

**Klausur zur Vorlesung
Datenbanken I
im Sommersemester 2003**

Name: Vorname:.....

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Wenn Sie extra Blätter für Antworten verwenden, versehen Sie diese bitte deutlich mit Ihrem Namen. Hilfsmittel sind **nicht** zugelassen.

Aufgabe 1:

Man sagt, relationale Datenbanksysteme seien wertbasiert, d.h. Beziehungen würden über Fremdschlüssel modelliert. Deshalb enthält ein Fremdschlüsselattribut

- () den Wert eines invarianten Record-Identifizier.
- () den Primärschlüsselwert einer anderen Relation.
- () den Wert eines eindeutigen Objekt-Identifiziers (OID).
- () in der Schemadefinition des Attributs das Schlüsselwort VALUE IS.
- () nie einen Nullwert.

Aufgabe 2:

Sind Sie reif für die Insel? Dann geben Sie ein ER-Diagramm für die folgende Miniwelt der Charterverbindungen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen.

Es gibt Abflughäfen in Deutschland, kurz **Abflug**. Jeder Flughafen hat bekanntlich einen weltweit eindeutigen dreibuchstabigen Code (**ACode**) und einen Namen (**AName**), also z. B. (ACode: PAD, AName: Paderborn-Lippstadt).

Es gibt Zielflughäfen in den Feriengebieten, kurz **Ziel**, auch mit Code (**ZCode**) und Name (**ZName**), also z.B. (ZCode: PMI, ZName: Palma de Mallorca).

Drittens gibt es Fluggesellschaften, kurz **Flugline**, die bekanntlich einen zweibuchstabigen Airline-Code **FID** und einen Namen **FName** haben, also z.B. (FID: HF, FName: Hapag-Lloyd).

Zwischen diesen drei Entitätenmengen gibt es einen **ternären** Beziehungstyp **verbindet**, der selbst ein Attribut hat, nämlich den Flugtag **FTag** (Mo, Di, Mi, ...). Eine Instanz gibt dann z.B. an, daß es samstags eine Verbindung der Hapag-Lloyd von Paderborn-Lippstadt nach Rhodos gibt.

Wie zu erwarten sind in unserer Welt nur solche Flughäfen aufgeführt, die auch wirklich abgehende oder ankommende Flüge haben. Nicht jeder Abflughafen bedient jedes Ziel, aber in der Regel viele und auch Fluggesellschaften haben viele Ziele im Angebot, aber nicht jede fliegt überallhin, aber mindestens zu einem Ziel. Viele Ziele werden von mehreren Airlines angeflogen, oft mehrmals die Woche, ggf. mehrfach am selben Wochentag, was hier nicht modelliert wird.

Aufgabe 3:

Es gebe eine weitere Entitätsmenge **Land** mit Ländercode (**LCode**) und Name (**LName**), also z.B. (LCode: GR, LName: Griechenland) oder (TR, Türkei); (ES, Spanien). Tragen Sie entweder in das obige Diagramm oder getrennt in ein neues Diagramm den Entity-Typ **Land** und die Beziehung **liegt-in** zu Zielen ein. Dabei werden natürlich nur die Länder aufgeführt, die in der Miniwelt einen Zielflughafen für den Charterverkehr aus Deutschland haben, also z.B. Griechenland wegen den Flughäfen Rhodos oder Heraklion, die Türkei für Antalya und Bodrum, Spanien unter anderem für Palma de Mallorca. Die Kardinalitäten sollen wieder in (min, max)-Notation angegeben werden.

Aufgabe 4:

Welche der Beziehungen **verbindet** und/oder **liegt-in** ist **hierarchisch-existentiell**? Begründung mit einem oder zwei Sätzen!

Aufgabe 5:

Ein wohlmeinender Datenbankadministrator hat **Ziel** und **Land** zusammengefaßt zu einer Relation **Ziele** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 2, 3 und 4 widerspiegelt.

Ziele

ZCode	ZName	LCode	LName
RHO	Rhodos	GR	Griechenland
HER	Heraklion	GR	Griechenland
AYT	Antalya	TR	Türkei
BXN	Bodrum	TR	Türkei
PMI	Palma de Mallorca	ES	Spanien

- (a) Was sind die Schlüsselkandidaten?

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

- (c) Wenn die Schlüsselkandidaten nicht zusammengesetzt sind, ist eine Relation in 2. Normalform. Ist **Ziele** in 2NF?

Aufgabe 6:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial, X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Ist **Ziele** aus Frage 5 in 3NF? Begründung!

Aufgabe 7:

Um in der Relation **Ziele** ganz sicher 3NF zu erreichen, schlägt ein Freund vor, das Attribut **ZName** zu streichen. Damit wäre man sogar in BCNF. Ist **Ziele** jetzt in 3NF? Wenn ja, ist es sogar in BCNF?

Hinweis: Gegenüber 3NF entfällt bei BCNF die Alternative „... oder A ist prim.“

Aufgabe 8:

Eine Freundin sagt, das sei Quatsch, man müsse in **Ziele** entweder **LCode** oder **LName** streichen, sagen wir **LName**. Ist **Ziele2** unten in 3NF, vielleicht sogar in BCNF?

Ziele2

ZCode	ZName	LCode
RHO	Rhodos	GR
HER	Heraklion	GR
AYT	Antalya	TR
BXN	Bodrum	TR
PMI	Palma de Mallorca	ES

Begründung!

Aufgabe 9:

Die folgende, recht große Tabelle realisiert die Relation **verbindet** aus Aufgabe 2. Schlüsselattribute sind unterstrichen. Anzahl steht für die Anzahl der **Verbindungen einer Fluglinie** an diesem Tag, etwa 2, wenn die Air Berlin (AB) zweimal samstags von PAD nach PMI fliegt.

verbindet

<u>ACode</u>	<u>ZCode</u>	<u>FID</u>	<u>FTag</u>	Anzahl
PAD	RHO	HF	Sa	1
PAD	RHO	AB	Do	1
PAD	AYT	AB	Sa	1
PAD	PMI	HF	Fr	1
PAD	PMI	HF	Sa	1
PAD	PMI	AB	Sa	2
ERF	HER	HF	Mo	1
ERF	HER	AB	Mi	1
ERF	BXN	AB	Sa	1
ERF	PMI	HF	Fr	1
ERF	PMI	AB	So	1
ERF	PMI	HF	So	1

Hinweis: Die Relation ist in BCNF.

Weil die Relation **verbindet** so groß ist, teilt sie der Administrator in $R_1 = (\text{ACode}, \text{ZCode}, \text{FID})$ und $R_2 = (\text{ACode}, \text{ZCode}, \text{FTag}, \text{Anzahl})$. Bekanntlich ist aber eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$.

- (a) Ist die Aufteilung verlustfrei? Begründung mit dem formalen Argument $R_1 \cap R_2 \dots$
- (b) Wenn nein, wie zeigt sich der Verlust? Wenn ja, welcher Join bringt R zurück?

Aufgabe 10:

Was liefert generell die folgende Abfrage?

```
SELECT DISTINCT FID
FROM verbindet v
WHERE NOT EXISTS (SELECT * FROM verbindet
WHERE ZCode = 'AYT' AND FID = v.FID)
```

Aufgabe 11:

Gesucht ist eine SQL-Abfrage, die zu jeder Fluglinie FID die Anzahl der verschiedenen angefliegenen Ziele unabhängig vom Ablughafen nennt. Ergänzen Sie die fehlenden Teile!

```
SELECT FID, COUNT (_____ )
FROM verbindet
GROUP BY _____
```

Aufgabe 12:

Zwei Fluglinien machen sich direkte Konkurrenz, wenn sie vom selben Abflughafen, am selben Tag, zum selben Ziel fliegen. Eine Managerin der „HF“ sucht mit einer SQL-Abfrage in der Tabelle **verbindet** alle Tupel eigener Verbindungen, die in direkter Konkurrenz zu einer anderen Fluglinie stehen. Wie lautet die Abfrage?

Hinweis: Sie können einen Join von **verbindet** mit sich selbst (... FROM verbindet V1, verbindet V2 WHERE ...) machen.

Aufgabe 13:

Betrachten Sie die folgende Ablauffolge:

Transaktion 1

Lesen A
Schreiben A

Lesen C
Schreiben C

Transaktion 2

Lesen A

Lesen B
Schreiben A

Schreiben B

Transaktion 3

Lesen B
Schreiben B

Lesen C

Schreiben C

Ist die Ablauffolge konfliktserialisierbar? Begründung durch Abhängigkeitsgraphen!

**Musterlösung zur Klausur
Datenbanken I
im Sommersemester 2003**

Aufgabe 1:

Man sagt, relationale Datenbanksysteme seien wertbasiert, d.h. Beziehungen würden über Fremdschlüssel modelliert. Deshalb enthält ein Fremdschlüsselattribut

- den Wert eines invarianten Record-Identifizier.
- den Primärschlüsselwert einer anderen Relation.
- den Wert eines eindeutigen Objekt-Identifizierers (OID).
- in der Schemadefinition des Attributs das Schlüsselwort `VALUE IS`.
- nie einen Nullwert.

Aufgabe 2:

Sind Sie reif für die Insel? Dann geben Sie ein ER-Diagramm für die folgende Miniwelt der Charterverbindungen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen.

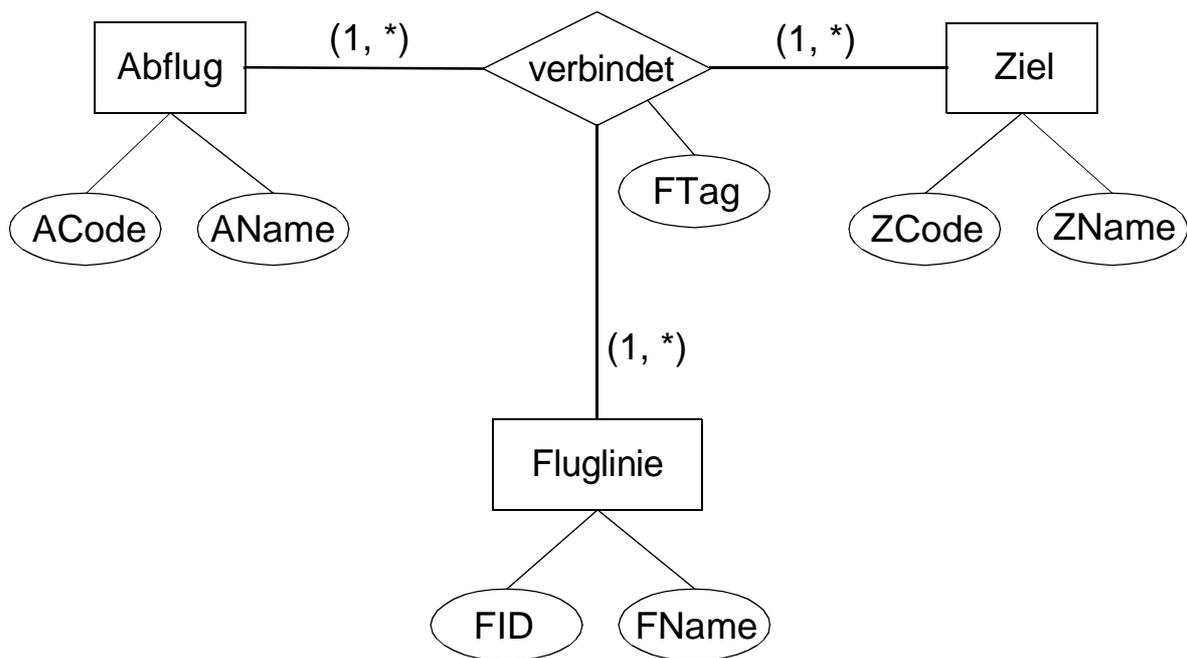
Es gibt Abflughäfen in Deutschland, kurz **Abflug**. Jeder Flughafen hat bekanntlich einen weltweit eindeutigen dreibuchstabigen Code (**ACode**) und einen Namen (**AName**), also z. B. (ACode: PAD, AName: Paderborn-Lippstadt).

Es gibt Zielflughäfen in den Feriengebieten, kurz **Ziel**, auch mit Code (**ZCode**) und Name (**ZName**), also z.B. (ZCode: PMI, ZName: Palma de Mallorca).

Drittens gibt es Fluggesellschaften, kurz **Fluglinie**, die bekanntlich einen zweibuchstabigen Airline-Code **FID** und einen Namen **FName** haben, also z.B. (FID: HF, FName: Hapag-Lloyd).

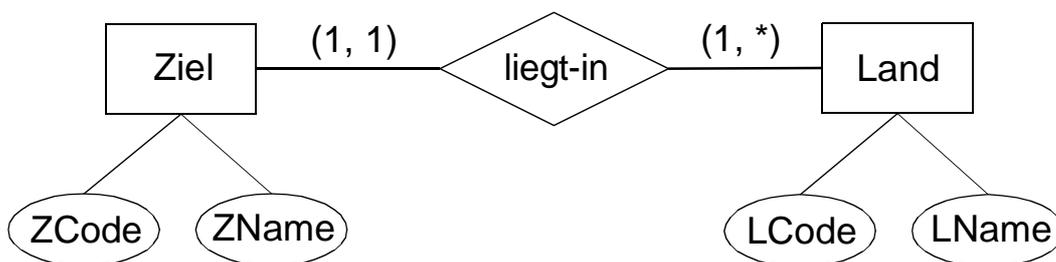
Zwischen diesen drei Entitätenmengen gibt es einen **ternären** Beziehungstyp **verbindet**, der selbst ein Attribut hat, nämlich den Flugtag **FTag** (Mo, Di, Mi, ...). Eine Instanz gibt dann z.B. an, daß es samstags eine Verbindung der Hapag-Lloyd von Paderborn-Lippstadt nach Rhodos gibt.

Wie zu erwarten sind in unserer Welt nur solche Flughäfen aufgeführt, die auch wirklich abgehende oder ankommende Flüge haben. Nicht jeder Abflughafen bedient jedes Ziel, aber in der Regel viele und auch Fluggesellschaften haben viele Ziele im Angebot, aber nicht jede fliegt überallhin, aber mindestens zu einem Ziel. Viele Ziele werden von mehreren Airlines angefliegen, oft mehrmals die Woche, ggf. mehrfach am selben Wochentag, was hier nicht modelliert wird.



Aufgabe 3:

Es gebe eine weitere Entitätsmenge **Land** mit Ländercode (**LCode**) und Name (**LName**), also z.B. (LCode: GR, LName: Griechenland) oder (TR, Türkei); (ES, Spanien). Tragen Sie entweder in das obige Diagramm oder getrennt in ein neues Diagramm den Entity-Typ **Land** und die Beziehung **liegt-in** zu Zielen ein. Dabei werden natürlich nur die Länder aufgeführt, die in der Miniwelt einen Zielflughafen für den Charterverkehr aus Deutschland haben, also z.B. Griechenland wegen den Flughäfen Rhodos oder Heraklion, die Türkei für Antalya und Bodrum, Spanien unter anderem für Palma de Mallorca. Die Kardinalitäten sollen wieder in (min, max)-Notation angegeben werden.



Aufgabe 4:

Welche der Beziehungen **verbindet** und/oder **liegt-in** ist **hierarchisch-existentiell**? Begründung mit einem oder zwei Sätzen!

- „verbindet“ ist nicht hierarchisch-existentiell.
- „liegt-in“ ist hierarchisch-existentiell, da:
 - Einem Land werden ein oder mehrere Zielflughäfen zugeordnet.
Also: hierarchisch (1:n).
 - Den Zielflughäfen muss ein Land zugeordnet werden.
Also: existentiell.

Aufgabe 5:

Ein wohlmeinender Datenbankadministrator hat **Ziel** und **Land** zusammengefaßt zu einer Relation **Ziele** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 2, 3 und 4 widerspiegelt.

Ziele

ZCode	ZName	LCode	LName
RHO	Rhodos	GR	Griechenland
HER	Heraklion	GR	Griechenland
AYT	Antalya	TR	Türkei
BXN	Bodrum	TR	Türkei
PMI	Palma de Mallorca	ES	Spanien

(a) Was sind die Schlüsselkandidaten?

{ZCode} und {ZName}

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

Die wichtigsten funktionalen Abhängigkeiten sind:

{ZCode} → {ZName, LCode, LName}

{ZName} → {ZCode, LCode, LName}

{LCode} → {LName}

{LName} → {LCode}

- (c) Wenn die Schlüsselkandidaten nicht zusammengesetzt sind, ist eine Relation in 2. Normalform. Ist **Ziele** in 2NF?

Ja, Ziele ist in 2NF.

Begründung: Alle Schlüsselkandidaten sind einelementig (siehe (a)).

Aufgabe 6:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial, X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Ist **Ziele** aus Frage 5 in 3NF? Begründung!

Nein, Ziele ist nicht in 3NF.

Begründung: Es gilt: $\{ZCode\} \rightarrow \{LCode\} \rightarrow \{LName\}$

Dies ist eine transitive Abhängigkeit des nicht-primen Attributs LName vom Schlüsselkandidaten {ZCode}.

Alternative Begründung: Es gilt: $\{LCode\} \rightarrow \{LName\}$

Diese funktionale Abhängigkeit ist nicht trivial, {LCode} ist kein Schlüssel und LName ist nicht-prim.

Aufgabe 7:

Um in der Relation **Ziele** ganz sicher 3NF zu erreichen, schlägt ein Freund vor, das Attribut **ZName** zu streichen. Damit wäre man sogar in BCNF. Ist Ziele jetzt in 3NF? Wenn ja, ist es sogar in BCNF?

Hinweis: Gegenüber 3NF entfällt bei BCNF die Alternative „... oder A ist prim.“

Ziele ist auch ohne ZName nicht in 3NF mit der gleichen Begründung wie in Aufgabe 6. Also ist Ziele auch nicht in BCNF.

Aufgabe 8:

Eine Freundin sagt, das sei Quatsch, man müsse in **Ziele** entweder **LCode** oder **LName** streichen, sagen wir **LName**. Ist **Ziele2** unten in 3NF, vielleicht sogar in BCNF?

Ziele2

ZCode	ZName	LCode
RHO	Rhodos	GR
HER	Heraklion	GR
AYT	Antalya	TR
BXN	Bodrum	TR
PMI	Palma de Mallorca	ES

Begründung!

Ziele2 ist sogar in BCNF (und damit auch in 3NF).

Begründung: Es existieren die folgenden funktionalen Abhängigkeiten:

$\{ZCode\} \rightarrow \{ZName, LCode\}$

$\{ZName\} \rightarrow \{ZCode, LCode\}$

Alle nicht-trivialen funktionalen Abhängigkeiten gehen von einem Schlüsselkandidaten aus. Somit ist Ziele2 in BCNF.

Aufgabe 9:

Die folgende, recht große Tabelle realisiert die Relation **verbindet** aus Aufgabe 2. Schlüsselattribute sind unterstrichen. Anzahl steht für die Anzahl der **Verbindungen einer Fluglinie** an diesem Tag, etwa 2, wenn die Air Berlin (AB) zweimal samstags von PAD nach PMI fliegt.

verbindet

<u>ACode</u>	<u>ZCode</u>	<u>FID</u>	<u>FTag</u>	Anzahl
PAD	RHO	HF	Sa	1
PAD	RHO	AB	Do	1
PAD	AYT	AB	Sa	1
PAD	PMI	HF	Fr	1
PAD	PMI	HF	Sa	1
PAD	PMI	AB	Sa	2
ERF	HER	HF	Mo	1
ERF	HER	AB	Mi	1
ERF	BXN	AB	Sa	1
ERF	PMI	HF	Fr	1
ERF	PMI	AB	So	1
ERF	PMI	HF	So	1

Hinweis: Die Relation ist in BCNF.

Weil die Relation **verbindet** so groß ist, teilt sie der Administrator in $R_1 = (ACode, ZCode, FID)$ und $R_2 = (ACode, ZCode, FTag, Anzahl)$. Bekanntlich ist aber eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$.

(a) Ist die Aufteilung verlustfrei? Begründung mit dem formalen Argument $R_1 \cap R_2 \dots$

Nein, die Aufteilung ist nicht verlustfrei.

Begründung: $R_1 \cap R_2 = \{ACode, ZCode\}$, aber:

- $\{ACode, ZCode\} \not\rightarrow R_1$, da verschiedene Fluglinien die gleiche Strecke fliegen können und
- $\{ACode, ZCode\} \not\rightarrow R_2$, da die gleiche Strecke an verschiedenen Tagen geflogen werden kann.

(b) Wenn nein, wie zeigt sich der Verlust? Wenn ja, welcher Join bringt R zurück?

Nach einem Natural Join von R_1 und R_2 über die Attribute ACode und ZCode zeigt sich der Verlust durch ein Zuviel an Tupeln. So würde zum Beispiel neben dem gewünschten Tupel

(PAD, RHO, HF, Sa, 1)

auch das Tupel

(PAD, RHO, HF, Do, 1)

auftreten.

Aufgabe 10:

Was liefert generell die folgende Abfrage?

```
SELECT DISTINCT FID
FROM verbindet v
WHERE NOT EXISTS (SELECT * FROM verbindet
WHERE ZCode = 'AYT' AND FID = v.FID)
```

Diese Abfrage liefert die Airline-Codes (FID) aller Fluglinien, die nicht nach Antalya (AYT) fliegen.

Aufgabe 11:

Gesucht ist eine SQL-Abfrage, die zu jeder Fluglinie FID die Anzahl der verschiedenen angefliegenen Ziele unabhängig vom Abflughafen nennt. Ergänzen Sie die fehlenden Teile!

```
SELECT FID, COUNT ( DISTINCT ZCode )
FROM verbindet
GROUP BY FID
```

Aufgabe 12:

Zwei Fluglinien machen sich direkte Konkurrenz, wenn sie vom selben Abflughafen, am selben Tag, zum selben Ziel fliegen. Eine Managerin der „HF“ sucht mit einer SQL-Abfrage in der Tabelle **verbindet** alle Tupel eigener Verbindungen, die in direkter Konkurrenz zu einer anderen Fluglinie stehen. Wie lautet die Abfrage?

Hinweis: Sie können einen Join von **verbindet** mit sich selbst (... FROM verbindet V1, verbindet V2 WHERE ...) machen.

```
SELECT DISTINCT v1.*
FROM verbindet v1, verbindet v2
WHERE v1.ACode = v2.ACode AND
v1.ZCode = v2.ZCode AND
v1.FTag = v2.FTag AND
v1.FID = "HF" AND
v2.FID <> "HF"
```

Aufgabe 13:

Betrachten Sie die folgende Ablauffolge:

Transaktion 1

Lesen A
Schreiben A

Lesen C
Schreiben C

Transaktion 2

Lesen A

Lesen B
Schreiben A

Schreiben B

Transaktion 3

Lesen B
Schreiben B

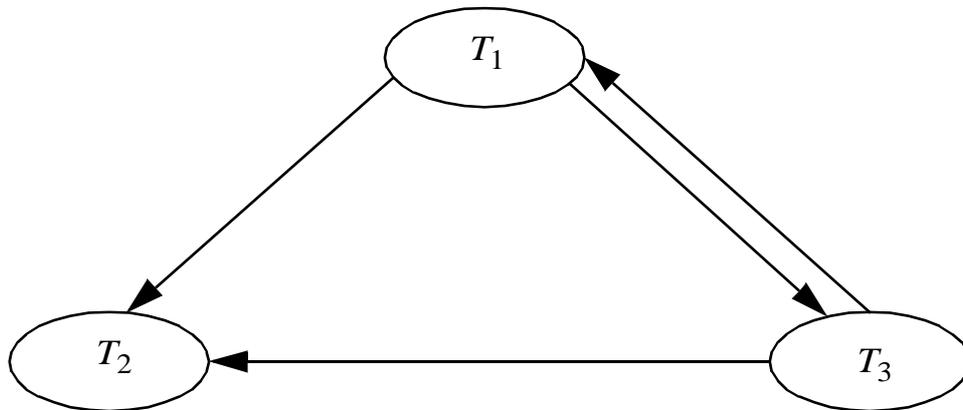
Lesen C

Schreiben C

Ist die Ablauffolge konfliktserialisierbar? Begründung durch Abhängigkeitsgraphen!

Nein, die Ablauffolge ist nicht serialisierbar.

Begründung: Der Abhängigkeitsgraph enthält einen Zyklus.



**Klausur zur Vorlesung
Datenbanken I
vom Sommersemester 2003**

Name: Vorname:.....

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Wenn Sie extra Blätter für Antworten verwenden, versehen Sie diese bitte deutlich mit Ihrem Namen. Hilfsmittel sind **nicht** zugelassen.

Nur für den Prüfer:

Punkte

1		6
2		2
3		2
4		2+2+2
5		4
6		3+2
7		3+2
8		3
9		3
10		3
11		3
12		3
Summe		45 max.

Aufgabe 1:

Wir helfen Ihnen aufs Rad! Erstellen Sie ein ER-Diagramm für die Miniwelt der Tour de France. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen.

Es gibt **Fahrer**, eindeutig gekennzeichnet sowohl durch Fahrernamen (**FName**) als auch durch ihre Rückennummer (**RNr**), die während der Tour gleich bleibt. Jeder Fahrer hat eine Nationalität (**FNAT**), abgekürzt mit GER, USA, FRA, usw.

Fahrer **gehören zu** einem **Team**, natürlich immer nur zu einem. Jedes Team besteht aus genau neun Fahrern. Teams sind durch den eindeutigen Teamnamen (**TName**) gekennzeichnet und haben auch eine Nationalität (**TNAT**).

In jedem Team erfüllt genau ein Fahrer die **Rolle des Kapitäns**, daneben gibt es noch sportliche Leiter, was wir hier nicht betrachten.

Die Tour de France besteht aus mehreren **Etappen**. Eine Etappe hat eine Nummer (**ENr**), die gewöhnlich als Schlüssel dient, einen Tag (**ETag**), der zwar auch Schlüsseleigenschaft hat, weil es an einem Tag keine zwei Etappen gibt, aber wegen unterschiedlicher Schreibweisen problematisch ist, einen **Startort** und einen **Zielort**.

Zwischen Fahrern und Etappen gibt es eine Beziehung **nimmt_teil**, die angibt, welche Fahrer an welchen Etappen teilgenommen haben. Ein Attribut **Platz** gibt den erlangten Platz beim Zieleinlauf an. Jeder Fahrer nimmt an wenigstens einer Etappe teil, auch wenn er schon bei der ersten ausfallen kann. Etappen, die wegen externer Umstände insgesamt abgesagt werden, berücksichtigen wir hier nicht, d.h. jede hat mindestens einen Teilnehmer.

Aufgabe 2:

Nennen Sie mindestens eine Beziehung aus Aufgabe 1, bei der eine Mitgliedschaft für ein Entity **optional ist**?

Aufgabe 3:

Nehmen wir an, wir hätten eine Tabelle **nimmt_teil** mit dem zusammengesetzten Schlüssel (PRIMARY KEY) Rückennummer **RNr** und Etappennummer **ENr**, sowie dem Attribut **Platz**, für den Rang, den der Fahrer auf dieser Etappe belegt. Eine Datenbank-Null darf eingetragen werden

- () für einen **Platz**-Wert, wenn er während des Rennens noch *unbekannt* ist.
- () für einen **Platz**-Wert, wenn er wegen Ausscheidens des Fahrers, z.B. nach Zeitüberschreitung, *nicht zutreffend* ist.
- () für **RNr**, wenn der Sieger noch nicht feststeht, ein Siegetupel mit Platz-Wert 1 aber schon existiert.
- () für **ENr**, wenn man mit diesem Trick das Gesamtklassement in der Tabelle abspeichern will.
- () in keiner der drei Spalten.

Aufgabe 4:

Ein wohlmeinender Datenbankadministrator hat **Fahrer** und **Team** zusammengefaßt zu einer Relation **Teilnehmer** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 1 widerspiegelt, wobei die Rückennummer entfallen ist.

Teilnehmer

FName	FNAT	TName	TNAT
Lance Armstrong	USA	US Postal	USA
Roberto Heras	ESP	US Postal	USA
Jan Ullrich	GER	Team Bianchi	GER
Jörg Jaksche	GER	O.N.C.E.	ESP
Joseba Beloki	ESP	O.N.C.E.	ESP

- (a) Was sind die Schlüsselkandidaten?
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
- (c) Wenn die Schlüsselkandidaten nicht zusammengesetzt sind, ist eine Relation in 2. Normalform. Ist **Teilnehmer** in 2NF? Begründung!

Aufgabe 5:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial, X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Ist **Teilnehmer** aus Aufgabe 4 in 3NF? Begründung!

Aufgabe 6:

In der neuen Relation **Fahrer2** fliegt die Teamnationalität raus, dafür kommt **RNr**, die Rückennummer des Fahrers, rein. Einen Ausschnitt zeigen wir unten.

Fahrer2

FName	RNr	FNAT	TName
Lance Armstrong	1	USA	US Postal
Roberto Heras	2	ESP	US Postal
Jan Ullrich	131	GER	Team Bianchi
Jörg Jaschke	15	GER	O.N.C.E.
Joseba Beloki	11	ESP	O.N.C.E.
Tyler Hamilton	71	USA	CSC

(a) Ist **Fahrer2** in 3NF? Begründung!

(b) Wenn ja, ist **Fahrer2** sogar in BCNF? Wenn nein, welches Attribut ist zu streichen?
Begründung! Hinweis: Gegenüber 3NF entfällt bei BCNF die Alternative „... oder A ist prim.“

Aufgabe 7:

Die Relation $R = \mathbf{Fahrer2}$ soll in zwei Relationen aufgeteilt werden. Zur Auswahl stehen die beiden Aufteilungen

Alternative 1: $R_1 = (\text{RNr}, \text{FName})$ und $R_2 = (\text{RNr}, \text{FNAT}, \text{TName})$, oder

Alternative 2: $R_1 = (\text{FName}, \text{TName})$ und $R_2 = (\text{RNr}, \text{FNAT}, \text{TName})$

- (a) Sind die Aufteilungen verlustfrei? Begründung mit dem formalen Argument $R_1 \cap R_2 \dots$

Hinweis: Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$.

- (b) Bei Alternative 1 kann man lokal nicht mehr die funktionale Abhängigkeit $\text{FName} \rightarrow \text{FNAT}$ überprüfen, weil sich beide Attribute nicht mehr in der selben Relation befinden. Wie kann man direkt mit einem Armstrongschen Axiom zeigen, daß die Aufteilung zumindestens bezüglich dieser Abhängigkeit trotzdem abhängigkeitsbewahrend ist.

Hinweis: Die drei Armstrongschen Axiome sind

Reflexivität: Wenn $Y \subseteq X$, dann folgt $X \rightarrow Y$, die natürlichen oder trivialen Abhängigkeiten.

Erweiterung: Wenn $X \rightarrow Y$ gilt und Z eine Teilmenge von R ist, dann gilt auch $XZ \rightarrow YZ$ (XZ ist kurz für $X \cup Z$).

Transitivität: Wenn $X \rightarrow Y$ und $Y \rightarrow Z$, dann auch $X \rightarrow Z$.

Aufgabe 8:

Die folgende Tabelle zeigt einen Ausschnitt aus der Relation **nimmt_teil**, die aus der ER-Modellierung in Aufgabe 1 abgeleitet wurde. Schlüsselattribute sind unterstrichen. Die Bemerkungen sind nur als menschliche Lesehilfe gedacht.

nimmt_teil

<u>RNr</u>	<u>ENr</u>	Platz	Bemerkung
1	15	1	Lance gewinnt die Bergankunft in Luz-Ardiden
1	0	7	Lance 7. im Prolog und verliert 5 s auf Jan
1	8	3	Lance Dritter in L'Alpe d'Huez
131	12	1	Jan gewinnt das Zeitfahren (12. Etappe)
131	0	4	Jan 4. im Prolog (0. Etappe)
131	8	13	Jan 13. in L'Alpe d'Huez
2	8	12	Roberto Heras (US Postal) vor Jan in L'Alpe d'Huez
2	9	15	Heras 15. in der 9. Etappe über den Col d'Izoard
11	8	6	Beloki 6. in L'Alpe d'Huez, zweiter im Gesamtklass.
11	9	Null	Joseba Beloki scheidet tragisch mit Sturz aus
71	0	6	Tyler Hamilton 6. im Prolog knapp vor Lance
71	16	1	Tyler Hamilton gewinnt in Bayonne (16. Etappe)

Enthält **nimmt_teil** Fremdschlüssel? Wenn ja, in welchen Relationen, die sich aus der ER-Modellierung in Aufgabe 1 ergeben, sind die zugehörigen Primärschlüssel enthalten? Wenn nein, warum tritt kein Fremdschlüssel auf?

Aufgabe 9:

Gesucht ist eine SQL-Abfrage, die aus **nimmt_teil** in Verbindung mit **Fahrer2** für alle Etappen die drei Erstplatzierten liefert. Jedes Ergebnistupel soll also die Rückennummer, den Namen, den Platz sowie die Etappennummer liefern. Sortieren Sie das Ergebnis nach Etappen.

Aufgabe 10:

Was liefert generell die folgende Abfrage?

```
SELECT FNAT, COUNT(*)
FROM nimmt_teil n, Fahrer2 F
WHERE n.RNr = F.RNr AND Platz = 1
GROUP BY FNAT
```

Aufgabe 11:

Gesucht ist eine SQL-Abfrage, die alle Teamnamen (genau einmal) der Teams in **Fahrer2** ausgibt, die keinen deutschen Fahrer haben (wo es keinen Fahrer gibt mit FNAT = 'GER').

```
SELECT _____
FROM Fahrer2 F
WHERE _____ (SELECT _____ FROM _____
WHERE _____ AND
TName = F.TName)
```

Aufgabe 12:

Gezeigt sind drei Ablauffolgen. Welche davon ist **Lost Update**, **Dirty Read**, **Unrepeatable Read** oder **OK**?

- (a) T2 Lesen A
T1 Schreiben A
T2 Schreiben A _____

- (b) T1 Lesen A
T2 Schreiben A
T1 Lesen A _____

- (c) T2 Schreiben A
T1 Lesen A
T2 Schreiben A _____

MUSTERLÖSUNG

Klausur zur Vorlesung Datenbanken I vom Sommersemester 2003

Name: Vorname:.....

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Wenn Sie extra Blätter für Antworten verwenden, versehen Sie diese bitte deutlich mit Ihrem Namen. Hilfsmittel sind **nicht** zugelassen.

Nur für den Prüfer:

Punkte

1		6
2		2
3		2
4		2+2+2
5		4
6		3+2
7		3+2
8		3
9		3
10		3
11		3
12		3
Summe		45 max.

Aufgabe 1:

Wir helfen Ihnen aufs Rad! Erstellen Sie ein ER-Diagramm für die Miniwelt der Tour de France. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen.

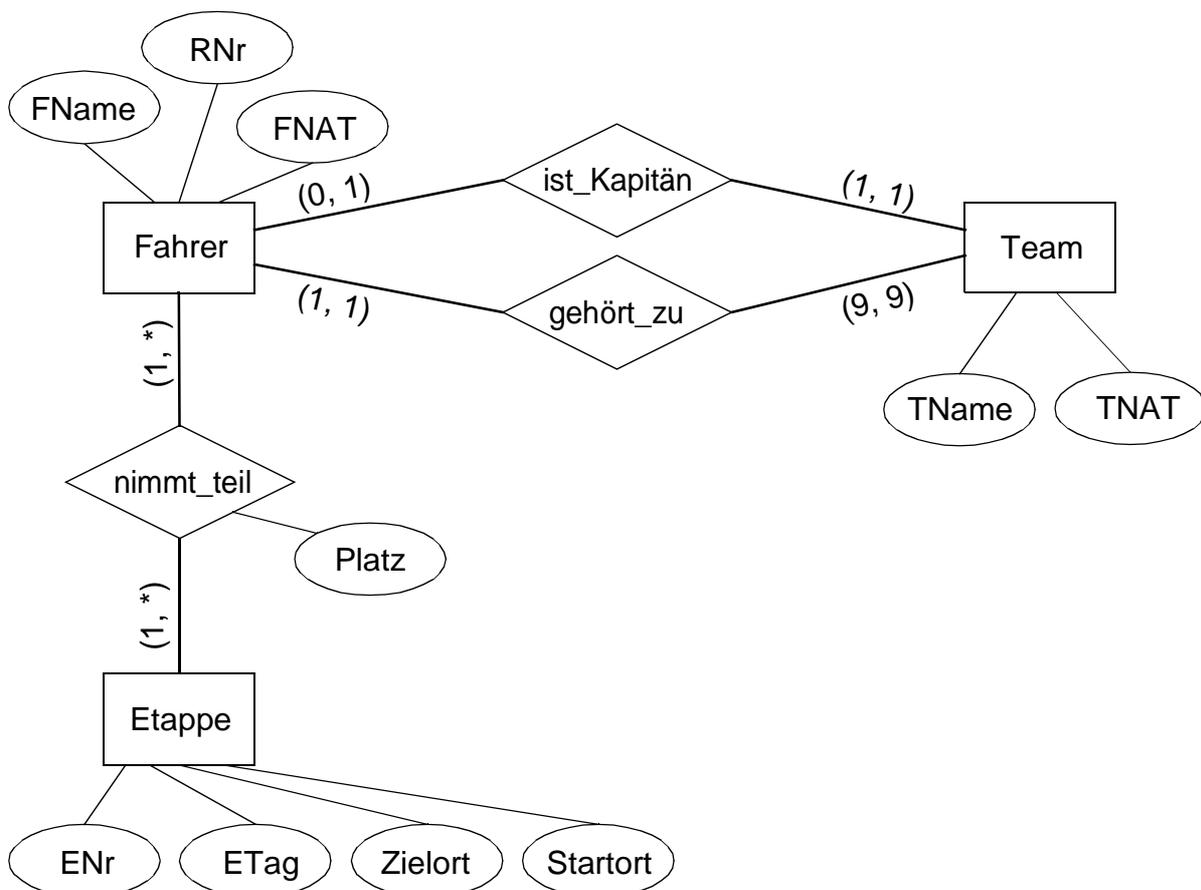
Es gibt **Fahrer**, eindeutig gekennzeichnet sowohl durch Fahrernamen (**FName**) als auch durch ihre Rückennummer (**RNr**), die während der Tour gleich bleibt. Jeder Fahrer hat eine Nationalität (**FNAT**), abgekürzt mit GER, USA, FRA, usw.

