

**Klausur Einführung in Datenbanken (mit Systemanalyse)
im WS 2013/14**

Musterlösung

Prüfen Sie bitte zuerst, ob sie die für Sie richtige Klausur vorliegen haben.

Beachten Sie bitte auch, dass die Verwendung unerlaubter Hilfsmittel einen Täuschungsversuch darstellt, der entsprechend geahndet wird.

Studiengänge: B_BWL 10.0, 10.1, 10.5; B_Wing 4.0, 11.0

Bearbeitungszeit: 60 Minuten von 120 Minuten

Erlaubte Hilfsmittel: Blatt mit Beispieldatenbank *Firma* auf der letzte Seite darf abgetrennt werden.

Als Schmierpapier stehen Ihnen die Rückseiten zur Verfügung. Die Rückseiten werden **nicht** bewertet In der Regel stehen einige Zeilen / Spalten / Tableau mehr zur Verfügung als benötigt.

Jede Teilaufgabe wird selbständig bewertet. Aufgabenlösungen werden nur korrigiert und gewertet, wenn der Rechen- bzw. Lösungsweg nachvollziehbar ist. Denken Sie an Kurzkomentare oder Kurzbegründungen innerhalb Ihrer Lösungswege! Die Zeitangaben sind nur zur Groborientierung geeignet.

Viel Erfolg!

Aufgabe 1: Definitionen und Begriffe (5 Minuten)

Kreuzen Sie bitte die richtigen Lösungen an:

a) Welche Bestandteile einer Tabelle gehören zum zeitinvarianten Teil einer Tabelle?

- | | |
|--|--|
| <input type="checkbox"/> Zeilen (Tupel) | <input checked="" type="checkbox"/> Tabellenname |
| <input checked="" type="checkbox"/> Spaltenüberschriften | <input type="checkbox"/> Anzahl der Zeilen |

b) In SQL-Ausdrücken ist $s \text{ IN } m$ gleichbedeutend mit

- | | |
|---|---|
| <input checked="" type="checkbox"/> $s = \text{ANY } m$ | <input type="checkbox"/> $s = \text{ALL } m$ |
| <input type="checkbox"/> $s <> \text{ANY } m$ | <input type="checkbox"/> $s <> \text{ALL } m$ |

c) Welches sind die Bestandteile eines Datenbanksystems?

- | | |
|---|---------------------------------------|
| <input checked="" type="checkbox"/> Datenbank | <input type="checkbox"/> Modell |
| <input checked="" type="checkbox"/> Datenbankmanagementsystem | <input type="checkbox"/> SQL-Anfragen |

d) Grundelemente von ER-Diagrammen sind

- | | |
|---|--|
| <input checked="" type="checkbox"/> Ovale für Attribute | <input checked="" type="checkbox"/> Rauten für Beziehungen |
| <input type="checkbox"/> Herzen für Beziehungen | <input type="checkbox"/> Rechtecke für Attribute |

e) Welches sind Standard-Operationen über einzelnen Relationen?

- | | |
|--|---|
| <input type="checkbox"/> Konsternierung | <input checked="" type="checkbox"/> Selektion |
| <input checked="" type="checkbox"/> Projektion | <input type="checkbox"/> Verdichtung |

f) Welches sind Teilsprachen von SQL — Structured Query Language?

- | | |
|--|--|
| <input checked="" type="checkbox"/> DDL (Data Definition Language) | <input checked="" type="checkbox"/> DML (Data Manipulation Language) |
| <input type="checkbox"/> DAL (Data Access Language) | <input type="checkbox"/> DEL (Data Extraction Language) |

Aufgabe 2: SQL (30 Minuten)

Wir betrachten die in der Vorlesung behandelte Datenbank *Firma* mit den Tabellen *Maschine*, *Personal*, *Gehalt*, *Kind*, *Abteilung* und *Prämie*. Beispieltabellen, aus denen sich auch das Datenbankschema ablesen lässt, finden sich auf der letzten Seite dieser Aufgabenstellungen. Sie dürfen dieses Blatt gerne abtrennen.

Schreiben Sie bitte SQL-Anweisungen, um die folgenden „Fragen“ zu beantworten. Wo gefragt, geben Sie bitte auch an, welche Antworten das Datenbanksystem auf Ihre Anfrage hin basierend auf den Beispieltabellen geben würde.

- a) Wer sind die Mitarbeiter der Firma? Geben Sie bitte die Personalnummer, den Vor- und den Nachnamen der Mitarbeiter aus.

