

*Modul Datenmanagement (DMG)***Übung S2: Advanced SQL****1. Installation von PostgreSQL**

Viele fortgeschrittene Konzepte von SQL werden von MySQL nicht unterstützt. Deshalb arbeiten wir auch mit dem PostgreSQL Datenbankserver.

- ☞ Installieren Sie PostgreSQL. Sie finden Die Installationsdateien auf dem Web, oder auf unserem FTP Server:

<ftp://ftp.enterpriselab.ch/DMG/Software/PostgreSQL/>

- ☞ Sie können auch die neuste Version aus dem Internet laden.
- ☞ Erstellen Sie eine neue Datenbank „uni“ und erstellen Sie die Uni-Datenbank mit den SQL-Skripten auf ILIAS.

2. SQL-Aufgaben

Mit der Installation der Uni-DB auf ihrem lokalen PostgreSQL-Server können Sie nun die folgenden Aufgaben lösen:

2.1. Mengenvergleich (In)

- ☞ Finden Sie die Namen der Professoren, die noch keine Prüfung abgenommen haben.

name
Kopernikus
Popper
Augustinus
Curie

```
SELECT name FROM professoren LEFT OUTER JOIN pruefen  
ON professoren.`PersNr` = pruefen.`PersNr` WHERE pruefen.`Note` is null
```

2.2. Left outer join

- ☞ Geben Sie eine Liste aller Professoren (alle Attribute) aus. Sofern ein Professor einen Assistenten hat, soll dessen Name und Fachgebiet ebenfalls ausgegeben werden.

```
SELECT professoren.PersNr, professoren.Name, Rang, Raum, assistenten.Name, assistenten.fachgebiet  
FROM professoren LEFT OUTER JOIN assistenten  
ON professoren.`PersNr` = assistenten.`Boss`
```

PersNr	Name	Rang	Raum	Name	Fachgebiet
2125	Sokrates	C4	226	Platon	Ideenlehre
2125	Sokrates	C4	226	Aristoteles	Syllogistik
2126	Russel	C4	232	Wittgenstein	Sprachtheorie
2127	Kopernikus	C3	310	Rhetikus	Planetenbewegung
2127	Kopernikus	C3	310	Newton	Keplersche Gesetze
2133	Popper	C3	52	NULL	NULL
2134	Augustinus	C3	309	Spinoza	Gott und Natur
2136	Curie	C4	36	NULL	NULL
2137	Kant	C4	7	NULL	NULL

2.3. Full outer join

- ☞ Geben Sie eine Liste aller Studenten (MatrNr und Name) und aller Vorlesungen (VorINr und Titel), die sie hören, aus. Sofern ein Student keine Vorlesung hört oder eine Vorlesung nicht besucht wird, sollen die entsprechenden Informationen des Studenten trotzdem ausgegeben werden.

Result Grid		Filter Rows:	Search	Export:
MatrNr	Name	VorINr	Titel	
26120	Fichte	5001	Grundzuege	
27550	Schopenhauer	5001	Grundzuege	
29120	Theophrastos	5001	Grundzuege	
29555	Feuerbach	5001	Grundzuege	
28106	Carnap	5041	Ethik	
29120	Theophrastos	5041	Ethik	
29120	Theophrastos	5049	Maeeutik	
27550	Schopenhauer	4052	Logik	
28106	Carnap	5052	Wissenschafts...	
28106	Carnap	5216	Bioethik	
28106	Carnap	5259	Der Wiener Kreis	
29555	Feuerbach	5022	Glaube und W...	
25403	Jonas	5022	Glaube und W...	
24002	Xenokrates	NULL	NULL	
26830	Aristoxenos	NULL	NULL	
NULL	NULL	5043	Erkenntnistheorie	
NULL	NULL	4630	Die 3 Kritiken	

```
SELECT * FROM studenten LEFT OUTER JOIN hoeren
ON studenten.MatrNr = hoeren.`MatrNr`
FULL OUTER JOIN vorlesungen
ON hoeren.`VorINr` = vorlesungen.`VorINr`
```

2.4. Right outer join

- ☞ Schauen sie sich folgenden äusseren join an.

```
select v.Titel, s.name
from vorlesungen v
```

left outer join hören h **on** h.VorlNr = v.VorlNr

right outer join Studenten s **on** s.MatrNr = h.MatrNr

- ? Was geschieht, wenn Sie „right outer join“ durch „left outer join“ ersetzen?
Anstatt alle studenten werden dann alle Vorlesungen angezeigt
- ? Was geschieht, wenn Sie in der letzten Zeile „full outer join“ verwenden?
wird beides angezeigt... wie Aufgabe 2.3

3. Zusammenhang Relationenalgebra und SQL: Mengenorientierte Operatoren

- ☞ Für die folgenden Aufgaben, geben Sie die Lösung jeweils (1) als SQL-Code und (2) als mathematische Formel der Relationenalgebra an.