Fahrer **gehören zu** einem **Team**, natürlich immer nur zu einem. Jedes Team besteht aus genau neun Fahrern. Teams sind durch den eindeutigen Teamnamen (**TName**) gekennzeichnet und haben auch eine Nationalität (**TNAT**).

In jedem Team erfüllt genau ein Fahrer die **Rolle des Kapitäns**, daneben gibt es noch sportliche Leiter, was wir hier nicht betrachten.

Die Tour de France besteht aus mehreren **Etappen**. Eine Etappe hat eine Nummer (**ENr**), die gewöhnlich als Schlüssel dient, einen Tag (**ETag**), der zwar auch Schlüsseleigenschaft hat, weil es an einem Tag keine zwei Etappen gibt, aber wegen unterschiedlicher Schreibweisen problematisch ist, einen **Startort** und einen **Zielort**.

Zwischen Fahrern und Etappen gibt es eine Beziehung **nimmt_teil**, die angibt, welche Fahrer an welchen Etappen teilgenommen haben. Ein Attribut **Platz** gibt den erlangten Platz beim Zieleinlauf an. Jeder Fahrer nimmt an wenigstens einer Etappe teil, auch wenn er schon bei der ersten ausfallen kann. Etappen, die wegen externer Umstände insgesamt abgesagt werden, berücksichtigen wir hier nicht, d.h. jede hat mindestens einen Teilnehmer.



Aufgabe 2:

Nennen Sie mindestens eine Beziehung aus Aufgabe 1, bei der eine Mitgliedschaft für ein Entity **optional ist**?

ist_Kapitän (Nicht jeder Fahrer ist Kapitän eines Teams.)

Aufgabe 3:

Nehmen wir an, wir hätten eine Tabelle **nimmt_teil** mit dem zusammengesetzten Schlüssel (PRIMARY KEY) Rückennummer **RNr** und Etappennummer **ENr**, sowie dem Attribut **Platz**, für den Rang, den der Fahrer auf dieser Etappe belegt. Eine Datenbank-Null darf eingetragen werden

- für einen **Platz**-Wert, wenn er während des Rennens noch *unbekannt* ist.
- für einen **Platz**-Wert, wenn er wegen Ausscheidens des Fahrers, z.B. nach Zeitüberschreitung, *nicht zutreffend* ist.
- für **RNr**, wenn der Sieger noch nicht feststeht, ein Siegertupel mit Platz-Wert 1 aber schon existiert.
- für **ENr**, wenn man mit diesem Trick das Gesamtklassement in der Tabelle abspeichern will.
- in keiner der drei Spalten.

Aufgabe 4:

Ein wohlmeinender Datenbankadministrator hat **Fahrer** und **Team** zusammengefaßt zu einer Relation **Teilnehmer** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 1 widerspiegelt, wobei die Rückennummer entfallen ist.

Teilnehmer

FName	FNAT	TName	TNAT
Lance Armstrong	USA	US Postal	USA
Roberto Heras	ESP	US Postal	USA
Jan Ullrich	GER	Team Bianchi	GER
Jörg Jaksche	GER	O.N.C.E.	ESP
Joseba Beloki	ESP	O.N.C.E.	ESP

- (a) Was sind die Schlüsselkandidaten?

{FName}

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

$\{\text{FName}\} \rightarrow \{\text{FNAT}, \text{TName}, \text{TNAT}\}$

$\{\text{TName}\} \rightarrow \{\text{TNAT}\}$

(c) Wenn die Schlüsselkandidaten nicht zusammengesetzt sind, ist eine Relation in 2. Normalform. Ist **Teilnehmer** in 2NF? Begründung!

Ja, Teilnehmer ist in 2NF.

Nur $\{\text{FName}\}$ ist Schlüsselkandidat. Dieser ist nicht zusammengesetzt.

Aufgabe 5:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial, X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Ist **Teilnehmer** aus Aufgabe 4 in 3NF? Begründung!

Nein, Teilnehmer ist nicht in 3NF.

Begründung: $\{\text{TName}\} \rightarrow \{\text{TNAT}\}$ ist eine nicht-triviale funktionale Abhängigkeit eines nicht-primen Attributs von einem Nicht-Schlüssel.

Alternative Begründung: $\{\text{FName}\} \xrightarrow{\neq} \{\text{TName}\} \rightarrow \{\text{TNAT}\}$ ist eine transitive Abhängigkeit eines nicht-primen Attributs von einem Schlüssel.

Aufgabe 6:

In der neuen Relation **Fahrer2** fliegt die Teamnationalität raus, dafür kommt **RNr**, die Rückennummer des Fahrers, rein. Einen Ausschnitt zeigen wir unten.

Fahrer2

FName	RNr	FNAT	TName
Lance Armstrong	1	USA	US Postal
Roberto Heras	2	ESP	US Postal
Jan Ullrich	131	GER	Team Bianchi
Jörg Jaschke	15	GER	O.N.C.E.
Joseba Beloki	11	ESP	O.N.C.E.
Tyler Hamilton	71	USA	CSC

(a) Ist **Fahrer2** in 3NF? Begründung!

Ja, Fahrer2 ist in 3NF.

Begründung: Alle nicht-trivialen funktionalen Abhängigkeiten gehen von FName oder RNr aus. Beides sind Schlüsselkandidaten für die Relation Fahrer2. Es gibt keine nicht-trivialen funktionalen Abhängigkeiten, die von FNAT oder TName ausgehen.

Damit ist das in Aufgabe 5 angegebene Kriterium für das Vorliegen von 3NF erfüllt.

(b) Wenn ja, ist **Fahrer2** sogar in BCNF? Wenn nein, welches Attribut ist zu streichen?

Begründung! Hinweis: Gegenüber 3NF entfällt bei BCNF die Alternative „... oder A ist prim.“

Ja, Fahrer2 ist sogar in BCNF.

Begründung: Da es nur nicht-triviale funktionale Abhängigkeiten gibt, die von Schlüsselkandidaten ausgehen, ist das in Aufgabe 5 angegebene Kriterium auch ohne die Alternative „... oder A ist prim.“ erfüllt.

Aufgabe 7:

Die Relation $R = \mathbf{Fahrer2}$ soll in zwei Relationen aufgeteilt werden. Zur Auswahl stehen die beiden Aufteilungen

Alternative 1: $R_1 = (RNr, FName)$ und $R_2 = (RNr, FNAT, TName)$, oder

Alternative 2: $R_1 = (FName, TName)$ und $R_2 = (RNr, FNAT, TName)$

(a) Sind die Aufteilungen verlustfrei? Begründung mit dem formalen Argument $R_1 \cap R_2 \dots$

Hinweis: Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$.

Alternative 1 ist verlustfrei.

Begründung: $R_1 \cap R_2 = \{RNr\}$ und $\{RNr\} \rightarrow R_1$

Alternative 2 ist nicht verlustfrei.

Begründung: $R_1 \cap R_2 = \{TName\}$ und $\{TName\} \not\rightarrow R_1$ und $\{TName\} \not\rightarrow R_2$

(b) Bei Alternative 1 kann man lokal nicht mehr die funktionale Abhängigkeit $FName \rightarrow FNAT$ überprüfen, weil sich beide Attribute nicht mehr in der selben Relation befinden. Wie kann man direkt mit einem Armstrongschen Axiom zeigen, daß die Aufteilung zumindestens bezüglich dieser Abhängigkeit trotzdem abhängigkeitsbewahrend ist.

Hinweis: Die drei Armstrongschen Axiome sind

Reflexivität: Wenn $Y \subseteq X$, dann folgt $X \rightarrow Y$, die natürlichen oder trivialen Abhängigkeiten.

Erweiterung: Wenn $X \rightarrow Y$ gilt und Z eine Teilmenge von R ist, dann gilt auch $XZ \rightarrow YZ$ (XZ ist kurz für $X \cup Z$).

Transitivität: Wenn $X \rightarrow Y$ und $Y \rightarrow Z$, dann auch $X \rightarrow Z$.

Es gilt: {FName} \rightarrow {RNr} (in Relation R_1) und
 {RNr} \rightarrow {FNAT} (in Relation R_2)

Mit Hilfe der Transitivität erhält man: {FName} \rightarrow {FNAT}

Aufgabe 8:

Die folgende Tabelle zeigt einen Ausschnitt aus der Relation **nimmt_teil**, die aus der ER-Modellierung in Aufgabe 1 abgeleitet wurde. Schlüsselattribute sind unterstrichen. Die Bemerkungen sind nur als menschliche Lesehilfe gedacht.

nimmt_teil

<u>RNr</u>	<u>ENr</u>	Platz	Bemerkung
1	15	1	Lance gewinnt die Bergankunft in Luz-Ardiden
1	0	7	Lance 7. im Prolog und verliert 5 s auf Jan
1	8	3	Lance Dritter in L'Alpe d'Huez
131	12	1	Jan gewinnt das Zeitfahren (12. Etappe)
131	0	4	Jan 4. im Prolog (0. Etappe)
131	8	13	Jan 13. in L'Alpe d'Huez
2	8	12	Roberto Heras (US Postal) vor Jan in L'Alpe d'Huez
2	9	15	Heras 15. in der 9. Etappe über den Col d'Izoard
11	8	6	Beloki 6. in L'Alpe d'Huez, zweiter im Gesamtklass.
11	9	Null	Joseba Beloki scheidet tragisch mit Sturz aus
71	0	6	Tyler Hamilton 6. im Prolog knapp vor Lance
71	16	1	Tyler Hamilton gewinnt in Bayonne (16. Etappe)

Enthält **nimmt_teil** Fremdschlüssel? Wenn ja, in welchen Relationen, die sich aus der ER-Modellierung in Aufgabe 1 ergeben, sind die zugehörigen Primärschlüssel enthalten? Wenn nein, warum tritt kein Fremdschlüssel auf?

Ja, **nimmt_teil** enthält zwei Fremdschlüssel:

- 1.) RNr: Der zugehörige Primärschlüssel befindet sich in der Relation Fahrer.
- 2.) ENr: Der zugehörige Primärschlüssel befindet sich in der Relation Etappe.

Aufgabe 9:

Gesucht ist eine SQL-Abfrage, die aus **nimmt_teil** in Verbindung mit **Fahrer2** für alle Etappen die drei Erstplatzierten liefert. Jedes Ergebnistupel soll also die Rückennummer, den Namen, den Platz sowie die Etappennummer liefern. Sortieren Sie das Ergebnis nach Etappen.

```

SELECT F.RNr, FName, Platz, ENr
FROM nimmt_teil n, Fahrer2 F
WHERE n.RNr = F.RNr AND Platz <= 3
ORDER BY ENr

```

Aufgabe 10:

Was liefert generell die folgende Abfrage?

```

SELECT FNAT, COUNT(*)
FROM nimmt_teil n, Fahrer2 F
WHERE n.RNr = F.RNr AND Platz = 1
GROUP BY FNAT

```

Diese Abfrage liefert eine zweispaltige Tabelle mit den verschiedenen Fahernationalitäten und ihrer jeweiligen Anzahl an Siegen. Fahernationalitäten ohne Siege werden nicht aufgelistet.

Aufgabe 11:

Gesucht ist eine SQL-Abfrage, die alle Teamnamen (genau einmal) der Teams in **Fahrer2** ausgibt, die keinen deutschen Fahrer haben (wo es keinen Fahrer gibt mit FNAT = 'GER').

```

SELECT DISTINCT F.TName
FROM Fahrer2 F
WHERE NOT EXISTS (SELECT * FROM Fahrer2
                     WHERE FNAT = 'GER' AND
                     TName = F.TName)

```

Aufgabe 12:

Gezeigt sind drei Ablauffolgen. Welche davon ist **Lost Update**, **Dirty Read**, **Unrepeatable Read** oder **OK**?

- | | | |
|-----|--|--------------------------|
| (a) | T2 Lesen A
T1 Schreiben A
T2 Schreiben A | <u>Lost Update</u> |
| (b) | T1 Lesen A
T2 Schreiben A
T1 Lesen A | <u>Unrepeatable Read</u> |
| (c) | T2 Schreiben A
T1 Lesen A
T2 Schreiben A | <u>Dirty Read</u> |

Klausur zur Vorlesung Datenbanken I im Sommersemester 2008

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		8	6	
2	5		9	6	
3	2+2+3		10	5	
4	7		11	4	
5	3+3		12	3	
6	4+4		13	5	
7	5		Summe	70	

Aufgabe 1:

Was gilt für die Behandlung von DB-Nullwerten in den großen relationalen Datenbanksystemen, z. B. Oracle oder DB2?

- () Es wird zwischen der Null für „nicht zutreffend“ und „unbekannt“ unterschieden.
- () Bei der Tabellendefinition läßt sich das Auftreten von Nullwerten für jede Spalte mit der Angabe „NOT NULL“ verhindern.
- () Mit der NVL- oder COALESCE-Funktion kann man einen Ersatzwert für Null angeben.
- () Der Vergleich zweier Spaltenwerte, die beide DB-Null sind, ergibt wahr.
- () Bei COUNT(S) auf einer einzelnen Spalte S werden Nullwerte nicht gezählt.

Aufgabe 2:

Wir legen uns eine kleine Datenbank für unsere Lieblingsfilme an.

Für **Filme** gelte: Es gibt einen eindeutigen **Filmtitel**, mitwirkende **Schauspieler**, einen **Regisseur**, ein Erscheinungsjahr (**Jahr**). Jeder Film habe mindestens einen/eine Schauspieler/in und genau einen Regisseur, wovon es in der Realität Ausnahmen gibt, die hier nicht zugelassen sind. Schauspieler und Regisseure sind in der Regel an mehreren Filmen beteiligt, sogar innerhalb eines Jahres.

Man wird aber erst zum Schauspieler oder Regisseur, wenn man mindestens bei einem Film mitgespielt bzw. bei ihm Regie geführt hat. Schauspieler und Regisseure werden durch ihren **Namen** eindeutig identifiziert, sie haben ein Geburtsjahr (**GJahr**) und einen Geburtsort (**GOrt**).

Geben Sie ein ER-Diagramm für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Schlüssel.

Aufgabe 3:

Im ersten Entwurf einer Datenbank entsteht eine Tabelle **Filme** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 2 widerspiegelt.

Filme

Filmtitel	Schauspieler	Regisseur	Jahr
The Terminator	Arnie Schwarzenegger	James Cameron	1984
Titanic	Leonardo DiCaprio	James Cameron	1997
Titanic	Kate Winslet	James Cameron	1997
Departed	Leonardo DiCaprio	Martin Scorsese	2006
Departed	Jack Nickolson	Martin Scorsese	2006
Departed	Matt Damon	Martin Scorsese	2006
Departed	Alec Baldwin	Martin Scorsese	2006
Cape Fear	Robert de Niro	Martin Scorsese	1991
Cape Fear	Nick Nolte	Martin Scorsese	1991
Cape Fear	Jessica Lange	Martin Scorsese	1991
True Lies	Arnie Schwarzenegger	James Cameron	1994
True Lies	Jamie Lee Curtis	James Cameron	1994
Taxi Driver	Robert de Niro	Martin Scorsese	1976
Taxi Driver	Judie Foster	Martin Scorsese	1976

- (a) Was sind die Schlüsselkandidaten?
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
- (c) Wenn alle Schlüsselkandidaten nicht zusammengesetzt sind, ist eine Relation in 2. Normalform. Allgemein ist eine Relation in der 2. Normalform (2NF), wenn jedes Attribut entweder prim oder **voll funktional** abhängig von jedem Schlüsselkandidat ist. Ist Filme in 2NF? Begründung!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Aus **Filme** haben wir durch Weglassen der Schauspieler-Spalte die Relation **FRJ** gewonnen. Beachten Sie, dass Regisseure potentiell mehr als einen Film pro Jahr drehen können, auch wenn das hier nicht gezeigt ist. Weiterhin habe jeder Film nur einen Regisseur, wie angenommen.

Ist **FRJ** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

FRJ

Filmtitel	Regisseur	Jahr
The Terminator	James Cameron	1984
Titanic	James Cameron	1997
Departed	Martin Scorsese	2006
Cape Fear	Martin Scorsese	1991
True Lies	James Cameron	1994
Taxi Driver	Martin Scorsese	1976

Aufgabe 5:

Für die folgenden Aufgaben haben wir die Schauspieler mit den Regisseuren zu einer Relation **SR(Schauspieler, Regisseur)** zusammengefaßt. Alternativ haben wir aus Filmtitel und Schauspieler eine Relation **FS(Filmtitel, Schauspieler)** gebildet.

SR

Schauspieler	Regisseur
Arnie Schwarzenegger	James Cameron
Leonardo DiCaprio	James Cameron
Kate Winslet	James Cameron
Leonardo DiCaprio	Martin Scorsese
Jack Nickolson	Martin Scorsese
Matt Damon	Martin Scorsese
Alec Baldwin	Martin Scorsese
Robert de Niro	Martin Scorsese
Nick Nolte	Martin Scorsese
Jessica Lange	Martin Scorsese
Jamie Lee Curtis	James Cameron
Judie Foster	Martin Scorsese

FS

Filmtitel	Schauspieler
The Terminator	Arnie Schwarzenegger
Titanic	Leonardo DiCaprio
Titanic	Kate Winslet
Departed	Leonardo DiCaprio
Departed	Jack Nickolson
Departed	Matt Damon
Departed	Alec Baldwin
Cape Fear	Robert de Niro
Cape Fear	Nick Nolte
Cape Fear	Jessica Lange
True Lies	Arnie Schwarzenegger
True Lies	Jamie Lee Curtis
Taxi Driver	Robert de Niro
Taxi Driver	Judie Foster

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

(b) Sind die Relationen **SR** und **FS** in BCNF? Begründung!

Aufgabe 6:

Die Relationen **FRJ** und **SR** sowie **FRJ** und **FS** von oben bilden jeweils eine Aufteilung von **Filme**. Bekanntlich ist eine solche Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$.

(a) Welche der Aufteilungen **FRJ** und **SR** bzw. **FRJ** und **FS** ist verlustfrei? Begründung jeweils mit dem formalen Argument $R_1 \cap R_2 \dots$

(b) Wenn nein, wie zeigt sich der Verlust? Wenn ja, welcher Join bringt **Filme** zurück?

Aufgabe 7:

Zur Einstimmung auf SQL eine einfache Abfrage: Wie lautet die SQL-Abfrage, die genau die Schauspieler liefert, die in der Tabelle **Filme** aus Aufgabe 3 mit Filmen vertreten sind, bei denen „James Cameron“ Regie geführt hat? Dabei soll jeder Schauspieler maximal einmal erscheinen.

Aufgabe 8:

Wie lautet die gleichwertige SQL-Abfrage zum folgenden Ausdruck der relationalen Algebra?

$$\pi_{\text{Jahr}} (\sigma_{\text{Schauspieler}=\text{„Leonardo DiCaprio“}} (\text{FS} \bowtie \text{FRJ}))$$

Aufgabe 9:

Wenn wie oben **FS(Filmtitel, Schauspieler)** und **FRJ(Filmtitel, Regie, Jahr)** gegeben sind, wie vereinbart man dann eine Relation **RolleAb1990(Schauspieler)** als Sicht (view), die alle die Schauspieler enthält, die 1990 oder später noch in einem Film gespielt haben.

Aufgabe 10:

In der Tabelle **FS** soll der Name des Schauspielers Schwarzenegger berichtigt werden, also „Arnold Schwarzenegger“ statt „Arnie Schwarzenegger“. Geben Sie das entsprechende SQL-Kommando dazu an.

Aufgabe 11:

Betrachten Sie die folgende Abfrage:

```
SELECT Regisseur, Schauspieler, COUNT(Filmtitel)
FROM Filme
GROUP BY Regisseur, Schauspieler
HAVING COUNT(Filmtitel) > 1;
```

Beschreiben Sie in ein bis zwei Sätzen, welche Ausgabe diese Abfrage auf einer beliebigen **Filme**-Tabelle liefert!

Aufgabe 12:

Ordnen Sie die Begriffe links den Aussagen rechts zu! Zu einem Begriff kann es auch keinen oder mehrere zugehörige Aussagen geben.

WAL	Wachstums-/Reduzierungsphase
steal	bei commit alle Seiten durchschreiben
cursor stability	Operationen zuerst ins Log schreiben
2PL	letzte Lesezeit
RTM(x)	Transaktion kann einer anderen Seiten wegnehmen
RAID	Disk Array

Aufgabe 13:

Betrachten Sie die folgende Ablauffolge:

Transaktion 1	Transaktion 2	Transaktion 3
Lesen A Schreiben A		
	Lesen A Lesen B Schreiben A Schreiben B	
		Lesen B Lesen C
Lesen C		Schreiben C Schreiben B
Schreiben C		

Geben Sie den vollständigen Abhängigkeitsgraphen an! Ist die Ablauffolge serialisierbar?

**Klausur zur Vorlesung Datenbanken I
im Sommersemester 2008**

Name:

Matr.Nr.: Studiengang:

Musterlösung

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		8	6	
2	5		9	6	
3	2+2+3		10	5	
4	7		11	4	
5	3+3		12	3	
6	4+4		13	5	
7	5		Summe	70	

Aufgabe 1:

Was gilt für die Behandlung von DB-Nullwerten in den großen relationalen Datenbanksystemen, z. B. Oracle oder DB2?

- () Es wird zwischen der Null für „nicht zutreffend“ und „unbekannt“ unterschieden.
- (**X**) Bei der Tabellendefinition läßt sich das Auftreten von Nullwerten für jede Spalte mit der Angabe „NOT NULL“ verhindern.
- (**X**) Mit der NVL- oder COALESCE-Funktion kann man einen Ersatzwert für Null angeben.
- () Der Vergleich zweier Spaltenwerte, die beide DB-Null sind, ergibt wahr.
- (**X**) Bei COUNT(S) auf einer einzelnen Spalte S werden Nullwerte nicht gezählt.

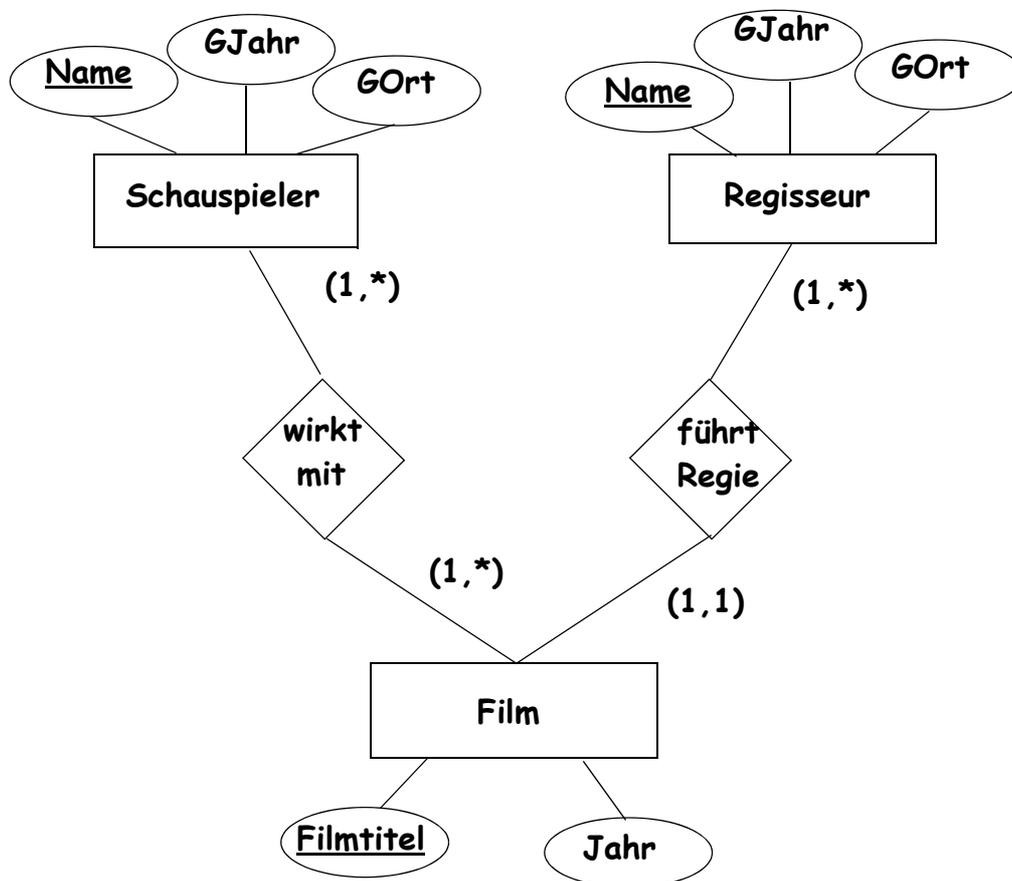
Aufgabe 2:

Wir legen uns eine kleine Datenbank für unsere Lieblingsfilme an.

Für **Filme** gelte: Es gibt einen eindeutigen **Filmtitel**, mitwirkende **Schauspieler**, einen **Regisseur**, ein Erscheinungsjahr (**Jahr**). Jeder Film habe mindestens einen/eine Schauspieler/in und genau einen Regisseur, wovon es in der Realität Ausnahmen gibt, die hier nicht zugelassen sind. Schauspieler und Regisseure sind in der Regel an mehreren Filmen beteiligt, sogar innerhalb eines Jahres.

Man wird aber erst zum Schauspieler oder Regisseur, wenn man mindestens bei einem Film mitgespielt bzw. bei ihm Regie geführt hat. Schauspieler und Regisseure werden durch ihren **Namen** eindeutig identifiziert, sie haben ein Geburtsjahr (**GJahr**) und einen Geburtsort (**GOrt**).

Geben Sie ein ER-Diagramm für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Schlüssel.



Andere Varianten sind möglich.

Aufgabe 3:

Im ersten Entwurf einer Datenbank entsteht eine Tabelle **Filme** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 2 widerspiegelt.

Filme

Filmtitel	Schauspieler	Regisseur	Jahr
The Terminator	Arnold Schwarzenegger	James Cameron	1984
Titanic	Leonardo DiCaprio	James Cameron	1997
Titanic	Kate Winslet	James Cameron	1997
Departed	Leonardo DiCaprio	Martin Scorsese	2006
Departed	Jack Nicholson	Martin Scorsese	2006
Departed	Matt Damon	Martin Scorsese	2006
Departed	Alec Baldwin	Martin Scorsese	2006
Cape Fear	Robert de Niro	Martin Scorsese	1991
Cape Fear	Nick Nolte	Martin Scorsese	1991
Cape Fear	Jessica Lange	Martin Scorsese	1991
True Lies	Arnold Schwarzenegger	James Cameron	1994
True Lies	Jamie Lee Curtis	James Cameron	1994
Taxi Driver	Robert de Niro	Martin Scorsese	1976
Taxi Driver	Judie Foster	Martin Scorsese	1976

- (a) Was sind die Schlüsselkandidaten?

Nur ein Schlüsselkandidat: {Filmtitel, Schauspieler}

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

Filmtitel -> {Jahr, Regisseur}

{Filmtitel, Schauspieler} -> Filme (aus Schlüsseleigenschaft)

- (c) Wenn alle Schlüsselkandidaten nicht zusammengesetzt sind, ist eine Relation in 2. Normalform. Allgemein ist eine Relation in der 2. Normalform (2NF), wenn jedes Attribut entweder prim oder **voll funktional** abhängig von jedem Schlüsselkandidat ist. Ist Filme in 2NF? Begründung!

Nein, nicht in 2NF, denn Jahr ist z.B. nicht prim und nicht voll funktional abhängig vom Schlüsselkandidaten {Filmtitel, Schauspieler}; es gilt bereits Filmtitel -> Jahr.

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Aus **Filme** haben wir durch Weglassen der Schauspieler-Spalte die Relation **FRJ** gewonnen. Beachten Sie, dass Regisseure potentiell mehr als einen Film pro Jahr drehen können, auch wenn das hier nicht gezeigt ist. Weiterhin habe jeder Film nur einen Regisseur, wie angenommen.

Ist **FRJ** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

FRJ

Filmtitel	Regisseur	Jahr
The Terminator	James Cameron	1984
Titanic	James Cameron	1997
Departed	Martin Scorsese	2006
Cape Fear	Martin Scorsese	1991
True Lies	James Cameron	1994
Taxi Driver	Martin Scorsese	1976

Schlüsselkandidaten: Filmtitel

**Nicht-triviale fkt. Abhängigkeiten: Filmtitel -> FRJ (Schlüsseleigenschaft)
also insbesondere Filmtitel -> Regisseur und Filmtitel -> Jahr**

3NF liegt vor, da alle nicht-trivialen fkt. Abhängigkeiten von Filmtitel ausgehen und Filmtitel Schlüssel ist.

Im übrigen ist auch 1NF erfüllt, da alle Attribute atomar sind (kein Abzug, wenn nicht erwähnt).

Aufgabe 5:

Für die folgenden Aufgaben haben wir die Schauspieler mit den Regisseuren zu einer Relation **SR(Schauspieler, Regisseur)** zusammengefaßt. Alternativ haben wir aus Filmtitel und Schauspieler eine Relation **FS(Filmtitel, Schauspieler)** gebildet.

SR

Schauspieler	Regisseur
Arnie Schwarzenegger	James Cameron
Leonardo DiCaprio	James Cameron
Kate Winslet	James Cameron
Leonardo DiCaprio	Martin Scorsese
Jack Nickolson	Martin Scorsese
Matt Damon	Martin Scorsese
Alec Baldwin	Martin Scorsese
Robert de Niro	Martin Scorsese
Nick Nolte	Martin Scorsese
Jessica Lange	Martin Scorsese
Jamie Lee Curtis	James Cameron
Judie Foster	Martin Scorsese

FS

Filmtitel	Schauspieler
The Terminator	Arnie Schwarzenegger
Titanic	Leonardo DiCaprio
Titanic	Kate Winslet
Departed	Leonardo DiCaprio
Departed	Jack Nickolson
Departed	Matt Damon
Departed	Alec Baldwin
Cape Fear	Robert de Niro
Cape Fear	Nick Nolte
Cape Fear	Jessica Lange
True Lies	Arnie Schwarzenegger
True Lies	Jamie Lee Curtis
Taxi Driver	Robert de Niro
Taxi Driver	Judie Foster

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

SR: Schlüsselkandidaten {Schauspieler, Regisseur}, keine nicht-trivialen fkt. Abhängigkeiten

FS: Schlüsselkandidaten {Filmtitel, Schauspieler}, keine nicht-trivialen fkt. Abhängigkeiten

(b) Sind die Relationen **SR** und **FS** in BCNF? Begründung!

SR und FS sind in BCNF, da alle Relationen mit nur 2 Attributen automatisch in BCNF sind

Aufgabe 6:

Die Relationen **FRJ** und **SR** sowie **FRJ** und **FS** von oben bilden jeweils eine Aufteilung von **Filme**. Bekanntlich ist eine solche Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$.

(a) Welche der Aufteilungen **FRJ** und **SR** bzw. **FRJ** und **FS** ist verlustfrei? Begründung jeweils mit dem formalen Argument $R_1 \cap R_2 \dots$

FRJ und SR: $FRJ \cap SR = \text{Regisseur}$

Regisseur bestimmt nicht FRJ und Regisseur bestimmt nicht SR, daher nicht verlustfrei

FRJ und FS: $FRJ \cap FS = \text{Filmtitel}$

Filmtitel bestimmt FRJ (und Filmtitel bestimmt nicht FS), daher verlustfrei!

(b) Wenn nein, wie zeigt sich der Verlust? Wenn ja, welcher Join bringt **Filme** zurück?

bei FRJ und SR zeigt sich Verlust, indem zu Filmen eines Regisseurs Schauspieler angegeben werden, die zwar mal mit ihm gearbeitet haben, aber nicht in diesem Film.

bei FRJ und FS bringt der Natural Join (also der Join über Filmtitel) alle richtigen Tupel zurück.

Aufgabe 7:

Zur Einstimmung auf SQL eine einfache Abfrage: Wie lautet die SQL-Abfrage, die genau die Schauspieler liefert, die in der Tabelle **Filme** aus Aufgabe 3 mit Filmen vertreten sind, bei denen „James Cameron“ Regie geführt hat? Dabei soll jeder Schauspieler maximal einmal erscheinen.

```
SELECT DISTINCT Schauspieler  
FROM Filme  
WHERE Regisseur = 'James Cameron'
```

Aufgabe 8:

Wie lautet die gleichwertige SQL-Abfrage zum folgenden Ausdruck der relationalen Algebra?

$$\pi_{\text{Jahr}} (\sigma_{\text{Schauspieler}=\text{'Leonardo DiCaprio'}} (\text{FS} \bowtie \text{FRJ}))$$

```
SELECT DISTINCT Jahr kein Abzug, wenn DISTINCT fehlt  
FROM FS, FRJ  
WHERE FS.Filmtitel = FRJ.Filmtitel AND  
FS.Schauspieler = 'Leonardo DiCaprio'
```

auch möglich: ... **FROM FS NATURAL JOIN FRJ** ...
... **FROM FS JOIN FRJ ON (FS.Filmtitel = FRJ.Filmtitel)** ...
... **FROM FS JOIN FRJ USING (Filmtitel)** ...

Aufgabe 9:

Wenn wie oben **FS(Filmtitel, Schauspieler)** und **FRJ(Filmtitel, Regie, Jahr)** gegeben sind, wie vereinbart man dann eine Relation **RolleAb1990(Schauspieler)** als Sicht (view), die alle die Schauspieler enthält, die 1990 oder später noch in einem Film gespielt haben.

```
CREATE VIEW RolleAb1990 AS  
  SELECT DISTINCT Schauspieler  
  FROM FS, FRJ  
  WHERE FS.Filmtitel = FRJ.Filmtitel AND FRJ.Jahr >= 1990;
```

alternative Formulierungen zum Join wie oben

Aufgabe 10:

In der Tabelle **FS** soll der Name des Schauspielers Schwarzenegger berichtigt werden, also „Arnold Schwarzenegger“ statt „Arnie Schwarzenegger“. Geben Sie das entsprechende SQL-Kommando dazu an.

```
UPDATE FS  
SET Schauspieler = 'Arnold Schwarzenegger'  
WHERE Schauspieler = 'Arnie Schwarzenegger';
```

Aufgabe 11:

Betrachten Sie die folgende Abfrage:

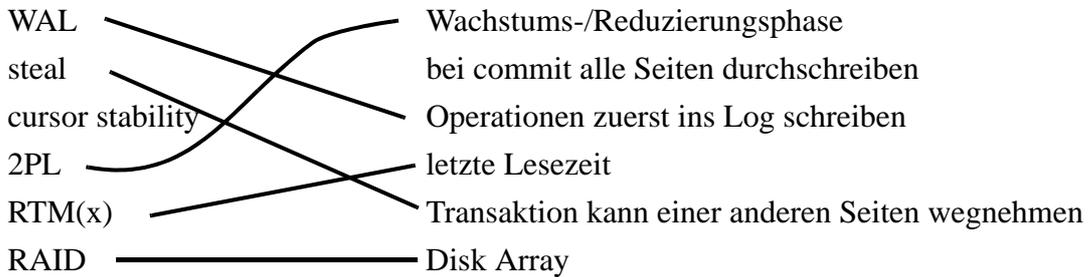
```
SELECT Regisseur, Schauspieler, COUNT(Filmtitel)  
FROM Filme  
GROUP BY Regisseur, Schauspieler  
HAVING COUNT(Filmtitel) > 1;
```

Beschreiben Sie in ein bis zwei Sätzen, welche Ausgabe diese Abfrage auf einer beliebigen **Filme**-Tabelle liefert!

