

**Fachprüfung Datenbanksysteme
 Wintersemester 2015/16**

Studiengang: Bachelor Medieninformatik

Prüfungsdauer: 90 min.

Prüfungstermin: 21.3.2016

Letzter Prüfungsversuch?	<input type="checkbox"/> ja
	Punkte aus den Übungen: 25 / 25 Übung bestanden im WS/SS <u>15/16</u> Punkte aus der Prüfung: 66,5 / 75 Gesamtpunktzahl: 91,5 / 100
Die Klausur ist mit 37,5 Punkten bestanden. Bewertung der Leistungen für das Fach mit Note:	Unterschrift Prüfer:

Folgende Hilfsmittel dürfen benutzt werden:

- Eine A5 Seite, beidseitig beschriftet.

Andere Hilfsmittel sind nicht erlaubt!

Prüfungsteile:	Zu erreichende Punkte:	Erreichte Punkte:
I. Theoretische Fragen	15	13
II. Abfragen mit SQL / Relationenalgebra	30	27,5
III. DB-Entwicklungsaufgabe	30	26

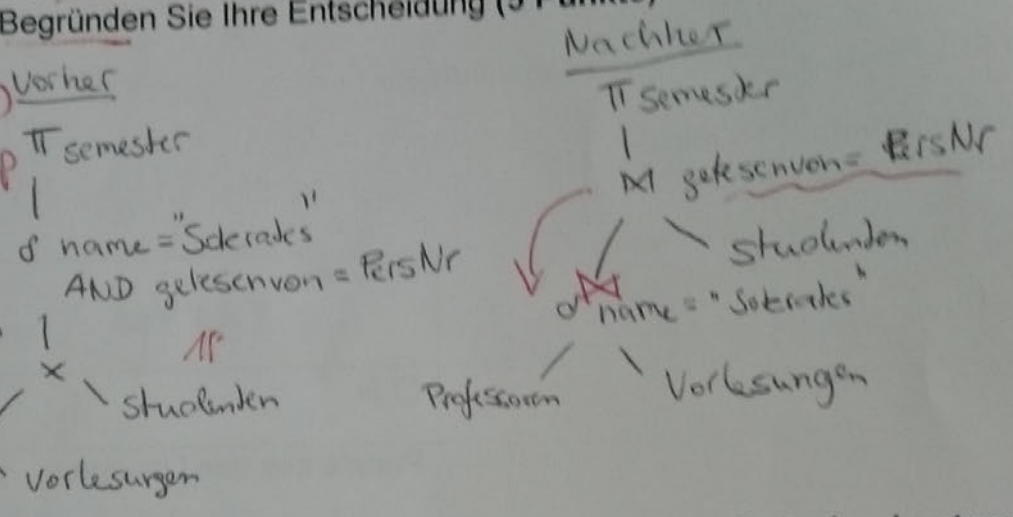
Viel Erfolg 😊

I. Theoretische Fragen (je 5 Punkte)

1. **Optimierung.** Bitte definieren Sie einen nicht optimierten Anfrageplan Ihrer Wahl. Schlagen Sie zwei konzeptionell unterschiedliche logische Optimierungen für den Plan vor. Begründen Sie Ihre Entscheidung (5 Punkte)

3/5

- Selektion so weit wie möglich nach unten
- Selektionen mit AND aufteilen und separat verschleiben
- Projektion soweit wie möglich nach unten
- Kreuzprodukt mit Selektion zu Join zusammenfassen



2. **Datenbankentwurf:** Nennen Sie vier Phasen des Datenbankentwurfs. Beschreiben Sie eine Phase bitte mit 3 kurzen Sätzen. (5 Punkte)

5/5

Datendefinition, Anforderungsanalyse, konzeptioneller Entwurf, logischer Entwurf, Verteilungsentwurf, ...

Beim konzeptionellen Entwurf werden Ergebnisse in ein Modell übertragen. Meistens ist dies das ER-Modell. Dort werden Entitäten (Begriffe) und Relationships (Beziehungen) zwischen den Entitäten, sowie weitere Nebenbedingungen (Kardinalitäten, Totalitäten, Schlüssel, Datentyp) aufgenommen.

3. **SQL Union:** Wie sieht die Ergebnismenge dieser beiden Tabellen nach einer „union all“-Operation aus? Zeichnen Sie die Ergebnistabelle. (5 Punkte)

5/5

Tabelle 1

A	B	C
1	2	3
3	4	5
9	8	7

Tabelle 2

A	B	C
1	2	3
6	7	8
5	4	3

Ergebnistabelle

A	B	C
1	2	3
3	4	5
6	7	8
9	8	7
5	4	3
1	2	3

II. Abfragen mit SQL / Relationenalgebra (je 5 Punkte)

Gegeben ist folgendes Relationenschema eines Ihnen bekannten Computerspiels:

Dorf (d_id, name, hauptling_ID)
Bewohner (b_id, name, dorf_id, geschlecht, beruf, gold, status)
Gegenstand (g_id, name, besitzer_id)

Formulieren Sie auf dem obigen Relationenschema folgende Anfragen als Ausdruck in SQL (Oracle 11g R2 Dialekt):

a) Zeigen Sie alle Gegenstände die männlichen Besitzern in „Gurkendorf“ gehören.

