



## Modul Datenbanken I — Klausur —

**Name:** \_\_\_\_\_

**Studiengang:** \_\_\_\_\_

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

Fühlen Sie sich gesundheitlich in der Lage die Prüfung abzulegen?  Ja,  Nein

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

(Diese Daten werden zur Ausstellung des Leistungsnachweises benötigt.)

Aufgabe	Punkte	Max. Punkte	Zeit
1 (Logik)		3	10 min
2 (Relationale Algebra)		2	5 min
3 (SQL-Anfragen)		18	60 min
4 (ER-Entwurf)		4	15 min
5 (Relationales Modell)		4	15 min
6 (Normalformen)		3	10 min
7 (Serialisierbarkeit)		2	5 min
Summe		36	120 min

### Hinweise:

- Bearbeitungsdauer: 120 Minuten
- Skript, Bücher, Notizen sind erlaubt.  
Notebooks, PDAs, Handys etc. dürfen nicht verwendet werden.
- Die Klausur hat 16 Seiten. Bitte prüfen Sie die Vollständigkeit.
- Verwenden Sie weder Rot- noch Bleistift zum Bearbeiten der Aufgaben.
- Bitte benutzen Sie den vorgegebenen Platz. Wenn Sie auf die Rückseite ausweichen müssen, markieren Sie klar, dass es eine Fortsetzung gibt.

## Beispiel-Datenbank

Die folgende Datenbank dient zur Verwaltung einer Sammlung von Rezepten für Cocktails und deren Zutaten, zu der verschiedene Personen beigetragen haben. Das Schema der Datenbank ist

- REZEPT(RID, Cocktail, EingebenVon)
- ZUTAT(ZID, ZName, Woher, Alkoholgehalt, Brennwert)
- MIX(RID→ REZEPT, ZID→ ZUTAT, Quantitaet)

Die Tabellen sind mit den folgenden Inhalten gefüllt. Die Inhalte dienen als Beispiele um die Beziehungen zwischen den Tabellen zu illustrieren.

- Die Tabelle **REZEPT** beschreibt die bisher gesammelten Rezepte. Die **RID** identifiziert ein Mixgetränk. **Cocktail** bezeichnet den Namen des Getränks. Der Name ist nicht eindeutig, weil es mehrere Varianten des Getränks unter dem gleichen Namen geben kann. **EingegebenVon** bezeichnet die Person, die das Rezept zur Sammlung beigetragen hat. Beliebige Personen können Rezepte für Cocktails beitragen.

REZEPT		
<u>RID</u>	Cocktail	EingegebenVon
1	Daiquiri	Otto
2	Brooklyn Lamp	Bene
3	Mai Tai	Otto
4	Brooklyn Lamp	Bene
5	Brooklyn Lamp	Otto
6	Drivers Glow	Alfred

- In **ZUTAT** sind Fakten über die verwendeten Komponenten gespeichert. Die **ZID** ist Primärschlüssel. **ZName** beschreibt die Zutat, aber der Name ist kein Schlüssel. **Woher** gibt an, wo die Zutat hergestellt wird. **Alkoholgehalt** gibt die Menge Alkohol in Gramm pro 100 ml an. **Brennwert** gibt die Kilokalorien (kcal) pro 100 ml an.

ZUTAT				
<u>ZID</u>	ZName	Woher	Alkoholgehalt	Brennwert
1	Obstgeist	Deutschland	28	205
2	Kuba-Rum	Kuba	32	260
3	Limettensaft	Spanien	0	100
4	Obstgeist	Deutschland	32	240
5	Grapefruit Bitter	Deutschland	42	290
6	Zitronensaft	Spanien	0	163
7	Jamaika-Rum	Jamaika	48	334
8	Martinique-Rum	Martinique	54	312
9	Curaçao	Venezuela	16	179
10	Mandelsirup	Italien	25	234
11	Grapefruitsaft	Spanien	0	163
12	Brombeersirup	Deutschland	0	80

- In **MIX** wird die Mischung der Zutaten für eine bestimmte Variante eines Cocktails beschrieben. Die Kombination aus **RID** und **ZID** ist Primärschlüssel. Die **Quantität** beschreibt, wieviel man von einer Zutat für ein spezielles Getränk braucht. Die

Quantitäten sind in Milliliter (ml).