Auflistung der Regisseur-Schauspieler-Kombinationen, die in mehr als einem Film zusammengearbeitet haben. Zusätzlich erscheint jeweils die Anzahl der gemeinsamen Filme.

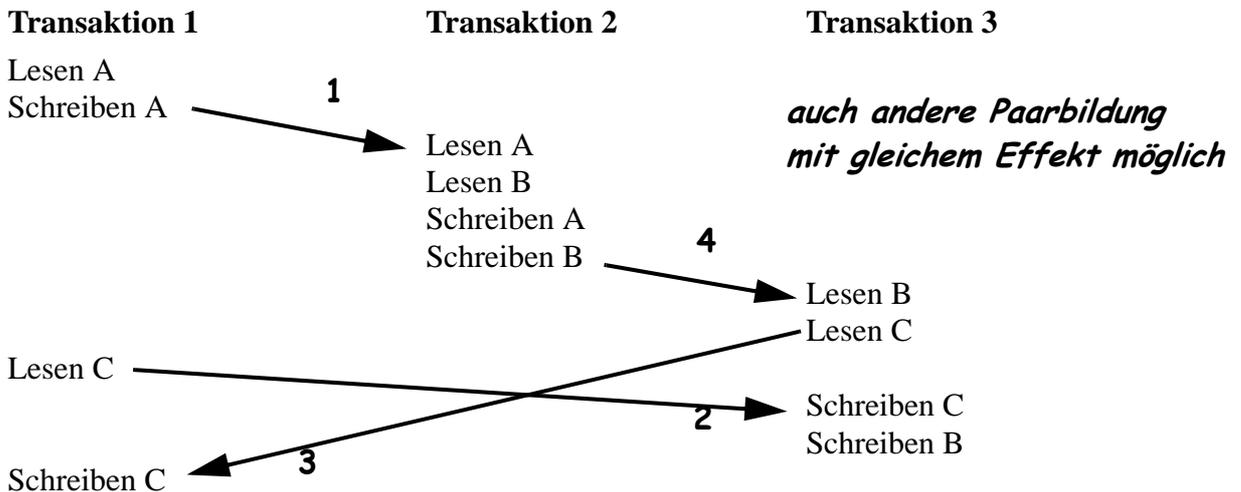
Aufgabe 12:

Ordnen Sie die Begriffe links den Aussagen rechts zu! Zu einem Begriff kann es auch keinen oder mehrere zugehörige Aussagen geben.

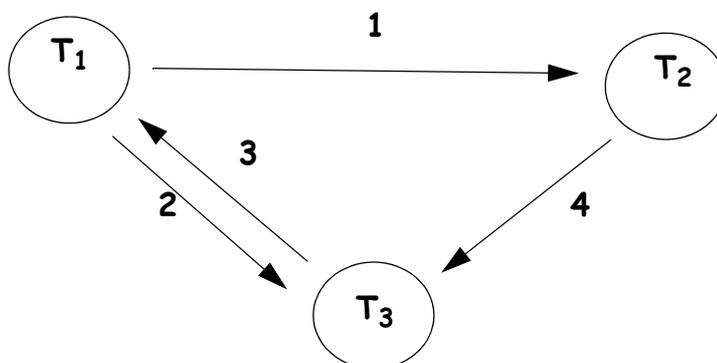


Aufgabe 13:

Betrachten Sie die folgende Ablauffolge:



Geben Sie den vollständigen Abhängigkeitsgraphen an! Ist die Ablauffolge serialisierbar?



Graph enthält (zwei) Zyklen, damit ist die Folge nicht serialisierbar.

Klausur zur Vorlesung Datenbanken I vom Wintersemester 2008/09

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		7	5	
2	6		8	6	
3	2+2+2		9	4	
4	6		10	4	
5	3+3+1		11	5	
6	4		12	6	
			Summe	62	

Aufgabe 1:

Das ANSI/SPARC-Architekturmodell für Datenbanksysteme aus dem Jahr 1975 schlägt drei Schichten vor. Verbinden Sie durch Linien die fünf Datenbankbegriffe links mit jeweils einer der drei Schichten (Modelle, Schemata) rechts.

- | | |
|--|--|
| <ul style="list-style-type: none"> SQL CREATE VIEW • Hash-Join • Normalisierung • Indexoptimierung • logische Gesamtsicht • | <ul style="list-style-type: none"> • externes Schema • konzeptuelles Schema • internes Schema |
|--|--|

Schriftfamilien und einige ihrer Schnitte

Renaissance
Antiqua

Times Roman
Times Roman fett
Times Roman kursiv
Times Roman fettkursiv

Garamond
Garamond fett
Garamond kursiv
Garamond fettkursiv

Klassizistische
Antiqua

Bodoni
Bodoni condensed

Grotesk

Arial
Arial fett
Arial kursiv
Arial fettkursiv
Arial extrafett
Arial extrafettkursiv

Univers
Univers fett
Univers kursiv
Univers fettkursiv

Nicht-Proportional

Courier New
Courier New kursiv

Aufgabe 2:

Ein Verlag verwaltet *Schriften* in einer Datenbank. In der Welt der Schriften gelten die folgenden, hier etwas vereinfachten, Zusammenhänge (siehe auch die Tabelle in Aufgabe 3 und die Illustration auf der gegenüberliegenden Seite).

Es gibt **Schriftfamilien** und **Schriftschnitte**, auch Schriftstile genannt. Eine Schriftfamilie ist eindeutig durch ihren **Schriftnamen** (z. B. *Times Roman*, *MS Arial*, *Courier New*) identifiziert. Außerdem hat jede Schriftfamilie ein Entstehungsjahr **EJahr**.

Zu jeder Schriftfamilie gibt es mindestens einen, unter Umständen sehr viele Schriftschnitte, die wir mit einem eindeutigen abgekürzten **SchriftschnittID** versehen. Jeder Schriftschnitt gehört immer eindeutig zu einer Schriftfamilie. Jeder Schriftschnitt einer Schriftfamilie hat als Eigenschaft einen **Stil** (*normal*, *fett*, *kursiv*, *leicht*, *fettkursiv* usw.). Ferner werde kein Stil einer Schriftfamilie mit zwei Schnitten dargestellt (funktionale Abhängigkeit).

Schriftfamilien wiederum lassen sich klassifizieren, etwa nach der DIN 16518, in sogenannte **Schriftklassen**. Die bekanntesten sind *Renaissance Antiqua Schriften*, *Klassizistische Antiqua Schriften*, *Grotesk Schriften*, auch *Linear Schriften* genannt, *Nicht-proportionale Schriften*, *Schreibschriften*, *Zierschriften*. Eine Schriftklasse hat mindestens eine Schriftfamilie, die bekannten großen Klassen sogar einige hundert Schriftfamilien. Auch wenn die Zuordnung manchmal umstritten ist, behaupten wir, dass jede Schriftfamilie zu genau einer Schriftklasse gehört.

Eine Schriftklasse habe bei uns einen eindeutigen Klassenidentifizier **KlassenID**, einen eindeutigen Klassennamen **Klassenname** und ein Attribut **Serife**, das die Werte *mit*, *ohne* oder einen *Nullwert* für „nicht anwendbar“ annehmen kann, je nachdem ob es eine Schriftklasse für Schriften mit Serifen (Anstrichen, Füßchen) oder für serifenlose (Grotesk-)Schriften ist.

Geben Sie ein ER-Diagramm für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. **Unterstreichen Sie in jeder Entitätsklasse einen Schlüssel!**

Aufgabe 3:

Im ersten Entwurf einer Datenbank entsteht eine Tabelle **Schriften** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 2 widerspiegelt.

Schriften

SchriftschnittID	Schriftname	Klassenname	Stil
TR	Times Roman	Renaissance Antiqua	normal
TRbd	Times Roman	Renaissance Antiqua	fett
TRit	Times Roman	Renaissance Antiqua	kursiv
TRbdit	Times Roman	Renaissance Antiqua	fettkursiv
GMOND	Garamond	Renaissance Antiqua	normal
GMONDbd	Garamond	Renaissance Antiqua	fett
GMONDit	Garamond	Renaissance Antiqua	kursiv
GMONDbdit	Garamond	Renaissance Antiqua	fettkursiv
BOD	Bodoni	Klassizistische Antiqua	normal
BODcnd	Bodoni	Klassizistische Antiqua	eng
ARIAL	MS Arial	Grotesk	normal
ARIALbd	MS Arial	Grotesk	fett
ARIALit	MS Arial	Grotesk	kursiv
ARIALbdit	MS Arial	Grotesk	fettkursiv
ARIALblack	MS Arial	Grotesk	extrafett
ARIALblackit	MS Arial	Grotesk	extrafettkursiv

- (a) Was sind die Schlüsselkandidaten?
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
- (c) Eine Relation ist in der 2. Normalform (2NF), wenn jedes Attribut entweder prim oder **voll funktional** abhängig von jedem Schlüsselkandidat ist. Ist Schriften in 2NF? Begründung auf der Rückseite!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform (3NF), wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Für *Schriftklassen* und den enthaltenen *Schriftfamilien* haben wir eine Relation **SK** entworfen. Ist **SK** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

SK

Schriftname	KlassenID	Klassenname
Times Roman	RA	Renaissance Antiqua
Garamond	RA	Renaissance Antiqua
Bodoni	KA	Klassizistische Antiqua
MS Arial	LG	Linear Grotesk
Helvetica	LG	Linear Grotesk
Univers	LG	Linear Grotesk

Aufgabe 5:

Für die folgenden Aufgaben haben wir zwei Relationen **Schriftklassen**(**KlassenID**, **Klassenname**) und **Schriftfamilien**(**Schriftname**, **KlassenID**, **Serife**, **EJahr**) gebildet. Beachten Sie: für jede Klasse ist eindeutig bestimmt, ob alle darin enthaltenen Schriften Serifen haben oder nicht, bzw. man legt durch einen Nullwert fest, dass dieses Kriterium nicht anwendbar ist.

Schriftklassen

Schriftfamilien

Klassen-ID	Klassenname	Schriftname	Klassen-ID	Serife	EJahr
RA	Renaissance Antiqua	Times Roman	RA	mit	1931
KA	Klassizistische Antiqua	Garamond	RA	mit	1530
LG	Linear Grotesk	Bodoni	KA	mit	1798
ZS	Zierschrift	MS Arial	LG	ohne	1989
SC	Schreibschrift	Helvetica	LG	ohne	1956
TT	Nicht-Proportional	Univers	LG	ohne	1956
SYM	Zeichen und Symbole				

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

(b) Sind die Relationen **Schriftklassen** und **Schriftfamilien** in 3NF? Wenn ja, sogar in BCNF? Begründung!

(c) Welches Attribut ist offensichtlich in der falschen Relation?

Aufgabe 6:

Zur Einstimmung auf SQL eine einfache Abfrage: Wie lautet die SQL-Abfrage auf der Tabelle **Schriften** aus Aufgabe 3, die uns alle Namen der Schriftfamilien liefert, für die ein Schnitt mit Stil „fett“ vorhanden ist.

Aufgabe 7:

Wie lautet die gleichwertige SQL-Abfrage zum folgenden Ausdruck der relationalen Algebra mit den Relationen aus Aufgabe 5?

$$\pi_{\text{Schriftname, Klassenname}} (\sigma_{\text{Serife}='mit'} (\text{Schriftklassen} \bowtie \text{Schriftfamilien}))$$

Aufgabe 8:

Wenn wie oben **Schriftfamilien(Schriftname, KlassenID, Serife, EJahr)** und **Schriftklassen(KlassenID, Klassenname)** gegeben sind, wie vereinbart man dann eine Relation **ModerneKlasse(KName)** als Sicht (view), die nur die Klassen aufführt (**jeweils einmal**), die Schriften enthalten, die gemäß **EJahr**-Wert nach 1900 entstanden sind.

Aufgabe 9:

In der Tabelle **Schriften** aus Aufgabe 3 soll der Name der Schriftfamilie *Times Roman* in *Times New Roman* geändert werden. Geben Sie das entsprechende SQL-Kommando dazu an.

Aufgabe 10:

Betrachten Sie die folgende Abfrage:

```
SELECT Schriftname, COUNT(Stil)
FROM Schriften
GROUP BY Schriftname
HAVING COUNT(Stil) > 2;
```

Beschreiben Sie in zwei bis drei Sätzen, welche Ausgabe diese Abfrage auf einer beliebigen **Schriften**-Tabelle liefert!

Aufgabe 11:

Im Zeitungslayout kombiniert man gerne zwei Schriften mit starkem Kontrast, typischerweise eine Serifenantiqua für den Mengentext mit einer Grotesk für die Überschriften.

Man gebe dazu eine SQL-Anfrage an, die aus der Tabelle **Schriftfamilien** in Aufgabe 5 alle die Schriftpaare (**Schriftname1**, **Schriftname2**) liefert, von denen Schriftname1 zu den Serifenschriften gehört und die andere serifenlos ist. Schriften mit einem Nullwert in der **Serife**-Spalte sollen nicht zu diesen Paaren gehören.

Aufgabe 12:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktion T_i :

Transaktion T_1 (10:00 Uhr)	Transaktion T_2 (10:05 Uhr)	Transaktion T_3 (10:10 Uhr)
Lesen A Schreiben A	Lesen A Lesen B Schreiben A Schreiben B	
Lesen C		Lesen B Lesen C
Schreiben C		Schreiben C Schreiben B

Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle ein. Markieren Sie oben die erste Operation, die zurückgewiesen wird.

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann ignoriere Schreiboperation

sonst Schreiben, $WTM(x) := TS$

**Klausur zur Vorlesung Datenbanken I
vom Wintersemester 2008/09**

Name: Vorname:

Matr.Nr.: Studiengang:

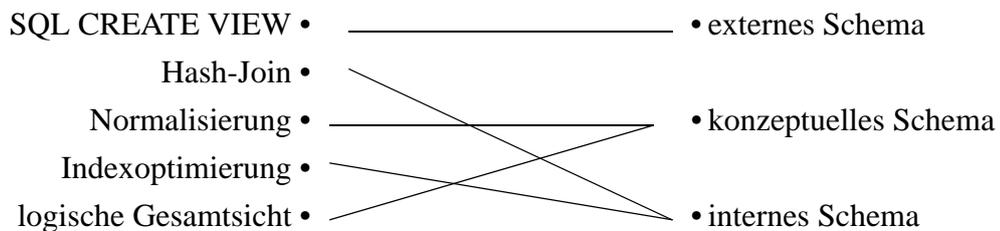
MUSTERLÖSUNG

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		7	5	
2	6		8	6	
3	2+2+2		9	4	
4	6		10	4	
5	3+3+1		11	5	
6	4		12	6	
			Summe	62	

Aufgabe 1:

Das ANSI/SPARC-Architekturmodell für Datenbanksysteme aus dem Jahr 1975 schlägt drei Schichten vor. Verbinden Sie durch Linien die fünf Datenbankbegriffe links mit jeweils einer der drei Schichten (Modelle, Schemata) rechts.



Schriftfamilien und einige ihrer Schnitte

Renaissance
Antiqua

Times Roman
Times Roman fett
Times Roman kursiv
Times Roman fettkursiv

Garamond
Garamond fett
Garamond kursiv
Garamond fettkursiv

Klassizistische
Antiqua

Bodoni
Bodoni condensed

Grotesk

Arial
Arial fett
Arial kursiv
Arial fettkursiv
Arial extrafett
Arial extrafettkursiv

Univers
Univers fett
Univers kursiv
Univers fettkursiv

Nicht-Proportional

Courier New
Courier New kursiv

Aufgabe 2:

Ein Verlag verwaltet *Schriften* in einer Datenbank. In der Welt der Schriften gelten die folgenden, hier etwas vereinfachten, Zusammenhänge (siehe auch die Tabelle in Aufgabe 3 und die Illustration auf der gegenüberliegenden Seite).

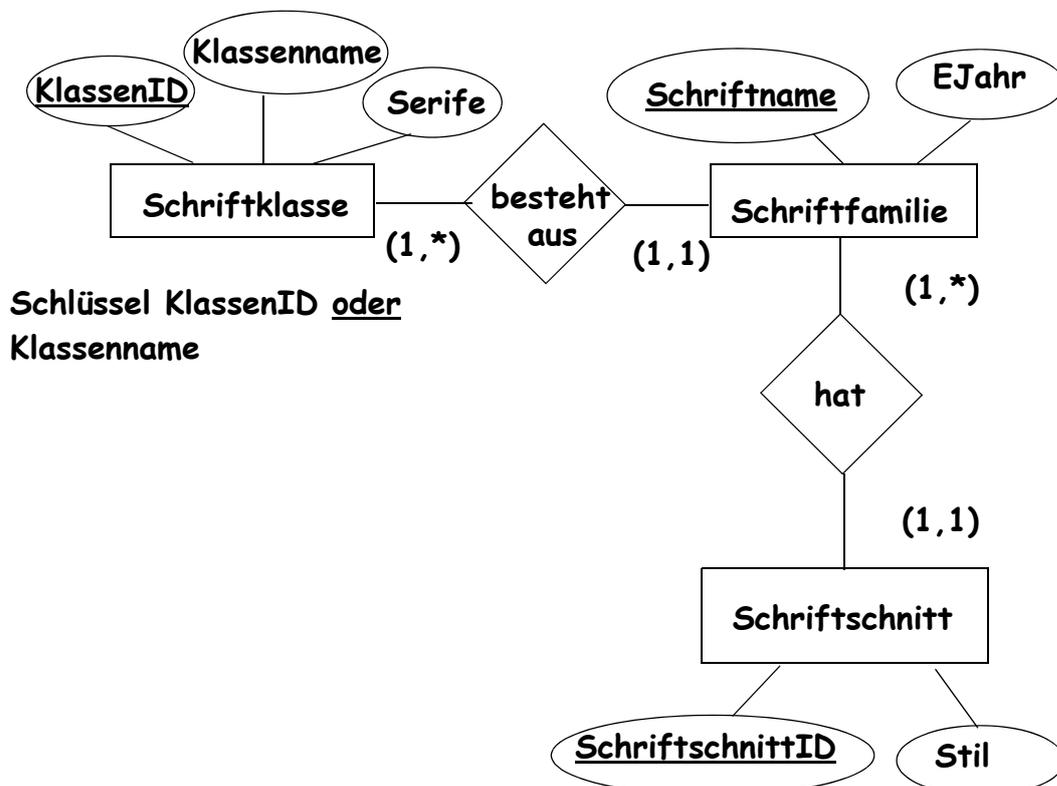
Es gibt **Schriftfamilien** und **Schriftschnitte**, auch Schriftstile genannt. Eine Schriftfamilie ist eindeutig durch ihren **Schriftnamen** (z. B. *Times Roman*, *MS Arial*, *Courier New*) identifiziert. Außerdem hat jede Schriftfamilie ein Entstehungsjahr **EJahr**.

Zu jeder Schriftfamilie gibt es mindestens einen, unter Umständen sehr viele Schriftschnitte, die wir mit einem eindeutigen abgekürzten **SchriftschnittID** versehen. Jeder Schriftschnitt gehört immer eindeutig zu einer Schriftfamilie. Jeder Schriftschnitt einer Schriftfamilie hat als Eigenschaft einen **Stil** (*normal*, *fett*, *kursiv*, *leicht*, *fettkursiv* usw.). Ferner werde kein Stil einer Schriftfamilie mit zwei Schnitten dargestellt (funktionale Abhängigkeit).

Schriftfamilien wiederum lassen sich klassifizieren, etwa nach der DIN 16518, in sogenannte **Schriftklassen**. Die bekanntesten sind *Renaissance Antiqua Schriften*, *Klassizistische Antiqua Schriften*, *Grotesk Schriften*, auch *Linear Schriften* genannt, *Nicht-proportionale Schriften*, *Schreibschriften*, *Zierschriften*. Eine Schriftklasse hat mindestens eine Schriftfamilie, die bekannten großen Klassen sogar einige hundert Schriftfamilien. Auch wenn die Zuordnung manchmal umstritten ist, behaupten wir, dass jede Schriftfamilie zu genau einer Schriftklasse gehört.

Eine Schriftklasse habe bei uns einen eindeutigen Klassenidentifizier **KlassenID**, einen eindeutigen Klassennamen **Klassenname** und ein Attribut **Serife**, das die Werte *mit*, *ohne* oder einen *Nullwert* für „nicht anwendbar“ annehmen kann, je nachdem ob es eine Schriftklasse für Schriften mit Serifen (Anstrichen, Füßchen) oder für serifenlose (Grotesk-)Schriften ist.

Geben Sie ein ER-Diagramm für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. **Unterstreichen Sie in jeder Entitätsklasse einen Schlüssel!**



Aufgabe 3:

Im ersten Entwurf einer Datenbank entsteht eine Tabelle **Schriften** mit einer Belegung wie unten gezeigt, die auch die Miniweltannahmen von Aufgabe 2 widerspiegelt.

Schriften

SchriftschnittID	Schriftname	Klassenname	Stil
TR	Times Roman	Renaissance Antiqua	normal
TRbd	Times Roman	Renaissance Antiqua	fett
TRit	Times Roman	Renaissance Antiqua	kursiv
TRbdit	Times Roman	Renaissance Antiqua	fettkursiv
GMOND	Garamond	Renaissance Antiqua	normal
GMONDbd	Garamond	Renaissance Antiqua	fett
GMONDit	Garamond	Renaissance Antiqua	kursiv
GMONDbdit	Garamond	Renaissance Antiqua	fettkursiv
BOD	Bodoni	Klassizistische Antiqua	normal
BODcnd	Bodoni	Klassizistische Antiqua	eng
ARIAL	MS Arial	Grotesk	normal
ARIALbd	MS Arial	Grotesk	fett
ARIALit	MS Arial	Grotesk	kursiv
ARIALbdit	MS Arial	Grotesk	fettkursiv
ARIALblack	MS Arial	Grotesk	extrafett
ARIALblackit	MS Arial	Grotesk	extrafettkursiv

- (a) Was sind die Schlüsselkandidaten?

{SchriftschnittID}, {Schriftname, Stil}

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

{SchriftschnittID} -> Schriften

{Schriftname, Stil} -> Schriften

{Schriftname} -> {Klassenname}

- (c) Eine Relation ist in der 2. Normalform (2NF), wenn jedes Attribut entweder prim oder **voll funktional** abhängig von jedem Schlüsselkandidat ist. Ist Schriften in 2NF? Begründung auf der Rückseite!

Schriften ist nicht in 2NF, weil Klassenname nicht prim ist und nicht voll funktional abhängig von jedem Schlüsselkandidaten, da {Schriftname} -> {Klassenname}

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform (3NF), wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Für *Schriftklassen* und den enthaltenen *Schriftfamilien* haben wir eine Relation **SK** entworfen. Ist **SK** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

SK

Schriftname	KlassenID	Klassenname
Times Roman	RA	Renaissance Antiqua
Garamond	RA	Renaissance Antiqua
Bodoni	KA	Klassizistische Antiqua
MS Arial	LG	Linear Grotesk
Helvetica	LG	Linear Grotesk
Univers	LG	Linear Grotesk

Schlüsselkandidaten: {Schriftname}

Nicht-triviale FA: {Schriftname} \rightarrow SK

{KlassenID} \rightarrow {Klassenname}

{Klassenname} \rightarrow {KlassenID}

SK ist nicht in 3NF, da {Klassenname} \rightarrow {KlassenID} nicht trivial, {Klassenname} kein Schlüssel und KlassenID nicht prim ist.

alternative Begründung:

{Schriftname} $\begin{matrix} \rightarrow \\ \leftarrow \end{matrix}$ {Klassenname} \rightarrow {KlassenID}

Dies ist eine transitive Abhängigkeit eines nicht-primen Attributs von einem Schlüssel. Also ist SK nicht in 3NF.

Aufgabe 5:

Für die folgenden Aufgaben haben wir zwei Relationen **Schriftklassen**(**KlassenID**, **Klassenname**) und **Schriftfamilien**(**Schriftname**, **KlassenID**, **Serife**, **EJahr**) gebildet. Beachten Sie: für jede Klasse ist eindeutig bestimmt, ob alle darin enthaltenen Schriften Serifen haben oder nicht, bzw. man legt durch einen Nullwert fest, dass dieses Kriterium nicht anwendbar ist.

Schriftklassen

Schriftfamilien

Klassen-ID	Klassenname	Schriftname	Klassen-ID	Serife	EJahr
RA	Renaissance Antiqua	Times Roman	RA	mit	1931
KA	Klassizistische Antiqua	Garamond	RA	mit	1530
LG	Linear Grotesk	Bodoni	KA	mit	1798
ZS	Zierschrift	MS Arial	LG	ohne	1989
SC	Schreibschrift	Helvetica	LG	ohne	1956
TT	Nicht-Proportional	Univers	LG	ohne	1956
SYM	Zeichen und Symbole				

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

Schriftklassen: Schlüsselkandidaten sind {KlassenID} und {Klassenname}

Nicht-triviale FA sind {KlassenID} → {Klassenname}

{Klassenname} → {KlassenID}

Schriftfamilien: Schlüsselkandidaten sind {Schriftname}

Nicht-triviale FA sind {Schriftname} → Schriftfamilien

{KlassenID} → {Serife}

(b) Sind die Relationen **Schriftklassen** und **Schriftfamilien** in 3NF? Wenn ja, sogar in BCNF? Begründung!

Schriftklassen ist in 3NF und in BCNF, da Relationen mit nur 2 Attributen stets in 3NF und BCNF sind.

Schriftfamilien ist nicht in 3NF (und damit auch nicht in BCNF). 3NF wird verletzt durch nicht-triviale FA {KlassenID} → {Serife}, {KlassenID} ist kein Schlüssel und Serife nicht prim.

(c) Welches Attribut ist offensichtlich in der falschen Relation?

Serife sollte nach Schriftklassen verschoben werden (dann beide in BCNF).

Aufgabe 6:

Zur Einstimmung auf SQL eine einfache Abfrage: Wie lautet die SQL-Abfrage auf der Tabelle **Schriften** aus Aufgabe 3, die uns alle Namen der Schriftfamilien liefert, für die ein Schnitt mit Stil „fett“ vorhanden ist.

```
SELECT Schriftname
FROM Schriften
WHERE Stil = 'fett'
```

Aufgabe 7:

Wie lautet die gleichwertige SQL-Abfrage zum folgenden Ausdruck der relationalen Algebra mit den Relationen aus Aufgabe 5?

π Schriftname, Klassenname (σ Serife='mit' (Schriftklassen \bowtie Schriftfamilien))

*auch andere Varianten
möglich*

```
SELECT Schriftname, Klassenname
FROM Schriftklassen, Schriftfamilien
WHERE Schriftklassen.KlassenID = Schriftfamilien.KlassenID AND
      Serife = 'mit'
```

Aufgabe 8:

Wenn wie oben **Schriftfamilien(Schriftname, KlassenID, Serife, EJahr)** und **Schriftklassen(KlassenID, Klassenname)** gegeben sind, wie vereinbart man dann eine Relation **ModerneKlasse(KName)** als Sicht (view), die nur die Klassen aufführt (**jeweils einmal**), die Schriften enthalten, die gemäß **EJahr**-Wert nach 1900 entstanden sind.

```
CREATE VIEW ModerneKlasse(KName) AS
  SELECT DISTINCT Klassenname
  FROM Schriftenklassen, Schriftenfamilien
  WHERE Schriftenklassen.KlassenID = Schriftenfamilien.KlassenID
        AND EJahr > 1900
```

*hier auch AS KName
statt (KName) darüber*

Aufgabe 9:

In der Tabelle **Schriften** aus Aufgabe 3 soll der Name der Schriftfamilie *Times Roman* in *Times New Roman* geändert werden. Geben Sie das entsprechende SQL-Kommando dazu an.

```
UPDATE Schriften
  SET Schriftname = 'Times New Roman'
  WHERE Schriftname = 'Times Roman'
```

Aufgabe 10:

Betrachten Sie die folgende Abfrage:

```
SELECT Schriftname, COUNT(Stil)
FROM Schriften
GROUP BY Schriftname
HAVING COUNT(Stil) > 2;
```

Beschreiben Sie in zwei bis drei Sätzen, welche Ausgabe diese Abfrage auf einer beliebigen **Schriften**-Tabelle liefert!

Die Ausgabe ist eine zweispaltige Tabelle mit Namen von Schriftfamilien und Anzahl der Stile (Schnitte) dieser Familie für alle die Familien, die mehr als 2 Stile besitzen. In unserem Beispiel wären das

Schriftname	COUNT(Stil)
Times Roman	4
MS Arial	6
Garamond	4

Die Angabe einer Tabelle war nicht verlangt

Aufgabe 11:

Im Zeitungslayout kombiniert man gerne zwei Schriften mit starkem Kontrast, typischerweise eine Serifenantiqua für den Mengentext mit einer Grotesk für die Überschriften.

Man gebe dazu eine SQL-Anfrage an, die aus der Tabelle **Schriftfamilien** in Aufgabe 5 alle die Schriftpaare (**Schriftname1**, **Schriftname2**) liefert, von denen Schriftname1 zu den Serifenschriften gehört und die andere serifenlos ist. Schriften mit einem Nullwert in der **Serife**-Spalte sollen nicht zu diesen Paaren gehören.

```
SELECT s1.Schriftname, s2.Schriftname
FROM Schriftfamilien s1, Schriftfamilien s2
WHERE s1.Serife = 'mit' AND s2.Serife = 'ohne'
```

Aufgabe 12:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktion T_i :

Transaktion T_1 (10:00 Uhr)	Transaktion T_2 (10:05 Uhr)	Transaktion T_3 (10:10 Uhr)
Lesen A Schreiben A	Lesen A Lesen B Schreiben A Schreiben B	Lesen B Lesen C
Lesen C		Schreiben C Schreiben B
Schreiben C ← Auslöser		

Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle ein. Markieren Sie oben die erste Operation, die zurückgewiesen wird.

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)
10:00 10:05	10:00 10:05	10:05 10:10	10:05 10:10	10:10 Konflikt	10:10

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann ignoriere Schreiboperation

sonst Schreiben, $WTM(x) := TS$

Klausur zur Vorlesung Datenbanken I im Sommersemester 2009

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		8	4	
2	6		9	4	
3	2+2+2+2		10	4	
4	4		11	3	
5	2+2+2		12	5	
6	4		Summe	55	
7	4				

Aufgabe 1:

Welche der folgenden Aussagen sind richtig im Zusammenhang mit dem im Skript besprochenen Record Identifier/Tuple Identifier (RID/TID) Konzept bei Datenbanken?

- () Aus einem RID lässt sich direkt eine Seitennummer ablesen.
- () Muss ein Speicherobjekt umziehen, kommt in die Seite mit dem ursprünglichen RID ein Verweis auf die neue Seite.
- () Durch mehrfachen Umzug eines Speicherobjekts können lange Verweisketten entstehen, daher ist eine Reorganisation von Zeit zu Zeit nötig.
- () Durch Zusammenschieben der Speicherobjekte in einer Seite kann man erreichen, dass der freie Platz in einer Seite zusammenhängend ist, ohne dass sich RIDs ändern.
- () In jeder Seite werden die Speicherobjekte mittels eines B-Baums geordnet.

Aufgabe 2:

Weil wir uns nicht einigen konnten, ob Schweinegrippe, Finanzkrise oder die Bundestagswahl das Klausurthema sind, kommt hier eine ganz phantasielose Miniwelt.

Es geht um **Geräte** und um **Räume** in unserem Fachgebiet. Jedes Gerät (z. B. PCs, Monitore, Notebooks, Drucker, Beamer) hat bei uns eine eindeutige Inventarnummer (**InvNr**), eine Kurzbezeichnung (**KurzBez**), ein Anschaffungsjahr (**AJahr**). Räume haben eine eindeutige Raumnummer (**RNr**), eine Raumbezeichnung (**RBez**), z. B. Büro, Labor, Materialraum, und eine vorgesehene **Belegung**, z. B. WB, Prof, Studenten, Sekretärin.

Die Inventarisierungsvorschriften verlangen, dass jedes Gerät in einem Raum **steht** (natürlich nur in einem), auch wenn tatsächlich Notebooks und portable Beamer hin- und hergetragen werden. Klar ist auch, dass in einem Raum auch mehrere Geräte stehen dürfen, ggf. auch gar keines, etwa wenn der Raum ein Flur ist. In letzterem Fall ist für die Belegung dann auch ein Nullwert eingetragen.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung) für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Schlüssel.

Aufgabe 3:

Im ersten Entwurf der Inventar-Datenbank wurde eine Tabelle **Inventar**, ohne das Attribut **Belegung**, vorgesehen. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

Inventar

InvNr	KurzBez	AJahr	RNr	RBez
6711	PC FS	2008	1304	Büro groß
8719	Monitor FS LCD 26"	2009	1304	Büro groß
5567	Notebook IBM	2008	1304	Büro groß
4490	PC FS	2006	1308	Büro einzel
4498	Monitor Eizo LCD 19"	2006	1308	Büro einzel
4551	Tablet PC HP	2007	1310	Büro einzel
5201	Monitor Eizo LCD 19"	2007	1307	Studentenlabor
5202	Monitor Eizo LCD 19"	2007	1307	Studentenlabor
...
5208	Monitor Eizo LCD 19"	2007	1307	Studentenlabor
5209	Deckenbeamer Sanyo	2007	1307	Studentenlabor

- (a) Was sind die Schlüsselkandidaten?

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

- (c) Gibt es eine transitive Abhängigkeit in **Inventar**? Wenn ja, welche?

- (d) Ist **Inventar** in 3NF?

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Geräte und **Räume** haben wir jetzt getrennt. Die Raumnummer haben wir als Fremdschlüssel in **Geraete** aufgenommen. Allerdings hat ein wohlmeinender Datenbankadministrator die eigentlich schon aus der Kurzbezeichnung ersichtliche Geräteart als Attribut **Kategorie** eingeführt, das angibt, ob ein Gerät zu den Monitoren, PCs, Notebooks oder Beamern gehört. Damit lassen sich z. B. Abfragen der Art „Welche Monitore wurden vor 2008 beschafft?“ beantworten.

Geraete

InvNr	KurzBez	Kategorie	AJahr	RNr
6711	PC FS	PC	2008	1304
8719	Monitor FS LCD 26“	MON	2009	1304
5567	Notebook IBM 15“	NB	2008	1304
5730	Notebook Sony Vaio 11“	NB	2008	1304
4490	PC FS	PC	2006	1308
4498	Monitor Eizo LCD 19“	MON	2006	1308
4551	Tablet PC HP	NB	2007	1310
5201	Monitor Eizo LCD 19“	MON	2007	1307
5202	Monitor Eizo LCD 19“	MON	2007	1307
...
5208	Monitor Eizo LCD 19“	MON	2007	1307
5209	Deckenbeamer Sanyo	BEAM	2007	1307

Ist **Geraete** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Aufgabe 5:

Für die folgenden Aufgaben haben wir Räume und deren Zuordnung zu Fachgebieten erfaßt. Wir nehmen an, daß jeder Raum zu genau einem Fachgebiet gehört. Dazu wurden in einer Relation **RBF** die Raumnummer, die Raumbezeichnung und die Zuordnung zu dem Fachgebiet festgehalten. In einer zweiten Relation **FN** haben wir die Fachgebietsabkürzung mit zugehörigem vollen Namen gespeichert.

RBF

FN

RNr	Bezeichnung	FGID
1304	Büro groß	DB
1308	Büro einzel	DB
1307	Studentenlabor	DB
1314	Flur	DB
1310	Büro einzel	DB
1343	Büro einzel	SE
1342	Büro groß	SE
0435	Seminarraum	WV
0439	Büro groß	WV
1315	Büro einzel	NN
...

FGID	NameFachgebiet
DB	Datenbanken
WV	Wissenverarbeitung
NN	Neuronale Netze
SE	Software Engineering
...	...

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

(b) Sind die Relationen **RBF** und **FN** in BCNF? Kurze Begründung!

