sigma_{\text{ pruefen.VorlNr=Vorlesungen.VorlNr }}(\text{ Vorlesungen x } (\sigma_{\text{ pruefen.MatrNr=Studenten.MatrNr }}(\text{ pruefen x } (\sigma_{\text{ Name='Carnap' }}(\text{ Studenten}))))))$$

```
select v.Titel, p.Note from Vorlesungen v, Studenten s, pruefen p, Vorlesungen v where  
v.VorlNr=p.VorlNr and s.MatrNr=p.MatrNr and s.Name='Carnap';
```