MIX		
<u>RID</u>	<u>ZID</u>	Quantitaet
1	2	60
1	3	30
2	4	40
2	5	10
2	6	20
3	7	30
3	8	30
3	9	15
3	10	8
3	3	20
4	4	40
4	5	10
4	3	15
5	4	40
5	5	10
5	3	10
5	6	10
6	11	150
6	12	30

**Aufgabe 1 (Logik)****3 Punkte**

Die Informationen über die Cocktails können durch folgende Signatur im Tupelkalkül beschrieben werden. Gehen Sie davon aus, dass alle Primär- und Fremdschlüssel aus den Tabellen auf Seite 2 und 3 auch hier erfüllt sind.

- Sorten  $S = \{\text{int, string, rezept, zutat, mix}\}$
  - Funktionen der Sorte **rezept**:
    - $\text{rid}(\text{rezept}): \text{int}$
    - $\text{cocktail}(\text{rezept}): \text{string}$
    - $\text{eingegebenvon}(\text{rezept}): \text{string}$
  - Funktionen der Sorte **zutat**:
    - $\text{zid}(\text{zutat}): \text{int}$
    - $\text{zname}(\text{zutat}): \text{string}$
    - $\text{woher}(\text{zutat}): \text{string}$
    - $\text{alkoholgehalt}(\text{zutat}): \text{int}$
    - $\text{brennwert}(\text{zutat}): \text{int}$
  - Funktionen der Sorte **mix**:
    - $\text{rid}(\text{mix}): \text{int}$
    - $\text{zid}(\text{mix}): \text{int}$
    - $\text{quantitaet}(\text{mix}): \text{int}$
  - Für die Sorten **int** und **string** gibt es die üblichen Datentyp-Funktionen/Prädikate.
- a) Formulieren Sie folgende Integritätsbedingung als logische Formel: Unterschiedliche Cocktails mit gleichem Namen müssen sich in mindestens einer Zutat unterscheiden, d.h. sie dürfen nicht nur Quantitäten von Zutaten variieren.  $\forall \text{rezept}X, \text{rezept}Y : X.\text{name} = Y.\text{name} \wedge X.\text{rid} \neq Y.\text{rid} \rightarrow (\exists \text{mix}A : A.\text{rid} = X.\text{rid} \wedge \neg \exists \text{mix}B : B.\text{rid} = Y.\text{rid} \wedge A.\text{zid} = B.\text{zid}) \vee (\exists \text{mix}A : A.\text{rid} = Y.\text{rid} \wedge \neg \exists \text{mix}B : B.\text{rid} = X.\text{rid} \wedge A.\text{zid} = B.\text{zid})$
- b) Formulieren Sie folgende Anfrage in der Notation der Logik (Tupel-Kalkül): Geben Sie alle Namen von Getränken aus, zu denen mindestens zwei Version von der gleichen Person eingetragen wurden.

Ergebnis
Cocktail
Brooklyn Lamp

$$\{X.\text{Cocktail}[\text{rezept}X] \mid \exists \text{rezept}Y : X.\text{rid} \neq Y.\text{rid} \wedge X.\text{Cocktail} = Y.\text{Cocktail} \wedge X.\text{EingebenVon} = Y.\text{EingebenVon}\}$$

**Platz zur Lösung von Aufgabe 1**

**Aufgabe 2 (Relationale Algebra)****2 Punkte**

Geben Sie eine Anfrage in relationaler Algebra für die folgende Aufgabe an. Die Primär- und Fremdschlüsselbedingungen aus dem relationalen Schema gelten ebenfalls.

Welche Cocktails enthalten alkoholfreie Zutaten aus Deutschland oder Spanien?

