

--	--	--

Name

Vorname

Matrikelnummer

# Klausur

## KV Datenbanksysteme WS 2001/2002

---

Bitte freilassen!

1	2	3	4	5	6	7	8	9

Summe

--

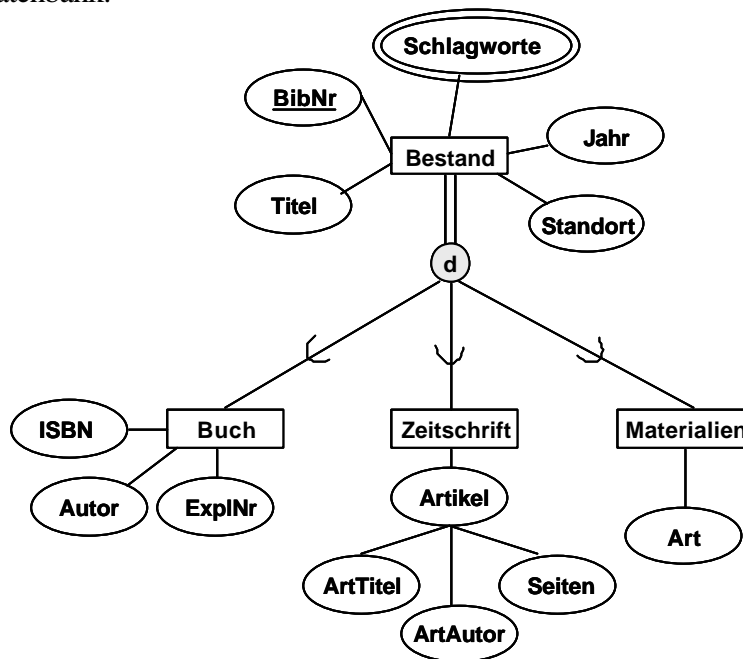
### Aufgabe 1: Abbildung EGBM auf relationales Modell (18 Punkte)

Pro Regel sind nur jene Relationenschemata aufzuführen, die neu hinzugekommen sind oder sich geändert haben. Die einzelnen Relationenschemata sind in der Form:

*Relationenschemaname (Attributname1, ..., AttributnameN)*

aufzuschreiben. Primärschlüssel sind zu unterstreichen. Attributdomänen werden dagegen nicht verlangt.

Gegeben sei folgender Ausschnitt des Gegenstands-Beziehungs-Diagramms einer Bibliotheksdatenbank:



Bilden Sie mit Hilfe **einer möglichen Variante der achten Regel** die dargestellte Spezialisierung auf das relationale Modell ab.

Welche der anderen Varianten wären auch anwendbar (*nur Nennung bzw. kurze Charakterisierung, keine vollständige Anwendung!*)? Nennen Sie Vor- und Nachteile dieser möglichen Varianten.



**Aufgabe 2: Normalisierung****(10 Punkte)**

Gegeben sei folgendes Relationenschema mit Informationen über Vorlesungen und Studierende, sowie die nachfolgende Ausprägung:

Vorlesungen (Vorlesungsnummer, Leginummer, Vorlesungsname, StudentenName)

Vorlesungen

Vorlesungsnummer	Leginummer	VorlesungsName	StudentenName
100	500	Datenbanken	Markus Mustermann
100	501	Datenbanken	Hans Iseli
101	501	Programmiersprachen	Hans Iseli
101	502	Programmiersprachen	Barbara Schniderlin
200	500	Rechnungswesen	Markus Mustermann

Frage 2.1:

(4 Punkte)

Befindet sich die oben aufgeführte Relation in zweiter Normalform? Begründen Sie Ihre Aussage.

Frage 2.2:

(6 Punkte)

Sie haben folgende Update-Anomalien kennengelernt: INSERT-, DELETE-, UPDATE-Anomalie. Erklären Sie jede dieser Anomalien an dem oben genannten Relationenschema.