Lösung:

```
SELECT pnr, vorname, name
FROM personal
```

- b) Welches sind die Mitarbeiter der Firma, die in der *Barmer Ersatzkasse* (bek) versichert sind? Bitte geben sie wiederum die Personalnummer, den Vor- und den Nachnamen aus.

Lösung:

```
SELECT pnr, vorname, name
FROM personal
WHERE krankenkasse='bek'
```

Welche konkrete Antwort liefert diese Anfrage?

Lösung:

```
+-----+-----+-----+
| pnr | vorname | name   |
+-----+-----+-----+
| 168 | Egon    | Hahn   |
| 156 | Juergen | Hartmann |
+-----+-----+-----+
```

- c) Wie hoch ist das jeweilige Monatsgehalt der Mitarbeiter, die in den Abteilungen *d12* und *d15* arbeiten? Geben Sie bitte den monatlichen Betrag unter der Spaltenüberschrift **Monatsgehalt**, die Personalnummer, sowie Vorname und Nachname an. Sortieren Sie bitte das Ergebnis absteigend nach Höhe des Gehalts, bei gleichen Gehältern aufsteigend nach Nachname und Vorname des Mitarbeiters. Verwenden Sie bitte den IN-Operator.

Lösung:

```
SELECT betrag "Monatsgehalt", pnr, vorname, name, abt_nr
FROM personal join gehalt using (geh_stufe)
WHERE abt_nr in ('d12', 'd15')
ORDER BY betrag desc, name asc, vorname asc;
```

Welche konkrete Antwort liefert diese Anfrage?

Lösung:

Monatsgehalt	pnr	vorname	name	abt_nr
3027	167	Gustav	Krause	d12
2873	159	Petra	Osswald	d15
2523	127	Siegfried	Ehlert	d15

- d) Unter der Überschrift **Gesamtsumme** soll der Betrag ausgegeben werden, den unsere Firma insgesamt an Gehalt auszahlen muss.

Lösung:

```
SELECT sum(betrag) "Gesamtsumme"
FROM personal p, gehalt g
WHERE p.geh_stufe=g.geh_stufe;
```

- e) Für wie viele Maschinen sind die Mitarbeiter verantwortlich?
Geben Sie bitte *für jeden* Mitarbeiter (Personalnummer, Vorname, Nachname) unter der Überschrift **Anz-Maschinen** an, für wie viele Maschinen er oder sie verantwortlich ist. Für Mitarbeiter, die für keine Maschine verantwortlich sind, soll 0 ausgegeben werden. Sortierung aufsteigend nach Anzahl der Maschinen-Verantwortlichkeiten.

Lösung:

```
SELECT pnr, vorname, personal.name, count(mnr) "Anz-Maschinen"
FROM personal left outer join maschine using(pnr)
GROUP BY personal.pnr
ORDER BY 4;
```

- f) Benutzen Sie bitte Unterabfragen und vermeiden Sie Joins: Welche Mitarbeiter (Personalnummer, Vorname, Nachname) haben die höchste Einzelprämie erhalten?

Lösung:

```
SELECT pnr, vorname, name
FROM personal
WHERE pnr in
(SELECT pnr
FROM praemie
WHERE p_betrag >=ALL (SELECT p_betrag FROM praemie));
```

Welche konkrete Antwort liefert diese Anfrage?

Lösung:

```
+-----+-----+-----+
| pnr | vorname | name |
+-----+-----+-----+
| 168 | Egon    | Hahn |
+-----+-----+-----+
```

Sind Ihre Unterabfragen korreliert oder unkorreliert? Bitte begründen Sie Ihre Antwort.

Lösung:

Beide Unterabfragen sind hier unkorreliert, da sie keine Tabellen der äußeren Abfragen ansprechen.

- g) Welche Mitarbeiter (Personalnummer, Vorname, Nachname) haben Prämien erhalten? Jeder Mitarbeiter soll nur einmal genannt werden, selbst wenn er mehrere Prämien erhalten hat.

Lösung:

```
SELECT DISTINCT pnr, vorname, name
FROM personal NATURAL JOIN praemie
```

oder

```
SELECT pnr, vorname, name
FROM personal NATURAL JOIN praemie
GROUP BY pnr, vorname, name
```

Welche konkrete Antwort liefert diese Anfrage?

Lösung:

pnr	vorname	name
124	Richard	Meier
127	Siegfried	Ehlert
168	Egon	Hahn
227	Walter	Wagner
234	August	Krohn

- h) Tragen Sie bitte die Antwort auf die folgenden Anfrage in die Tabelle ein:

```
SELECT DISTINCT pnr, vorname, name
FROM personal JOIN kind USING (pnr)
WHERE abt_nr='d13';
```

Lösung:

pnr	vorname	name
123	Karl	Lehmann
133	Harry	Schulz
124	Richard	Meier

Welche Frage wird mit der oberen Anfrage beantwortet?

