

Informationssysteme (SS 05)



Übungsblatt 5

Ausgabe: 17.Mai 2004

Abgabe: 24.Mai 2004 in der Vorlesung

Aufgabe 1: Anfragen in SQL - Universitätsdatenbank

Gegeben sei das aus der ersten Übung bekannte Schema einer Universitätsdatenbank:




Professor	(P_Name, Fachrichtung_Nr, Gebäude, Raum, Tel)
Fachrichtung	(Fachrichtung_Nr, F_Name, Studiendekan)
Gebäude	(Gebäude, Hausmeister)
Student	(Matrikel_Nr, S_Name, Semester, Fachrichtung_Nr)
Prüfung	(Matrikel_Nr, Fach, Prüfer, Note)

Formulieren Sie die folgenden Anfragen in SQL:

- An welchen Hausmeister muß sich Prof. Weikum wenden, wenn er seinen Zimmerschlüssel vergessen hat?
- Welche Studenten (Matrikel_Nr) haben eine Prüfung beim augenblicklichen Studiendekan ihrer Fachrichtung abgelegt?
- Wo (Gebäude, Raum) fand die Prüfung von Hugo Meier im Fach Betriebssysteme statt (Annahme: Professoren prüfen in ihren Büros)?
- Welche Studenten (Matrikel_Nr) mit mindestens 4 Semestern haben noch keine Prüfung abgelegt?
- Welche Studenten (Matrikel_Nr) haben ausschließlich Prüfungen bei Professoren ihrer Fachrichtung abgelegt?
- Welche Studenten (Matrikel_Nr) haben alle ihre bisher abgelegten Prüfungen mit der Bestnote 1.0 bestanden?
- Bestimmen Sie die Durchschnittsnote für jeden Studenten.
- Welche Studenten (Matrikel_Nr) haben ein Prüfungsfach (Fach) besser abgeschlossen als der Durchschnitt in diesem Fach?
- Welches Prüfungsfach (Fach) hat die beste Durchschnittsnote?
- Welcher Student (Name) hat alle Prüfungen als Bester abgeschlossen?

Aufgabe 2: Anfragen in SQL - Musikdatenbank

Gegeben sei ein gegenüber Übung 1 erweitertes Schema der Musikdatenbank:



Disk	(DiskID, DiskTitel, Preis) <i>78462, W. A. Mozart: Klavierkonzerte, 29.99</i>
Musikstück	(DiskID, StückID, Titel, Länge) <i>78462, 4, Konzert für Klavier und Orchester Nr. 21, 2732</i>
Person	(PID, Name, Nationalität) <i>9362, W. A. Mozart, Österreich</i>
Interpret	(PID, DiskID, StückID, Funktion, Instrument) <i>15267, 78462, 4, Solist, Klavier</i>
Autor	(PID, DiskID, StückID, Tätigkeit) <i>9362, 78462, 4, Komponist</i>

Formulieren Sie die folgenden Anfragen in SQL:

- Welche Stücke (Titel) hat F. Chopin komponiert?
- Welches ist die teuerste Disk und was kostet sie?
- Welche Disk enthält das längste Stück unter den Disks, die nicht mehr als 20 (Mark) kosten?
- Welche Disks (DiskTitel) enthalten ausschließlich Stücke, die von F. Chopin komponiert wurden?
- Welche Disks enthalten kein Stück, das länger ist als 60 (Sekunden)?
- Welchen Durchschnittspreis haben Disks, auf denen Interpreten aus über 3 Nationen zu hören sind?
- Ermitteln Sie für jede Disk die Gesamtlänge der drei längsten Stücke.

Aufgabe 3: Abbildung von SQL auf TRK und RA



Geben Sie für die folgenden SQL-Anfragen auf der Musikdatenbank äquivalente Formulierungen in der Relationenalgebra und dem sicheren Tupelrelationenkalkül an.

- ```
Select D.DiskTitel
From Disk D
Where D.DiskID In
 (Select M.DiskID From Musikstück M
 Where M.Titel = 'I love you'
 And M.DiskID In
 (Select I.DiskID From Interpret I
 Where I.Instrument = 'Triangel'
 And I.StückID = M.StückID))
```
- ```
Select D.DiskTitel
From Disk D
Where D.Preis < 20
And Exists
      ( Select *
        From Interpret I
        Where I.Instrument = 'Sitar'
          And I.DiskID = D.DiskID )
And Not Exists
      ( Select *
        From Musikstück M
        Where M.Länge > 5
          And M.DiskID = D.DiskID )
```
- ```
Select D.DiskTitel
From Disk D
Where D.DiskID = All (Select M.DiskID
 From Musikstück M, Person P, Interpret I, Autor A
 Where (M.StückID = I.StückID Or M.StückID = A.StückID)
 And (P.PID = I.PID Or P.PID = A.PID)
 And P.Nationalität = 'Luxemburg')
```