**Aufgabe 3: SQL DML****(17 Punkte)**

Gegeben sei folgendes Schema einer Studentendatenbank:

Vorlesungen(Vorlesungsnummer, Vorlesungsname, Abteilung)Studenten(Leginummer, Name, Geschlecht, Kanton)Einschreibungen(Vorlesungsnummer, Leginummer)

mit folgender Ausprägung:

## STUDENTEN

Leginummer	Name	Geschlecht	Kanton
500	Markus Mustermann	M	ZH
501	Hans Iseli	M	ZH
502	Barbara Schniderlin	W	BL
503	Claudia Bachmann	W	AG

## VORLESUNGEN

Vorlesungsnummer	Vorlesungsname	Abteilung
100	Datenbanken	Informatik
101	Programmiersprachen	Informatik
200	Rechnungswesen	BWL

## EINSCHREIBUNGEN

Vorlesungsnummer	Leginummer
100	500
100	501
101	503
200	502
101	500

**ACHTUNG:** Berücksichtigen Sie bei der Anfrageformulierung nur das Datenbankschema, nicht die beispielhafte Ausprägung!

Frage 3.1:

(3 Punkte)

Welches Ergebnis liefert die folgende SQL-Abfrage:

```
select Leginummer, Name  
from Studenten  
where Geschlecht = 'W' or Kanton = 'BL'
```

Frage 3.2:

(4 Punkte)

Formulieren Sie eine SQL-Abfrage, welche für alle Studierenden die Leginummer und die Anzahl der eingeschriebenen Vorlesungen auflistet.

Frage 3.3:

(5 Punkte)

Formulieren Sie eine SQL-Abfrage, um für alle Vorlesungen die eingeschriebenen Studierenden herauszufinden. Sortieren Sie das Ergebnis alphabetisch nach dem Vorlesungsnamen.

Frage 3.4:

(5 Punkte)

Der Student Markus Mustermann schreibt sich auch für die Vorlesung Rechnungswesen ein. **Vervollständigen Sie folgenden SQL-Ausdruck**, so dass diese Daten in die Datenbank eingefügt werden.

***ACHTUNG: Berücksichtigen Sie bei der Anfrageformulierung nur das Datenbankschema, nicht die beispielhafte Ausprägung!***

**insert into**

**select**

**Aufgabe 4: Relationale Algebra****(15 Punkte)**

Gegeben sei das folgende logische Modell einer Bank-DB gegeben:

FILIALE (FNR, BLZ, NAME, LEITER)  
KONTO (KTNR, SALDO, DISPO, KNR, FNR)  
KREDIT (DNR, BETRAG, RATE, ZINS, KNR, FNR)  
SPARBUCH (SNR, GUTHABEN, ZINSSATZ, KNR, FNR)  
KUNDE (KNR, NAME, VORNAME, GEBDATUM)

Stellen Sie an die Datenbank nun folgende Anfragen mittels Ausdrücken der Relationalen Algebra

**Frage 4.1:****(4 Punkte)**

Listen Sie die Nummern (und nur diese) der Konten (KTNR) auf, welche den Kunden mit dem Namen (VORNAME und NAME) 'Anton Maier' gehören.

**Frage 4.2:****(4 Punkte)**

Ermitteln Sie die Namen der Filialen (NAME), die Kunden haben, die sowohl Konten als auch Sparbücher mit einem Guthaben (GUTHABEN) bzw. Saldo (SALDO) von mehr als 5000 CHF haben.

Frage 4.3:

(3 Punkte)

Was berechnet der folgende in der Relationalen Algebra formulierte Ausdruck?

Q1:  $\sigma_{\text{FNR} = '555'}(\text{FILIALE}) *_{\text{FNR}=\text{FNR}} \pi_{\text{KTNR}, \text{FNR}}(\text{KONTO})$

Q2:  $\pi_{\text{NAME}, \text{VORNAME}}(\text{Q1} *_{\text{KNR}=\text{KNR}} \text{KUNDE})$

Frage 4.4:

(4 Punkte)

Was berechnet der folgende in der Relationalen Algebra formulierte Ausdruck?