5/5

```
SELECT gegenstand.name NP
FROM gegenstand, bewohner, dorf
WHERE gegenstand.besitzer_id = bewohner.b_id NP
AND bewohner.dorf_id = dorf.d_id NP
AND bewohner.geschlecht = "männlich" NP
AND dorf.name = "Gurkendorf" NP
```

b) Zeigen Sie für jedes Dorf die dort lebenden vermögenden Berufsgruppen, geordnet pro Dorf nach dem meisten „Gold“.

5/5

```
SELECT dorf.name, bewohner.beruf, SUM(gold) NP
FROM dorf, bewohner NP
WHERE bewohner.dorf_id = dorf.d_id NP
AND gold > 0
GROUP BY dorf.name, bewohner.beruf NP
ORDER BY SUM(gold) DESC NP
```

c) Lesen Sie alle Gegenstände aus die keinem gehören und weisen Sie diese dem Häuptling von „Gurkendorf“ zu.

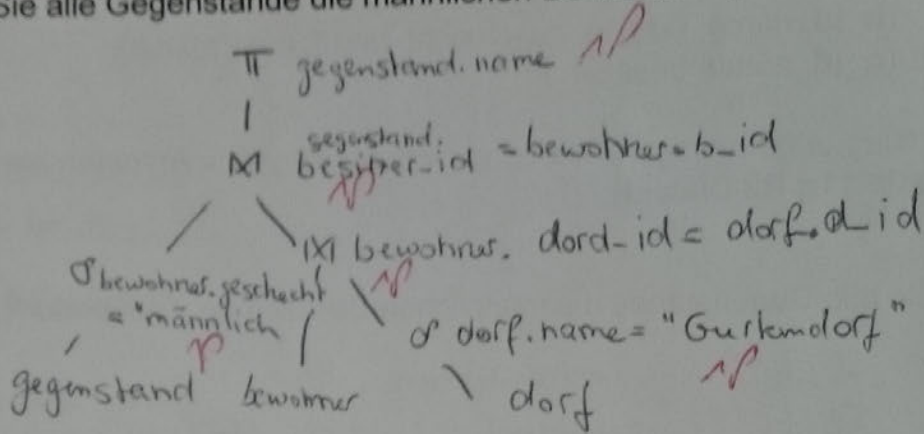
2,5/5

```
UPDATE NP gegenstand SET name name = "Gurkendorf" NP
dorf_id gegenstand.besitzer_id = dorf_id hauptling_id
WHERE gegenstand.besitzer_id IS NULL NP
```

Formulieren Sie auf dem obigen Relationenschema die obigen drei Anfragen in der relationalen Algebra bzw. formulieren Sie einen möglichen Anfrageplan:

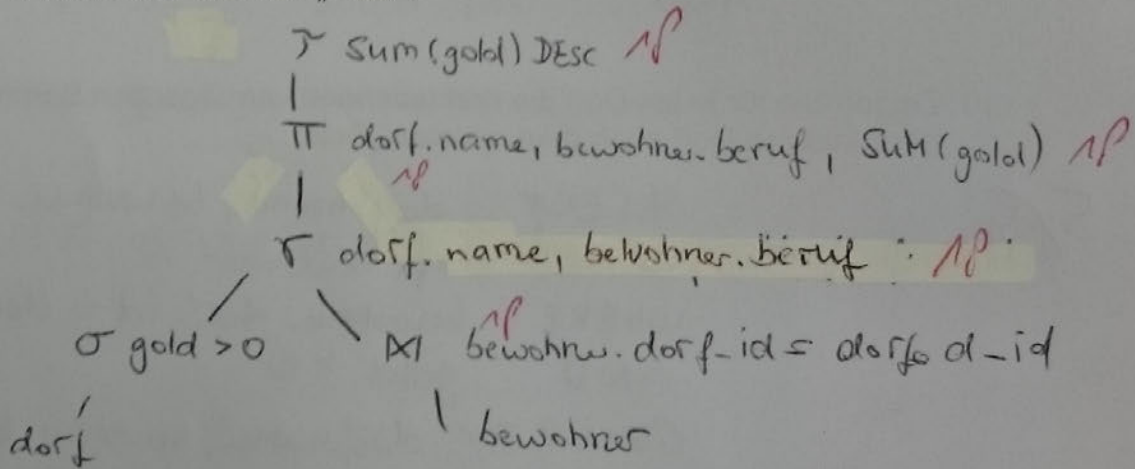
d) Zeigen Sie alle Gegenstände die männlichen Besitzern in „Gurkendorf“ gehören.

