



**KLAUSUR**

Beispielklausur 1 zum Modul:

**Datenbanken (DBA) in den Bachelor Studiengängen,  
Fakultät für Informatik**

Ich wünsche Ihnen viel Erfolg!

**Name, Vorname (in DRUCKBUCHSTABEN):**

\_\_\_\_\_

**Matrikelnummer:** \_\_\_\_\_

**Unterschrift:** \_\_\_\_\_

**Studiengang:** ☐ IB

☐ IMB

☐ UIB

**WICHTIGE HINWEISE !**

- Legen Sie einen gültigen Ausweis und einen gültigen Studentenausweis auf den leeren Platz neben sich.
- Schreiben Sie mit Füller oder mit Kugelschreiber, verwenden Sie **keinen Bleistift** und **nicht die Farbe Rot!**
- Schreiben Sie die Lösung zu einer Aufgabe auf das vorgesehene Blatt oder auf die Rückseite der vorherigen Aufgabe oder wenn der Platz nicht reichen sollte, benutzen Sie das zusätzliche leere Blatt am Ende der Klausur. **Geben Sie unbedingt einen Verweis auf das Blatt mit der Fortsetzung an** und kennzeichnen Sie dieses Blatt als zu der entsprechenden Aufgabe gehörig.
- Es sind keine Hilfsmittel erlaubt, also **keine Taschenrechner, Skripte, Bücher, eigenen Blätter, Tipp-Ex etc.**
- Die Klausur ist nicht zu zerlegen. **Lose Blätter werden nicht gewertet!**
- Jeder Täuschungsversuch wird mit 0 Punkten = Note 5.0 gewertet.

Aufgabe	1	2	3	4	5
Punkte					

**Gesamtpunkte:** \_\_\_\_\_ von 65

**Note:** \_\_\_\_\_

## AUFGABE 1

Die Universität DB-Stadt (mir ist gestern Nacht ehrlich nichts mehr besseres eingefallen) möchte die Verwaltung des Vorlesungsbetriebes automatisieren und plant deshalb die Benutzung einer Datenbank. Daher muss ein Schema erstellt werden, das die relevanten Daten beschreibt. Die folgenden Informationen wurden vom Datenbank-Designer zusammengetragen:

1. Jede Vorlesung wird organisatorisch von einem Lehrgebiet betreut.
2. Es gibt mehr Vorlesungen als Lehrgebiete.
3. Jeder Professor ist einem Lehrgebiet zugeordnet, es gibt auch Lehrgebiete mit mehreren Professoren.
4. Jede Vorlesung wird von einem einzelnen Professor gehalten.
5. Manche Professoren halten mehrere Vorlesungen pro Semester.
6. Zu jeder Vorlesung werden (im allgemeinen mehrere) Übungsgruppen angeboten.
7. Vorlesungen und Übungsgruppen werden nicht unbedingt jedes Mal im selben Raum angeboten. Beispielsweise wird im laufenden Semester die Vorlesung Datenbanken (DBA) montags von 13:40 Uhr bis 15:10 Uhr im Hörsaal A206, freitags von 08:00 Uhr bis 09:30 Uhr im Hörsaal A212 angeboten.
8. Studenten hören im allgemeinen mehrere Vorlesungen.

Ihre Aufgabe lautet nun wie folgt:

- a) Stellen Sie die oben angegebenen Informationen in Form eines Entity-Relationship-Diagramms dar. Begründen Sie (kurz!) alle Ihre Entscheidungen (warum Entity? warum Beziehung? warum gerade diese Kardinalität der Beziehung?). Überlegen Sie genau, wie Sie die Vorlesungszeiten im Diagramm unterbringen.
- b) Überlegen Sie, ob in Ihrem Diagramm ein Professor eine Vorlesung halten darf, die nicht von dem Lehrgebiet betreut wird, dem er zugeordnet ist. Wenn nein, erstellen Sie eine alternative Lösung, die dies zulässt. Wenn ja, erstellen Sie eine alternative Lösung, die dies nicht zulässt.

**(13 + 4 = 17 Punkte)**

## **FORTSETZUNG AUFGABE 1**

**AUFGABE 2**

Gegeben sei die Relation "Projekt-Material":

T-Nr	P-Nr	Name	Leiter	Menge
026	102	P01	Kemper	30
026	751	P29	Lenzen	40
216	102	P01	Kemper	205
216	319	P27	Lenzen	125
216	666	P89	Köllner	170

Mit den folgenden funktionalen Abhängigkeiten:

(T-Nr, P-Nr) → Menge  
P-Nr → Name  
P-Nr → Leiter

- Begründen Sie, warum die Relation "Projekt-Material" nicht in zweiter Normalform ist.
- Erzeugen Sie aus der Relation "Projekt-Material" ein Relationenschema in zweiter Normalform.