Lösung:

Welche Mitarbeiter der Abteilung *d13* haben Kinder?

Aufgabe 3: Datenbankentwurf (25 Minuten)

Ein Bootsverleih möchte sein Verleihgeschäft automatisieren und eine Datenbank für die Organisation einsetzen.

Es sollen u. a. Informationen über **Boote** verwaltet werden. Dazu soll der Name des Bootes, sein Baujahr, die Anzahl der Besatzungsmitglieder (**Besatzung**) gespeichert werden. Keine zwei Boote haben den selben Namen.

Kunden können Boote zu einem bestimmten Datum (**Leihbeginn**) für eine bestimmte Dauer leihen. Ein Kunde kann mehrere Boote leihen (etwa stellvertretend für eine Ausflugsgruppe). Ein Boot kann zu einer Zeit aber nur von einem einzigen Kunden geliehen werden, oder es ist nicht ausgeliehen.

Von Kunden sollen ihr Vorname, ihr Nachname, ihre Postadresse (**PLZ**, **Ort**, **Straße**, **Nr**) und ihre E-Mail-Adresse sowie ihre Telefonnummer festgehalten werden. Jeder Kunde erhält zudem eine Kundennummer.

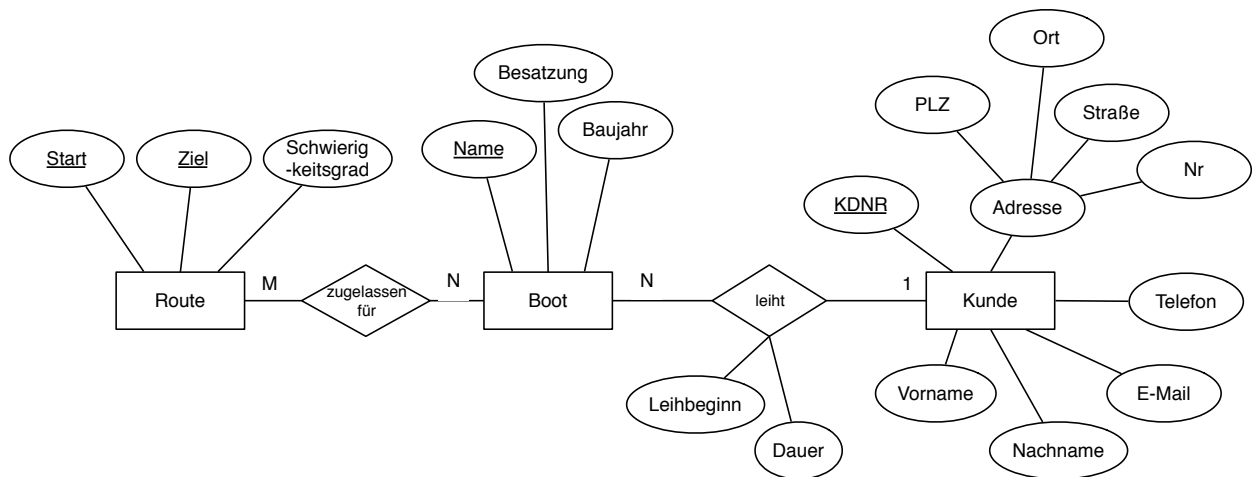
Boote sind nur für bestimmte Routen zugelassen, da etwa Tretboote nicht für Wildwasserfahrten geeignet sind. Ein Boot ist für mehrere Routen zugelassen und auch auf einer Route dürfen viele Boote eingesetzt werden.

Routen werden durch die Ortsbezeichnung für ihren Start und für ihre Ende identifiziert. Zudem wird ihr Schwierigkeitsgrad (leicht, mittel, schwer) festgehalten.

a) Entity-Relationship-Diagramm

Erstellen Sie bitte ein Entity-Relationship-Diagramm, das die oben skizzierten Sachverhalte wiedergibt. Charakterisieren Sie dabei bitte insbesondere die Beziehung zwischen Booten, Kunden und Routen genau.

Lösung:



b) Entity-Relationship-Modell

Leiten Sie aus dem ER-Diagramm bitte ein Entity-Relationship-Modell ab und geben Sie bitte die zugehörigen Entity- und Relationship-Deklarationen an.