Ergebnis
Cocktail
Daiquiri
Brooklyn Lamp
Mai Tai
Drivers Glow

$\pi_{\text{Cocktail}}(\sigma_{\text{alkoholgehalt}=0 \wedge (\text{woher}='Deutschland' \vee \text{woher}='Spanien')}(Rezept \bowtie Mix \bowtie Zutat))$

**Aufgabe 3 (SQL-Anfragen)****18 Punkte**

Formulieren Sie die folgenden Anfragen in SQL. Sie bekommen drei Punkte für jede korrekte Anfrage. Natürlich sollen Ihre Anfragen nicht nur mit den obigen Beispiel-Daten funktionieren, sondern für beliebige Tabelleninhalte. Beachten Sie, daß auch für unnötige Komplikationen Punkte abgezogen werden können. Die Anfragen sollen keine Duplikate liefern, aber für ein unnötiges `DISTINCT` oder ein unnötiges `UNION` statt `UNION ALL` werden auch Punkte abgezogen. Um Missverständnisse zu vermeiden, sind bei Aufgabe 3 jeweils die Ergebnisse der gesuchten Anfragen im Beispiel-Zustand angegeben. Falls nicht ausdrücklich eine bestimmte Spaltenüberschrift verlangt ist, müssen die Spaltenüberschriften Ihrer Anfrage nicht unbedingt mit dem Beispiel übereinstimmen.

- a) Welche Cocktails (RID und Cocktail) enthalten eine beliebige Sorte Rum, d.h. Rum wird im Namen der Zutat erwähnt?

Ergebnis	
RID	COCKTAIL
1	Daiquiri
3	Mai Tai

```
select distinct Rid, Cocktail
from Rezept join MIX using (rid) join zutat using (zid)
where lower(ZName) like '%rum%'
```

- b) Geben Sie alle Cocktails (RID und Cocktail) von Otto aus, für die Zutaten aus Kuba und Spanien benötigt werden.

Ergebnis	
RID	COCKTAIL
1	Daiquiri

```
select r.Rid, r.Cocktail
from Rezept r, Mix m1, Zutat z1, Mix m2, Zutat z2
where r.rid=m1.rid and r.rid=m2.rid and
      m1.zid =z1.zid and m2.zid =z2.zid and
      z1.woher = 'Kuba' and z2.woher='Spanien' and
      r.EingegebenVon='Otto'
```

- c) Geben Sie alle Personen aus, die Getränke eingetragen haben, welche keinen Alkohol enthalten.

Ergebnis
EingegebenVon
Alfred

```
select distinct EingegebenVon
from Rezept r
where rid not in
(
select rid
from mix join zutat using(zid)
where alkoholgehalt>0
)
```

- d) Geben Sie für alle Cocktails (RID und Cocktail) den Gesamtbrennwert pro 100 ml in einer neuen Spalte **Gesamtbrennwert** an. Beachten Sie, dass der Brennwert in der Zutaten-Tabelle pro 100 ml angegeben ist, aber nicht unbedingt 100 ml von einer Zutat in einem Cocktail verwendet werden. Außerdem ist der Gesamtbrennwert pro 100 ml des Cocktails gefragt, nicht einfach nur der Gesamtbrennwert des Cocktails. Dies ist ein Unterschied, weil ein Cocktail nicht unbedingt 100 ml Gesamtflüssigkeit hat.

Ergebnis		
RID	COCKTAIL	GESAMTBRENNWERT
3	Mai Tai	251.81
6	Drivers Glow	149.16
2	Brooklyn Lamp	225.14
4	Brooklyn Lamp	215.38
5	Brooklyn Lamp	216.14
1	Daiquiri	206.66

```
select r.rid, r.cocktail,
sum(z.brennwert/100*m.quantitaet) /
sum(m.quantitaet) *100 as Gesamtbrennwert
from rezept r, zutat z, mix m
where
```



---

```
r.rid=m.rid and z.zid=m.zid  
group by r.rid, r.cocktail
```

- e) Geben Sie für jeden Cocktail, von dem es mehrere Varianten mit gleichem Namen gibt, die minimale und die maximale Anzahl an Zutaten aus.