Die Beispiele beziehen sich auf die Uni-Datenbank.

3.1. Umbenennung

Projizieren Sie die Relation Professoren auf die Attribute Personennummer und Name. Benennen Sie dabei das Attribut „PersNr“ in „Nr“ um. Benennen Sie die neue Relation „P“, und speichern Sie sie für die weitere Verwendung.

Tun Sie das gleiche für die Relation Assistenten, und nennen Sie diese Relation „A“.

3.2. Vereinigung

Vereinigen Sie die beiden vorhergehend kreierte Relationen, und nennen Sie das Resultat Mitarbeiter.

3.3. Schnittmenge

Welches sind die Namen der Mitarbeiter, die gleichzeitig Professor und Assistent sind?

3.4. Differenz

Welches sind die Namen der Assistenten, welche nicht gleichzeitig Student sind?

3.5. Division

Welches sind die Namen der Studenten, welche *alle* Vorlesungen besucht haben?

Testen können Sie dies mit folgendem Studentendatensatz:

```
insert into Studenten values (123, 'Streber', 7);  
insert into hören select distinct 123, VorlNr from vorlesungen;
```

4. Abgabe der Übung

- Ergänzen Sie ihr Dokument mit den Namen der Teammitglieder, welche zur Lösung der Aufgabe beigetragen haben.
- Erstellen Sie ein PDF mit den Lösungen zu den Aufgaben: Übung_S2_Gruppe_<XY>.pdf
- Laden Sie die Datei als PDF auf ILIAS in den Briefkasten S2
- Abgabetermin: Siehe Semesterplan Detail (auf ILIAS > Organisatorisches)

Aufgabe 3: SQL/Relationale Algebra

3.1 Umbenennung

$$\Pi_{\text{persnr}, \text{name}} \rho_{\text{nr} \leftarrow \text{persnr}, \text{p} \leftarrow \text{professoren}}(\text{professoren})$$

```
select persnr as nr, name from professoren as p;
```

$$\Pi_{\text{persnr}, \text{name}} \rho_{\text{nr} \leftarrow \text{persnr}, \text{a} \leftarrow \text{assistenten}}(\text{professoren})$$

```
select persnr as nr, name from assistenten as a;
```

3.2 Vereinigung

$$\Pi_{\text{nr}, \text{name}}[\Pi_{\text{persnr}, \text{name}} \rho_{\text{nr} \leftarrow \text{persnr}}(\text{professoren} \cup \text{assistenten})]$$

```
select nr, name from (  
    select persnr as nr, name from professoren  
    union  
    select persnr as nr, name from assistenten  
) as mitarbeiter;
```

3.3 Schnittmenge

$$\Pi_{\text{nr}, \text{name}}[\Pi_{\text{persnr}, \text{name}} \rho_{\text{nr} \leftarrow \text{persnr}}(\text{professoren} \cup \text{assistenten})] \sigma_{\text{name} \in [\Pi_{\text{name}}(\text{professoren}) \wedge \Pi_{\text{name}}(\text{assistenten})]}$$

```
select name from (  
    select persnr as nr, name from professoren  
    union  
    select persnr as nr, name from assistenten  
) as mitarbeiter  
where name in (select name from professoren)  
and name in (select name from assistenten);
```

3.4 Differenz

$$\Pi_{\text{name}} \sigma_{\text{name} \notin [\Pi_{\text{name}}(\text{studenten})]}(\text{assistenten})$$

```
select name from assistenten  
where name not in (select name from studenten);
```

3.5 Division

$$\Pi_{\text{matnr}, \text{count}(\text{ hoeren.vorlnr})}(\text{ hoeren}) \Gamma_{\text{matnr}} \sigma_{\text{count}(\text{vorlnr}) = [P_{i_{\text{count}(\text{vorlnr})}(\text{vorlesungen})}]$$

```
select matnr, count(hoeren.vorlnr)
from hoeren
group by matnr
having count(vorlnr) = (select count(vorlnr) from vorlesungen)
```

(Da es in der relationalen Algebra kein Equivalent zu GROUP BY zu geben scheint, habe ich Γ dafür verwendet.)