Entity-Deklarationen:

Lösung:

Kunde = ({ KDNR, Adresse(PLZ, Ort, Strasse, Nr), Telefon, E-Mail, Vorname, Nachname }, { KDNR })
Boot = ({ Name, Besatzung, Baujahr }, { Name })
Route = ({ Start, Ziel, Schwierigkeitsgrad }, { Start, Ziel })

Relationship-Deklarationen:

Lösung:

leiht = ({ Boot, Kunde }, { Leihbeginn, Dauer }) Typ 1:N, PS: Name
zugelassen_für = ({ Route, Boot }, { }) Typ N:M, PS: Start, Ziel, Name

c) Relationales Modell

Transformieren Sie bitte das ER-Modell in ein relationales Modell und geben sie bitte entsprechende R-Schema-Definitionen sowie Integritätsbedingungen an.

Lösung:

Kunde = ({ KDNR, Adresse(PLZ, Ort, Strasse, Nr), Telefon, E-Mail, Vorname, Nachname }, { KDNR })
Boot = ({ Name, Besatzung, Baujahr }, { Name })
Route = ({ Start, Ziel, Schwierigkeitsgrad }, { Start, Ziel })

leiht = ({ Name, KDNR, Leihbeginn, Dauer }, { Name })
zugelassen_für = ({ Start, Ziel, Name }, { Start, Ziel, Name })

leiht[Name] \subseteq Boot[Name]
leiht[KDNR] \subseteq Kunde[KDNR]

zugelassen_für[Start, Ziel] \subseteq Route[Start, Ziel]
zugelassen_für[Name] \subseteq Boot[Name]

d) SQL-Datendefinitionen

Wie sieht die zugehörige Tabellendefinition (CREATE TABLE) in SQL für die Tabelle **Boot** aus?

Lösung:

```
CREATE TABLE boot (  
    name VARCHAR(30),  
    besatzung INTEGER(4),  
    baujahr INTEGER(4),  
    PRIMARY KEY (name)  
);
```

e) SQL-Anfrage

Wie sieht eine SQL-Anfrage aus, die die Boote ermittelt, die für schwere Routen zugelassen sind?

Lösung:

```
SELECT name  
FROM boot NATURAL JOIN zugelassen_fuer NATURAL JOIN route  
WHERE schwierigkeitsgrad='schwer';
```

Beispieldatenbank für Aufgabe 2. Diese Seite darf abgetrennt werden.

PERSONAL:

PNR	NAME	VOR-NAME	GEH_STUFE	ABT_NR	KRANKENKASSE
167	Krause	Gustav	it3	d12	dak
168	Hahn	Egon	it4	d11	bek
123	Lehmann	Karl	it3	d13	aok
133	Schulz	Harry	it1	d13	aok
124	Meier	Richard	it5	d13	aok
125	Wutschke	Oskar	it3	d13	aok
126	Schroeder	Karl-Heinz	it4	d13	aok
227	Wagner	Walter	it2	d13	dak
234	Krohn	August	it4	d13	aok
135	Tietze	Lutz	it2	d13	tkk
156	Hartmann	Juergen	it1	d14	bek
127	Ehlert	Siegfried	it1	d15	kkh
157	Schultze	Hans	it1	d14	aok
159	Osswald	Petra	it2	d15	dak
137	Haase	Gert	it1	d11	kkh
134	Meier	Gerd	it5	d11	tkk

GEHALT:

GEH_STUFE	BETRAG
it1	2523
it2	2873
it3	3027
it4	3341
it5	3782

ABTEILUNG:

ABT_NR	NAME
d11	Verwaltung
d12	Projektierung
d13	Produktion
d14	Lagerung
d15	Verkauf

PRAEMIE:

PNR	P_BETRAG
227	550
227	610
227	250
124	250
234	600
234	500
127	300
168	600
168	700

Kind:

PNR	K_NAME	K_VORN	K_GEB
167	Krause	Fritz	1997
167	Krause	Ida	1999
123	Lehmann	Sven	2002
123	Lehmann	Karl	2004
168	Hahn	Hans	1993
133	Wendler	Klaus	1996
124	Meier	Gustav	1999
124	Meier	Susi	2002
124	Meier	Dirk	2004

MASCHINE:

MNR	NAME	PNR	ANSCH_DATUM	NEUWERT	ZEITWERT
1	bohrmaschine	123	1995	30.000	15.000
2	bohrmaschine	123	2002	30.000	18.000
3	fräsmaschine	124	1998	40.000	10.000
11	hobelmaschine	127	2002	29.000	19.000
12	drehbank	126	1999	31.000	21.000
14	hobelmaschine	123	1998	32.000	22.000
16	drehbank	134	2001	32.000	23.000
17	bohrmaschine	127	2003	31.000	25.000