Ergebnis		
COCKTAIL	MIN(ANZAHLZUTATEN)	MAX(ANZAHLZUTATEN)
Brooklyn Lamp	3	4

```
select cocktail, min(anzahlZutaten), max(anzahlZutaten)
from
(
select rid, cocktail, count(distinct zid) as anzahlZutaten
from rezept natural join mix
group by rid, cocktail
)
group by cocktail
having count(distinct rid)>1
```

- f) Geben Sie für alle Zutaten (ZID, ZName) in einer neuen Spalte Verwendbarkeit aus, ob sie sich vielseitig verwenden lassen oder eine Spezialzutat sind. Eine Zutat ist vielseitig verwendbar, wenn sie in mindestens drei Cocktails mit verschiedenem Namen vorkommt. Ansonsten ist sie eine Spezialzutat. Ordnen Sie die Ausgabe alphabetisch nach dem Namen der Zutat.

Ergebnis		
ZID	ZNAME	VERWENDBARKEIT
12	Brombeersirup	Spezialzutat
9	Curaçao	Spezialzutat
5	Grapefruit Bitter	Spezialzutat
11	Grapefruitsaft	Spezialzutat
7	Jamaika-Rum	Spezialzutat
2	Kuba-Rum	Spezialzutat
3	Limettensaft	Vielseitig
10	Mandelsirup	Spezialzutat
8	Martinique-Rum	Spezialzutat
4	Obstgeist	Spezialzutat
6	Zitronensaft	Spezialzutat

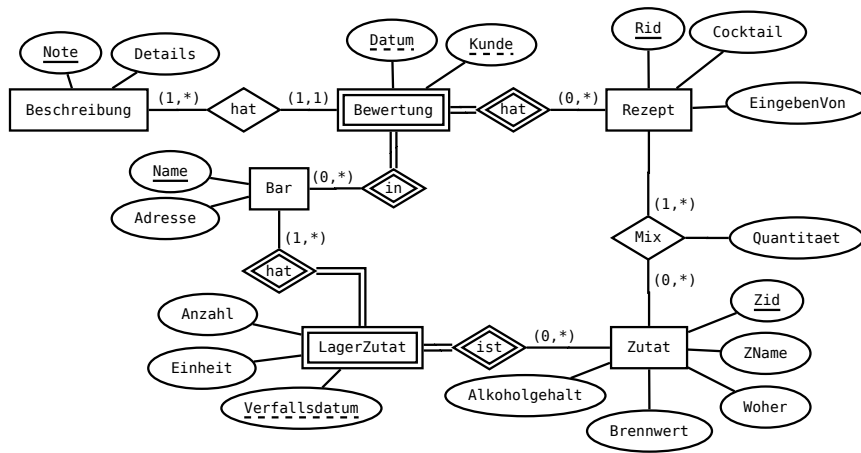
```

select z.zid, z.zname, 'Vielseitig' as Verwendbarkeit
from zutat z, mix m, rezept r
where z.zid=m.zid and r.rid=m.rid
group by z.zid, z.zname
having count(distinct Cocktail)>=3
union all
select z.zid, z.zname, 'Spezialzutat' as Verwendbarkeit
from zutat z, mix m, rezept r
where z.zid=m.zid and r.rid=m.rid
group by z.zid, z.zname
having count(distinct Cocktail)<3
order by Zname;