(c) Die oben genannten Relationen **RBF** und **FN** entstanden durch Aufteilung der Relation **RBFN**, die Raumnummer, Bezeichnung, FGID und Fachgebietsname enthielt. Ist diese Aufteilung verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

Aufgabe 6:

Mit Hilfe der Relationen **Geraete**, **RBF** und **FN** sollen die Namen aller Fachgebiete ermittelt werden, die einen Beamer (Kategorie BEAM) haben. Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und zwei Joins.

Aufgabe 7:

Wie lautet die SQL-Abfrage, mit der man aus **RBF** und **FN** die Namen der Fachgebiete mit der Anzahl der ihnen zugeordneten Räume erhält?

Aufgabe 8:

Mit welchem SQL-Befehl erstellt man aus der Relation **Geraete** in Aufgabe 4 den View **MPneu**, der alle Monitore und PCs liefert, die 2007 oder danach angeschafft wurden. Im View sollen **Inventarnummer**, **Kurzbezeichnung** und **Anschaffungsjahr** erscheinen.

Hinweis: **AJahr** sei als Integer deklariert.

Aufgabe 9:

Wir suchen alle Raumnummern von Räumen, in denen keine Geräte stehen. Dazu verwenden wir in der Abfrage die Tabellen **Geraete** und **RBF**. Hinweis: Wir brauchen **RBF**, weil dort auch Räume genannt sind, in denen keine Geräte stehen. Wie lautet die SQL-Abfrage?

Aufgabe 10:

Betrachten Sie die folgende Abfrage:

```
SELECT AJahr, COUNT(*)
FROM Geraete
GROUP BY AJahr
HAVING COUNT(*) > (SELECT AVG(COUNT(*)) FROM Geraete GROUP BY AJahr);
```

Beschreiben Sie kurz, welche Ausgabe diese Abfrage auf einer beliebigen **Geraete**-Tabelle liefert!

Aufgabe 11:

Für **Geräte** betrachten wir zwei Spezialisierungen, „ausgemustertes Gerät“ und „Drittmittelgerät“. Es gilt die Annahme, dass ein nicht mehr brauchbares Gerät zu einem bestimmten Datum auf Antrag bei der Verwaltung ausgemustert wird. Zweitens kann ein Gerät aus Drittmitteln beschafft worden sein, dann gibt es genau einen Drittmittelgeber (DFG, EU, Industrie, ...). Alle anderen Angaben sind davon unbetroffen. Welche der folgenden Aussagen sind grundsätzlich richtig?

Die zwei Spezialisierungen können ...

- () als zusätzliche Attribute (**Ausgemustert-seit, Drittmittelgeber**) in der **Geraete**-Tabelle behandelt werden mit einem Nullwert für noch nicht ausgemusterte Geräte oder Geräte aus Landesmitteln (Nichtdrittmittelgeräte).
- () mit zwei zweiseitigen Tabellen **Ausgemustert(InvNr, seit)** und **Drittmittelgeraete(InvNr, Drittmittelgeber)** oder einer dreispaltigen Tabelle **AD(InvNr, ausgemustert_seit, Drittmittelgeber)** behandelt werden, wobei **InvNr** Fremdschlüssel in die **Geraete**-Tabelle hinein ist.
- () mit vier disjunkten Tabellen, **GeraeteAktivNichtDrittmittel**, **GeraeteAusgemustertNichtDrittmittel**, **GeraeteAktivDrittmittel**, **GeraeteAusgemustertDrittmittel** behandelt werden, jeweils mit passenden zusätzlichen Attributen.
- () so nicht behandelt werden, Spezialisierungen und Vererbung geht nur mit objekt-orientierten Datenbanken.

Aufgabe 12:

Betrachten Sie die folgende Ablauffolge:

Transaktion 1	Transaktion 2	Transaktion 3	Transaktion 4
Lesen A Schreiben A			
Lesen B Schreiben B		Lesen A	
	Lesen C Lesen D		
		Lesen B Schreiben B	
		Lesen D Schreiben D	Lesen C
	Schreiben C		
			Lesen D Schreiben D

Geben Sie den vollständigen Abhängigkeitsgraphen an! An jeder gerichteten Kante soll der Datensatz stehen (A, B, C oder D), der den Konflikt verursacht. Ist die Ablauffolge serialisierbar? Wenn nein, welche Transaktionen stehen in Konflikt?

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Sommersemester 2009**

Musterlösung

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		8	4	
2	6		9	4	
3	2+2+2+2		10	4	
4	4		11	3	
5	2+2+2		12	5	
6	4		Summe	55	
7	4				

Aufgabe 1:

Welche der folgenden Aussagen sind richtig im Zusammenhang mit dem im Skript besprochenen Record Identifier/Tuple Identifier (RID/TID) Konzept bei Datenbanken?

- () Aus einem RID läßt sich direkt eine Seitennummer ablesen.
- () Muss ein Speicherobjekt umziehen, kommt in die Seite mit dem ursprünglichen RID ein Verweis auf die neue Seite.
- () Durch mehrfachen Umzug eines Speicherobjekts können lange Verweisketten entstehen, daher ist eine Reorganisation von Zeit zu Zeit nötig.
- () Durch Zusammenschieben der Speicherobjekte in einer Seite kann man erreichen, dass der freie Platz in einer Seite zusammenhängend ist, ohne dass sich RIDs ändern.
- () In jeder Seite werden die Speicherobjekte mittels eines B-Baums geordnet.

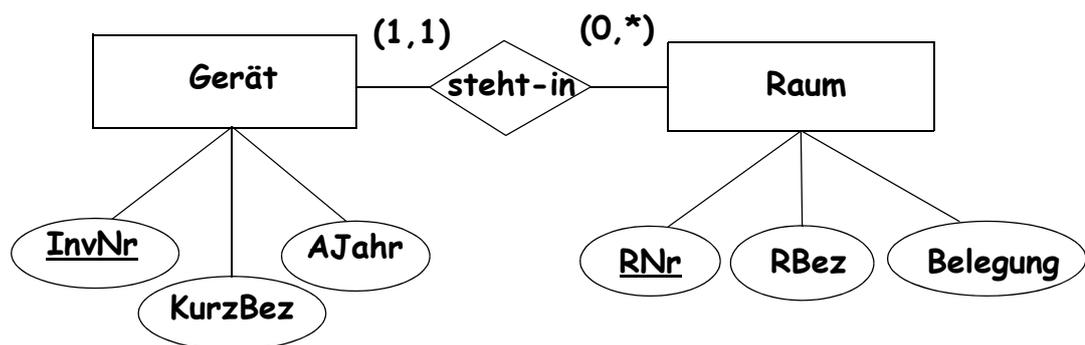
Aufgabe 2:

Weil wir uns nicht einigen konnten, ob Schweinegrippe, Finanzkrise oder die Bundestagswahl das Klausurthema sind, kommt hier eine ganz phantasielose Miniwelt.

Es geht um **Geräte** und um **Räume** in unserem Fachgebiet. Jedes Gerät (z. B. PCs, Monitore, Notebooks, Drucker, Beamer) hat bei uns eine eindeutige Inventarnummer (**InvNr**), eine Kurzbezeichnung (**KurzBez**), ein Anschaffungsjahr (**AJahr**). Räume haben eine eindeutige Raumnummer (**RNr**), eine Raumbezeichnung (**RBez**), z. B. Büro, Labor, Materialraum, und eine vorgesehene **Belegung**, z. B. WB, Prof, Studenten, Sekretärin.

Die Inventarisierungsvorschriften verlangen, dass jedes Gerät in einem Raum **steht** (natürlich nur in einem), auch wenn tatsächlich Notebooks und portable Beamer hin- und hergetragen werden. Klar ist auch, dass in einem Raum auch mehrere Geräte stehen dürfen, ggf. auch gar keines, etwa wenn der Raum ein Flur ist. In letzterem Fall ist für die Belegung dann auch ein Nullwert eingetragen.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung) für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Schlüssel.



Aufgabe 3:

Im ersten Entwurf der Inventar-Datenbank wurde eine Tabelle **Inventar**, ohne das Attribut **Belegung**, vorgesehen. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

Inventar

InvNr	KurzBez	AJahr	RNr	RBez
6711	PC FS	2008	1304	Büro groß
8719	Monitor FS LCD 26"	2009	1304	Büro groß
5567	Notebook IBM	2008	1304	Büro groß
4490	PC FS	2006	1308	Büro einzel
4498	Monitor Eizo LCD 19"	2006	1308	Büro einzel
4551	Tablet PC HP	2007	1310	Büro einzel
5201	Monitor Eizo LCD 19"	2007	1307	Studentenlabor
5202	Monitor Eizo LCD 19"	2007	1307	Studentenlabor
...
5208	Monitor Eizo LCD 19"	2007	1307	Studentenlabor
5209	Deckenbeamer Sanyo	2007	1307	Studentenlabor

- (a) Was sind die Schlüsselkandidaten?

{InvNr}

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

InvNr -> Inventar (Schlüsselabhängigkeit)

RNr -> RBez

- (c) Gibt es eine transitive Abhängigkeit in **Inventar**? Wenn ja, welche?

Ja: InvNr ----> RNr ----> RBez

<-/-

- (d) Ist **Inventar** in 3NF?

Nein, da die in (c) genannte Abhängigkeit eine transitive Abhängigkeit eines nicht primen Attributs von einem Schlüssel ist.

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Geräte und **Räume** haben wir jetzt getrennt. Die Raumnummer haben wir als Fremdschlüssel in **Geraete** aufgenommen. Allerdings hat ein wohlmeinender Datenbankadministrator die eigentlich schon aus der Kurzbezeichnung ersichtliche Geräteart als Attribut **Kategorie** eingeführt, das angibt, ob ein Gerät zu den Monitoren, PCs, Notebooks oder Beamern gehört. Damit lassen sich z. B. Abfragen der Art „Welche Monitore wurden vor 2008 beschafft?“ beantworten.

Geraete

InvNr	KurzBez	Kategorie	AJahr	RNr
6711	PC FS	PC	2008	1304
8719	Monitor FS LCD 26“	MON	2009	1304
5567	Notebook IBM 15“	NB	2008	1304
5730	Notebook Sony Vaio 11“	NB	2008	1304
4490	PC FS	PC	2006	1308
4498	Monitor Eizo LCD 19“	MON	2006	1308
4551	Tablet PC HP	NB	2007	1310
5201	Monitor Eizo LCD 19“	MON	2007	1307
5202	Monitor Eizo LCD 19“	MON	2007	1307
...
5208	Monitor Eizo LCD 19“	MON	2007	1307
5209	Deckenbeamer Sanyo	BEAM	2007	1307

Ist **Geraete** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Schlüsselkandidaten: {InvNr}

Nicht-triviale fkt. Abh.: InvNr ---> Geraete (Schlüsselabhängigkeit)

KurzBez ---> Kategorie

Geraete ist nicht in 3NF, da KurzBez ---> Kategorie nicht trivial, KurzBez kein Schlüssel und Kategorie nicht prim ist.

**Alternative Begründung: Es gibt eine transitive Abhängigkeit
InvNr ---> KurzBez ---> Kategorie eines nicht-primen Attributs von einem Schlüssel.**

Aufgabe 5:

Für die folgenden Aufgaben haben wir Räume und deren Zuordnung zu Fachgebieten erfaßt. Wir nehmen an, daß jeder Raum zu genau einem Fachgebiet gehört. Dazu wurden in einer Relation **RBF** die Raumnummer, die Raumbezeichnung und die Zuordnung zu dem Fachgebiet festgehalten. In einer zweiten Relation **FN** haben wir die Fachgebietsabkürzung mit zugehörigem vollen Namen gespeichert.

RBF

RNr	Bezeichnung	FGID
1304	Büro groß	DB
1308	Büro einzel	DB
1307	Studentenlabor	DB
1314	Flur	DB
1310	Büro einzel	DB
1343	Büro einzel	SE
1342	Büro groß	SE
0435	Seminarraum	WV
0439	Büro groß	WV
1315	Büro einzel	NN
...

FN

FGID	NameFachgebiet
DB	Datenbanken
WV	Wissenverarbeitung
NN	Neuronale Netze
SE	Software Engineering
...	...

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

RBF: Schlüsselkandidaten nur {RNr}

Nicht-tr. fkt. Abhängigkeiten: RNr ---> RBF (Schlüsselabhängigkeit)

FN: Schlüsselkandidaten {FGID}, {NameFachgebiet}

**Nicht-tr. fkt. Abhängigkeiten: FGID ---> FN,
NameFachgebiet ---> FN**

beide Schlüsselabhängigkeiten

(b) Sind die Relationen **RBF** und **FN** in BCNF? Kurze Begründung!

Ja, beide Relationen sind in BCNF, da alle nicht-trivialen Abhängigkeiten von einem Schlüssel ausgehen. Eine alternative Begründung bei FN wäre die Tatsache, dass FN nur zwei Attribute besitzt und damit automatisch in BCNF ist.

(c) Die oben genannten Relationen **RBF** und **FN** entstanden durch Aufteilung der Relation **RBFN**, die Raumnummer, Bezeichnung, FGID und Fachgebietsname enthielt. Ist diese Aufteilung verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

Die Aufteilung von RBFN in RBF und FN ist verlustfrei, denn

$R_1 \cap R_2$ ergibt {FGID} und FGID \rightarrow FN, also R_2 .

Aufgabe 6:

Mit Hilfe der Relationen **Geraete**, **RBF** und **FN** sollen die Namen aller Fachgebiete ermittelt werden, die einen Beamer (Kategorie BEAM) haben. Geben Sie den Ausdruck der Relationenalgebra an. Sie brauchen Projektion, Selektion und zwei Joins.

$\Pi_{\text{NameFachgebiet}}(\sigma_{\text{Kategorie}='BEAM'}(\text{Geraete}) \bowtie \text{RBF} \bowtie \text{FN})$

Aufgabe 7:

Wie lautet die SQL-Abfrage, mit der man aus **RBF** und **FN** die Namen der Fachgebiete mit der Anzahl der ihnen zugeordneten Räume erhält?

```
SELECT NameFachgebiet, COUNT(*)
FROM RBF, FN
WHERE RBF.FGID = FN.FGID
GROUP BY NameFachgebiet;
```

Auch alternative Formulierungen des Joins möglich: NATURAL JOIN, JOIN .. ON, ...

Aufgabe 8:

Mit welchem SQL-Befehl erstellt man aus der Relation **Geraete** in Aufgabe 4 den View **MPneu**, der alle Monitore und PCs liefert, die 2007 oder danach angeschafft wurden. Im View sollen **Inventarnummer**, **Kurzbezeichnung** und **Anschaffungsjahr** erscheinen.

Hinweis: **AJahr** sei als Integer deklariert.

```
CREATE VIEW MPneu (Inventarnummer, Kurzbezeichnung,
                  Anschaffungsjahr) AS

SELECT InvNr, KurzBez, AJahr
FROM Geraete
WHERE (Kategorie = 'PC' OR Kategorie = 'MON') AND AJahr >= 2007;
```

Aufgabe 9:

Wir suchen alle Raumnummern von Räumen, in denen keine Geräte stehen. Dazu verwenden wir in der Abfrage die Tabellen **Geraete** und **RBF**. Hinweis: Wir brauchen **RBF**, weil dort auch Räume genannt sind, in denen keine Geräte stehen. Wie lautet die SQL-Abfrage?

```
SELECT RNr
FROM RBF
WHERE RNr NOT IN (SELECT RNr FROM Geraete);

alternative Formulierungen möglich, etwa
SELECT RNr
FROM RBF r
WHERE NOT EXISTS (SELECT * FROM Geraete g
                  WHERE g.RNr = r.RNr);

oder als left outer join mit
SELECT r.RNr
FROM RBF r LEFT OUTER JOIN Geraete g ON r.RNr = g.RNr
WHERE g.RNr IS NULL;
```

Aufgabe 10:

Betrachten Sie die folgende Abfrage:

```
SELECT AJahr, COUNT(*)
FROM Geraete
GROUP BY AJahr
HAVING COUNT(*) > (SELECT AVG(COUNT(*)) FROM Geraete GROUP BY AJahr);
```

Beschreiben Sie kurz, welche Ausgabe diese Abfrage auf einer beliebigen **Geraete**-Tabelle liefert!

Das Ergebnis ist eine zweispaltige Tabelle mit Anschaffungsjahr und der Anzahl der in diesem Jahr beschafften Geräte. Es erscheinen aber nur die Jahre, in denen mehr Geräte als im Durchschnitt aller Jahre angeschafft wurden.

Aufgabe 11:

Für **Geräte** betrachten wir zwei Spezialisierungen, „ausgemustertes Gerät“ und „Drittmittelgerät“. Es gilt die Annahme, dass ein nicht mehr brauchbares Gerät zu einem bestimmten Datum auf Antrag bei der Verwaltung ausgemustert wird. Zweitens kann ein Gerät aus Drittmitteln beschafft worden sein, dann gibt es genau einen Drittmittelgeber (DFG, EU, Industrie, ...). Alle anderen Angaben sind davon unbetroffen. Welche der folgenden Aussagen sind grundsätzlich richtig?

Die zwei Spezialisierungen können ...

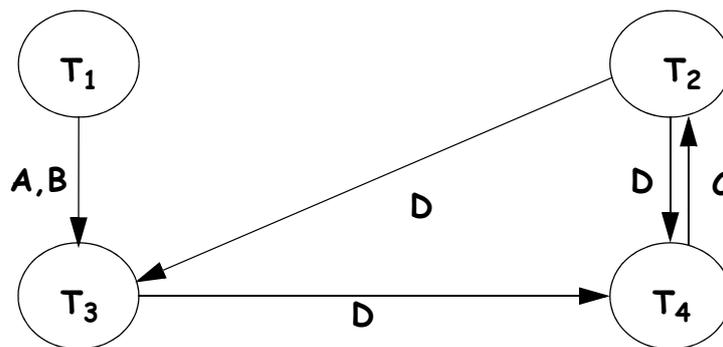
- () als zusätzliche Attribute (**Ausgemustert-seit, Drittmittelgeber**) in der **Geraete**-Tabelle behandelt werden mit einem Nullwert für noch nicht ausgemusterte Geräte oder Geräte aus Landesmitteln (Nichtdrittmittelgeräte).
- () mit zwei zweispaltigen Tabellen **Ausgemustert(InvNr, seit)** und **Drittmittelgeraete(InvNr, Drittmittelgeber)** oder einer dreispaltigen Tabelle **AD(InvNr, ausgemustert_seit, Drittmittelgeber)** behandelt werden, wobei **InvNr** Fremdschlüssel in die **Geraete**-Tabelle hinein ist.
- () mit vier disjunkten Tabellen, **GeraeteAktivNichtDrittmittel, GeraeteAusgemustert-NichtDrittmittel, GeraeteAktivDrittmittel, GeraeteAusgemustertDrittmittel** behandelt werden, jeweils mit passenden zusätzlichen Attributen.
- () so nicht behandelt werden, Spezialisierungen und Vererbung geht nur mit objekt-orientierten Datenbanken.

Aufgabe 12:

Betrachten Sie die folgende Ablauffolge:

Transaktion 1	Transaktion 2	Transaktion 3	Transaktion 4
Lesen A Schreiben A			
Lesen B Schreiben B		Lesen A	
	Lesen C Lesen D	Lesen B Schreiben B	
		Lesen D Schreiben D	Lesen C
	Schreiben C		
			Lesen D Schreiben D

Geben Sie den vollständigen Abhängigkeitsgraphen an! An jeder gerichteten Kante soll der Datensatz stehen (A, B, C oder D), der den Konflikt verursacht. Ist die Ablauffolge serialisierbar? Wenn nein, welche Transaktionen stehen in Konflikt?



Es sind zwei Zyklen vorhanden, damit ist die Ablauffolge nicht serialisierbar. Die Transaktionen 2 und 4 sowie 2, 3 und 4 stehen in Konflikt.

ENDE DER KLAUSUR

Klausur zur Vorlesung Datenbanken I im Wintersemester 2009/10

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		7	4	
2	6		8	4	
3	1+1+2+2		9	4	
4	2+2+2		10	4	
5	2+2+2+2		11	4	
6	4		Summe	53	

Aufgabe 1:

Der Isolationsgrad eines Transaktionssystems verläuft vom Grad 0 (Chaos) bis zum höchsten Grad 3 (vollständige Isolation). Tragen Sie für jede der unten genannten Eigenschaften des Transaktionssystems den passenden Grad (0 bis 3) in die Klammern [] ein.

Hinweis: Jeder Grad taucht genau einmal auf.

- [] READ COMMITTED, Cursorstabilität, kein Verlust von Updates.
- [] READ UNCOMMITTED, nur Schreibsperrern, diese bis EOT, liest u. U. schmutzige Daten.
- [] SERIALIZABLE, Lesezugriffe wiederholbar.
- [] Daten nach Schreibzugriff sofort sichtbar, nur Schreibsperrern, Update kann verloren gehen.

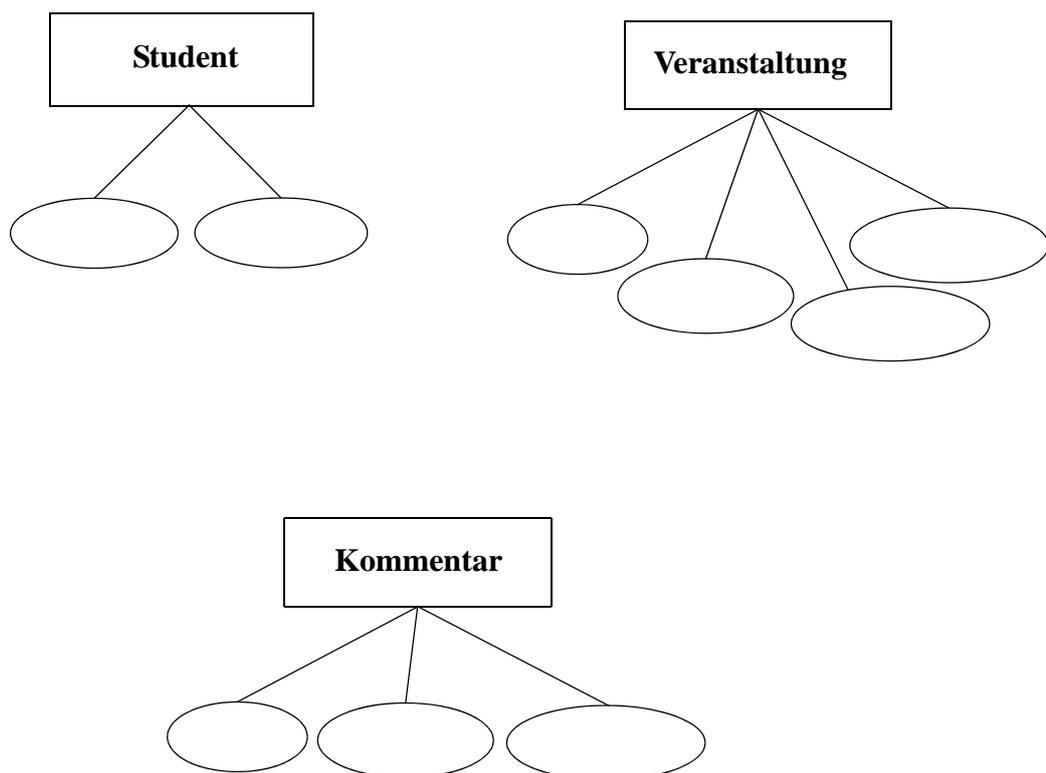
Aufgabe 2:

Bewertungsportale sind derzeit (noch) groß in Mode. Als Miniwelt betrachten wir hier deshalb ein fiktives Portal zur Bewertung unserer Veranstaltungen.

Veranstaltungen haben einen eindeutigen **Veranstaltungsidentifizier (VID)**, ferner **Titel**, **Dozent** und **Semester**, in dem die Veranstaltung stattfand. **Studenten besuchen Veranstaltungen**; wir nehmen hier an, mindestens eine und in der Regel natürlich mehrere. Veranstaltungen dagegen brauchen mindestens drei Hörer. Jeder Student hat ein **Pseudonym** (frei gewählter Spitzname **PName**), der nur einmal vergeben wird. Ferner muss er seine **E-Mail-Adresse** angeben, die es auch nur einmal geben darf und die bei den Kommentaren nicht angezeigt wird.

Studenten können dann **Kommentare verfassen**. Jeder Kommentar wird von genau einem Studenten verfasst und **betrifft** genau eine Veranstaltung. Ein Kommentar habe einen eindeutigen **Kommentaridentifizier (KID)**, eine **Note** (1 = sehr gut, 5 = mangelhaft) und einen **Text (KText)** mit bis zu 1000 Zeichen. Klar ist auch, dass ein Student viele Kommentare abgeben kann, nicht jeder Student kommentieren möchte, nicht jede Veranstaltung kommentiert wird, dass eine Veranstaltung aber häufig viele Kommentare hat.

Vervollständigen Sie das ER-Diagramm unten (ohne Spezialisierung/Generalisierung) für diese Miniwelt. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.



Aufgabe 3:

Im ersten Entwurf der Bewertungsdatenbank wurde eine Tabelle **Bewertung** vorgesehen. Allerdings haben die Kommentare keinen Kommentaridentifizier. Jeder Verfasser kann zu jeder Veranstaltung höchstens einen Kommentar abgeben. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

Bewertung

PName	VID	Note	KText
Goldie	DB1	3	Prof nervt
Goldie	XML	4	Klausur geschenkt
Bullie	DB1	2	Uebungen toll
Bullie	Unix	3	geht so
Queen	XML	1	Spitze
Queen	Unix	2	Augen zu und durch
Rex123	DB1	3	langweilig
Rex123	XML	4	langweilig
Rex123	Tcl/Tk	2	kurzweilig

- (a) Was sind die Schlüsselkandidaten?

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

- (c) Gibt es eine transitive Abhängigkeit in **Bewertung**? Wenn ja, welche?

- (d) Ist **Bewertung** in 3NF? Begründung!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In die Tabelle **Bewertung** nehmen wir noch die eindeutigen E-Mail-Adressen hinzu und erhalten dadurch die Tabelle **Kommentare**.

Kommentare

PName	E-Mail	VID	Note	KText
Goldie	gabi@uni-kassel.de	DB1	3	Prof nervt
Goldie	gabi@uni-kassel.de	XML	4	Klausur geschenkt
Bullie	bernd@uni-kassel.de	DB1	2	Uebungen toll
Bullie	bernd@uni-kassel.de	Unix	3	geht so
Queen	anne@uni-kassel.de	XML	1	Spitze
Queen	anne@uni-kassel.de	Unix	2	Augen zu und durch
Rex123	rolf@uni-kassel.de	DB1	3	langweilig
Rex123	rolf@uni-kassel.de	XML	4	langweilig
Rex123	rolf@uni-kassel.de	Tcl/Tk	2	kurzweilig

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

(b) Ist **Kommentare** in 3NF? Begründung!

(c) Ist **Kommentare** in BCNF? Begründung!

Aufgabe 5:

Für die folgenden Aufgaben betrachten wir neben der **Bewertung**-Tabelle aus Aufgabe 3 noch eine **Votum**-Tabelle. Dort wird festgehalten, wer (Abstimmernamen **AName**) positiv oder negativ über einen Kommentar abgestimmt hat. Jeder darf nur einmal zu einem Kommentar ein **Votum** abgeben.

Bewertung

PName	VID	Note	KText
Goldie	DB1	3	Prof nervt
Goldie	XML	4	Klausur geschenkt
Bullie	DB1	2	Uebungen toll
Bullie	Unix	3	geht so
Queen	XML	1	Spitze
Queen	Unix	2	Augen zu
Rex123	DB1	3	langweilig
Rex123	XML	4	langweilig
Rex123	Tcl/ Tk	2	kurzweilig

Votum

VID	PName	AName	Urteil
DB1	Goldie	Rex123	P
DB1	Goldie	OldHenry	N
XML	Goldie	Rex123	P
XML	Queen	Goldie	N
XML	Rex123	Goldie	N
DB1	Bullie	Goldie	N
DB1	Rex123	OldHenry	N
DB1	Rex123	Goldie	P
Unix	Bullie	Queen	P
Unix	Bullie	Rex123	P
Unix	Bullie	OldHenry	N

(a) Geben Sie die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten zu **Votum** an! Denken Sie daran, dass jeder Student zu jeder Veranstaltung nur einmal einen Kommentar verfassen und man zu jedem Kommentar nur eine **Votum** abgeben darf.

(b) Sind die Relationen **Bewertung** und **Votum** in BCNF? Kurze Begründung!

(c) Die oben genannten Relationen **Bewertung** und **Votum** entstanden durch Aufteilung der nichtnormalisierten Relation **BV** unten.

BV

PName	VID	Note	KText	AName	Urteil
Goldie	DB1	3	Prof nervt	Rex123	P
Goldie	DB1	3	Prof nervt	OldHenry	N
Goldie	XML	4	Klausur geschenkt	Rex123	P
Bullie	DB1	2	Uebungen toll	Goldie	N
Bullie	Unix	3	geht so	Queen	P
Bullie	Unix	3	geht so	Rex123	P
Bullie	Unix	3	geht so	OldHenry	N
Queen	XML	1	Spitze	Goldie	N
Queen	Unix	2	Augen zu		
Rex123	DB1	3	langweilig	OldHenry	N
Rex123	DB1	3	langweilig	Goldie	P
Rex123	XML	4	langweilig	Goldie	N
Rex123	Tel/Tk	2	kurzweilig		

Ist die Aufteilung von **BV** in **Bewertung** und **Votum** verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

(d) Bei einem natürlichen Join von **Bewertung** und **Votum** fehlen einige der Tupel aus **BV**. Welche andere Art von Join von **Bewertung** und **Votum** erzeugt **BV** von oben?

Aufgabe 6:

Mit Hilfe der Relationen **Student(PName, EMail)** und **Bewertung** von oben wollen Sie die E-Mail-Adressen all der Studenten ermitteln, die einen Kommentar verfasst haben mit einer Note schlechter als 3. Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

Aufgabe 7:

Welches Tupel wird bzw. welche Tupel werden durch den folgenden SQL-Befehl in **Votum** eingefügt? Genaue Angabe des/r Tupel!

```
INSERT INTO Votum (VID, PName, AName, Urteil)
  SELECT VID, PName, 'Bullie', 'P'
  FROM Bewertung
  WHERE PName = 'Goldie'
```

Aufgabe 8:

Der Anreiz in Bewertungsportalen mitzumachen besteht darin, möglichst viele positive Urteile anderer zu erhalten. Wie lautet die SQL-Abfrage, mit der man aus **Votum** in Aufgabe 5 ein zweispaltiges Ergebnis bekommt, das alle Verfasser und die Anzahl der P-Urteile, die sie in der Summe auf ihre Kommentare bekommen haben, enthält?

Aufgabe 9:

Mit welchem SQL-Befehl erstellt man aus der Relation **Bewertung** in Aufgabe 5 den View **Ranking**, der alle Vorlesungen (**VID**) und zu jeder Vorlesung die aus den Kommentaren ermittelte Durchschnittsnote (**DNote**) enthält?

Aufgabe 10:

Wir suchen alle Kommentare, über die noch niemand abgestimmt hat. Konkret benötigen wir aus **Bewertung** alle vier Attribute. Allerdings darf es dazu in **Votum** keine Treffer geben.

Wie lautet die SQL-Abfrage?

Aufgabe 11:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktion T_i :

Transaktion T_1 (10:00 Uhr)	Transaktion T_2 (10:05 Uhr)	Transaktion T_3 (10:10 Uhr)
	Lesen A	
Lesen B	Schreiben A	
	Lesen C	Lesen B
		Lesen C
		Schreiben C
	Schreiben C	Schreiben B
Schreiben B		

Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle ein. Markieren Sie die erste Operation oben und die relevante Zeitmarke unten, die zusammen eine Zurückweisung auslösen.

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann ignoriere Schreiboperation

sonst Schreiben, $WTM(x) := TS$

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Wintersemester 2009/10**

Name: Vorname:

Matr.Nr.: Studiengang:

MUSTERLÖSUNG

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		7	4	
2	6		8	4	
3	1+1+2+2		9	4	
4	2+2+2		10	4	
5	2+2+2+2		11	4	
6	4		Summe	53	

Aufgabe 1:

Der Isolationsgrad eines Transaktionssystems verläuft vom Grad 0 (Chaos) bis zum höchsten Grad 3 (vollständige Isolation). Tragen Sie für jede der unten genannten Eigenschaften des Transaktionssystems den passenden Grad (0 bis 3) in die Klammern [] ein.

Hinweis: Jeder Grad taucht genau einmal auf.

- [**2**] READ COMMITTED, Cursorstabilität, kein Verlust von Updates.
- [**1**] READ UNCOMMITTED, nur Schreibsperrern, diese bis EOT, liest u. U. schmutzige Daten.
- [**3**] SERIALIZABLE, Lesezugriffe wiederholbar.
- [**0**] Daten nach Schreibzugriff sofort sichtbar, nur Schreibsperrern, Update kann verloren gehen.

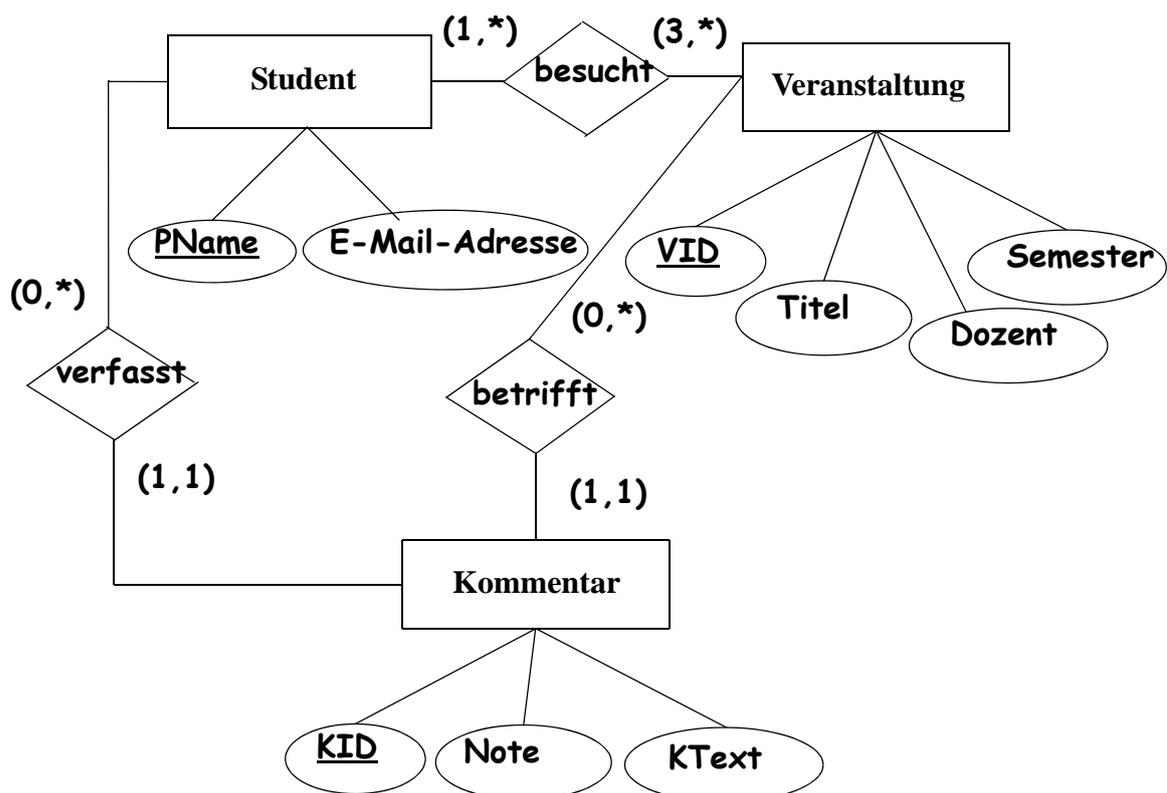
Aufgabe 2:

Bewertungsportale sind derzeit (noch) groß in Mode. Als Miniwelt betrachten wir hier deshalb ein fiktives Portal zur Bewertung unserer Veranstaltungen.

Veranstaltungen haben einen eindeutigen **Veranstaltungsidentifizier (VID)**, ferner **Titel**, **Dozent** und **Semester**, in dem die Veranstaltung stattfand. **Studenten besuchen Veranstaltungen**; wir nehmen hier an, mindestens eine und in der Regel natürlich mehrere. Veranstaltungen dagegen brauchen mindestens drei Hörer. Jeder Student hat ein **Pseudonym (frei gewählter Spitzname PName)**, der nur einmal vergeben wird. Ferner muss er seine **E-Mail-Adresse** angeben, die es auch nur einmal geben darf und die bei den Kommentaren nicht angezeigt wird.