Q1:  $\sigma_{\text{ZINSSATZ} > '3\%'}(\text{SPARBUCH}) *_{\text{FNR}=\text{FNR}} \sigma_{\text{NAME} = 'Neumünster'}(\text{FILIALE})$

Q2:  $\pi_{\text{KNR}}(\text{Q1}) *_{\text{KNR}=\text{KNR}} \text{KUNDE}$

**Aufgabe 5: Konsistenzbedingungen des  
Relationenmodells**

**(23 Punkte)**

Frage 5.1: SQL DDL

(9 Punkte)

Gegeben sei folgendes Relationenschema einer Studentendatenbank:

Vorlesungen(Vorlesungsnummer, Vorlesungsname, Wochentag, Uhrzeit, Abteilung)

Wie würden Sie folgende Konsistenzbedingungen mittels SQL ausdrücken?

- a) Wochentag darf nur als 'Mo', 'Di', 'Mi', 'Do', 'Fr' angegeben werden.
- b) Die Vorlesungen finden zwischen 8:00 und 20:00 Uhr statt.
- c) Die Vorlesungsnummer identifizieren eindeutig ein Tupel.
- d) Das Feld Wochentag muss gefüllt sein, d.h. es darf keine leeren Einträge geben.

Frage 5.2: Implizite Konsistenzbedingungen

(8 Punkte)

Sie haben in der Vorlesung mehrere Arten impliziter Konsistenzbedingungen des Relationenmodells kennengelernt. Zählen sie zwei davon auf. Geben Sie für jede genannte Konsistenzbedingung ein Beispiel in SQL an. Benutzen Sie dabei das folgende Schema einer Studentendatenbank:

Vorlesungen(Vorlesungsnummer, Vorlesungsname, Wochentag, Uhrzeit, Abteilung)

Studenten(Leginummer, Name, Geschlecht, Kanton)

Einschreibungen(Vorlesungsnummer, Leginummer)

**Frage 5.3: Explizite Konsistenzbedingungen (5 Punkte)**

Gegeben sei folgendes Schema einer Studentendatenbank:

Vorlesungen(Vorlesungsnummer, Vorlesungsname, Wochentag, Uhrzeit, Abteilung)

Studenten(Leginummer, Name, Geschlecht, Kanton)

Einschreibungen(Vorlesungsnummer, Leginummer)

Definieren Sie in SQL folgende Konsistenzbedingung :

„Ein Student darf sich nicht für eine Vorlesungen einschreiben, in der bereits 120 Studenten eingeschrieben sind“.

**Aufgabe 6: Serialisierbarkeit****(16 Punkte)**

Gegeben seien folgende Transaktionen  $T_1$ ,  $T_2$ ,  $T_3$  sowie folgende Ausführungspläne  $A_1$ ,  $A_2$ :

$$T_1: r_1(A) \ r_1(C) \ w_1(A)$$
$$T_2: r_2(C) \ r_2(B) \ w_2(C) \ w_2(B)$$
$$T_3: r_3(A) \ r_3(B) \ w_3(B)$$
$$A_1: r_1(A) \ r_2(C) \ r_1(C) \ r_3(A) \ r_3(B) \ w_1(A) \ w_3(B) \ r_2(B) \ w_2(C) \ w_2(B)$$
$$A_2: r_1(A) \ r_2(C) \ r_3(A) \ r_1(C) \ r_2(B) \ r_3(B) \ w_1(A) \ w_2(C) \ w_3(B) \ w_2(B)$$

Zeichnen Sie für jeden Ausführungsplan den Abhängigkeitsgraphen und geben Sie an, ob er serialisierbar ist oder nicht. Falls er serialisierbar ist, geben Sie einen äquivalenten Ausführungsplan an.