**(4 + 4 = 8 Punkte)**

## **FORTSETZUNG AUFGABE 2**

## AUFGABE 3

Gegeben seien die folgenden Relationen:

r1

a	b	c
20	40	60
80	100	120
140	160	160
20	100	180

r2

a	b	c	d
22	76	60	38
80	4	120	128
14	176	196	208
20	102	58	22

Welche Ergebnisse liefern die folgenden Abfragen:

- a) `select b, a  
from r1  
where c in (select c  
from r2  
where d < (select max(d)  
from r2));`
- b) `select max(c)  
from r1  
where b > (select min(c)  
from r2  
where d > 20);`
- c) `select avg(c)  
from r1;`

Erstellen Sie bitte SQL-Statements für folgende Aufgaben:

- d) Erstellen Sie eine Abfrage, die alle unterschiedlichen Tupel der Relation r2 ausgibt, bei denen gleichzeitig der Wert des Attributs d um mehr als 10 größer ist als der eines anderen Tupels der Relation r2 und bei denen der Wert des Attributs a um mehr als 10 größer ist als der eines anderen Tupels der Relation r2. Geben Sie zusätzlich das Ergebnis der Abfrage an.
- e) Erstellen Sie eine Abfrage, die das Attribut b aller Tupel der Relation r1 ausgibt, deren Werte in jeder Spalte größer sind als der Durchschnitt dieser Komponenten über alle Tupel der Relation r1. Geben Sie zusätzlich das Ergebnis der Abfrage an.

**(3 + 2 + 1 + 5 + 4 = 15 Punkte)**

## **FORTSETZUNG AUFGABE 3**

## AUFGABE 4

Gegeben sei die folgende Datenbasis: Ein Pizzeria-Großhandel hat Lieferverträge mit Pizzerien, an die er jede Woche dieselben Produkte ausliefert. Es gilt hierbei, dass es keine zwei Pizzerien gleichen Namens gibt, daher ist Name der Schlüssel der Relation Kunde.

### Kunde

<u>Name</u>	<u>Ort</u>	<u>Strasse</u>
Bella Italia	Göttingen	Weender Weg
Casino Royale	Kassel	Karlsplatz
Da MaflA	Göttingen	Lotzenstr.
Venezia	Kassel	Europaplatz

### Preisliste

<u>Produkt</u>	<u>Preis</u>
Pizza	5,00
Lasagne	6,00
Gnocchi	4,50
Salat	3,00

### Liefervertrag

<u>Pizzeria</u>	<u>Produkt</u>	<u>Anzahl</u>
Bella Italia	Pizza	10
Bella Italia	Lasagne	15
Bella Italia	Salat	20
Casino Royale	Pizza	12
Casino Royale	Salat	15
Da MaflA	Gnocchi	60
Venezia	Pizza	20

- Geben Sie eine SQL-Abfrage an, die die Namen aller Pizzerien liefert, die Lasagne geliefert bekommen.
- Geben Sie eine SQL-Abfrage an, die die Menge aller Paare (Ort, Produkt) liefert, so dass dieses Produkt an den angegebenen Ort geliefert wird.
- Geben Sie eine SQL-Abfrage an, die für jede Stadt angibt, wie viele Portionen insgesamt in diese Stadt geliefert werden.
- Geben Sie eine SQL-Abfrage an, die alle Namen und Orte der Pizzerien liefert, die keinen Salat geliefert bekommen.
- Geben Sie in natürlicher Sprache an, was die folgende Abfrage ergibt:  

```
select ort, sum(preisliste.preis * liefervertrag.anzahl)
from kunde, preisliste, liefervertrag
where kunde.name = liefervertrag.pizzeria
      and preisliste.produkt = liefervertrag.produkt
group by ort;
```
- Geben Sie eine Zeile des Ergebnisses der obigen Abfrage aus e) an.

(2 + 3 + 4 + 4 + 2 + 1 = 16 Punkte)



## **FORTSETZUNG AUFGABE 4**

## AUFGABE 5

Es gilt die gleiche Datenbasis Pizzeria Großhandel, wie in Aufgabe 5.

- a) Geben Sie in Relationenalgebra eine Abfrage an, die die Menge der Namen aller Pizzerien liefert, die Lasagne oder Gnocchi geliefert bekommen.
- b) Geben Sie in Relationenalgebra eine Abfrage an, die alle Paare (Ort, Produkt) ergibt, so dass dieses Produkt an den angegebenen Ort geliefert wird.
- c) Geben Sie in Relationenalgebra eine Abfrage an, die die Namen aller Kunden angibt, deren Lieferung alle angebotenen Produkte enthält.

**(3 + 2 + 4 = 9 Punkte)**

## NOTIZEN