Studenten können dann **Kommentare verfassen**. Jeder Kommentar wird von genau einem Studenten verfasst und **betrifft** genau eine Veranstaltung. Ein Kommentar habe einen eindeutigen **Kommentaridentifizier (KID)**, eine **Note** (1 = sehr gut, 5 = mangelhaft) und einen **Text (KText)** mit bis zu 1000 Zeichen. Klar ist auch, dass ein Student viele Kommentare abgeben kann, nicht jeder Student kommentieren möchte, nicht jede Veranstaltung kommentiert wird, dass eine Veranstaltung aber häufig viele Kommentare hat.

Vervollständigen Sie das ER-Diagramm unten (ohne Spezialisierung/Generalisierung) für diese Miniwelt. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.



Aufgabe 3:

Im ersten Entwurf der Bewertungsdatenbank wurde eine Tabelle **Bewertung** vorgesehen. Allerdings haben die Kommentare keinen Kommentaridentifizier. Jeder Verfasser kann zu jeder Veranstaltung höchstens einen Kommentar abgeben. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

Bewertung

PName	VID	Note	KText
Goldie	DB1	3	Prof nervt
Goldie	XML	4	Klausur geschenkt
Bullie	DB1	2	Uebungen toll
Bullie	Unix	3	geht so
Queen	XML	1	Spitze
Queen	Unix	2	Augen zu und durch
Rex123	DB1	3	langweilig
Rex123	XML	4	langweilig
Rex123	Tcl/Tk	2	kurzweilig

- (a) Was sind die Schlüsselkandidaten?
{PName, VID}
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
{PName, VID} --> Bewertung
- (c) Gibt es eine transitive Abhängigkeit in **Bewertung**? Wenn ja, welche?
Nein
- (d) Ist **Bewertung** in 3NF? Begründung!
Ja, Bewertung ist in 1NF und es sind keine transitiven Abhängigkeiten vorhanden. Andere Begründungen möglich.

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In die Tabelle **Bewertung** nehmen wir noch die eindeutigen E-Mail-Adressen hinzu und erhalten dadurch die Tabelle **Kommentare**.

Kommentare

PName	E-Mail	VID	Note	KText
Goldie	gabi@uni-kassel.de	DB1	3	Prof nervt
Goldie	gabi@uni-kassel.de	XML	4	Klausur geschenkt
Bullie	bernd@uni-kassel.de	DB1	2	Uebungen toll
Bullie	bernd@uni-kassel.de	Unix	3	geht so
Queen	anne@uni-kassel.de	XML	1	Spitze
Queen	anne@uni-kassel.de	Unix	2	Augen zu und durch
Rex123	rolf@uni-kassel.de	DB1	3	langweilig
Rex123	rolf@uni-kassel.de	XML	4	langweilig
Rex123	rolf@uni-kassel.de	Tcl/Tk	2	kurzweilig

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Schlüsselkandidaten: {PName, VID}

{E-Mail, VID}

Nicht-triviale fkt. Abh.: {PName, VID} --> Kommentare

{E-Mail, VID} --> Kommentare

PName --> E-Mail

E-Mail --> PName

(b) Ist **Kommentare** in 3NF? Begründung!

Ja, es gibt keine „echten“ transitiven Abhängigkeiten, speziell ist

{PName, VID} --> {E-Mail, VID} --> Kommentare nicht transitiv

gem. Definition, weil auch {E-Mail, VID} --> {PName, VID}

Alternativ: Alle FA gehen von einem Schlüssel aus oder bestimmen die primen Attribute PName und E-Mail.

(c) Ist **Kommentare** in BCNF? Begründung!

Nein, z. B. ist PName --> E-Mail nicht trivial und PName kein Schlüssel.

Aufgabe 5:

Für die folgenden Aufgaben betrachten wir neben der **Bewertung**-Tabelle aus Aufgabe 3 noch eine **Votum**-Tabelle. Dort wird festgehalten, wer (Abstimmernamen **AName**) positiv oder negativ über einen Kommentar abgestimmt hat. Jeder darf nur einmal zu einem Kommentar ein **Votum** abgeben.

Bewertung

Votum

PName	VID	Note	KText
Goldie	DB1	3	Prof nervt
Goldie	XML	4	Klausur geschenkt
Bullie	DB1	2	Uebungen toll
Bullie	Unix	3	geht so
Queen	XML	1	Spitze
Queen	Unix	2	Augen zu
Rex123	DB1	3	langweilig
Rex123	XML	4	langweilig
Rex123	Tcl/ Tk	2	kurzweilig

VID	PName	AName	Urteil
DB1	Goldie	Rex123	P
DB1	Goldie	OldHenry	N
XML	Goldie	Rex123	P
XML	Queen	Goldie	N
XML	Rex123	Goldie	N
DB1	Bullie	Goldie	N
DB1	Rex123	OldHenry	N
DB1	Rex123	Goldie	P
Unix	Bullie	Queen	P
Unix	Bullie	Rex123	P
Unix	Bullie	OldHenry	N

(a) Geben Sie die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten zu **Votum** an! Denken Sie daran, dass jeder Student zu jeder Veranstaltung nur einmal einen Kommentar verfassen und man zu jedem Kommentar nur eine **Votum** abgeben darf.

Schlüsselkandidaten: {VID, PName, AName}

Nicht-tr. fkt. Abh.: {VID, PName, AName} --> **Votum**

auch richtig {VID, PName, AName} --> **Urteil**

(b) Sind die Relationen **Bewertung** und **Votum** in BCNF? Kurze Begründung!

Ja, beide. Alle funktionalen Abhängigkeiten gehen vom Schlüssel aus (bei *Bewertung* siehe Aufgabe 3b, bei *Votum* siehe oben).

(c) Die oben genannten Relationen **Bewertung** und **Votum** entstanden durch Aufteilung der nichtnormalisierten Relation **BV** unten.

BV

PName	VID	Note	KText	AName	Urteil
Goldie	DB1	3	Prof nervt	Rex123	P
Goldie	DB1	3	Prof nervt	OldHenry	N
Goldie	XML	4	Klausur geschenkt	Rex123	P
Bullie	DB1	2	Uebungen toll	Goldie	N
Bullie	Unix	3	geht so	Queen	P
Bullie	Unix	3	geht so	Rex123	P
Bullie	Unix	3	geht so	OldHenry	N
Queen	XML	1	Spitze	Goldie	N
Queen	Unix	2	Augen zu		
Rex123	DB1	3	langweilig	OldHenry	N
Rex123	DB1	3	langweilig	Goldie	P
Rex123	XML	4	langweilig	Goldie	N
Rex123	Tel/Tk	2	kurzweilig		

Ist die Aufteilung von **BV** in **Bewertung** und **Votum** verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

R1 = Bewertung

R2 = Votum

$R_1 \cap R_2 = \{PName, VID\}$

$\{PName, VID\} \rightarrow$ Bewertung, also ist die Aufteilung verlustfrei!

(d) Bei einem natürlichen Join von **Bewertung** und **Votum** fehlen einige der Tupel aus **BV**. Welche andere Art von Join von **Bewertung** und **Votum** erzeugt **BV** von oben?

Der äußere Join (outer join, genauer der left outer join)

Aufgabe 6:

Mit Hilfe der Relationen **Student(PName, EMail)** und **Bewertung** von oben wollen Sie die E-Mail-Adressen all der Studenten ermitteln, die einen Kommentar verfasst haben mit einer Note schlechter als 3. Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

$\Pi_{EMail} (\sigma_{Note > 3} (Student \bowtie Bewertung))$

Aufgabe 7:

Welches Tupel wird bzw. welche Tupel werden durch den folgenden SQL-Befehl in **Votum** eingefügt? Genaue Angabe des/r Tupel!

```
INSERT INTO Votum (VID, PName, AName, Urteil)
  SELECT VID, PName, 'Bullie', 'P'
  FROM Bewertung
  WHERE PName = 'Goldie'
```

2 Zeilen werden eingefügt:

VID	PName	AName	Urteil
DB1	Goldie	Bullie	P
XML	Goldie	Bullie	P

Aufgabe 8:

Der Anreiz in Bewertungsportalen mitzumachen besteht darin, möglichst viele positive Urteile anderer zu erhalten. Wie lautet die SQL-Abfrage, mit der man aus **Votum** in Aufgabe 5 ein zweispaltiges Ergebnis bekommt, das alle Verfasser und die Anzahl der P-Urteile, die sie in der Summe auf ihre Kommentare bekommen haben, enthält?

```
SELECT PName, COUNT(Urteil)
  FROM Votum
  WHERE Urteil = 'P'
  GROUP BY PName
```

Aufgabe 9:

Mit welchem SQL-Befehl erstellt man aus der Relation **Bewertung** in Aufgabe 5 den View **Ranking**, der alle Vorlesungen (**VID**) und zu jeder Vorlesung die aus den Kommentaren ermittelte Durchschnittsnote (**DNote**) enthält?

```
CREATE VIEW Ranking (VID, DNote) AS
  SELECT VID, AVG(Note)
  FROM Bewertung
  GROUP BY VID
```

Aufgabe 10:

Wir suchen alle Kommentare, über die noch niemand abgestimmt hat. Konkret benötigen wir aus **Bewertung** alle vier Attribute. Allerdings darf es dazu in **Votum** keine Treffer geben.

Wie lautet die SQL-Abfrage?

```
SELECT * FROM Bewertung
  WHERE (PName, VID) NOT IN
  (SELECT PName, VID
   FROM Votum)
```

geht auch mit WHERE NOT EXISTS ...

Aufgabe 11:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktion T_i :

Transaktion T_1
(10:00 Uhr)

Lesen B

Schreiben B

Transaktion T_2
(10:05 Uhr)

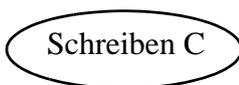
Lesen A
Schreiben A

Lesen C

Schreiben C

Transaktion T_3
(10:10 Uhr)

Lesen B
Lesen C
Schreiben C
Schreiben B



dieser Schreibbefehl löst den Konflikt aus!

Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle ein. Markieren Sie die erste Operation oben und die relevante Zeitmarke unten, die zusammen eine Zurückweisung auslösen.

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)
10:05	10:05	10:00 10:10	10:10	10:05 10:10	10:10

Hier ist $TS = 10:05 < RTM(C)$

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann ignoriere Schreiboperation

sonst Schreiben, $WTM(x) := TS$

ENDE DER KLAUSUR

Klausur zur Vorlesung Datenbanken I im Sommersemester 2010

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		7	3	
2	6		8	4	
3	1+2+2+2		9	4	
4	2+2		10	2+2	
5	3+2+3		11	4	
6	3		Summe	50	

Aufgabe 1:

Bringen Sie die Datenmodelle links mit den Begriffen rechts durch Linien in Verbindung.

Hinweis: Ein Datenmodell kann einen Bezug zu mehreren Begriffen haben, aber jeder Begriff gehört zu genau einem Datenmodell.

Hierarchisches Modell •

Relationales Modell •

Objektorientiertes Modell •

• Tupel

• OID

• Klassenhierarchie

• virtueller Satztyp *T

• Fünf notwendige und hinreichende Operationen der Algebra

Aufgabe 2:

Wie beim Kindergeburtstag spielen wir hier Stadt-Land-Fluss. In unserer Miniwelt gibt es **Flüsse**, die einen eindeutigen Flussnamen (**FName**) haben und eine **Länge** (in km). Dann gibt es **Städte**, von denen wir annehmen, dass sie durch ihren Städtenamen (**SName**) eindeutig identifiziert sind. Ein weiteres Merkmal ist, ob sie **Hauptstadt** sind (ja/nein). Städte **liegen in** einem **Land** und jedes Land hat mindestens eine Stadt. Die Länder sind gekennzeichnet durch einen eindeutigen Ländernamen (**LName**), einen eindeutigen LänderId (**LID**), den wir vom internationalen Kfz-Kennzeichen für das Land ableiten, eine **Einwohnerzahl** (gerundet auf volle Millionen) und eine **Währung**.

Häufig, aber nicht immer, **liegt** eine Stadt **an** einem oder mehreren Flüssen, die wiederum keine oder mehrere Städte durchfließen.

Geben Sie ein ER-Diagramm für diese Miniwelt mit drei Entity-Typen an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.

Aufgabe 3:

Im ersten Entwurf für einen Teil des Datenbankschemas der Miniwelt aus Aufgabe 2 wurde eine Tabelle **SF** für Städte, die an Flüssen liegen, vorgesehen.

SF

SName	Hauptstadt	LID	FName	Länge
Cottbus	Nein	D	Spree	380
Berlin	Ja	D	Spree	380
Berlin	Ja	D	Havel	325
Köln	Nein	D	Rhein	1233
Basel	Nein	CH	Rhein	1233
Bern	Ja	CH	Aare	288
Rom	Ja	I	Tiber	405
Paris	Ja	F	Seine	777
Rotterdam	Nein	NL	Rhein	1233

- (a) Was sind die Schlüsselkandidaten?
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
- (c) Gibt es transitive Abhängigkeiten in **SF**? Wenn ja, nennen Sie eine!
- (d) Ist **SF** in 3NF? Begründung!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat. Für BCNF entfällt „ $A \in R$ ist prim“.

In die Tabelle **Land** haben wir nur Angaben über Länder aufgenommen.

Land

LName	LID	Einwohner	Waehrung
Deutschland	D	82	EUR
Frankreich	F	65	EUR
Italien	I	60	EUR
Niederlande	NL	17	EUR
Schweiz	CH	8	SFR

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

(b) Was ist die höchste Normalform, in der **Land** ist? Begründung!

(c) Wenn wie oben eine Relation R , hier **SF**, in drei Relationen R_1, R_2 und R_3 aufgespalten wird (generell n Relationen), dann prüft man die Verlustlosigkeit der Zerlegung so, dass man zuerst R in R_1 und R_2' teilt und jetzt wie gewohnt feststellt, ob diese Zerlegung verlustfrei ist, ob also $R_1 \cap R_2' \rightarrow R_1$ oder $R_1 \cap R_2' \rightarrow R_2'$. Danach spaltet man R_2' in R_2 und R_3 und prüft wiederum nach derselben Methode die Verlustlosigkeit dieser Aufteilung.

Prüfen Sie mit obiger Methode, ob die Zerlegung von **SF** in **Stadt**, **Fluss** und **Liegt-an** verlustfrei ist. Geben Sie zuerst das Schema von R_2' an und bilden Sie dann die Durchschnitte.

Aufgabe 6:

Mit Hilfe der Relation **Land(LName, LID, Einwohner, Waehrung)** aus Aufgabe 4 suchen wir die Ländernamen aller Länder, die als Währung den Euro und mehr als 59 Mio. Einwohner haben. Geben Sie den Ausdruck der Relationenalgebra an!

Aufgabe 7:

Wie lautet der SQL-Befehl, um in die Tabelle **Land** die Vereinigten Staaten von Amerika (USA) einzufügen, die derzeit ca. 308 Mio. Einwohner haben und deren Währung (US-Dollar) die Abkürzung USD hat?

Aufgabe 8:

Gegeben sei die **Land**-Tabelle aus Aufgabe 4 und die **Stadt**-Tabelle aus Aufgabe 5. Wir suchen die SQL-Abfrage, die den Namen der Hauptstadt und den Namen des Landes für genau die Länder liefert, die den Euro als Währung und mehr als 59 Mio. Einwohner haben.

Aufgabe 9:

Mit welchem SQL-Befehl erhält man aus der Relation **SF** in Aufgabe 3 den View **Flussstaedte**, der alle Städtenamen (**SName**) genau einmal und zu jedem Namen die Anzahl der durch die Stadt fließenden Flüsse als Spalte **FlussAnz** enthält?

Aufgabe 10:

Was liefert die folgende SQL-Abfrage?

```
SELECT DISTINCT s1.FName
FROM SF s1, SF s2
WHERE s1.FName = s2.FName AND NOT (s1.LID = s2.LID)
```

(a) Geben Sie die Ergebnistupel auf der SF-Tabelle aus Aufgabe 3 an.

(b) Beschreiben Sie das Ergebnis auf beliebigen SF-Tabellen!

Aufgabe 11:

Betrachten Sie die unten gezeigte Ablauffolge. Geben Sie den vollständigen Abhängigkeitsgraphen dazu an! Ist die Ablauffolge serialisierbar? Begründung!

Transaktion T_1

Lesen B

Schreiben B

Transaktion T_2

Lesen A

Schreiben A

Lesen C

Schreiben C

Transaktion T_3

Lesen B

Lesen C

Schreiben C

Schreiben B

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Sommersemester 2010**

MUSTERLÖSUNG

Name: Vorname:

Matr.Nr.: Studiengang:

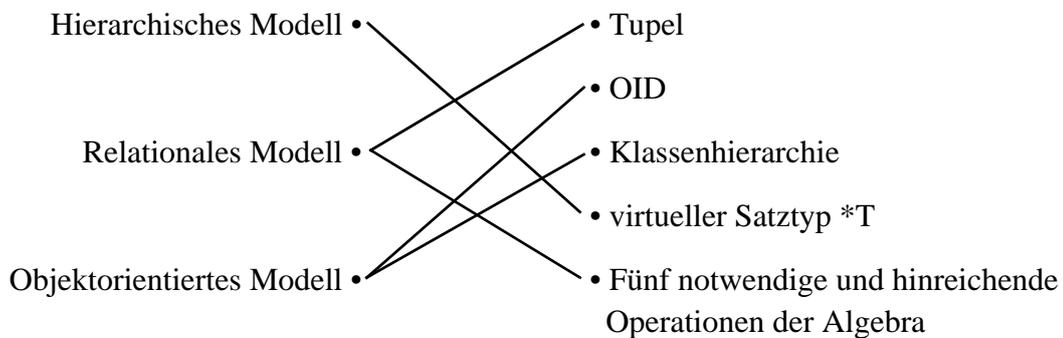
Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	3		7	3	
2	6		8	4	
3	1+2+2+2		9	4	
4	2+2		10	2+2	
5	3+2+3		11	4	
6	3		Summe	50	

Aufgabe 1:

Bringen Sie die Datenmodelle links mit den Begriffen rechts durch Linien in Verbindung.

Hinweis: Ein Datenmodell kann einen Bezug zu mehreren Begriffen haben, aber jeder Begriff gehört zu genau einem Datenmodell.

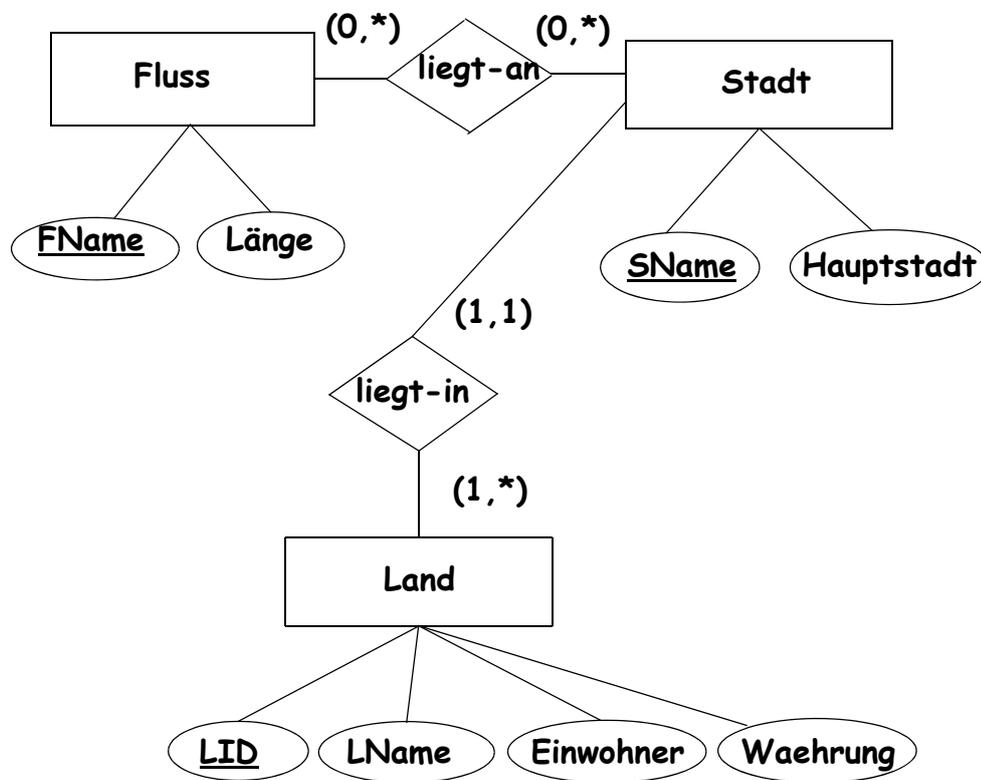


Aufgabe 2:

Wie beim Kindergeburtstag spielen wir hier Stadt-Land-Fluss. In unserer Miniwelt gibt es **Flüsse**, die einen eindeutigen Flussnamen (**FName**) haben und eine **Länge** (in km). Dann gibt es **Städte**, von denen wir annehmen, dass sie durch ihren Städtenamen (**SName**) eindeutig identifiziert sind. Ein weiteres Merkmal ist, ob sie **Hauptstadt** sind (ja/nein). Städte **liegen in** einem **Land** und jedes Land hat mindestens eine Stadt. Die Länder sind gekennzeichnet durch einen eindeutigen Ländernamen (**LName**), einen eindeutigen LänderId (**LID**), den wir vom internationalen Kfz-Kennzeichen für das Land ableiten, eine **Einwohnerzahl** (gerundet auf volle Millionen) und eine **Währung**.

Häufig, aber nicht immer, **liegt** eine Stadt **an** einem oder mehreren Flüssen, die wiederum keine oder mehrere Städte durchfließen.

Geben Sie ein ER-Diagramm für diese Miniwelt mit drei Entity-Typen an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.



auch LName als Primärschlüssel

Aufgabe 3:

Im ersten Entwurf für einen Teil des Datenbankschemas der Miniwelt aus Aufgabe 2 wurde eine Tabelle **SF** für Städte, die an Flüssen liegen, vorgesehen.

SF

SName	Hauptstadt	LID	FName	Länge
Cottbus	Nein	D	Spree	380
Berlin	Ja	D	Spree	380
Berlin	Ja	D	Havel	325
Köln	Nein	D	Rhein	1233
Basel	Nein	CH	Rhein	1233
Bern	Ja	CH	Aare	288
Rom	Ja	I	Tiber	405
Paris	Ja	F	Seine	777
Rotterdam	Nein	NL	Rhein	1233

(a) Was sind die Schlüsselkandidaten?

{SName, FName}

Hinweis: Die Angabe der Klammern ist hier zwingend erforderlich!

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

SName -> Hauptstadt

FName -> Länge

SName -> LID

{SName, FName} -> SF (wegen Schlüsseleigenschaft, ist in den drei FAs oben enthalten)

(c) Gibt es transitive Abhängigkeiten in **SF**? Wenn ja, nennen Sie eine!

**Ja, {SName, FName} --> SName --> Hauptstadt
 <-/-**

(d) Ist **SF** in 3NF? Begründung!

Nein, transitive Abhängigkeit eines nichtprimen Attributs (Hauptstadt) von einem Schlüssel. Auch andere Begründungen über Def. 3NF möglich.

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat. Für BCNF entfällt „ $A \in R$ ist prim“.

In die Tabelle **Land** haben wir nur Angaben über Länder aufgenommen.

Land

LName	LID	Einwohner	Waehrung
Deutschland	D	82	EUR
Frankreich	F	65	EUR
Italien	I	60	EUR
Niederlande	NL	17	EUR
Schweiz	CH	8	SFR

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Schlüsselkandidaten: {LID}, {LName}

Nicht-triviale FA: LID \rightarrow Land (Schlüssel)

LName \rightarrow Land (Schlüssel)

(b) Was ist die höchste Normalform, in der **Land** ist? Begründung!

Land ist in BCNF. Alle nicht-trivialen FA gehen von einem Schlüssel aus.

Hinweis: Häufig wurde behauptet, Land habe eine transitive FA, nämlich LName \rightarrow LID \rightarrow Waehrung. Dies ist aber keine transitive FA, weil auch LID \rightarrow LName, was bei Transitivität nicht erlaubt ist (siehe oben 3 c).

Aufgabe 5:

Für die folgenden Aufgaben haben wir die **SF**-Tabelle aus Aufgabe 3 in drei Relationen aufgespalten. Ergänzen Sie die Einträge in der Tabelle **Liegt-an** gemäß **SF** aus Aufgabe 3!

Hinweis: Die Tabelle hat mehr Zeilen als Sie benötigen.

Stadt

SName	Hauptstadt	LID
Cottbus	Nein	D
Berlin	Ja	D
Köln	Nein	D
Basel	Nein	CH
Bern	Ja	CH
Rom	Ja	I
Paris	Ja	F
Rotterdam	Nein	NL

Fluss

FName	Länge
Spree	380
Havel	325
Rhein	1233
Aare	288
Tiber	405
Seine	777

Liegt-an

SName	FName
Cottbus	Spree
Berlin	Spree
Berlin	Havel
Köln	Rhein
Basel	Rhein
Bern	Aare
Rom	Tiber
Paris	Seine
Rotterdam	Rhein

(a) Welche sind die sinnvollen Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten in den Relationen **Stadt**, **Fluss** und **Liegt-an**?

Stadt: SK ist {SName}

FA ist SName --> Stadt (Schlüssel)

Fluss: SK ist {FName}

FA ist FName --> Fluss (Schlüssel)

Liegt-an: SK ist {SName, FName}

FA ist {SName, FName} --> Liegt-an (Schlüssel)

auch richtig, wenn hier keine FA genannt wird, da es eine triviale FA ist

(b) Sind die Relationen **Stadt**, **Fluss** und **Liegt-an** in BCNF? Kurze Begründung!

Ja, alle drei, weil nur FA existieren, die jeweils vom Schlüssel ausgehen.

(c) Wenn wie oben eine Relation R , hier **SF**, in drei Relationen R_1, R_2 und R_3 aufgespalten wird (generell n Relationen), dann prüft man die Verlustlosigkeit der Zerlegung so, dass man zuerst R in R_1 und R_2' teilt und jetzt wie gewohnt feststellt, ob diese Zerlegung verlustfrei ist, ob also $R_1 \cap R_2' \rightarrow R_1$ oder $R_1 \cap R_2' \rightarrow R_2'$. Danach spaltet man R_2' in R_2 und R_3 und prüft wiederum nach derselben Methode die Verlustlosigkeit dieser Aufteilung.

Prüfen Sie mit obiger Methode, ob die Zerlegung von **SF** in **Stadt**, **Fluss** und **Liegt-an** verlustfrei ist. Geben Sie zuerst das Schema von R_2' an und bilden Sie dann die Durchschnitte.

$R = \text{SF}$ aufgespalten in $R_1 = \text{Stadt}$ und $R_2' = \{\text{FName, Länge, SName}\}$

Verlustfrei, da $R_1 \cap R_2' = \{\text{SName}\}$ und $\text{SName} \rightarrow R_1 = \text{Stadt}$.

$R_2' = \{\text{FName, Länge, SName}\}$ aufgespalten in $R_2 = \text{Fluss}$ und $R_3 = \text{Liegt-an}$

Verlustfrei, da $R_2 \cap R_3 = \{\text{FName}\}$ und $\text{FName} \rightarrow R_2 = \text{Fluss}$

Also ist die Zerlegung von **SF in R_1, R_2 und R_3 insgesamt verlustfrei.**

Aufgabe 6:

Mit Hilfe der Relation **Land(LName, LID, Einwohner, Waehrung)** aus Aufgabe 4 suchen wir die Ländernamen aller Länder, die als Währung den Euro und mehr als 59 Mio. Einwohner haben. Geben Sie den Ausdruck der Relationenalgebra an!

$\Pi_{\text{LName}} (\sigma_{\text{Waehrung}='EUR' \wedge \text{Einwohner} > 59} (\text{Land}))$

Aufgabe 7:

Wie lautet der SQL-Befehl, um in die Tabelle **Land** die Vereinigten Staaten von Amerika (USA) einzufügen, die derzeit ca. 308 Mio. Einwohner haben und deren Währung (US-Dollar) die Abkürzung USD hat?

INSERT INTO Land

VALUES ('Vereinigte Staaten von Amerika', 'USA', 308, 'USD')

Aufgabe 8:

Gegeben sei die **Land**-Tabelle aus Aufgabe 4 und die **Stadt**-Tabelle aus Aufgabe 5. Wir suchen die SQL-Abfrage, die den Namen der Hauptstadt und den Namen des Landes für genau die Länder liefert, die den Euro als Währung und mehr als 59 Mio. Einwohner haben.

SELECT Stadt.SName, Land.LName

FROM Stadt, Land

WHERE Stadt.LID = Land.LID AND

Stadt.Hauptstadt = 'Ja' AND

Land.Waehrung = 'EUR' AND

Land.Einwohner > 59

Aufgabe 9:

Mit welchem SQL-Befehl erhält man aus der Relation **SF** in Aufgabe 3 den View **Flussstaedte**, der alle Städtenamen (**SName**) genau einmal und zu jedem Namen die Anzahl der durch die Stadt fließenden Flüsse als Spalte **FlussAnz** enthält?

```
CREATE VIEW Flussstaedte AS  
    SELECT SName, COUNT (FName) AS FlussAnz  
    FROM SF  
    GROUP BY SName
```

oder **CREATE VIEW Flussstaedte (SName, FlussAnz) AS ...**

Aufgabe 10:

Was liefert die folgende SQL-Abfrage?

```
SELECT DISTINCT s1.FName  
FROM SF s1, SF s2  
WHERE s1.FName = s2.FName AND NOT (s1.LID = s2.LID)
```

(a) Geben Sie die Ergebnistupel auf der SF-Tabelle aus Aufgabe 3 an.

(Rhein)

(b) Beschreiben Sie das Ergebnis auf beliebigen SF-Tabellen!

Die Abfrage liefert die Flussnamen aller Flüsse, die durch mehr als ein Land fließen.

Aufgabe 11:

Betrachten Sie die unten gezeigte Ablauffolge. Geben Sie den vollständigen Abhängigkeitsgraphen dazu an! Ist die Ablauffolge serialisierbar? Begründung!

Transaktion T_1

Lesen B

Schreiben B

Transaktion T_2

Lesen A

Schreiben A

Lesen C

Schreiben C

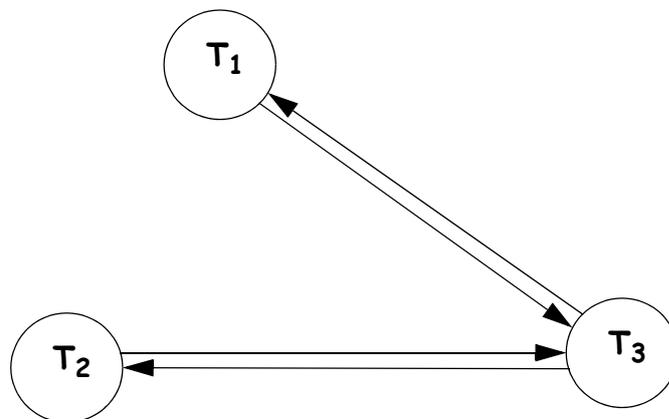
Transaktion T_3

Lesen B

Lesen C

Schreiben C

Schreiben B



Die Ablauffolge ist nicht serialisierbar, da der Graph einen Zyklus hat (sogar zwei Zyklen).

ENDE DER KLAUSUR

Klausur zur Vorlesung Datenbanken I im Wintersemester 2010/11

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	6		8	4	
3	1+1+2+1		9	4	
4	3+2+2		10	4	
5	2+2+2		11	4	
6	4		Summe	50	

Aufgabe 1:

Welche der folgenden Begriffe sind Teil der Abkürzung ACID (Eigenschaften eines Transaktionssystems)?

- Atomicity (atomares Verhalten)
- Connectivity (Zusammenhalt).
- Isolation (unabhängige Ausführung).
- Data Independence (Datenunabhängigkeit).

Aufgabe 2:

Als Miniwelt betrachten wir heute die Pension Sonnenschein, die Gästezimmer vermietet. Das geplante DBMS soll sowohl Reservierungen in die Zukunft als auch Buchungen der Vergangenheit speichern.

Dazu haben **Zimmer** eine eindeutige **Zimmernummer (ZNr)**, ferner eine **Beschreibung** und einen **Preis** pro Nacht für das Zimmer. Die Zimmerpreise seien also immer fest an das Zimmer gebunden, unabhängig von Dauer, Termin und Belegung.

Gäste haben einen eindeutigen **Gast-Identifizier (GastID)**, einen **Namen** und eine **Adresse**. Gäste können keine oder mehrere **Aufenthalte** buchen bzw. gebucht haben. Ein Aufenthalt hat genau einen Gast, der ihn bucht. Dazu bekommt jeder Aufenthalt eine eindeutige **Buchungsnummer (BNr)**, das **Anreisedatum** und die gebuchte Anzahl **Nächte**.

Demnach **buchen** Gäste Aufenthalte, nicht direkt Zimmer. Ein Aufenthalt muss mindestens ein oder mehrere Zimmer **belegen**, jeweils für dieselbe Anzahl an Nächten. Aus Modellsicht muss nicht jedes Zimmer belegt sein. Wie schon gesagt, kann ein Gast dem System bekannt sein, auch wenn er noch nie einen Aufenthalt gebucht hat, genauso kann er beliebig viele Aufenthalte buchen.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung) für diese vereinfachte Miniwelt an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.

Aufgabe 3:

Im ersten Entwurf der Datenbank wurde eine Tabelle **ZIMMER** vorgesehen. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

ZIMMER

ZNR	BESCHREIBUNG	PREIS
1	EZ EG	35,00
2	EZ EG	35,00
3	DZ EG	65,00
11	DZ 1. OG	65,00
12	DZ 1. OG	70,00
21	DZ 2. OG, Bad nur Dusche	65,00
22	DZ 2. OG, Bad nur Wanne	65,00
23	DZ 2. OG, Bad Dusche und Wanne	70,00
24	3-Bett 2. OG	80,00

- (a) Was sind die Schlüsselkandidaten?
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
- (c) Gibt es eine transitive Abhängigkeit in **ZIMMER**? Wenn ja, welche?
- (d) Ist **ZIMMER** in 3NF? Begründung!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In der Tabelle **RESERVIERUNG** haben wir beispielhaft Aufenthalte mit Zimmernummern, Preis und Namen der Gäste verknüpft. Durch unterschiedliche Bezahlung oder eine Nachbuchung können für einen Gast bei selbem Anreisedatum und gleicher Anzahl an Nächten zwei Buchungssätze entstehen.

RESERVIERUNG

BNR	ANREISE	NAECHTE	ZNR	PREIS	GASTNAME
2011-001	19.03.2011	1	3	65,00	Dr. Musterfrau
2011-002	19.03.2011	1	11	65,00	Mustermann
2011-002	19.03.2011	1	21	65,00	Mustermann
2011-003	20.03.2011	2	1	35,00	Dr. Otto Matic
2011-004	20.03.2011	2	3	65,00	Dr. Otto Matic
...

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

(b) Ist **RESERVIERUNG** in 2NF? Begründung!

(c) Ist **RESERVIERUNG** in 3NF? Begründung!

Aufgabe 5:

Für die folgenden Aufgaben betrachten wir eine **AUFENTHALT**-Tabelle und eine **GAST**-Tabelle. Wie in Aufgabe 4 bereits angedeutet, kann es durch unterschiedliche Bezahlung oder Nachbuchungen dazu kommen, dass für einen Gast bei gleicher An- und Abreise zwei Aufenthaltstupel entstehen.

AUFENTHALT

BNR	ANREISE	NAECHTE	GASTID
2011-001	19.03.2011	1	4711
2011-002	19.03.2011	1	4712
2011-003	20.03.2011	2	4710
2011-004	20.03.2011	2	4710
2011-005	24.03.2011	1	4711

GAST

GASTID	NAME	ADRESSE
4711	Dr. Musterfrau	Kassel
4710	Dr. Otto Matic	Frankfurt
4712	Mustermann	München
4713	Mustermann	Hamburg
4709	Prof. Niemand	Hamburg

(a) Geben Sie die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten zu **AUFENTHALT** an!

(b) Ist die Relationen **AUFENTHALT** in BCNF? Kurze Begründung!

