

5. Übung zur Vorlesung “Datenbanken” im Sommersemester 2004

Prof. Dr. Gerd Stumme, Dipl.-Inform. Christoph Schmitz

1. Juni 2004

Hinweise zu den Übungen am Rechner: Für die Übungen steht das Datenbanksystem IBM DB2 8.1 bereit. Zum Zugriff auf die Datenbank steht ein Web-Frontend zur Verfügung, das unter der Adresse <http://www.kde.cs.uni-kassel.de/sql> aufgerufen werden kann.

Verwenden Sie den Benutzernamen db04_XXX und das entsprechende Passwort.

Aufgabe 1

Rufen Sie in einem Internet-Browser das Frontend über die oben angegebene Adresse auf und melden Sie sich an.

Machen Sie sich mit der DB2-Hilfe im DB2 Information Center vertraut. Der Link dazu steht im Web-Frontend:

<http://publib.boulder.ibm.com/infocenter/db2help/index.jsp>

Schauen Sie sich insbesondere die Rubrik “Referenzinformationen → SQL” an und merken Sie sich diese Quelle für späteres Nachschlagen ;-)

Aufgabe 2

Verwenden Sie den Befehl `CREATE TABLE`, um eine Beispieltabelle mit dem Namen `TEST` anzulegen, die eine numerische Spalte, eine Text- sowie eine Datumsspalte besitzt.

Informieren Sie sich dazu im DB2 Infocenter über die von DB2 unterstützten Datentypen.

Fügen Sie nun mittels `INSERT` einige Zeilen in Ihre Tabelle ein und schauen Sie sich diese mit `SELECT` wieder an.

Bemerkungen: Das SQL-Interface erlaubt es, mehrere Befehle nacheinander mit Semikolon separiert einzugeben:

```
create table foo (bar integer);
drop table foo;
```

Bei Befehlen, die keine Resultate liefern (etwa `SELECT`) ist eine Ausgabe wie diese ein Zeichen, das alles funktioniert hat:

```
Results:  
create table foo (bar integer)  
0 columns, 0 rows.
```

Aufgabe 3

Um Tabellen für das Schulbeispiel anzulegen, steht unter der Adresse <http://www.kde.cs.uni-kassel.de/lehre/ss2004/datenbanken/daten/schule.sql> ein SQL-Skript zur Verfügung (siehe unten).

Laden Sie das Skript herunter und kleben es per Cut&Paste in das Query-Interface ein. Machen Sie sich die Bedeutung der einzelnen Tabellen und Attribute klar und führen Sie das Skript aus. Beachten Sie hierbei, dass das Skript eventuell bereits vorhandene gleichnamige Tabellen zunächst löscht. Dies ist auch der Grund dafür, dass das Skript beim erstmaligen Ausführen einige Fehler wegen nicht vorhandener Tabellen ausgibt. Sie können diese Meldungen ignorieren.

- Fügen Sie in das Schema mindestens einen Lehrer, drei Schüler und drei Unterrichtseinheiten ein.
- Ergänzen Sie dazu auch die weiteren Tabellen mit passenden Datensätzen.
- Geben Sie den Schülern Noten.
- Schauen Sie sich Ihre Ergebnisse mit Hilfe geeigneter `SELECT`-Anweisungen an.

```
drop table lehrer;  
drop table fach;  
drop table lehrbefugnis;  
drop table klasse;  
drop table schueler;  
drop table unterricht;  
drop table zeugnis;  
  
create table lehrer (  
    name varchar(40) not null,  
    gebdat date,  
    adresse varchar(80),  
    gehalt integer,  
    primary key(name)  
);  
  
create table fach (  
    fach varchar(40) not null,
```

```

    primary key (fach)
);

create table lehrbefugnis (
    lehrername varchar(40) not null,
    fach varchar(40) not null,
    primary key (lehrername,fach),
    foreign key (lehrername) references lehrer(name),
    foreign key (fach) references fach
);

create table klasse (
    jahrgang integer not null,
    buchst char not null,
    klassenlehrer varchar(40) not null,
    klassensprecher varchar(40),
    primary key (jahrgang,buchst),
    foreign key (klassenlehrer) references lehrer(name)
);

create table schueler (
    name varchar(40) not null,
    gebdat date,
    adresse varchar(80),
    jahrgang integer not null,
    buchst char not null,
    primary key (name),
    foreign key (jahrgang,buchst) references klasse
);

create table unterricht (
    jahrgang integer not null,
    buchst char not null,
    lehrername varchar(40) not null,
    fach varchar(40) not null,
    stunden integer,
    primary key (jahrgang,buchst,fach),
    foreign key (jahrgang,buchst) references klasse(jahrgang,buchst),
    foreign key (lehrername,fach) references lehrbefugnis(lehrername,fach)
);

create table zeugnis (
    schuelername varchar(40) not null,
    fach varchar(40) not null,
    note char(10),
    primary key (schuelername,fach),
    foreign key (schuelername) references schueler(name),
    foreign key (fach) references fach
);

```

```
alter table klasse
  add foreign key (klassensprecher) references schueler(name);
```

Aufgabe 4

Probieren Sie die Queries vom letzten Aufgabenblatt auf den Daten der Philosophenuni aus. Diese Tabellen können Sie anfragen, indem Sie das Schema “uni” voranstellen, also etwa:

```
SELECT titel, name
FROM uni.professoren, uni.vorlesungen
WHERE gelesenvon = persnr
```

Sie können auch dauerhaft in das andere Schema wechseln mit

```
SET SCHEMA uni
```

Ihr eigenes Schema heißt entsprechend dem Usernamen db04_XXX.