```



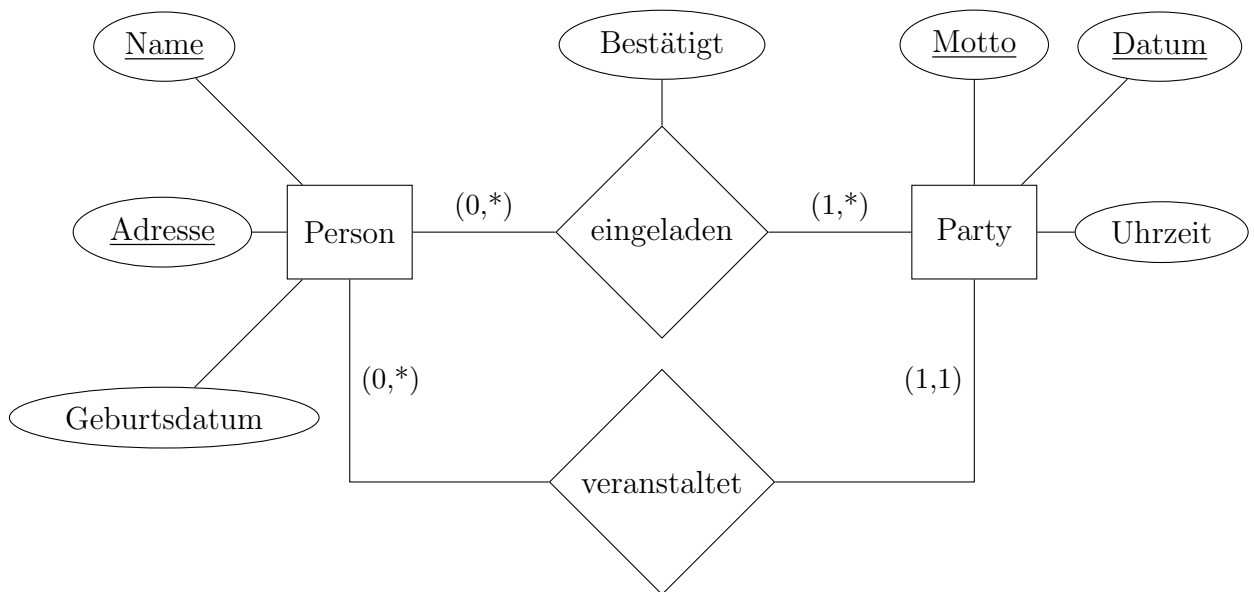
Lösung



**Aufgabe 5 (Relationales Modell)****4 Punkte**

Das folgende ER-Diagramm zeigt eine Datenbank für Einladungen zu Parties.

Ergänzen Sie die Übersetzung des gegebenen ER-Schemas ins Relationale Modell. Geben Sie das relationale Schema in Kurznotation (Tabelle, Attribute, Primär- und Fremdschlüssel) an. Falls es eine Spezifikation aus dem gegebenen ER-Schema gibt, die das relationale Schema nicht garantiert, geben Sie bitte explizit an, was noch geprüft werden müsste (natürliche Sprache reicht).

*Lösung*

- $Person(\underline{Name}, \underline{Adresse}, \underline{Geburtsdatum})$
- $Party(\underline{Motto}, \underline{Datum}, \underline{Uhrzeit}, (\underline{Name}, \underline{Adresse}) \rightarrow Person)$
- $Eingeladen((\underline{Name}, \underline{Adresse}) \rightarrow Person, (\underline{Motto}, \underline{Datum}) \rightarrow Party, \underline{Bestätigt})$

*Zusatzbedingungen*

- *Zu einer Party muss mindestens eine Person eingeladen sein.*

**Aufgabe 6 (Normalformen)****3 Punkte**

Gegeben ist eine Relation  $R(A, B, C, D)$  mit den funktionalen Abhängigkeiten

- $A, B \rightarrow D$
- $C \rightarrow B, D$
- $A \rightarrow C$

- a) Bestimmen Sie einen minimalen Schlüssel. *Lösung:  $\{A\}$*
- b) Ist die Relation in BCNF? Wenn ja, begründen Sie ihre Antwort. Wenn nein, begründen Sie ihre Antwort und spalten Sie die Relation auf, so dass die neuen Relationen in BCNF sind. Geben Sie Primär- und Fremdschlüssel der neuen Relationen mit an. *Lösung:  $R'(\underline{C}, B, D), R''(\underline{A}, C \rightarrow R')$*

**Aufgabe 7 (Serialisierbarkeit)****2 Punkte**

Gegeben sei folgender Schedule:

T1	T2	T3
	read(B)	read(B)
read(C)		
	write(A)	
read(A)	read(D)	
	write(D)	read(A)
write(C)		
	write(B)	write(D)
	read(B)	write(B)
commit	commit	
		commit

- a) Geben Sie den Konfliktgraphen für den Schedule an.  $T2 \rightarrow T3; T3 \rightarrow T2; T2 \rightarrow T1$ ;
- b) Ist der Schedule serialisierbar? *Nein.*