(c) Die oben genannten Relationen **AUFENTHALT** und **GAST** entstanden durch Aufteilung einer nichtnormalisierten Relation **AG** (**BNR, ANREISE, NAECHTE, GASTID, NAME, ADRESSE**). Ist die Aufteilung von **AG** in **AUFENTHALT** und **GAST** verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

Aufgabe 6:

Mit Hilfe der Relationen **AUFENTHALT** und **GAST** von oben wollen Sie den Namen aller Gäste ermitteln, die am 19.03.2011 anreisen. Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

Aufgabe 7:

Wie storniert (löscht) man dem Gast mit GASTID 4712 seine Buchung für den 19.03.2011, d. h. wie lautet der SQL-Befehl auf der **AUFENTHALT**-Tabelle dazu?

Aufgabe 8:

Nehmen wir an, es gebe eine Verbindungstabelle **ZA(ZNR, BNR)**, die **ZIMMER** aus Aufgabe 3 mit **AUFENTHALT** aus Aufgabe 5 verbindet.

Wie lautet die SQL-Abfrage über **ZIMMER**, **ZA** und **AUFENTHALT** zur Ausgabe aller Zimmernummern und Beschreibungen für die Zimmer, die bei Anreise am 19.03.2011 belegt werden?

Aufgabe 9:

Ein Nachteil der **ZIMMER**-Tabelle aus Aufgabe 3 ist, dass der Zimmertyp nur in der Beschreibung versteckt ist. Wir behelfen uns auf der Suche nach Mehrbettzimmern (die nicht EZ in der Beschreibung haben) mit dem **LIKE**-Vergleich (**NOT BESCHREIBUNG LIKE '%EZ%'**). Mit welchem SQL-Befehl erstellt man aus der Relation **ZIMMER** in Aufgabe 3 dann den View **MEHRBETTZIMMER** (mit allen Attributen von **ZIMMER**) für Zimmer mit obiger Eigenschaft?

Aufgabe 10:

Welche Tupel liefert die folgende SQL-Abfrage über die Tabellen **GAST** und **AUFENTHALT** von Aufgabe 5?

```
SELECT GAST.GASTID, NAME, ADRESSE, COUNT(*) AS ANZAHL
FROM GAST, AUFENTHALT
WHERE GAST.GASTID = AUFENTHALT.GASTID
GROUP BY GAST.GASTID, NAME, ADRESSE
HAVING COUNT(*) > 1
```

Aufgabe 11:

Betrachten Sie die unten gezeigte Ablauffolge. Geben Sie den vollständigen Abhängigkeitsgraphen dazu an! Ist die Ablauffolge serialisierbar? Begründung!

Transaktion T_1

Lesen A
Schreiben A

Transaktion T_2

Lesen A
Lesen B

Lesen C
Schreiben C

Transaktion T_3

Lesen B

Schreiben B
Lesen A
Schreiben A

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Wintersemester 2010/11**

Name: Vorname:

Matr.Nr.: Studiengang:

MUSTERLÖSUNG

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	6		8	4	
3	1+1+2+1		9	4	
4	3+2+2		10	4	
5	2+2+2		11	4	
6	4		Summe	50	

Aufgabe 1:

Welche der folgenden Begriffe sind Teil der Abkürzung ACID (Eigenschaften eines Transaktionssystems)?

- [] Atomicity (atomares Verhalten)
- [] Connectivity (Zusammenhalt).
- [] Isolation (unabhängige Ausführung).
- [] Data Independence (Datenunabhängigkeit).

Aufgabe 2:

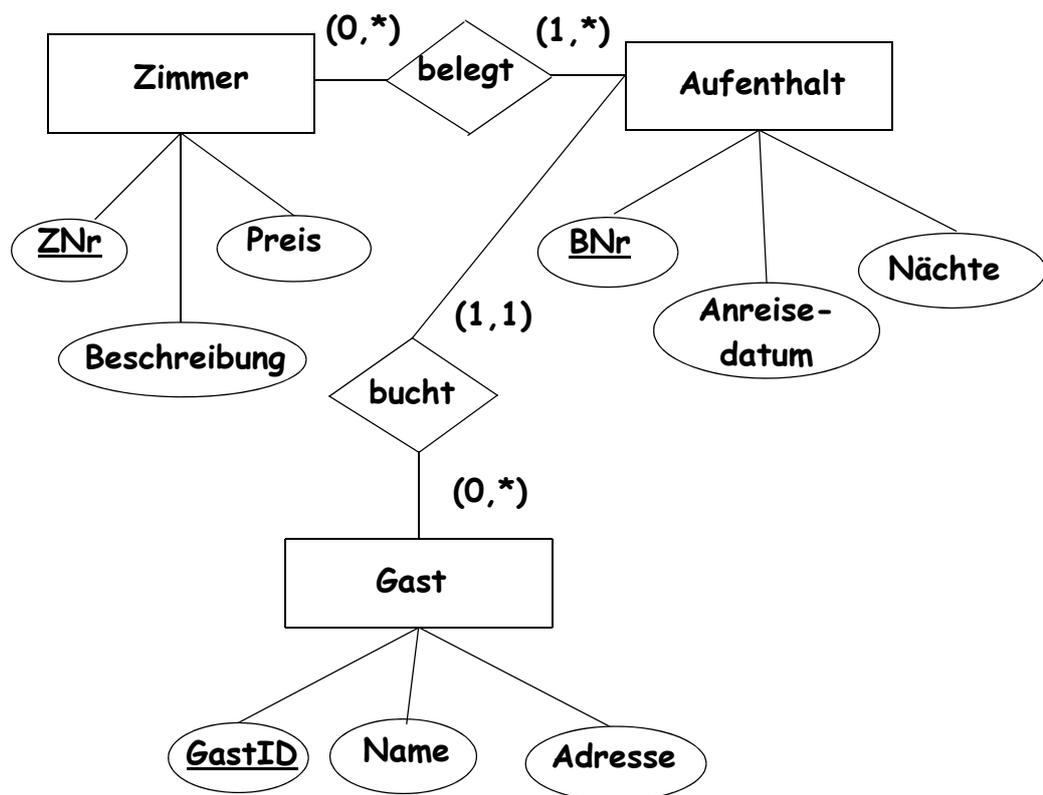
Als Miniwelt betrachten wir heute die Pension Sonnenschein, die Gästezimmer vermietet. Das geplante DBMS soll sowohl Reservierungen in die Zukunft als auch Buchungen der Vergangenheit speichern.

Dazu haben **Zimmer** eine eindeutige **Zimmernummer (ZNr)**, ferner eine **Beschreibung** und einen **Preis** pro Nacht für das Zimmer. Die Zimmerpreise seien also immer fest an das Zimmer gebunden, unabhängig von Dauer, Termin und Belegung.

Gäste haben einen eindeutigen **Gast-Identifizier (GastID)**, einen **Namen** und eine **Adresse**. Gäste können keine oder mehrere **Aufenthalte** buchen bzw. gebucht haben. Ein Aufenthalt hat genau einen Gast, der ihn bucht. Dazu bekommt jeder Aufenthalt eine eindeutige **Buchungsnummer (BNr)**, das **Anreisedatum** und die gebuchte Anzahl **Nächte**.

Demnach **buchen** Gäste Aufenthalte, nicht direkt Zimmer. Ein Aufenthalt muss mindestens ein oder mehrere Zimmer **belegen**, jeweils für dieselbe Anzahl an Nächten. Aus Modellsicht muss nicht jedes Zimmer belegt sein. Wie schon gesagt, kann ein Gast dem System bekannt sein, auch wenn er noch nie einen Aufenthalt gebucht hat, genauso kann er beliebig viele Aufenthalte buchen.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung) für diese vereinfachte Miniwelt an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.



Aufgabe 3:

Im ersten Entwurf der Datenbank wurde eine Tabelle **ZIMMER** vorgesehen. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

ZIMMER

ZNR	BESCHREIBUNG	PREIS
1	EZ EG	35,00
2	EZ EG	35,00
3	DZ EG	65,00
11	DZ 1. OG	65,00
12	DZ 1. OG	70,00
21	DZ 2. OG, Bad nur Dusche	65,00
22	DZ 2. OG, Bad nur Wanne	65,00
23	DZ 2. OG, Bad Dusche und Wanne	70,00
24	3-Bett 2. OG	80,00

- (a) Was sind die Schlüsselkandidaten?
{ZNR}
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
{ZNR} -> ZIMMER also nur Schlüsselabhängigkeit
- (c) Gibt es eine transitive Abhängigkeit in **ZIMMER**? Wenn ja, welche?
Nein
- (d) Ist **ZIMMER** in 3NF? Begründung!
Ja, keine transitiven Abhängigkeiten, wäre sogar in BCNF

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In der Tabelle **RESERVIERUNG** haben wir beispielhaft Aufenthalte mit Zimmernummern, Preis und Namen der Gäste verknüpft. Durch unterschiedliche Bezahlung oder eine Nachbuchung können für einen Gast bei selbem Anreisedatum und gleicher Anzahl an Nächten zwei Buchungssätze entstehen.

RESERVIERUNG

BNR	ANREISE	NAECHTE	ZNR	PREIS	GASTNAME
2011-001	19.03.2011	1	3	65,00	Dr. Musterfrau
2011-002	19.03.2011	1	11	65,00	Mustermann
2011-002	19.03.2011	1	21	65,00	Mustermann
2011-003	20.03.2011	2	1	35,00	Dr. Otto Matic
2011-004	20.03.2011	2	3	65,00	Dr. Otto Matic
...

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Schlüssel:

{BNR, ZNR}, {ANREISE, ZNR}

Funktionale Abhängigkeiten:

{BNR, ZNR} \rightarrow RESERVIERUNG

{ANREISE, ZNR} \rightarrow RESERVIERUNG

{ZNR} \rightarrow {PREIS}

{BNR} \rightarrow {ANREISE, NAECHTE, GASTNAME}

(b) Ist **RESERVIERUNG** in 2NF? Begründung!

Nein, bei ZNR \rightarrow PREIS partielle Abhängigkeit eines nicht-primen Attributs

(c) Ist **RESERVIERUNG** in 3NF? Begründung!

Nein, weil noch nicht mal in 2NF

Aufgabe 5:

Für die folgenden Aufgaben betrachten wir eine **AUFENTHALT**-Tabelle und eine **GAST**-Tabelle. Wie in Aufgabe 4 bereits angedeutet, kann es durch unterschiedliche Bezahlung oder Nachbuchungen dazu kommen, dass für einen Gast bei gleicher An- und Abreise zwei Aufenthaltstupel entstehen.

AUFENTHALT

GAST

BNR	ANREISE	NAECHTE	GASTID	GASTID	NAME	ADRESSE
2011-001	19.03.2011	1	4711	4711	Dr. Musterfrau	Kassel
2011-002	19.03.2011	1	4712	4710	Dr. Otto Matic	Frankfurt
2011-003	20.03.2011	2	4710	4712	Mustermann	München
2011-004	20.03.2011	2	4710	4713	Mustermann	Hamburg
2011-005	24.03.2011	1	4711	4709	Prof. Niemand	Hamburg

(a) Geben Sie die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten zu **AUFENTHALT** an!

Schlüssel: {BNR}

Fkt. Abhängigkeiten: {BNR} -> AUFENTHALT

(b) Ist die Relationen **AUFENTHALT** in BCNF? Kurze Begründung!

Ja, alle nicht-trivialen FA gehen von einem Schlüssel aus.

(c) Die oben genannten Relationen **AUFENTHALT** und **GAST** entstanden durch Aufteilung einer nichtnormalisierten Relation **AG** (**BNR, ANREISE, NAECHTE, GASTID, NAME, ADRESSE**). Ist die Aufteilung von **AG** in **AUFENTHALT** und **GAST** verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

Ja, Aufteilung ist verlustfrei, da $AUFENTHALT \cap GAST = \{GASTID\}$ und $\{GASTID\} \rightarrow GAST$

Aufgabe 6:

Mit Hilfe der Relationen **AUFENTHALT** und **GAST** von oben wollen Sie den Namen aller Gäste ermitteln, die am 19.03.2011 anreisen. Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

$\Pi_{\text{NAME}}((\sigma_{\text{ANREISE} = \text{"19.03.2011"}}(\text{AUFENTHALT})) \bowtie \text{GAST})$

Aufgabe 7:

Wie storniert (löscht) man dem Gast mit GASTID 4712 seine Buchung für den 19.03.2011, d. h. wie lautet der SQL-Befehl auf der **AUFENTHALT**-Tabelle dazu?

```
DELETE FROM AUFENTHALT
WHERE GASTID = 4712 AND
      ANREISE = '19.03.2011';
```

Oder: ... AND ANREISE = TO_DATE('19.03.2011', 'DD.MM.YYYY');

Aufgabe 8:

Nehmen wir an, es gebe eine Verbindungstabelle **ZA(ZNR, BNR)**, die **ZIMMER** aus Aufgabe 3 mit **AUFENTHALT** aus Aufgabe 5 verbindet.

Wie lautet die SQL-Abfrage über **ZIMMER**, **ZA** und **AUFENTHALT** zur Ausgabe aller Zimmernummern und Beschreibungen für die Zimmer, die bei Anreise am 19.03.2011 belegt werden?

```
SELECT ZIMMER.ZNR, BESCHREIBUNG
FROM ZIMMER, ZA, AUFENTHALT
WHERE ZIMMER.ZNR = ZA.ZNR AND
      ZA.BNR = AUFENTHALT.BNR AND
      ANREISE = '19.03.2011';
```

Oder: ... AND ANREISE = TO_DATE('19.03.2011', 'DD.MM.YYYY');

Aufgabe 9:

Ein Nachteil der **ZIMMER**-Tabelle aus Aufgabe 3 ist, dass der Zimmertyp nur in der Beschreibung versteckt ist. Wir behelfen uns auf der Suche nach Mehrbettzimmern (die nicht EZ in der Beschreibung haben) mit dem **LIKE**-Vergleich (**NOT BESCHREIBUNG LIKE '%EZ%'**). Mit welchem SQL-Befehl erstellt man aus der Relation **ZIMMER** in Aufgabe 3 dann den View **MEHRBETTZIMMER** (mit allen Attributen von **ZIMMER**) für Zimmer mit obiger Eigenschaft?

```
CREATE VIEW MEHRBETTZIMMER AS  
  SELECT *  
  FROM ZIMMER  
  WHERE NOT BESCHREIBUNG LIKE '%EZ%';
```

Oder: ... **WHERE BESCHREIBUNG NOT LIKE '%EZ%';**

Aufgabe 10:

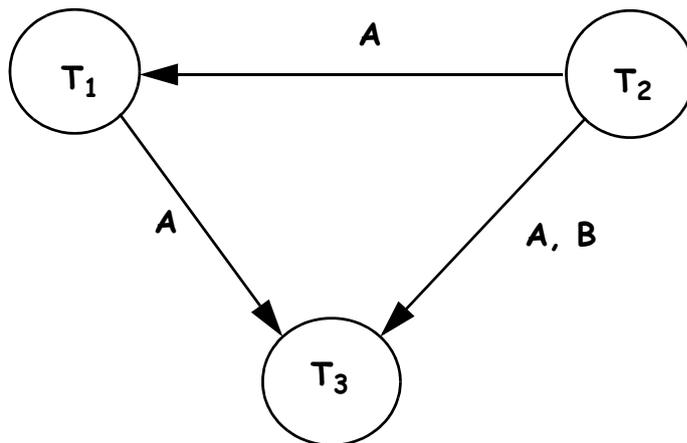
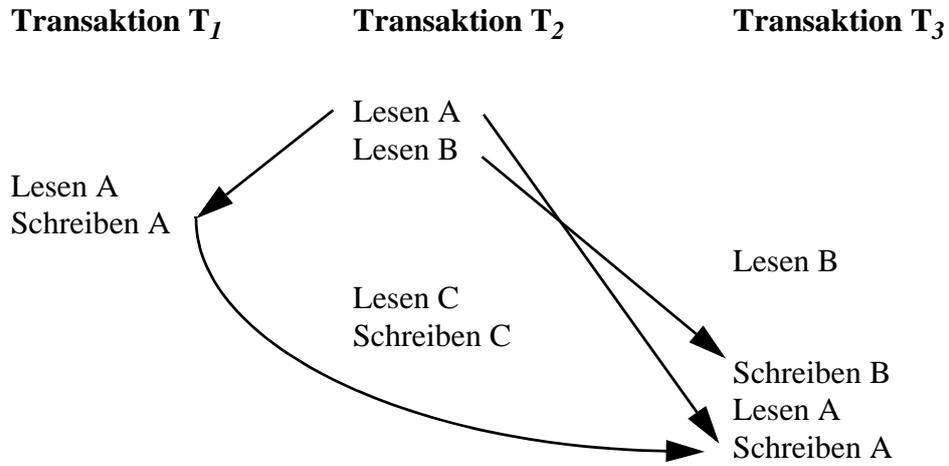
Welche Tupel liefert die folgende SQL-Abfrage über die Tabellen **GAST** und **AUFENTHALT** von Aufgabe 5?

```
SELECT GAST.GASTID, NAME, ADRESSE, COUNT(*) AS ANZAHL  
FROM GAST, AUFENTHALT  
WHERE GAST.GASTID = AUFENTHALT.GASTID  
GROUP BY GAST.GASTID, NAME, ADRESSE  
HAVING COUNT(*) > 1
```

GASTID	NAME	ADRESSE	ANZAHL
4710	Dr. Otto Matic	Frankfurt	2
4711	Dr. Musterfrau	Kassel	2

Aufgabe 11:

Betrachten Sie die unten gezeigte Ablauffolge. Geben Sie den vollständigen Abhängigkeitsgraphen dazu an! Ist die Ablauffolge serialisierbar? Begründung!



serialisierbar, da kein Zyklus.

ENDE DER KLAUSUR

Klausur zur Vorlesung Datenbanken I im Sommersemester 2011

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 90 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	6		8	4	
3	1+1+1+2		9	3	
4	2+2+2		10	2+1+1	
5	2+2		Summe	42	
6	4				

Aufgabe 1:

Bringen Sie die Begriffe der linken Spalte mit den Aussagen der rechten Spalte in Verbindung. Zu jedem Begriff gehört genau eine Aussage. Tragen Sie rechts den jeweiligen Buchstaben des Begriffs ein.

- | | |
|----------------------|--|
| (a) RID/TID | () Cursor-Stabilität / READ COMMITTED |
| (b) Isolationsgrad 2 | () disk interleaving ohne Redundanz |
| (c) RAID0 | () WITHIN LINK-DATA-AREA |
| (d) IS | () Tupeladresse |
| (e) Netzwerkmodell | () Sperre für gleichzeitiges Lesen angemeldet |

Aufgabe 2:

Als Miniwelt betrachten wir Atomkraftwerke, deren Standorte und Betreiber in Deutschland. Jedes Atomkraftwerk (**AKW**) habe einen eindeutigen Kraftwerksidentifizier (**KWID**), eine eindeutige **Bezeichnung**, ein Jahr der Inbetriebnahme (**IBJahr**). Ein Beispiel wäre (*KKI2, Isar/Ohu 2, 1988*).

Daneben gibt es **Standorte**, diese besitzen einen eindeutigen **Gemeindenamen** und ein **Bundesland**. Zwischen AKWs und Standorten gibt es die Beziehung **steht-in**. Jedes AKW hat genau einen Standort, an einem Standort muss mindestens ein, es können aber auch mehrere AKWs stehen. So stehen z. B. *Isar 1* und *Isar 2* in der Gemeinde *Markt Essenbach* in Bayern (*BY*).

Zuletzt gibt es Atomkraftwerksbetreiber (**Betreiber**). Diese sind durch einen eindeutigen Kurzbezeichner (**KBez**), Firmennamen (**FName**) und Hauptverwaltungsstandort (**HVOrt**) gekennzeichnet. Ein Beispiel wäre (*EnBW, Energie Baden-Württemberg AG, Karlsruhe*).

Jedes AKW wird von genau einem Betreiber betrieben, bekanntlich **betreiben** die Betreiber mindestens ein, in der Regel mehrere Kernkraftwerke.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung) für diese vereinfachte Miniwelt an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.

Aufgabe 3:

Im ersten Entwurf einer Datenbank für Atomkraftwerke wurde eine Tabelle **ATOMKW** vorgesehen. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

ATOMKW

KWID	KWBEZEICHNUNG	GEMEINDE
KBR	Brokdorf	Brokdorf
KWBB	Biblis B	Biblis
KWBA	Biblis A	Biblis
KGGC	Grundremmingen C	Grundremmingen
KGGB	Grundremmingen B	Grundremmingen
KKP1	Philippsburg 1	Philippsburg
KKI2	Isar 2	Markt Essenbach
KKI1	Isar 1	Markt Essenbach

(a) Was sind die Schlüsselkandidaten?

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

(c) Gibt es eine transitive Abhängigkeit in **ATOMKW**? Wenn ja, welche?
Hinweis: Es muss gelten $X \rightarrow Y \rightarrow Z$ und $Y \not\rightarrow X$!

(d) Ist **ATOMKW** in 3NF? Begründung!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In der Tabelle **AKW** haben wir jetzt beispielhaft Kraftwerksidentifizier, Gemeinde, Bundesländer und Betreiber verknüpft. Man beachte, dass an einem Standort zwar mehrere AKWs stehen können, diese dann aber immer vom selben Betreiber betrieben werden.

AKW

KWID	GEMEINDE	BUNDES- LAND	BETREIBER
KBR	Brokdorf	SH	E.ON
KWBB	Biblis	HE	RWE
KWBA	Biblis	HE	RWE
KGGC	Grundremmingen	BY	RWE
KGGB	Grundremmingen	BY	RWE
KKP1	Philippsburg	BW	EnBW
KKI2	Markt Essenbach	BY	E.ON
KKI1	Markt Essenbach	BY	E.ON

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

(b) Ist **AKW** in 2NF? Begründung!

(c) Ist **AKW** in 3NF? Begründung!

Aufgabe 5:

Für die folgenden Aufgaben betrachten wir eine **AKW_ORT**-, eine **ORT_BLAND**- und eine **AKW_BETREIBER**-Tabelle. Es gelten die Annahmen aus Aufgabe 2.

AKW_ORT

KWID	KWBEZEICHNUNG	GEMEINDE
KBR	Brokdorf	Brokdorf
KWBB	Biblis B	Biblis
KWBA	Biblis A	Biblis
KGGC	Grundremmingen C	Grundremmingen
...
KKI1	Isar 1	Markt Essenbach

ORT_BLAND

GEMEINDE	BUNDES-LAND
Brokdorf	SH
Biblis	HE
Grundremmingen	BY
Philippsburg	BW
Markt Essenbach	BY

AKW_BETREIBER

KWID	BETREIBER
KBR	E.ON
KWBB	RWE
KWBA	RWE
...	...
KKI1	E.ON

(a) Geben Sie die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten zu **AKW_ORT** an!

(b) Welche der drei Relationen oben sind in BCNF? Kurze Begründung!

Aufgabe 6:

Sei die Relation **AKW2** die Erweiterung von **AKW** aus Aufgabe 4 um das Attribut **KWBEZEICHNUNG**. Die drei Relationen **AKW_ORT**, **ORT_BLAND** und **AKW_BETREIBER** aus Aufgabe 5 sind dann die Aufteilung dieser Relation **AKW2**. Ist diese Aufteilung von **AKW2** verlustfrei? Begründung!

Hinweis: Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Trennen Sie zuerst **AKW2** in **AKW3** und **AKW_BETREIBER** und prüfen Sie, ob diese Aufteilung verlustfrei ist. Wenn ja, dann trennen Sie **AKW3** in **AKW_ORT** und **ORT_BLAND** und prüfen auch diese Aufteilung auf Verlustfreiheit.

Aufgabe 7:

Gesucht sind die Bundesländer (ohne Duplikate), in denen der Betreiber E.ON ein AKW betreibt. Wie lautet die SQL-Abfrage über die drei Relationen **AKW_ORT**, **ORT_BLAND** und **AKW_BETREIBER** aus Aufgabe 5 hierfür?

Aufgabe 8:

Wie lautet die SQL-Abfrage auf der Relation **AKW_ORT** aus Aufgabe 5, die genau die Gemeinden liefert, an denen mehr als ein AKW betrieben wird?

Aufgabe 9:

Welche Tupel liefert die folgende SQL-Abfrage über die Tabelle **AKW** aus Aufgabe 4?

```
SELECT BUNDESLAND, COUNT(*) AS ANZAHL  
FROM AKW  
GROUP BY BUNDESLAND  
ORDER BY BUNDESLAND ASC;
```

Aufgabe 10:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktionen T_i :

Transaktion T_1 (10:00 Uhr)	Transaktion T_2 (10:05 Uhr)	Transaktion T_3 (10:10 Uhr)
Lesen A Schreiben A	Lesen A Lesen B	Lesen A Lesen B
	Lesen C	Schreiben C
Schreiben B	Schreiben C	

(a) Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle solange ein, bis es ggf. einen Konflikt gibt.

(b) Geht die Ablauffolge durch? Wenn nein, markieren Sie die zurückgewiesene Operation!

(c) Gibt es eine Schreib-Operation, die „ignoriert“ wird, d. h. ohne Folgen nicht ausgeführt wird? Wenn ja, markieren Sie diese!

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)
9:00	9:00	9:00	9:00	9:00	9:00

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann ignoriere Schreiboperation

sonst Schreiben, $WTM(x) := TS$

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Sommersemester 2011**

Name: Vorname:

Matr.Nr.: Studiengang:

MUSTERLÖSUNG

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 90 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	6		8	4	
3	1+1+1+2		9	3	
4	2+2+2		10	2+1+1	
5	2+2		Summe	42	
6	4				

Aufgabe 1:

Bringen Sie die Begriffe der linken Spalte mit den Aussagen der rechten Spalte in Verbindung. Zu jedem Begriff gehört genau eine Aussage. Tragen Sie rechts den jeweiligen Buchstaben des Begriffs ein.

- | | |
|----------------------|---|
| (a) RID/TID | (b) Cursor-Stabilität / READ COMMITTED |
| (b) Isolationsgrad 2 | (c) disk interleaving ohne Redundanz |
| (c) RAID0 | (e) WITHIN LINK-DATA-AREA |
| (d) IS | (a) Tupeladresse |
| (e) Netzwerkmodell | (d) Sperre für gleichzeitiges Lesen angemeldet |

Aufgabe 2:

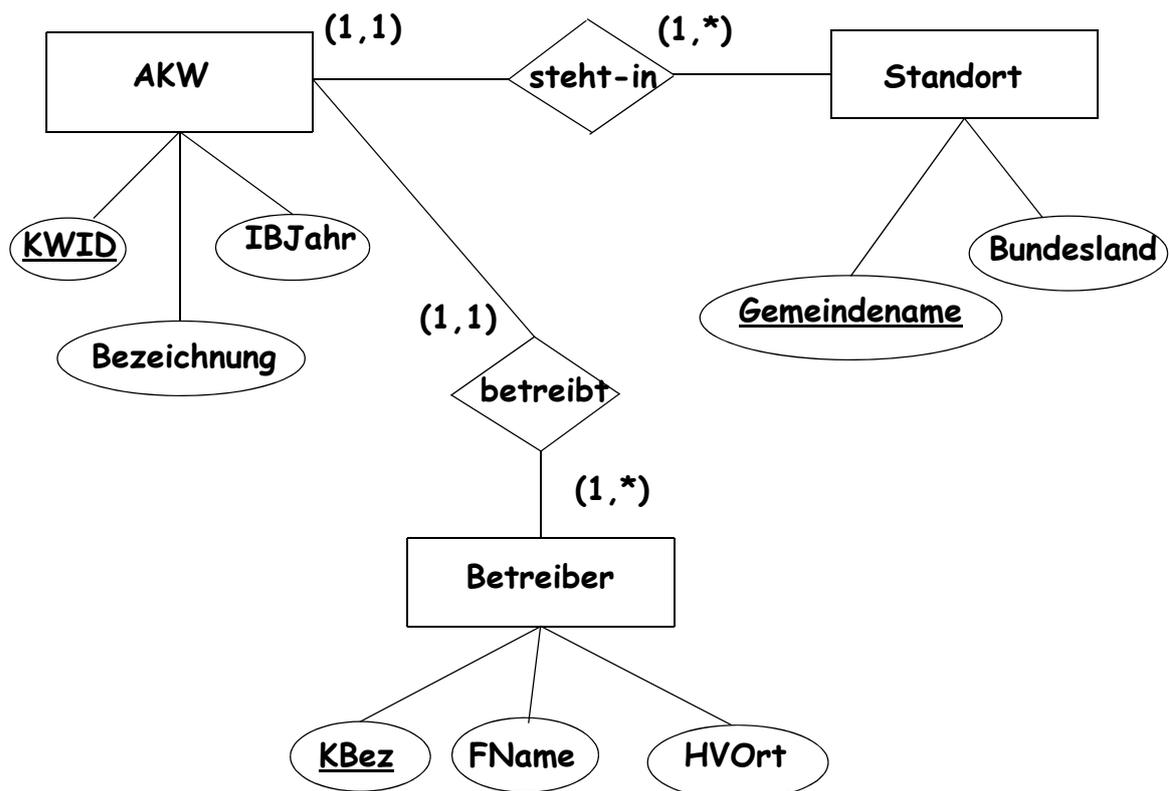
Als Miniwelt betrachten wir Atomkraftwerke, deren Standorte und Betreiber in Deutschland. Jedes Atomkraftwerk (**AKW**) habe einen eindeutigen Kraftwerksidentifizier (**KWID**), eine eindeutige **Bezeichnung**, ein Jahr der Inbetriebnahme (**IBJahr**). Ein Beispiel wäre (*KKI2, Isar/Ohu 2, 1988*).

Daneben gibt es **Standorte**, diese besitzen einen eindeutigen **Gemeindenamen** und ein **Bundesland**. Zwischen AKWs und Standorten gibt es die Beziehung **steht-in**. Jedes AKW hat genau einen Standort, an einem Standort muss mindestens ein, es können aber auch mehrere AKWs stehen. So stehen z. B. *Isar 1* und *Isar 2* in der Gemeinde *Markt Essenbach* in Bayern (*BY*).

Zuletzt gibt es Atomkraftwerksbetreiber (**Betreiber**). Diese sind durch einen eindeutigen Kurzbezeichner (**KBez**), Firmennamen (**FName**) und Hauptverwaltungsstandort (**HVOrt**) gekennzeichnet. Ein Beispiel wäre (*EnBW, Energie Baden-Württemberg AG, Karlsruhe*).

Jedes AKW wird von genau einem Betreiber betrieben, bekanntlich **betreiben** die Betreiber mindestens ein, in der Regel mehrere Kernkraftwerke.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung) für diese vereinfachte Miniwelt an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die Sie wählen würden.



Aufgabe 3:

Im ersten Entwurf einer Datenbank für Atomkraftwerke wurde eine Tabelle **ATOMKW** vorgesehen. Die Einträge sind beispielhaft für die Miniweltannahmen von Aufgabe 2.

ATOMKW

KWID	KWBEZEICHNUNG	GEMEINDE
KBR	Brokdorf	Brokdorf
KWBB	Biblis B	Biblis
KWBA	Biblis A	Biblis
KGCC	Grundremmingen C	Grundremmingen
KGGB	Grundremmingen B	Grundremmingen
KKP1	Philippsburg 1	Philippsburg
KKI2	Isar 2	Markt Essenbach
KKI1	Isar 1	Markt Essenbach

(a) Was sind die Schlüsselkandidaten?

{KWID}, {KWBEZEICHNUNG}

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

{KWID} -> ATOMKW

{KWBEZEICHNUNG} -> ATOMKW

(c) Gibt es eine transitive Abhängigkeit in **ATOMKW**? Wenn ja, welche?

Hinweis: Es muss gelten $X \rightarrow Y \rightarrow Z$ und $Y \not\rightarrow X$!

Nein!

(d) Ist **ATOMKW** in 3NF? Begründung!

Ja, da keine transitive Abhängigkeit (eines nicht-primen Attributs von einem Schlüssel) in ATOMKW existiert (siehe (c)).

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In der Tabelle **AKW** haben wir jetzt beispielhaft Kraftwerksidentifizier, Gemeinde, Bundesländer und Betreiber verknüpft. Man beachte, dass an einem Standort zwar mehrere AKWs stehen können, diese dann aber immer vom selben Betreiber betrieben werden.

AKW

KWID	GEMEINDE	BUNDES- LAND	BETREIBER
KBR	Brokdorf	SH	E.ON
KWBB	Biblis	HE	RWE
KWBA	Biblis	HE	RWE
KGGC	Grundremmingen	BY	RWE
KGGB	Grundremmingen	BY	RWE
KKP1	Philippsburg	BW	EnBW
KKI2	Markt Essenbach	BY	E.ON
KKI1	Markt Essenbach	BY	E.ON

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

SK: {KWID}

FA: {KWID} \rightarrow AKW (Schlüsselabhängigkeit)

GEMEINDE \rightarrow {BUNDESLAND, BETREIBER}

(b) Ist **AKW** in 2NF? Begründung!

Ja, da alle SK einelementig sind. Damit sind alle nicht-primen Attribute voll funktional von allen Schlüsselkandidaten abhängig.

(c) Ist **AKW** in 3NF? Begründung!

Nein, da KWID $\overset{\text{---}}{\rightarrow}$ GEMEINDE $\overset{\text{---}}{\rightarrow}$ BUNDESLAND

Schlüssel

nicht-primen Attribut

Aufgabe 5:

Für die folgenden Aufgaben betrachten wir eine **AKW_ORT**-, eine **ORT_BLAND**- und eine **AKW_BETREIBER**-Tabelle. Es gelten die Annahmen aus Aufgabe 2.

AKW_ORT

KWID	KWBEZEICHNUNG	GEMEINDE
KBR	Brokdorf	Brokdorf
KWBB	Biblis B	Biblis
KWBA	Biblis A	Biblis
KGGC	Grundremmingen C	Grundremmingen
...
KKI1	Isar 1	Markt Essenbach

ORT_BLAND

GEMEINDE	BUNDES-LAND
Brokdorf	SH
Biblis	HE
Grundremmingen	BY
Philippsburg	BW
Markt Essenbach	BY

AKW_BETREIBER

KWID	BETREIBER
KBR	E.ON
KWBB	RWE
KWBA	RWE
...	...
KKI1	E.ON

(a) Geben Sie die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten zu **AKW_ORT** an!

SK: {KWID}, {KWBEZEICHNUNG}

FA: {KWID} -> AKW_ORT

{KWBEZEICHNUNG} -> AKW_ORT

(b) Welche der drei Relationen oben sind in BCNF? Kurze Begründung!

AKW_ORT ist in BCNF, da alle nicht-trivialen FA von einem Schlüssel ausgehen. **ORT_BLAND** und **AKW_BETREIBER** sind auch in BCNF, da zweispaltig.

Aufgabe 6:

Sei die Relation **AKW2** die Erweiterung von **AKW** aus Aufgabe 4 um das Attribut **KWBEZEICHNUNG**. Die drei Relationen **AKW_ORT**, **ORT_BLAND** und **AKW_BETREIBER** aus Aufgabe 5 sind dann die Aufteilung dieser Relation **AKW2**. Ist diese Aufteilung von **AKW2** verlustfrei? Begründung!

Hinweis: Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Trennen Sie zuerst **AKW2** in **AKW3** und **AKW_BETREIBER** und prüfen Sie, ob diese Aufteilung verlustfrei ist. Wenn ja, dann trennen Sie **AKW3** in **AKW_ORT** und **ORT_BLAND** und prüfen auch diese Aufteilung auf Verlustfreiheit.

AKW2(KWID, GEMEINDE, BUNDESLAND, BETREIBER, KWBEZEICHNUNG)

AKW3(KWID, GEMEINDE, BUNDESLAND, KWBEZEICHNUNG)

AKW_BETREIBER(KWID, BETREIBER)

(1.) **AKW3 \cap AKW_BETREIBER = {KWID} und KWID \rightarrow AKW_BETREIBER.**
Damit ist die Aufteilung von **AKW2** in **AKW3** und **AKW_BETREIBER** verlustfrei.

(2.) **AKW_ORT \cap ORT_BLAND = {GEMEINDE} und GEMEINDE \rightarrow ORT_BLAND.**
Damit ist auch die Aufteilung von **AKW3** in **AKW_ORT** und **ORT_BLAND** verlustfrei und somit auch die Aufteilung von **AKW2** insgesamt.