**Aufgabe 7: Zugriffskontrolle****(6 Punkte)**

Die Datenbank einer Arztpraxis hat folgende Benutzer:

- Anton Schütze, der Arzt und Inhaber der Praxis
- Gudrun Meier, die Krankenschwester und Assistentin des Arztes
- Fritz Huber, der Arzthelfer der Praxis

Die Datenbank hat folgende Relationen:

- Patienten (PatNr, Name, Adresse, Versicherung) (gehört Schütze)
- Behandlungen (Bezeichnung, Kosten, Patient) (gehört Schütze)

Herr Schütze führt folgende Operation aus:

```
create view KostenPatient as
select PatNr, Name, Bezeichnung, Kosten
from Patienten; Behandlungen
where PatNr = Patient
```

Folgende Zugriffsmatrix sei für die Datenbank anfangs gegeben:

	Patienten	Behandlungen	KostenPatient
Schütze	all*	all*	all*
Meier	insert	insert	select
Huber	select	select	select

Herr Schütze führt nun nach und nach folgende Operationen aus:

```
grant insert on Patienten to Meier with grant option
grant insert on Behandlungen to Huber
grant all on Patienten to Meier
grant delete on Behandlungen to Meier
grant all on KostenPatient to Meier
```

Welche der folgenden Operationen können aufgrund der derart veränderten Zugriffsmatrix ausgeführt werden? Kreuzen sie die möglichen Operationen an.

*Kreuzen Sie die richtigen Antworten an. Pro Teilaufgabe können mehrere Antworten richtig sein. Bitte beachten Sie, dass für jede falsch angekreuzte Antwort gleich viele Punkte abgezogen, wie für eine richtige Antwort vergeben werden. Negative Punktzahlen ergeben null Punkte für die betreffende Frage.*

- Frau Meier gibt Herrn Huber die insert-Recht auf die Relation Patienten
- Herr Huber trägt einen neuen Patienten in die Relation Patienten ein
- Frau Meier nimmt eine Änderung an den Kosten eines Eintrages in der Relation KostenPatient vor
- Frau Meier fügt der Relation KostenPatient ein neues Tupel hinzu

**Aufgabe 8: Das Systemprotokoll**

**(10 Punkte)**

Erläutern Sie, welche Informationen in der Regel im Systemprotokoll (System Log od. DBMS Journal) gespeichert werden und wie diese im Falle einer undo- oder einer redo-Operation verwendet werden.

**Aufgabe 9: Multiple Choice**

**(6 Punkte)**

*Kreuzen Sie die richtigen Antworten an. Pro Teilaufgabe können mehrere Antworten richtig sein. Bitte beachten Sie, dass für jede falsch angekreuzte Antwort gleich viele Punkte abgezogen, wie für eine richtige Antwort vergeben werden. Negative Punktzahlen ergeben null Punkte für die betreffende Frage.*

**Frage 9.1: Datenunabhängigkeit**

**(3 Punkte)**

Um ein möglichst grosses Mass an Datenunabhängigkeit zu erreichen, sollten drei verschiedene Datenebenen realisiert werden. Dies sind:

- Externes, konzeptuelles und internes Schema
- Attribute, Entities und Beziehungen
- Datensystem, Zugriffssystem und Speichersystem

**Frage 9.2: Transaktionen**

**(3 Punkte)**

Eine Transaktion:

- wird entweder ganz oder ohne Hinterlassen einer Wirkung in der Datenbank ausgeführt
- ist eine atomare Datenbankoperation (wie select, delete, insert, update)
- ist die Menge aller Datenbankoperationen, die zu einem Zeitpunkt parallel ausgeführt werden dürfen
- ist eine Folge von Datenbankoperationen, die die DB von einem konsistenten DB-Zustand in einen konsistenten Zustand überführen
- ist die Einheit der Synchronisation derart, dass Transaktionen nur nacheinander ausgeführt werden dürfen