### Aufgabe 4: SQL Abfragen



Betrachten Sie die vereinfachte Universitätsdatenbank mit Informationen über Fachbereiche, Dozenten, Lehrangebote, Studenten und Prüfungen. Das Schema der Datenbank (mit Beispielausprägungen) ist unten aufgeführt (Primärschlüssel sind unterstrichen):

**Departments** (DName, Chair):

| <u>DName</u>                  | Chair              |
|-------------------------------|--------------------|
| <i>Computer Science</i>       | <i>Bob Smith</i>   |
| <i>Electrical Engineering</i> | <i>John Miller</i> |
| ...                           | ...                |

**Teachers** (TName, Office, Phone, DName):

| <u>TName</u>         | Office | Phone  | DName                         |
|----------------------|--------|--------|-------------------------------|
| <i>Bob Smith</i>     | 122    | 4819   | <i>Computer Science</i>       |
| <i>Mary Taylor</i>   | 245    | 4716   | <i>Computer Science</i>       |
| <i>John Miller</i>   | 312    | 223322 | <i>Electrical Engineering</i> |
| <i>Mike Franklin</i> | 444    | 4545   | <i>Electrical Engineering</i> |
| ...                  | ...    | ...    | ...                           |

**Courses** (CNo, Title, Semester, Room, Schedule, Lecturer):

| <u>CNo</u> | Title                         | Semester           | Room | Schedule | Lecturer             |
|------------|-------------------------------|--------------------|------|----------|----------------------|
| 1          | <i>Database Systems</i>       | <i>Summer 2001</i> | 322  | ...      | <i>Mary Taylor</i>   |
| 2          | <i>Wireless Communication</i> | <i>Winter 2001</i> | 455  | ...      | <i>John Miller</i>   |
| 3          | <i>Wireless Communication</i> | <i>Summer 2002</i> | 455  | ...      | <i>Mike Franklin</i> |
| ...        | ...                           | ...                | ...  | ...      | ...                  |

**Students** (SNo, SName, Address, Major, Minor)

| <u>SNo</u> | SName               | Address | Major                         | Minor                   |
|------------|---------------------|---------|-------------------------------|-------------------------|
| 1001       | <i>David Chang</i>  | ...     | <i>Computer Science</i>       | <i>Psychology</i>       |
| 1002       | <i>Sunita Singh</i> | ...     | <i>Electrical Engineering</i> | <i>Computer Science</i> |
| 1003       | <i>Joe Doe</i>      | ...     | <i>Electrical Engineering</i> | <i>Physics</i>          |
| ...        | ...                 | ...     | ...                           | ...                     |

**Exams** (SNo, CNo, EDate, Grade):

| <u>SNo</u> | <u>CNo</u> | EDate               | Grade  |
|------------|------------|---------------------|--------|
| 1001       | 1          | <i>27 July 2001</i> | 1.7    |
| 1002       | 2          | <i>15 July 2001</i> | 2.0    |
| 1002       | 1          | <i>28 July 2001</i> | 1.3    |
| 1003       | 3          | <NULL>              | <NULL> |
| ...        | ...        | ...                 | ...    |

Das Attribut *Chair* der Relation *Department* ist ein Fremdschlüssel bzgl. *Teacher.TName*; *Courses.Lecturer* ist Fremdschlüssel bzgl. *Teacher.TName*; in der Relation *Students* sind *Major* (Hauptfach) und *Minor* (Nebenfach) Fremdschlüssel bzgl. *Department.DName*.

Formulieren Sie folgende Anfragen in SQL :

- a) Wie heisst der Dekan (*Chair*) des Fachbereichs (*Department*) mit der besten Durchschnittsnote, berechnet über alle Kurse des Sommersemesters 2002?
- b) Welche Studenten sind im Nebenfach (*Minor*) ständig erfolgreicher als im Hauptfach (*Major*)? Auszugeben sind die Namen solcher Studenten sowie deren Haupt- und Nebenfach.

"Ständig erfolgreicher" bedeutet hier, dass alle Noten im Nebenfach besser sind als die beste Note im Hauptfach.