Aufgabe 7:

Gesucht sind die Bundesländer (ohne Duplikate), in denen der Betreiber E.ON ein AKW betreibt. Wie lautet die SQL-Abfrage über die drei Relationen **AKW_ORT**, **ORT_BLAND** und **AKW_BETREIBER** aus Aufgabe 5 hierfür?

```
SELECT DISTINCT BUNDESLAND  
FROM AKW_ORT, ORT_BLAND, AKW_BETREIBER  
WHERE AKW_ORT.KWID = AKW_BETREIBER.KWID  
AND AKW_ORT.GEMEINDE = ORT_BLAND.GEMEINDE  
AND BETREIBER = 'E.ON';
```

Aufgabe 8:

Wie lautet die SQL-Abfrage auf der Relation **AKW_ORT** aus Aufgabe 5, die genau die Gemeinden liefert, an denen mehr als ein AKW betrieben wird?

```
SELECT GEMEINDE  
FROM AKW_ORT  
GROUP BY GEMEINDE  
HAVING COUNT(KWID) > 1;
```

Aufgabe 9:

Welche Tupel liefert die folgende SQL-Abfrage über die Tabelle **AKW** aus Aufgabe 4?

```
SELECT BUNDESLAND, COUNT(*) AS ANZAHL  
FROM AKW  
GROUP BY BUNDESLAND  
ORDER BY BUNDESLAND ASC;
```

BUNDESLAND	ANZAHL
BW	1
BY	4
HE	2
SH	1

Aufgabe 10:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktionen T_i :

Transaktion T_1
(10:00 Uhr)

Lesen A
Schreiben A

Transaktion T_2
(10:05 Uhr)

Lesen A
Lesen B

Transaktion T_3
(10:10 Uhr)

Lesen A
Lesen B

Schreiben C

zurückgewiesene Op.



Lesen C



ignorierte Op.

(a) Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle solange ein, bis es ggf. einen Konflikt gibt.

(b) Geht die Ablauffolge durch? Wenn nein, markieren Sie die zurückgewiesene Operation!

(c) Gibt es eine Schreib-Operation, die „ignoriert“ wird, d. h. ohne Folgen nicht ausgeführt wird? Wenn ja, markieren Sie diese!

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)
9:00 10:00 10:05 10:10	9:00 10:00	9:00 10:05 10:10	9:00	9:00 10:05	9:00 10:10

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann ignoriere Schreiboperation

sonst Schreiben, $WTM(x) := TS$

ENDE DER KLAUSUR

Klausur zur Vorlesung Datenbanken I im Wintersemester 2011/12

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Rechtlicher Hinweis: Die Auswahl der in den Aufgaben genannten Kreditinstitute und Finanzprodukte ist willkürlich und dient nur der besseren Veranschaulichung der Miniwelt.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	6		8	5	
3	2+2+2+1+1		9	4	
4	4		10	4	
5	2+2+2+2		Summe	48	
6	3				

Aufgabe 1:

Welche der folgenden Aussagen sind richtig in Zusammenhang mit dem Transaktionskonzept?

- () Die *Logical Sequence Number* wird vom Anwender bei einer Insert-Operation angegeben.
- () Bei *optimistischer Nebenläufigkeitskontrolle* verzichtet man auf vorausschauende Sperren.
- () *Dirty Read* ist das Lesen von Daten aus fremden Transaktionen, die noch kein Commit gemacht haben.
- () Beim Zeitmarkenverfahren gibt es sowohl *ignorierbares Schreiben* als auch *ignorierbares Lesen*.
- () Wird bei einem Sperrkonflikt die betroffene Transaktion immer sofort abgebrochen, drohen Transaktionen auszuhungern, dafür gibt es keine Verklemmungen.

Aufgabe 2:

In dieser Klausur geht es um die Vermittlung von Baugeld. In unserer vereinfachten Miniwelt gibt es **Kreditgeber** für Baugeld. Kreditgeber sind gekennzeichnet durch eine eindeutige Kurzbezeichnung (**Kurzbez**), einen eindeutigen Namen (**Name**) und einen **Ort**, an dem sie firmieren.

Baugeld gibt es in verschiedenen Darlehensarten (**Dart**), die sich u. a. durch die Länge der Zinsbindung in Jahren (**Laufzeit**) und durch eine Beleihungsobergrenze (**Grenze**) in Prozent der Bausumme klassifizieren lassen.

Kreditgeber **bieten** Darlehensarten **an**, und zwar jeder mindestens eine, in der Regel mehrere. Umgekehrt kann eine Darlehensart von mehreren Kreditgebern angeboten werden und wir nehmen nur solche Darlehensarten auf, für die es mindestens einen Anbieter gibt. Ganz wesentlich ist dabei natürlich der effektive Zinssatz (**EZins**), zu dem ein spezieller Kreditgeber eine spezielle Darlehensart anbietet.

Zuletzt haben wir **Kunden** mit Kreditwünschen. Kunden sind durch einen eindeutigen Kundenamen (**KName**), einer gewünschten **Darlehenshöhe** (in TEuro) und einen Immobilienwert (**Immowert**) gekennzeichnet.

Kunden **verhandeln mit** Kreditgebern, auf beiden Seiten jeweils mit keinem oder mehreren. Kunden **interessieren sich** für mindestens eine oder auch mehrere Darlehensarten, eine Darlehensart kann momentan von keinem Kunden nachgefragt werden, in der Regel interessieren sich aber viele Kunden für eine Darlehensart.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung und nur mit binären Beziehungen) für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie für jede Entität Ihren bevorzugten Schlüssel!

Aufgabe 3:

Hier sehen wir eine Tabelle **Darlehen** mit Kreditgebern und angebotenen Konditionen. Die Einträge spiegeln die Miniweltannahmen von Aufgabe 2 wider.

Darlehen

Kurzbez	Name	Laufzeit	Grenze	EZins
KS SPK	Kasseler Sparkasse	5	40	2,75
KS SPK	Kasseler Sparkasse	5	60	2,75
Voba Baunatal	Volksbank Baunatal	5	60	2,80
Voba Baunatal	Volksbank Baunatal	5	80	2,85
KS SPK	Kasseler Sparkasse	10	40	3,05
KS SPK	Kasseler Sparkasse	10	80	3,20
Voba Baunatal	Volksbank Baunatal	10	60	3,00
Voba Baunatal	Volksbank Baunatal	10	80	3,20
DB KS	Deutsche Bank Kassel	10	80	3,15
DB KS	Deutsche Bank Kassel	15	80	3,15

- (a) Was sind die Schlüsselkandidaten?

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

- (c) Gibt es eine transitive Abhängigkeit eines nicht-primen Attributs in **Darlehen**?
Wenn ja, welche?

- (d) Ist **Darlehen** in 3NF? Begründung! Hinweis: Definition 3NF umseitig

- (e) Ist **Darlehen** in BCNF? Begründung!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Um sich einen Überblick zu verschaffen, welche Darlehensarten von welcher Bank überhaupt derzeit angeboten werden, entstand die folgende Tabelle **Anbieter**.

Anbieter

Kurzbez	Ort	Laufzeit	Grenze
KS SPK	Kassel	5	40
KS SPK	Kassel	5	60
Voba Baunatal	Baunatal	5	60
Voba Baunatal	Baunatal	5	80
KS SPK	Kassel	10	40
KS SPK	Kassel	10	80
Voba Baunatal	Baunatal	10	60
Voba Baunatal	Baunatal	10	80
DB KS	Kassel	10	80
DB KS	Kassel	15	80

Ist **Anbieter** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Aufgabe 5:

Wir trennen **Anbieter** von Aufgabe 4 mit zusätzlichen Attributen **Name** und **EZins** in **Angebote** und **Kreditgeber**.

Angebote

Kurzbez	Laufzeit	Grenze	EZins
KS SPK	5	40	2,75
KS SPK	5	60	2,75
Voba Baunatal	5	60	2,80
Voba Baunatal	5	80	2,85
KS SPK	10	40	3,05
KS SPK	10	80	3,20
Voba Baunatal	10	60	3,00
Voba Baunatal	10	80	3,20
DB KS	10	80	3,15
DB KS	15	80	3,15

Kreditgeber

Kurzbez	Name	Ort
DB KS	Deutsche Bank Kassel	Kassel
Voba Baunatal	Volksbank Baunatal	Baunatal
KS SPK	Kasseler Sparkasse	Kassel

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

(b) Sind die Relationen **Angebote** und **Kreditgeber** in BCNF? Kurze Begründung!

(c) Ist diese Aufteilung von **Anbieter** aus Aufgabe 4 (mit den zusätzlichen Attributen **Name** und **EZins**) verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

(d) Welche hierarchische Beziehung gibt es zwischen **Angebote** und **Kreditgeber**?

Fremdschlüssel _____ in Tabelle _____ verweist auf

Primärschlüssel _____ in Tabelle _____ .

Aufgabe 6:

Mit Hilfe der Relationen **Angebote** und **Kreditgeber** aus Aufgabe 5 sollen die Namen aller Kreditgeber ermittelt werden, die Angebote mit Zinsbindung (Laufzeit) von fünf Jahren haben. Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

Aufgabe 7:

Wie lautet die äquivalente SQL-Abfrage zu Aufgabe 6, wenn man zusätzlich noch den Zinssatz und die Beleihungsobergrenze bei jedem Kreditgeber will?

Aufgabe 8:

Mit welchem SQL-Befehl erstellt man aus der Relation **Darlehen** in Aufgabe 3 den zweispaltigen View **Zinsniveau**, der für alle Angebote mit einer Grenze ≥ 50 den durchschnittlichen Zinssatz als Spalte DSZins gruppiert nach Laufzeit ausgibt.

Aufgabe 9:

Betrachten Sie die folgende Abfrage für die Tabelle **Darlehen** aus Aufgabe 3:

```
SELECT Kurzbez, Laufzeit, Grenze, EZins
FROM Darlehen
WHERE (Laufzeit, Grenze, EZins) IN
      (SELECT Laufzeit, Grenze, MIN(EZins)
       FROM Darlehen
       WHERE Laufzeit <= 5
       GROUP BY Laufzeit, Grenze);
```

Wie lautet die Resultattabelle?

Aufgabe 10:

Betrachten Sie die folgende Ablauffolge:

Transaktion 1	Transaktion 2	Transaktion 3	Transaktion 4
Lesen A Schreiben A			
Lesen B Schreiben B	Lesen A		
	Lesen C Lesen D		
		Lesen C	Lesen B Schreiben B
	Schreiben C	Lesen D Schreiben D	Lesen D Schreiben D

Geben Sie den vollständigen Abhängigkeitsgraphen an! An jeder gerichteten Kante sollen die Datensätze stehen (A, B, C oder D), die einen Konflikt verursachen. Ist die Ablauffolge serialisierbar? Wenn nein, welche Transaktionen stehen in Konflikt?

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Wintersemester 2011/12**

Name: Vorname:

Matr.Nr.: Studiengang:

MUSTERLÖSUNG

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Rechtlicher Hinweis: Die Auswahl der in den Aufgaben genannten Kreditinstitute und Finanzprodukte ist willkürlich und dient nur der besseren Veranschaulichung der Miniwelt.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	6		8	5	
3	2+2+2+1+1		9	4	
4	4		10	4	
5	2+2+2+2		Summe	48	
6	3				

Aufgabe 1:

Welche der folgenden Aussagen sind richtig in Zusammenhang mit dem Transaktionskonzept?

- () Die *Logical Sequence Number* wird vom Anwender bei einer Insert-Operation angegeben.
- (X) Bei *optimistischer Nebenläufigkeitskontrolle* verzichtet man auf vorausschauende Sperren.
- (X) *Dirty Read* ist das Lesen von Daten aus fremden Transaktionen, die noch kein Commit gemacht haben.
- () Beim Zeitmarkenverfahren gibt es sowohl *ignorierbares Schreiben* als auch *ignorierbares Lesen*.
- (X) Wird bei einem Sperrkonflikt die betroffene Transaktion immer sofort abgebrochen, drohen Transaktionen auszuhungern, dafür gibt es keine Verklemmungen.

Aufgabe 2:

In dieser Klausur geht es um die Vermittlung von Baugeld. In unserer vereinfachten Miniwelt gibt es **Kreditgeber** für Baugeld. Kreditgeber sind gekennzeichnet durch eine eindeutige Kurzbezeichnung (**Kurzbez**), einen eindeutigen Namen (**Name**) und einen **Ort**, an dem sie firmieren.

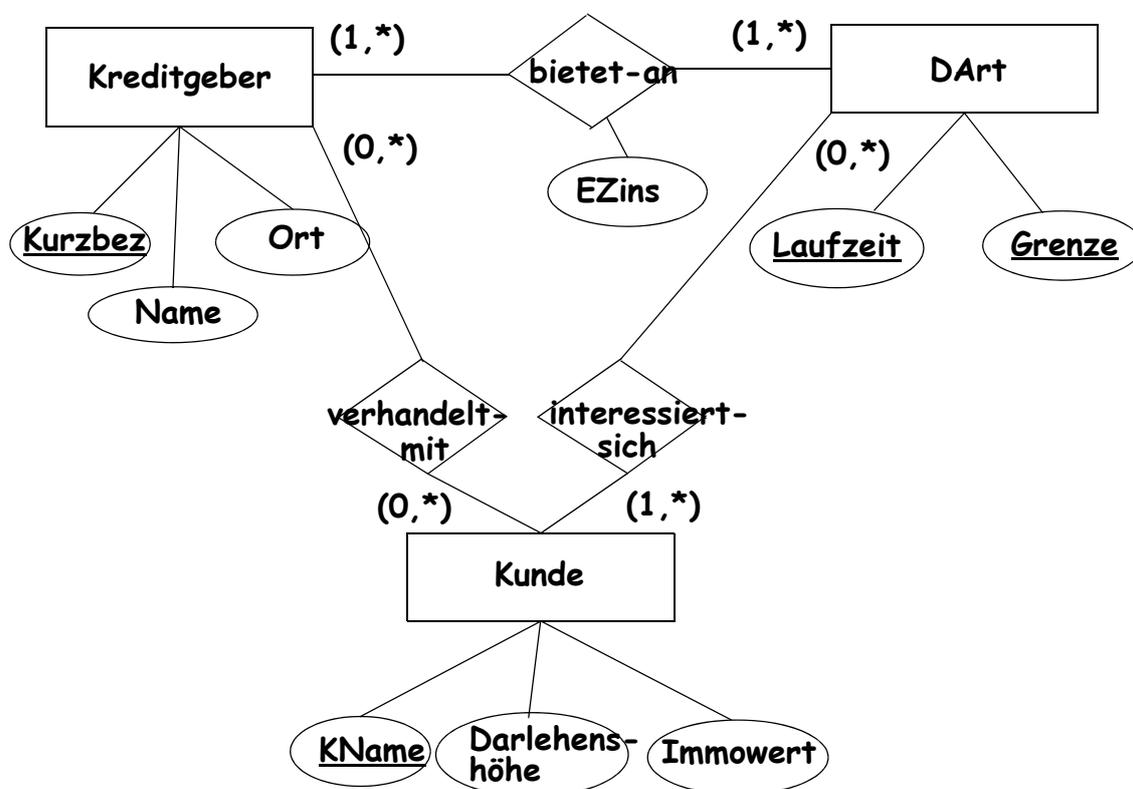
Baugeld gibt es in verschiedenen Darlehensarten (**DArt**), die sich u. a. durch die Länge der Zinsbindung in Jahren (**Laufzeit**) und durch eine Beleihungsobergrenze (**Grenze**) in Prozent der Bausumme klassifizieren lassen.

Kreditgeber **bieten** Darlehensarten **an**, und zwar jeder mindestens eine, in der Regel mehrere. Umgekehrt kann eine Darlehensart von mehreren Kreditgebern angeboten werden und wir nehmen nur solche Darlehensarten auf, für die es mindestens einen Anbieter gibt. Ganz wesentlich ist dabei natürlich der effektive Zinssatz (**EZins**), zu dem ein spezieller Kreditgeber eine spezielle Darlehensart anbietet.

Zuletzt haben wir **Kunden** mit Kreditwünschen. Kunden sind durch einen eindeutigen Kundenamen (**KName**), einer gewünschten **Darlehenshöhe** (in TEuro) und einen Immobilienwert (**Immowert**) gekennzeichnet.

Kunden **verhandeln mit** Kreditgebern, auf beiden Seiten jeweils mit keinem oder mehreren. Kunden **interessieren sich** für mindestens eine oder auch mehrere Darlehensarten, eine Darlehensart kann momentan von keinem Kunden nachgefragt werden, in der Regel interessieren sich aber viele Kunden für eine Darlehensart.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung und nur mit binären Beziehungen) für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie für jede Entität Ihren bevorzugten Schlüssel!



Aufgabe 3:

Hier sehen wir eine Tabelle **Darlehen** mit Kreditgebern und angebotenen Konditionen. Die Einträge spiegeln die Miniweltannahmen von Aufgabe 2 wider.

Darlehen

Kurzbez	Name	Laufzeit	Grenze	EZins
KS SPK	Kasseler Sparkasse	5	40	2,75
KS SPK	Kasseler Sparkasse	5	60	2,75
Voba Baunatal	Volksbank Baunatal	5	60	2,80
Voba Baunatal	Volksbank Baunatal	5	80	2,85
KS SPK	Kasseler Sparkasse	10	40	3,05
KS SPK	Kasseler Sparkasse	10	80	3,20
Voba Baunatal	Volksbank Baunatal	10	60	3,00
Voba Baunatal	Volksbank Baunatal	10	80	3,20
DB KS	Deutsche Bank Kassel	10	80	3,15
DB KS	Deutsche Bank Kassel	15	80	3,15

(a) Was sind die Schlüsselkandidaten?

{Kurzbez, Laufzeit, Grenze}

{Name, Laufzeit, Grenze}

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

{Kurzbez, Laufzeit, Grenze} -> Darlehen; Kurzbez -> Name

{Name, Laufzeit, Grenze} -> Darlehen; Name -> Kurzbez

(c) Gibt es eine transitive Abhängigkeit eines nicht-primen Attributs in **Darlehen**?
Wenn ja, welche?

Nein!

(d) Ist **Darlehen** in 3NF? Begründung! Hinweis: Definition 3NF umseitig

Ja! Folgt aus (c) oder weil für X -> A X Schlüssel ist (auch wenn A EZins ist) und im Fall Name -> Kurzbez bzw. Kurzbez -> Name, A prim ist.

(e) Ist **Darlehen** in BCNF? Begründung!

Nein, „A ist prim“ fällt weg und in Kurzbez -> Name ist Kurzbez kein Schlüssel.

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Um sich einen Überblick zu verschaffen, welche Darlehensarten von welcher Bank überhaupt derzeit angeboten werden, entstand die folgende Tabelle **Anbieter**.

Anbieter

Kurzbez	Ort	Laufzeit	Grenze
KS SPK	Kassel	5	40
KS SPK	Kassel	5	60
Voba Baunatal	Baunatal	5	60
Voba Baunatal	Baunatal	5	80
KS SPK	Kassel	10	40
KS SPK	Kassel	10	80
Voba Baunatal	Baunatal	10	60
Voba Baunatal	Baunatal	10	80
DB KS	Kassel	10	80
DB KS	Kassel	15	80

Ist **Anbieter** in 3NF? Begründung! Hinweis: Bestimmen Sie zuerst die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Schlüsselkandidaten: {Kurzbez, Laufzeit, Grenze}

Nicht-triviale FA: {Kurzbez, Laufzeit, Grenze} -> Anbieter (Schlüsselabh.)

Kurzbez -> Ort

Anbieter ist nicht in 3NF, da Kurzbez -> Ort nicht trivial, Kurzbez nicht Schlüssel und Ort nicht prim, d. h. nicht Teil eines Schlüssels ist.

Alternative Begründung: es existiert eine transitive Abhängigkeit

{Kurzbez, Laufzeit, Grenze} --> Kurzbez --> Ort

Schlüssel

<-/-

nicht prim

Aufgabe 5:

Wir trennen **Anbieter** von Aufgabe 4 mit zusätzlichen Attributen **Name** und **EZins** in **Angebote** und **Kreditgeber**.

Angebote

Kurzbez	Laufzeit	Grenze	EZins
KS SPK	5	40	2,75
KS SPK	5	60	2,75
Voba Baunatal	5	60	2,80
Voba Baunatal	5	80	2,85
KS SPK	10	40	3,05
KS SPK	10	80	3,20
Voba Baunatal	10	60	3,00
Voba Baunatal	10	80	3,20
DB KS	10	80	3,15
DB KS	15	80	3,15

Kreditgeber

Kurzbez	Name	Ort
DB KS	Deutsche Bank Kassel	Kassel
Voba Baunatal	Volksbank Baunatal	Baunatal
KS SPK	Kasseler Sparkasse	Kassel

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

Angebote: SK ist {Kurzbez, Laufzeit, Grenze}

FA ist {Kurzbez, Laufzeit, Grenze} -> Angebote (Schlüsselabh.)

Kreditgeber: SK sind {Kurzbez}, {Name} (nicht {Kurzbez, Name} !!)

FA sind {Kurzbez} -> Kreditgeber und {Name} -> Kreditgeber

beides Schlüsselabhängigkeiten

(b) Sind die Relationen **Angebote** und **Kreditgeber** in BCNF? Kurze Begründung!

Angebote: Ja! Alle nicht-trivialen FA gehen vom Schlüssel aus.

Kreditgeber: Ja! dito¹

1. dito = wie bereits gesagt. Also: alle nicht-trivialen FA ...

(c) Ist diese Aufteilung von **Anbieter** aus Aufgabe 4 (mit den zusätzlichen Attributen **Name** und **EZins**) verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

$R_1 = \text{Angebote}; R_2 = \text{Kreditgeber}$

$R_1 \cap R_2 = \{\text{Kurzbez}\} \text{ und } \text{Kurzbez} \rightarrow R_2$

Also ist die Aufteilung verlustfrei.

(d) Welche hierarchische Beziehung gibt es zwischen **Angebote** und **Kreditgeber**?

Fremdschlüssel **Kurzbez** in Tabelle **Angebote** verweist auf

Primärschlüssel **Kurzbez** in Tabelle **Kreditgeber** .

Aufgabe 6:

Mit Hilfe der Relationen **Angebote** und **Kreditgeber** aus Aufgabe 5 sollen die Namen aller Kreditgeber ermittelt werden, die Angebote mit Zinsbindung (Laufzeit) von fünf Jahren haben. Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

$\Pi_{\text{Name}}(\sigma_{\text{Laufzeit} = 5}(\text{Angebote}) \bowtie \text{Kreditgeber})$

Aufgabe 7:

Wie lautet die äquivalente SQL-Abfrage zu Aufgabe 6, wenn man zusätzlich noch den Zinssatz und die Beleihungsobergrenze bei jedem Kreditgeber will?

```
SELECT Name, EZins, Grenze
FROM Angebote, Kreditgeber
WHERE Angebote.Kurzbez = Kreditgeber.Kurzbez
      AND Laufzeit = 5;
```

Aufgabe 8:

Mit welchem SQL-Befehl erstellt man aus der Relation **Darlehen** in Aufgabe 3 den zweispaltigen View **Zinsniveau**, der für alle Angebote mit einer Grenze ≥ 50 den durchschnittlichen Zinssatz als Spalte DSZins gruppiert nach Laufzeit ausgibt.

```
CREATE VIEW ZinsNiveau AS
  SELECT Laufzeit, AVG(EZins) DSZins
  FROM Darlehen
  WHERE Grenze  $\geq$  50
  GROUP BY Laufzeit;
```

Aufgabe 9:

Betrachten Sie die folgende Abfrage für die Tabelle **Darlehen** aus Aufgabe 3:

```
SELECT Kurzbez, Laufzeit, Grenze, EZins
FROM Darlehen
WHERE (Laufzeit, Grenze, EZins) IN
  (SELECT Laufzeit, Grenze, MIN(EZins)
   FROM Darlehen
   WHERE Laufzeit  $\leq$  5
   GROUP BY Laufzeit, Grenze);
```

Wie lautet die Resultattabelle?

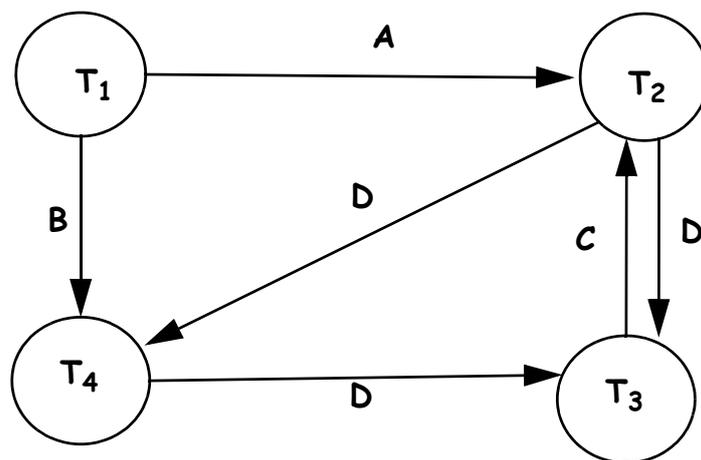
Kurzbez	Laufzeit	Grenze	EZins
KS SPK	5	40	2,75
KS SPK	5	60	2,75
Voba Baunatal	5	80	2,85

Aufgabe 10:

Betrachten Sie die folgende Ablauffolge:

Transaktion 1	Transaktion 2	Transaktion 3	Transaktion 4
Lesen A Schreiben A			
	Lesen A		
Lesen B Schreiben B			
	Lesen C Lesen D		
		Lesen C	Lesen B Schreiben B
	Schreiben C		Lesen D Schreiben D
		Lesen D Schreiben D	

Geben Sie den vollständigen Abhängigkeitsgraphen an! An jeder gerichteten Kante sollen die Datensätze stehen (A, B, C oder D), die einen Konflikt verursachen. Ist die Ablauffolge serialisierbar? Wenn nein, welche Transaktionen stehen in Konflikt?



Die Ablauffolge ist nicht serialisierbar. T₂, T₃ und T₄ stehen in Konflikt, es gibt zwei Zyklen.

ENDE DER KLAUSUR

Klausur zur Vorlesung Datenbanken I im Sommersemester 2012

Name: Vorname:

Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Bei Ankreuzaufgaben kann mehr als eine Antwort richtig sein. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Ferner sind die in dieser Klausur genannten Markennamen zufällig gewählt und dienen nur der Veranschaulichung. Das gezeigte Anwendungsbeispiel ist fiktiv.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	3	
2	6		8	4	
3	1+2+2+2		9	2+2	
4	2+2		10	4	
5	2+3+2+3		Summe	48	
6	4				

Aufgabe 1:

Welche Aussagen sind richtig in Zusammenhang mit dem Record Identifier/Tupel Identifier Speicherkonzept?

- () Ein RID/TID setzt sich zusammen aus einer Seitennummer und einer Indexnummer für die Adresstabelle in der Seite.
- () Ziehen Datensätze wegen Größenänderungen mehrfach um, können sehr lange Verweisketten entstehen.
- () Das RID-/TID-Speicherkonzept erfordert eine große Umsetztabelle, die vorzugsweise im Hauptspeicher gehalten werden sollte.
- () Ein für ein Datenobjekt einmal vergebener RID/TID bleibt auch bei Umzügen wegen Größenänderungen unverändert (invariant).
- () Der beim Löschen eines Datenobjekts in einer Seite freiwerdende Platz ist verloren und kann nicht wiederbenutzt werden. Dies erfordert periodisch eine komplette Reorganisation des Plattenspeichers.

Aufgabe 2:

Wir betrachten ein regionales Internet-Portal für den Gebrauchtwagenkauf. Darin gibt es **Händler**, die durch einen eindeutigen Händleridentifikator (**HID**), einen eindeutigen Händlernamen (**HNAME**) und einen Standort (**ORT**), an dem sich der Händler befindet, beschrieben werden.

Zweitens gibt es **Autos**. Diese sind gekennzeichnet durch einen vom Portal vergebenen Schlüssel **AID**, eine Modellbezeichnung (**MODELL**), die Laufleistung (**KM**), das Erstzulassungsjahr (**EZJAHR**) und einen Angebotspreis (**PREIS**).

Zwischen Händlern und Autos gibt es die Beziehung „**steht-bei**“, wobei ein Auto bei genau einem Händler steht und ein Händler mindestens ein Auto, in der Regel aber viele, anbietet.

Zuletzt gibt es noch interessierte Käufer (**Suchende**), die sich durch einen eindeutigen Namen (**SNAME**), einen **WOHNORT** und ein Preislimit (**PLIMIT**) auszeichnen. Interessiert sich im Portal ein Suchender für ein Auto, dann „parkt“ er es. Das führt zu einer Beziehung „**parken**“, bei der ein Suchender kein oder mehrere Autos parken kann und ein Auto von keinem oder mehreren Suchenden geparkt wird.

Geben Sie ein ER-Diagramm für diese Miniwelt mit drei Entity-Typen an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die in dieser Miniwelt sinnvoll sind.

Aufgabe 3:

Im ersten Entwurf für einen Teil des Datenbankschemas der Miniwelt aus Aufgabe 2 wurde eine Tabelle **AH** entworfen, die Autos mit Angabe des Händlers und des Orts, an dem sie stehen, zeigt.

AH

AID	MODELL	HID	HNAME	ORT
1001	Golf	G&G	Gut und Günstig	Kassel
1002	Focus	G&G	Gut und Günstig	Kassel
1003	Astra	HCD	Hot Car Deals	Baunatal
1004	Astra	HCD	Hot Car Deals	Baunatal
1005	Golf	HCD	Hot Car Deals	Baunatal
1006	Punto	PP	Prima preiswert	Kassel
1007	Passat	PP	Prima preiswert	Kassel

- (a) Was sind die Schlüsselkandidaten?
- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?
- (c) Gibt es transitive Abhängigkeiten in **AH**? Wenn ja, nennen Sie eine!
Hinweis: Es muss gelten $X \rightarrow Y \rightarrow Z$ und $Y \not\rightarrow X$!
- (d) Ist **AH** in 3NF? Begründung!

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Für BCNF entfällt „ $A \in R$ ist prim“. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In die Tabelle **Autos** haben wir zusätzlich zu den Angaben der Autos noch den Händleridentifikator aufgenommen, der angibt, bei welchem Händler das Auto steht.

Autos

AID	MODELL	KM	EZJAHR	PREIS	HID
1001	Golf	95000	2005	6700	G&G
1002	Focus	130000	2002	4300	G&G
1003	Astra	170000	1999	2100	HCD
1004	Astra	60000	2004	7800	HCD
1005	Golf	190000	2001	2800	HCD
1006	Punto	75000	2002	2800	PP
1007	Passat	280000	2001	3900	PP

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

(b) Was ist die höchste Normalform, in der **Autos** ist? Begründung!

Aufgabe 7:

Wir suchen Autos, die bei Händlern am eigenen Wohnort stehen. Wie lautet der SQL-Befehl dazu, der aus den Tabellen **AH** von Aufgabe 3 und **Suchende** von Aufgabe 5 die Namen der Suchenden (**SNAME**), ihren **WOHNORT**, den Automobilidentifizier **AID** und das Modell (**MODELL**) ausgibt, sofern eben das Auto bei einem Händler vor Ort steht?

Aufgabe 8:

Gesucht sind der Händleridentifizier **HID** und der Durchschnittspreis aller Fahrzeuge, die dieser Händler anbietet. Verwenden Sie für den SQL-Befehl die Tabelle **Autos** aus Aufgabe 4.

Aufgabe 9:

Was liefert die folgende SQL-Abfrage auf den Tabellen aus Aufgabe 5?

```
SELECT *  
FROM Modelle M1  
WHERE NOT EXISTS (SELECT * FROM Parken  
                  WHERE M1.AID = Parken.AID)
```

(a) Geben Sie die Ergebnistupel für die Tabellenbelegung aus Aufgabe 5 an.

(b) Beschreiben Sie in Worten das Ergebnis auf beliebigen Tabelleninhalten für die Schemata aus Aufgabe 5!

Aufgabe 10:

Betrachten Sie die unten gezeigte Ablauffolge. Geben Sie den vollständigen Abhängigkeitsgraphen dazu an! Ist die Ablauffolge serialisierbar? Begründung!

Transaktion T₁

Lesen B

Schreiben B

Transaktion T₂

Lesen A

Schreiben A

Lesen C

Schreiben C

Transaktion T₃

Lesen B

Lesen C

Schreiben C

Schreiben B

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Sommersemester 2012**

Name: Vorname:

Matr.Nr.: Studiengang:

MUSTERLÖSUNG

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Bei Ankreuzaufgaben kann mehr als eine Antwort richtig sein. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Ferner sind die in dieser Klausur genannten Markennamen zufällig gewählt und dienen nur der Veranschaulichung. Das gezeigte Anwendungsbeispiel ist fiktiv.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	3	
2	6		8	4	
3	1+2+2+2		9	2+2	
4	2+2		10	4	
5	2+3+2+3		Summe	48	
6	4				

Aufgabe 1:

Welche Aussagen sind richtig in Zusammenhang mit dem Record Identifier/Tupel Identifier Speicherkonzept?

- Ein RID/TID setzt sich zusammen aus einer Seitennummer und einer Indexnummer für die Adresstabelle in der Seite.
- Ziehen Datensätze wegen Größenänderungen mehrfach um, können sehr lange Verweisketten entstehen.
- Das RID-/TID-Speicherkonzept erfordert eine große Umsetztabelle, die vorzugsweise im Hauptspeicher gehalten werden sollte.
- Ein für ein Datenobjekt einmal vergebener RID/TID bleibt auch bei Umzügen wegen Größenänderungen unverändert (invariant).
- Der beim Löschen eines Datenobjekts in einer Seite freiwerdende Platz ist verloren und kann nicht wiederbenutzt werden. Dies erfordert periodisch eine komplette Reorganisation des Plattenspeichers.

Aufgabe 2:

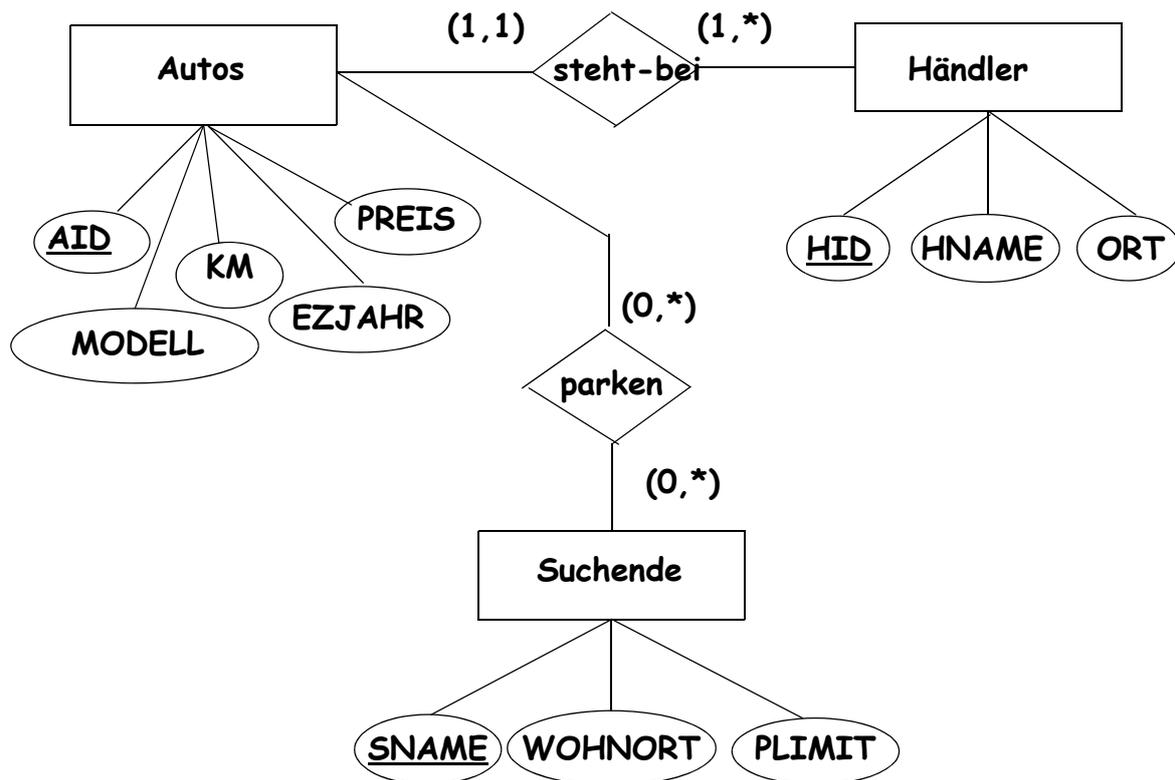
Wir betrachten ein regionales Internet-Portal für den Gebrauchtwagenkauf. Darin gibt es **Händler**, die durch einen eindeutigen Händleridentifikator (**HID**), einen eindeutigen Händlernamen (**HNAME**) und einen Standort (**ORT**), an dem sich der Händler befindet, beschrieben werden.

Zweitens gibt es **Autos**. Diese sind gekennzeichnet durch einen vom Portal vergebenen Schlüssel **AID**, eine Modellbezeichnung (**MODELL**), die Laufleistung (**KM**), das Erstzulassungsjahr (**EZJAHR**) und einen Angebotspreis (**PREIS**).

Zwischen Händlern und Autos gibt es die Beziehung „**steht-bei**“, wobei ein Auto bei genau einem Händler steht und ein Händler mindestens ein Auto, in der Regel aber viele, anbietet.

Zuletzt gibt es noch interessierte Käufer (**Suchende**), die sich durch einen eindeutigen Namen (**SNAME**), einen **WOHNORT** und ein Preislimit (**PLIMIT**) auszeichnen. Interessiert sich im Portal ein Suchender für ein Auto, dann „parkt“ er es. Das führt zu einer Beziehung „**parken**“, bei der ein Suchender kein oder mehrere Autos parken kann und ein Auto von keinem oder mehreren Suchenden geparkt wird.

Geben Sie ein ER-Diagramm für diese Miniwelt mit drei Entity-Typen an. Verwenden Sie nur binäre Relationen. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie die Primärschlüssel, die in dieser Miniwelt sinnvoll sind.



Aufgabe 3:

Im ersten Entwurf für einen Teil des Datenbankschemas der Miniwelt aus Aufgabe 2 wurde eine Tabelle **AH** entworfen, die Autos mit Angabe des Händlers und des Orts, an dem sie stehen, zeigt.

AH

AID	MODELL	HID	HNAME	ORT
1001	Golf	G&G	Gut und Günstig	Kassel
1002	Focus	G&G	Gut und Günstig	Kassel
1003	Astra	HCD	Hot Car Deals	Baunatal
1004	Astra	HCD	Hot Car Deals	Baunatal
1005	Golf	HCD	Hot Car Deals	Baunatal
1006	Punto	PP	Prima preiswert	Kassel
1007	Passat	PP	Prima preiswert	Kassel

- (a) Was sind die Schlüsselkandidaten?

{AID}

- (b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

{AID} -> AH -- die Schlüsselabhängigkeit

HID -> {HNAME, ORT} -- kann auch getrennt geschrieben werden

HNAME -> {HID, ORT}

- (c) Gibt es transitive Abhängigkeiten in **AH**? Wenn ja, nennen Sie eine!

Hinweis: Es muss gelten $X \rightarrow Y \rightarrow Z$ und $Y \not\rightarrow X$!

**Ja, AID ----> HID ----> ORT
<-/-**

- (d) Ist **AH** in 3NF? Begründung!

Nein, es existiert eine transitive Abhängigkeit eines nicht-primen Attributs von einem Schlüssel (siehe (c)).

Alternativ: HID --> HNAME ist eine nicht-triviale FA, HID ist kein Schlüssel von AH und HNAME ist nicht prim.

Aufgabe 4:

Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X ist Schlüssel oder $A \in R$ ist prim. Für BCNF entfällt „ $A \in R$ ist prim“. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

In die Tabelle **Autos** haben wir zusätzlich zu den Angaben der Autos noch den Händleridentifikator aufgenommen, der angibt, bei welchem Händler das Auto steht.

Autos

AID	MODELL	KM	EZJAHR	PREIS	HID
1001	Golf	95000	2005	6700	G&G
1002	Focus	130000	2002	4300	G&G
1003	Astra	170000	1999	2100	HCD
1004	Astra	60000	2004	7800	HCD
1005	Golf	190000	2001	2800	HCD
1006	Punto	75000	2002	2800	PP
1007	Passat	280000	2001	3900	PP

(a) Bestimmen Sie die Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten.

Schlüsselkandidaten: {AID}

Nicht-triviale FA: AID --> Autos -- die Schlüsselabhängigkeit

(b) Was ist die höchste Normalform, in der **Autos** ist? Begründung!

BCNF

Begründung: Wie in (a) gezeigt, gehen alle nicht-trivialen FA vom Schlüssel aus. Das ist die Definition der BCNF.

Aufgabe 5:

Für die folgenden Aufgaben betrachten wir Suchende und die Autos, die sie suchen (geparkt haben).

(a) Ergänzen Sie die Einträge in der Tabelle **Parken!** Dabei gilt: Rolf hat alle verfügbaren Golfs, Gabi alle Astras geparkt. Inge parkt den Focus und alle Golfs. Ernst hat nichts geparkt. **Hinweis:** Die Tabelle hat mehr Zeilen als Sie benötigen.

Suchende

SNAME	WOHN ORT	PLIMIT
Rolf	Kassel	4500
Gabi	Kassel	5000
Inge	Vellmar	3000
Ernst	Baunatal	6000

Modelle

AID	MODELL
1001	Golf
1002	Focus
1003	Astra
1004	Astra
1005	Golf
1006	Punto
1007	Passat

Parken

<u>SNAME</u>	AID
Rolf	1001
Rolf	1005
Gabi	1003
Gabi	1004
Inge	1002
Inge	1001
Inge	1005

(b) Welche sind die sinnvollen Schlüsselkandidaten und nicht-trivialen funktionalen Abhängigkeiten in den Relationen **Suchende**, **Modelle** und **Parken**?

Suchende: SK ist {SNAME}

FA ist SNAME --> Suchende

Modelle: SK ist {AID}

FA ist AID --> Modelle

Parken: SK ist {SNAME, AID}

FA ist {SNAME, AID} --> Parken

(c) Sind die Relationen **Suchende**, **Modelle** und **Parken** in BCNF? Kurze Begründung!

Ja, alle drei Relationen sind in BCNF, weil jeweils alle nicht-trivialen FA vom zugehörigen Schlüssel ausgehen.

Außerdem wären Modelle und Parken sowieso in BCNF, weil sie nur 2 Attribute haben.

(d) Wie lautet der zweifache Outer Join über die drei Tabellen **Suchende**, **Parken**, **Modelle** mit **SNAME** und **AID** als Verbindungsattribute mit den Tabelleninhalten von oben? Jedes Attribut soll dabei nur einmal auftauchen. Nullwerte deuten Sie mit „-“ an.

SNAME	WOHNORT	PLIMIT	AID	MODELL
Rolf	Kassel	4500	1001	Golf
Rolf	Kassel	4500	1005	Golf
Gabi	Kassel	5000	1003	Astra
Gabi	Kassel	5000	1004	Astra
Inge	Vellmar	3000	1002	Focus
Inge	Vellmar	3000	1001	Golf
Inge	Vellmar	3000	1005	Golf
Ernst	Baunatal	6000	-	-
-	-	-	1006	Punto
-	-	-	1007	Passat

Aufgabe 6:

Auf der Relation **Autos** aus Aufgabe 4 suchen wir alle Modelle (im Ergebnis tritt nur das Attribut **MODELL** auf), die preislich unter 4000 € liegen und ein Erstzulassungsjahr größer oder gleich 2002 haben. Geben Sie den Ausdruck der Relationenalgebra an!

$\Pi_{\text{MODELL}}(\sigma_{\text{PREIS} < 4000 \wedge \text{EZJAHR} \geq 2002}(\text{Autos}))$

Aufgabe 7:

Wir suchen Autos, die bei Händlern am eigenen Wohnort stehen. Wie lautet der SQL-Befehl dazu, der aus den Tabellen **AH** von Aufgabe 3 und **Suchende** von Aufgabe 5 die Namen der Suchenden (**SNAME**), ihren **WOHNORT**, den Automobilidentifizier **AID** und das Modell (**MODELL**) ausgibt, sofern eben das Auto bei einem Händler vor Ort steht?

```
SELECT Suchende.SNAME, Suchende.WOHNORT, AH.AID, AH.MODELL  
FROM Suchende, AH  
WHERE Suchende.WOHNORT = AH.ORT
```

Aufgabe 8:

Gesucht sind der Händleridentifizier **HID** und der Durchschnittspreis aller Fahrzeuge, die dieser Händler anbietet. Verwenden Sie für den SQL-Befehl die Tabelle **Autos** aus Aufgabe 4.

```
SELECT HID, AVG(PREIS)  
FROM Autos  
GROUP BY HID
```

Aufgabe 9:

Was liefert die folgende SQL-Abfrage auf den Tabellen aus Aufgabe 5?

```
SELECT *  
FROM Modelle M1  
WHERE NOT EXISTS (SELECT * FROM Parken  
                  WHERE M1.AID = Parken.AID)
```

(a) Geben Sie die Ergebnistupel für die Tabellenbelegung aus Aufgabe 5 an.

AID	MODELL
1006	Punto
1007	Passat

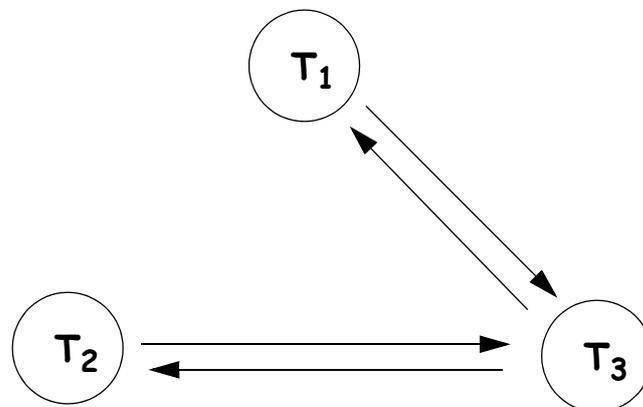
(b) Beschreiben Sie in Worten das Ergebnis auf beliebigen Tabelleninhalten für die Schemata aus Aufgabe 5!

Es wird die AID und das Modell der Autos ausgegeben, die momentan von keinem Suchenden geparkt sind.

Aufgabe 10:

Betrachten Sie die unten gezeigte Ablauffolge. Geben Sie den vollständigen Abhängigkeitsgraphen dazu an! Ist die Ablauffolge serialisierbar? Begründung!

Transaktion T₁	Transaktion T₂	Transaktion T₃
	Lesen A	
	Schreiben A	
Lesen B		
	Lesen C	
		Lesen B
		Lesen C
	Schreiben C	
		Schreiben C
		Schreiben B
Schreiben B		



Die Ablauffolge ist nicht serialisierbar, weil Zyklen im Abhängigkeitsgraphen vorhanden sind.

ENDE DER KLAUSUR

Klausur zur Vorlesung Datenbanken I im Wintersemester 2012/13

Name: Vorname:

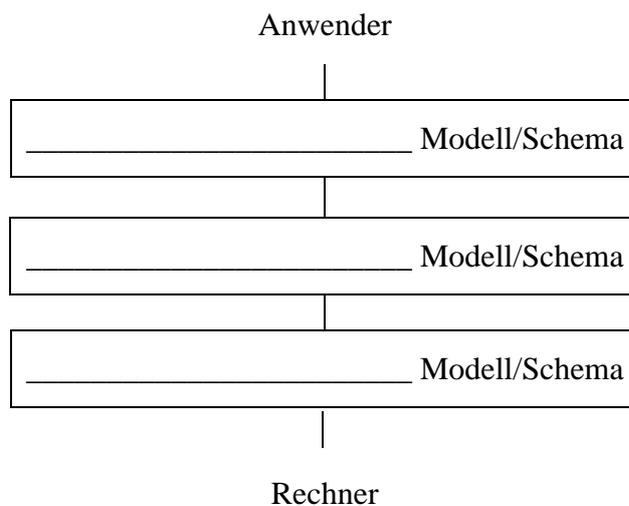
Matr.Nr.: Studiengang:

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	5		8	4	
3	1+2+2+1+1		9	2+3	
4	1+2+2		10	2+1	
5	2+2+2		Summe	44	
6	3				

Aufgabe 1:

Ergänzen Sie die Lücken für das 3-Schichtenmodell nach ANSI/SPARC!



Aufgabe 2:

In der letzten Datenbankklausur unseres Fachgebiets geht es um die Miniwelt der Datenbankklausuren!

Eine **Klausur** hat einen eindeutigen Klausuridentifizier (**KID**), z.B. *DBWS1213*, *DBSS12*, *DBWS1112*, ..., ein **Datum**, an dem die Klausur geschrieben wird (z.B. *19.03.2013*), eine Angabe Wiederholerklausur (**WKL**) mit Werten *ja* oder *nein* und ein Miniwelt-Thema (**Thema**).

Ferner gibt es eine Sammlung von Aufgaben. Eine **Aufgabe** hat einen künstlichen Schlüssel **AID**, einen **Typ** (z.B. *Ankreuz*, *Lücken*, *Programm*, *Ausgabe*, *Freitext*, *Diagramm*), ein abgefragtes **Stoffgebiet** (z.B. *ER-Modellierung*, *Transaktionen*, *SQL*, *Funktionale Abhängigkeiten*), eine zugeordnete Punktezahl (**Punkte**) und einen Aufgabentext (**Text**).

Eine Klausur **besteht aus** mindestens einer, theoretisch beliebig vielen Aufgaben. Eine Aufgabe ist immer in genau einer Klausur enthalten und hat dadurch eine Aufgabennummer (**Nummer**).

Zuletzt gibt es noch **Prüfer**. Die sind durch den **Namen** des Prüfers eindeutig bestimmt. Ferner hat ein Prüfer noch eine **Dienststellung** (*Professor*, *Wiss. Bediensteter*, *Dozent*). Für jede Klausur sind genau zwei Prüfer zuständig, ein Prüfer **ist** für mindestens eine, in der Regel viele Klausuren **zuständig**.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung und nur mit binären Beziehungen) für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie für jede Entität genau einen (den von Ihnen bevorzugten) Schlüssel!

Aufgabe 3:

Hier sehen wir eine Tabelle **Aufgaben**. Die Einträge spiegeln die Miniweltannahmen von Aufgabe 2 wider. Falls den Prüfern kein neues Klausurthema einfällt, darf ein Thema auch in mehreren Klausuren verwendet werden. Jede Klausur hat aber genau ein Thema.

Hinweis: Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X enthält einen Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Aufgaben

AID	Typ	Stoffgebiet	Punkte	KID	Thema
349644	Diagramm	ER-Modellierung	6	DBSS12	Gebrauchtwagen
579234	Diagramm	ER-Modellierung	6	DBWS1112	Baugeld
104566	Diagramm	Serialisierbarkeit	4	DBSS12	Gebrauchtwagen
782310	Programm	SQL	4	DBSS12	Gebrauchtwagen
556396	Programm	SQL	4	DBWS1112	Baugeld
230889	Programm	SQL	3	DBSS12	Gebrauchtwagen
654894	Programm	SQL	4	DBWS1112	Baugeld
436545	Programm	SQL	3	DBSS11	Atomkraftwerke
239827	Ausgabe	SQL	4	DBSS12	Gebrauchtwagen
165166	Ankreuz	Transaktionen	2	DBSS98	Gebrauchtwagen

(a) Was sind die Schlüsselkandidaten?

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

(c) Gibt es eine transitive Abhängigkeit eines nicht-primen Attributs in **Aufgaben**? Wenn ja, welche?

Aufgabenteile (d) und (e) umseitig.

(d) Ist **Aufgaben** in 2NF? Begründung!

(e) Ist **Aufgaben** in 3NF? Begründung!

Aufgabe 4:

Hier haben wir in die Tabelle **Klausuren** die Prüfer und das Klausurthema aufgenommen.

Klausuren

KID	Thema	Pruefername	Dienststellung
DBWS1213	Klausuren	Schweinsberg	Wiss. Bediensteter
DBSS12	Gebrauchtwagen	Schweinsberg	Wiss. Bediensteter
DBWS1112	Baugeld	Schweinsberg	Wiss. Bediensteter
DBSS11	Atomkraftwerke	Schweinsberg	Wiss. Bediensteter
DBSS98	Gebrauchtwagen	Wegner	Professor
DBWS1213	Klausuren	Wegner	Professor
DBSS12	Gebrauchtwagen	Wegner	Professor
DBWS1112	Baugeld	Wegner	Professor
DBSS11	Atomkraftwerke	Wegner	Professor
DBSS98	Gebrauchtwagen	Zirkelbach	Professor

(a) Bestimmen Sie die Schlüsselkandidaten.

(b) Bestimmen Sie die nicht-trivialen funktionalen Abhängigkeiten.

(c) Was ist die höchste Normalform für **Klausuren**? Begründung!

Aufgabe 5:

Wir trennen **Klausuren** von Aufgabe 4 auf und speichern die Daten in den Tabellen **KT** und **PT**.

KT

PT

KID	Thema	Pruefername	Dienststellung	Thema
DBWS1213	Klausuren	Schweinsberg	Wiss. Bediensteter	Klausuren
DBSS12	Gebrauchtwagen	Schweinsberg	Wiss. Bediensteter	Gebrauchtwagen
DBWS1112	Baugeld	Schweinsberg	Wiss. Bediensteter	Baugeld
DBSS11	Atomkraftwerke	Schweinsberg	Wiss. Bediensteter	Atomkraftwerke
DBSS98	Gebrauchtwagen	Wegner	Professor	Klausuren
		Wegner	Professor	Gebrauchtwagen
		Wegner	Professor	Baugeld
		Wegner	Professor	Atomkraftwerke
		Zirkelbach	Professor	Gebrauchtwagen

(a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

(b) Sind die Relationen **KT** und **PT** in BCNF? Kurze Begründung!

(c) Ist diese Aufteilung von **Klausuren** aus Aufgabe 4 verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

Aufgabe 6:

Mit Hilfe der Relationen **Aufgaben** aus Aufgabe 3 und **Klausuren** aus Aufgabe 4 sollen die Aufgabenidentifizier **AID** aller Aufgaben ermittelt werden, die in einer Klausur mit dem Thema „Gebrauchtwagen“ vorkommen, bei der „Wegner“ Prüfer ist.

Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

Aufgabe 7:

Wie lautet die äquivalente SQL-Abfrage zu Aufgabe 6?

Aufgabe 8:

Mit welchem SQL-Befehl erhält man aus der Relation **Aufgaben** in Aufgabe 3 die dreispaltige Ausgabe mit Klausuridentifizier (KID), Anzahl der Aufgaben und Summe der Punkte in dieser Klausur.

Aufgabe 9:

Betrachten Sie die folgende Abfrage für die Tabelle **Klausuren** aus Aufgabe 4:

```
SELECT KID, Thema
FROM Klausuren
WHERE Pruefername = 'Wegner' AND
      KID IN (SELECT KID FROM Klausuren
              WHERE Pruefername = 'Schweinsberg');
```

(a) Wie lautet die Resultattabelle?

(b) Formulieren Sie die Abfrage von oben um in eine SQL-Abfrage, die einen Selbst-Join verwendet!

Aufgabe 10:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktionen T_i :

Transaktion T_1 (10:00 Uhr)	Transaktion T_2 (10:05 Uhr)	Transaktion T_3 (10:10 Uhr)
Lesen A Schreiben A		
	Lesen A	
Lesen B Schreiben B		
	Lesen C	
		Lesen C
	Schreiben C Schreiben A	Schreiben C

- (a) Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle solange ein, bis es ggf. einen Konflikt gibt.
- (b) Geht die Ablauffolge durch? Wenn nein, markieren Sie oben die zurückgewiesene Operation!

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)
9:00	9:00	9:00	9:00	9:00	9:00

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann Schreiboperation ignorieren

sonst Schreiben, $WTM(x) := TS$

ENDE DER KLAUSUR

**Klausur zur Vorlesung Datenbanken I
im Wintersemester 2012/13**

Name: Vorname:

Matr.Nr.: Studiengang:

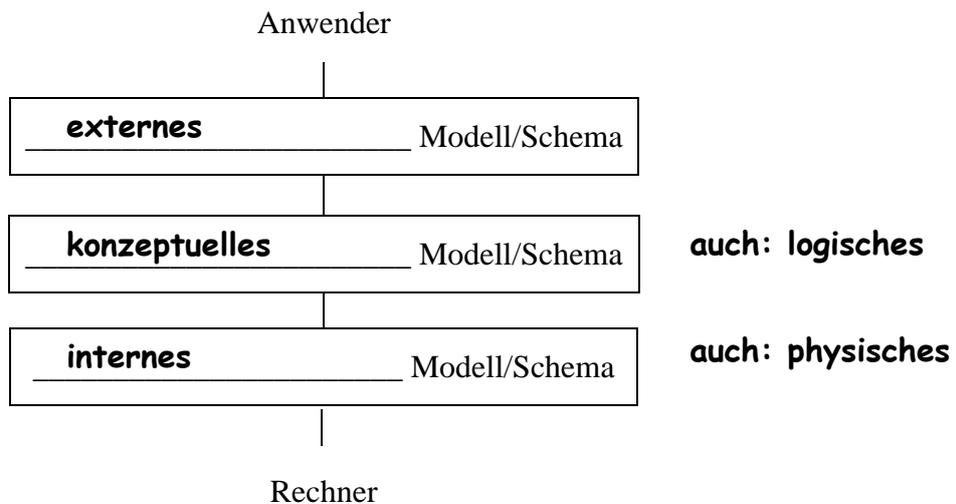
MUSTERLÖSUNG

Hinweis: Bearbeiten Sie alle Aufgaben. Schreiben Sie Ihre Antworten in den freien Platz oder auf die Rückseiten. Hilfsmittel sind **nicht** zugelassen. Die Bearbeitungszeit beträgt 120 Minuten.

Aufgabe	Punkte max.	Punkte err.	Aufgabe	Punkte max.	Punkte err.
1	2		7	4	
2	5		8	4	
3	1+2+2+1+1		9	2+3	
4	1+2+2		10	2+1	
5	2+2+2		Summe	44	
6	3				

Aufgabe 1:

Ergänzen Sie die Lücken für das 3-Schichtenmodell nach ANSI/SPARC!



Aufgabe 2:

In der letzten Datenbankklausur unseres Fachgebiets geht es um die Miniwelt der Datenbankklausuren!

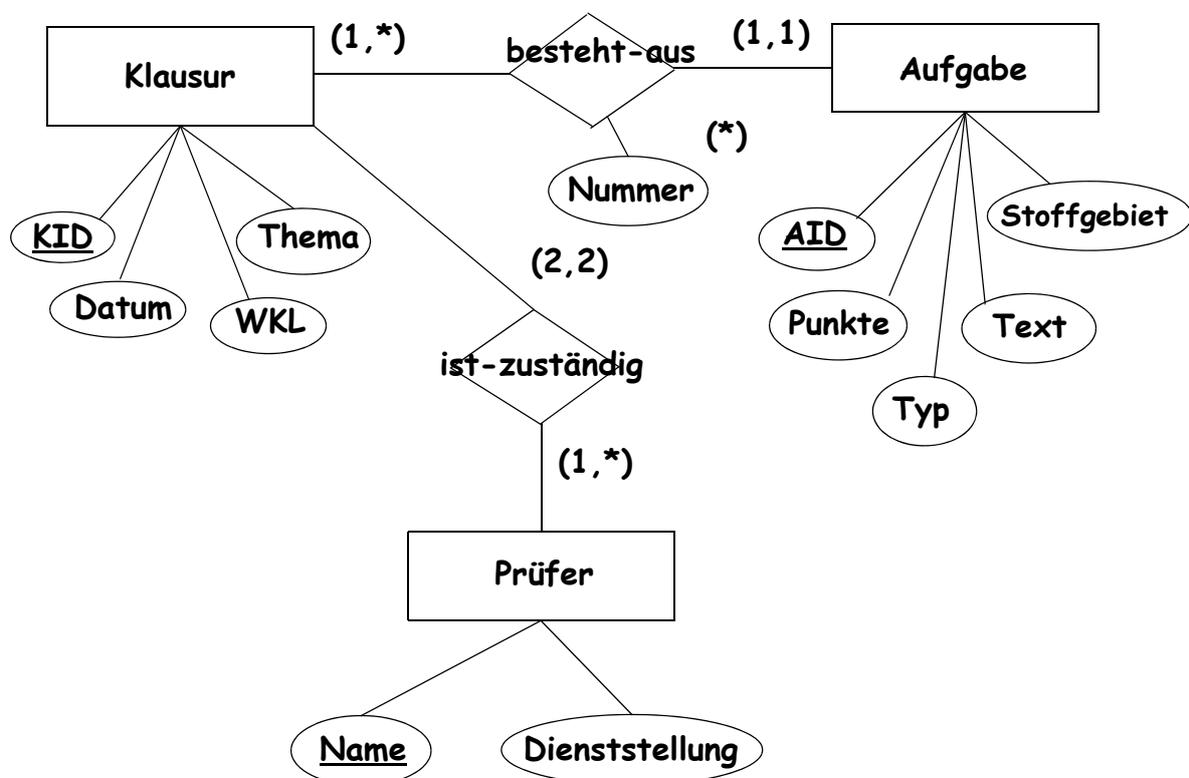
Eine **Klausur** hat einen eindeutigen Klausuridentifizier (**KID**), z.B. *DBWS1213*, *DBSS12*, *DBWS1112*, ..., ein **Datum**, an dem die Klausur geschrieben wird (z.B. *19.03.2013*), eine Angabe Wiederholerklausur (**WKL**) mit Werten *ja* oder *nein* und ein Miniwelt-Thema (**Thema**).

Ferner gibt es eine Sammlung von Aufgaben. Eine **Aufgabe** hat einen künstlichen Schlüssel **AID**, einen **Typ** (z.B. *Ankreuz*, *Lücken*, *Programm*, *Ausgabe*, *Freitext*, *Diagramm*), ein abgefragtes **Stoffgebiet** (z.B. *ER-Modellierung*, *Transaktionen*, *SQL*, *Funktionale Abhängigkeiten*), eine zugeordnete Punktezahl (**Punkte**) und einen Aufgabentext (**Text**).

Eine Klausur **besteht aus** mindestens einer, theoretisch beliebig vielen Aufgaben. Eine Aufgabe ist immer in genau einer Klausur enthalten und hat dadurch eine Aufgabennummer (**Nummer**).

Zuletzt gibt es noch **Prüfer**. Die sind durch den **Namen** des Prüfers eindeutig bestimmt. Ferner hat ein Prüfer noch eine **Dienststellung** (*Professor*, *Wiss. Bediensteter*, *Dozent*). Für jede Klausur sind genau zwei Prüfer zuständig, ein Prüfer **ist** für mindestens eine, in der Regel viele Klausuren **zuständig**.

Geben Sie ein ER-Diagramm (ohne Spezialisierung/Generalisierung und nur mit binären Beziehungen) für diese Miniwelt an. Die Kanten sollen die Kardinalitäten in der (min, max)-Notation zeigen. Unterstreichen Sie für jede Entität genau einen (den von Ihnen bevorzugten) Schlüssel!



(*) alternativ Nummer auch Attribut von Aufgabe

Aufgabe 3:

Hier sehen wir eine Tabelle **Aufgaben**. Die Einträge spiegeln die Miniweltannahmen von Aufgabe 2 wider. Falls den Prüfern kein neues Klausurthema einfällt, darf ein Thema auch in mehreren Klausuren verwendet werden. Jede Klausur hat aber genau ein Thema.

Hinweis: Eine Relation R ist bekanntlich in 3. Normalform, wenn für alle funktionalen Abhängigkeiten $X \rightarrow A$ gilt, $X \rightarrow A$ ist trivial oder X enthält einen Schlüssel oder $A \in R$ ist prim. Alternativ ist Relation R in 3NF, wenn sie keine transitiven Abhängigkeiten eines nicht-primen Attributs von einem Schlüssel hat.

Aufgaben

AID	Typ	Stoffgebiet	Punkte	KID	Thema
349644	Diagramm	ER-Modellierung	6	DBSS12	Gebrauchtwagen
579234	Diagramm	ER-Modellierung	6	DBWS1112	Baugeld
104566	Diagramm	Serialisierbarkeit	4	DBSS12	Gebrauchtwagen
782310	Programm	SQL	4	DBSS12	Gebrauchtwagen
556396	Programm	SQL	4	DBWS1112	Baugeld
230889	Programm	SQL	3	DBSS12	Gebrauchtwagen
654894	Programm	SQL	4	DBWS1112	Baugeld
436545	Programm	SQL	3	DBSS11	Atomkraftwerke
239827	Ausgabe	SQL	4	DBSS12	Gebrauchtwagen
165166	Ankreuz	Transaktionen	2	DBSS98	Gebrauchtwagen

(a) Was sind die Schlüsselkandidaten?

{AID}

(b) Welche nicht-trivialen funktionalen Abhängigkeiten existieren?

AID --> Aufgaben

KID --> Thema

(c) Gibt es eine transitive Abhängigkeit eines nicht-primen Attributs in **Aufgaben**? Wenn ja, welche?

Ja. AID --> KID --> Thema
<-/-

Aufgabenteile (d) und (e) umseitig.

(d) Ist **Aufgaben** in 2NF? Begründung!

Ja, weil alle Schlüssel einelementig sind. Alternativ: Ja, weil alle nicht-primen Attribute voll funktional von allen Schlüsseln abhängen.

(e) Ist **Aufgaben** in 3NF? Begründung!

Nein, wegen der FA in (c)

Aufgabe 4:

Hier haben wir in die Tabelle **Klausuren** die Prüfer und das Klausurthema aufgenommen.

Klausuren

KID	Thema	Pruefername	Dienststellung
DBWS1213	Klausuren	Schweinsberg	Wiss. Bediensteter
DBSS12	Gebrauchtwagen	Schweinsberg	Wiss. Bediensteter
DBWS1112	Baugeld	Schweinsberg	Wiss. Bediensteter
DBSS11	Atomkraftwerke	Schweinsberg	Wiss. Bediensteter
DBSS98	Gebrauchtwagen	Wegner	Professor
DBWS1213	Klausuren	Wegner	Professor
DBSS12	Gebrauchtwagen	Wegner	Professor
DBWS1112	Baugeld	Wegner	Professor
DBSS11	Atomkraftwerke	Wegner	Professor
DBSS98	Gebrauchtwagen	Zirkelbach	Professor

(a) Bestimmen Sie die Schlüsselkandidaten.

{KID, Pruefername}

(b) Bestimmen Sie die nicht-trivialen funktionalen Abhängigkeiten.

{KID, Pruefername} --> Klausuren

KID --> Thema

Pruefername --> Dienststellung

(c) Was ist die höchste Normalform für **Klausuren**? Begründung!

Nur in 1 NF. Nicht in 2NF, weil Thema ein nicht-primen Attribut ist und KID --> Thema, also ist Thema nicht voll funktional vom Schlüssel abhängig.

Aufgabe 5:

Wir trennen **Klausuren** von Aufgabe 4 auf und speichern die Daten in den Tabellen **KT** und **PT**.

KT

PT

KID	Thema	Pruefername	Dienststellung	Thema
DBWS1213	Klausuren	Schweinsberg	Wiss. Bediensteter	Klausuren
DBSS12	Gebrauchtwagen	Schweinsberg	Wiss. Bediensteter	Gebrauchtwagen
DBWS1112	Baugeld	Schweinsberg	Wiss. Bediensteter	Baugeld
DBSS11	Atomkraftwerke	Schweinsberg	Wiss. Bediensteter	Atomkraftwerke
DBSS98	Gebrauchtwagen	Wegner	Professor	Klausuren
		Wegner	Professor	Gebrauchtwagen
		Wegner	Professor	Baugeld
		Wegner	Professor	Atomkraftwerke
		Zirkelbach	Professor	Gebrauchtwagen

- (a) Geben Sie jeweils die Schlüsselkandidaten und alle nicht-trivialen funktionalen Abhängigkeiten an!

KT: SK ist {KID}, FA ist KID --> KT

PT: SK ist {Pruefername, Thema},

FA sind {Pruefername, Thema} --> PT und Pruefername --> Dienststellung

- (b) Sind die Relationen **KT** und **PT** in BCNF? Kurze Begründung!

KT ist in BCNF, da nur zwei Spalten

PT ist nicht in BCNF, denn Pruefername --> Dienststellung ist eine nicht-triviale FA, die nicht von einem Schlüssel ausgeht.

Alternativ: Dienststellung ist nicht-prim und nicht voll funktional abhängig vom Schlüssel, damit ist PT noch nicht einmal in 2NF und somit nicht in BCNF.

- (c) Ist diese Aufteilung von **Klausuren** aus Aufgabe 4 verlustfrei? Bekanntlich ist eine Aufteilung nur verlustfrei, wenn $R_1 \cap R_2 \rightarrow R_1$ oder $R_1 \cap R_2 \rightarrow R_2$. Begründung!

KT \cap PT = {Thema}, Thema -/-> KT und Thema -/-> PT

damit ist die Aufteilung nicht verlustfrei.

Aufgabe 6:

Mit Hilfe der Relationen **Aufgaben** aus Aufgabe 3 und **Klausuren** aus Aufgabe 4 sollen die Aufgabenidentifizier **AID** aller Aufgaben ermittelt werden, die in einer Klausur mit dem Thema „Gebrauchtwagen“ vorkommen, bei der „Wegner“ Prüfer ist.

Geben Sie den Ausdruck der Relationenalgebra an! Sie brauchen Projektion, Selektion und einen Join.

$$\Pi_{\text{AID}} (\sigma_{\text{Thema} = \text{"Gebrauchtwagen"} \wedge \text{Pruefername} = \text{"Wegner"}} (\text{Aufgaben} \bowtie \text{Klausuren}))$$

Aufgabe 7:

Wie lautet die äquivalente SQL-Abfrage zu Aufgabe 6?

```
SELECT a.AID
FROM Aufgaben a, Klausuren k
WHERE a.KID = k.KID AND
      k.Thema = 'Gebrauchtwagen' AND
      k.Pruefername = 'Wegner';
```

Aufgabe 8:

Mit welchem SQL-Befehl erhält man aus der Relation **Aufgaben** in Aufgabe 3 die dreispaltige Ausgabe mit Klausuridentifizier (KID), Anzahl der Aufgaben und Summe der Punkte in dieser Klausur.

```
SELECT KID, COUNT(*), SUM(Punkte)
FROM Aufgaben
GROUP BY KID;
```

Aufgabe 9:

Betrachten Sie die folgende Abfrage für die Tabelle **Klausuren** aus Aufgabe 4:

```
SELECT KID, Thema
FROM Klausuren
WHERE Pruefername = 'Wegner' AND
      KID IN (SELECT KID FROM Klausuren
              WHERE Pruefername = 'Schweinsberg');
```

(a) Wie lautet die Resultattabelle?

KID	Thema
DBWS1213	Klausuren
DBSS12	Gebrauchtwagen
DBWS1112	Baugeld
DBSS11	Atomkraftwerke

(b) Formulieren Sie die Abfrage von oben um in eine SQL-Abfrage, die einen Selbst-Join verwendet!

```
SELECT k1.KID, k1.Thema
FROM Klausuren k1, Klausuren k2
WHERE k1.KID = k2.KID AND k1.Pruefername = 'Schweinsberg'
      AND k2.Pruefername = 'Wegner';
```

Aufgabe 10:

Betrachten Sie die folgende Ablauffolge mit den Zeitmarken $TS(T_i)$ der Transaktionen T_i :

Transaktion T_1 (10:00 Uhr)	Transaktion T_2 (10:05 Uhr)	Transaktion T_3 (10:10 Uhr)
Lesen A Schreiben A		
	Lesen A	
Lesen B Schreiben B		
	Lesen C	
	Schreiben C	Lesen C
	Schreiben A	Schreiben C

- (a) Tragen Sie die Zeitmarken für Lese-/Schreibzugriffe gemäß dem Zeitmarkenalgorithmus in die Tabelle solange ein, bis es ggf. einen Konflikt gibt.
- (b) Geht die Ablauffolge durch? Wenn nein, markieren Sie oben die zurückgewiesene Operation!

RTM(A)	WTM(A)	RTM(B)	WTM(B)	RTM(C)	WTM(C)
9:00	9:00	9:00	9:00	9:00	9:00
10:00	10:00	10:00	10:00	10:05	
10:05				10:10	

Es gilt der folgende einfache *Zeitmarkenalgorithmus*:

Sei TS die Zeitmarke einer *Leseoperation* auf Datenobjekt x .

Wenn $TS < WTM(x)$

dann Zurückweisen der Leseoperation, Zurückrollen

und Neustart der auslösenden Transaktion mit neuer Zeitmarke

sonst Lesen ausführen, $RTM(x) := \max(RTM(x), TS)$

Sei TS die Zeitmarke einer *Schreiboperation* auf Datenobjekt x .

Wenn $TS < RTM(x)$

dann Zurückweisen der Schreiboperation, Zurückrollen und Neustart mit neuer Zeitmarke

sonst wenn $TS < WTM(x)$

dann Schreiboperation ignorieren

sonst Schreiben, $WTM(x) := TS$

ENDE DER KLAUSUR