5/5



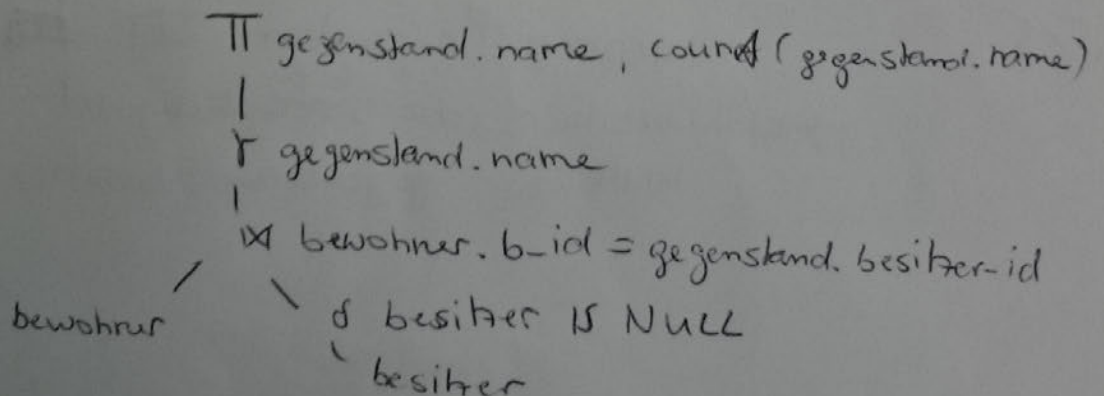
e) Zeigen Sie für jedes Dorf die dort lebenden vermögenden Berufsgruppen, geordnet pro Dorf nach dem meisten „Gold“.

5/5



f) Lesen Sie alle Gegenstände aus die keinem gehören und zählen sie diese.

5/5



III. DB-Entwicklungsaufgabe

Gehen Sie bitte von nachfolgend dargestelltem Sachverhalt aus und beantworten Sie die Fragen:

- 3.1 Bitte entwickeln Sie das Datenmodell für die Abbildung des Sachverhaltes. Stellen Sie das Datenmodell in Form eines **ER-Diagramms** dar. Machen Sie die Beziehungen entsprechend der in der Aufgabenstellung geforderten Kardinalitäten deutlich. Die Attribute können in dieser Darstellung weggelassen werden. Beachten Sie, dass die eingefügten Beispieltabellen nur zur Illustration dienen und **keine verwendbaren Relationen** darstellen. (15 Punkte)
- 3.2 Leiten Sie aus dem ER-Diagramm (aus 3.1) normalisierte Relationen gemäß der Ihnen bekannten Transformationsregeln ab. Notieren Sie die Relationen in **Relationenschreibweise**. Definieren Sie zu jeder Relation Primär- und Fremdschlüssel. Machen Sie die Fremdschlüssel kenntlich (z.B. Unterstreichung oder Hervorhebung). Ordnen Sie bitte **alle** angesprochenen Attribute zu. (7 Punkte)
- 3.3 Entwickeln Sie bitte für die Relationen „Produktvariation“ und „Motiv“ die entsprechenden **„create table“-Befehle**, mit Abbildung der Primär- und Fremdschlüssel und der Option zur Behandlung der kritischen delete-Operation. Begründen Sie Ihre Entscheidung für die gewählte Option kurz. Die weiteren Relationen sind **nicht** zu entwickeln! (6 Punkte)
- 3.4 Entwickeln Sie bitte die SQL-Befehle für das Anlegen von einem Datensatz in der in 3.3 erzeugten Tabelle „Motiv“. Verwenden Sie Daten aus den Beispieltabellen. (2 Punkte)
- insert into Motiv .. values ...*

Gegenstand der vorliegenden Aufgabe ist ein Produktkatalog mit -varianten:

Ein namhaftes Versandhaus bietet im Rahmen seiner verschiedenen Kataloge (young fashion, modern fashion, woman's fashion) Produkte häufig als sogenannte Produktvariationen an, die der Kunde bestellen kann.

Das heißt, dass ein bestimmtes Produkt, z.B. ein Shirt eines bestimmten Herstellers aus einem bestimmten Material, in verschiedenen Produktvariationen angeboten werden kann.

Ein Basisprodukt besteht dabei aus genau einem Material, wie im Beispiel „Baumwolle“.

Eine Produktvariation zu einem Basisprodukt besteht aus einer gewählten Farbe, einer Größe und einem Druckmotiv.

Die Preisbildung orientiert sich an der Größe.

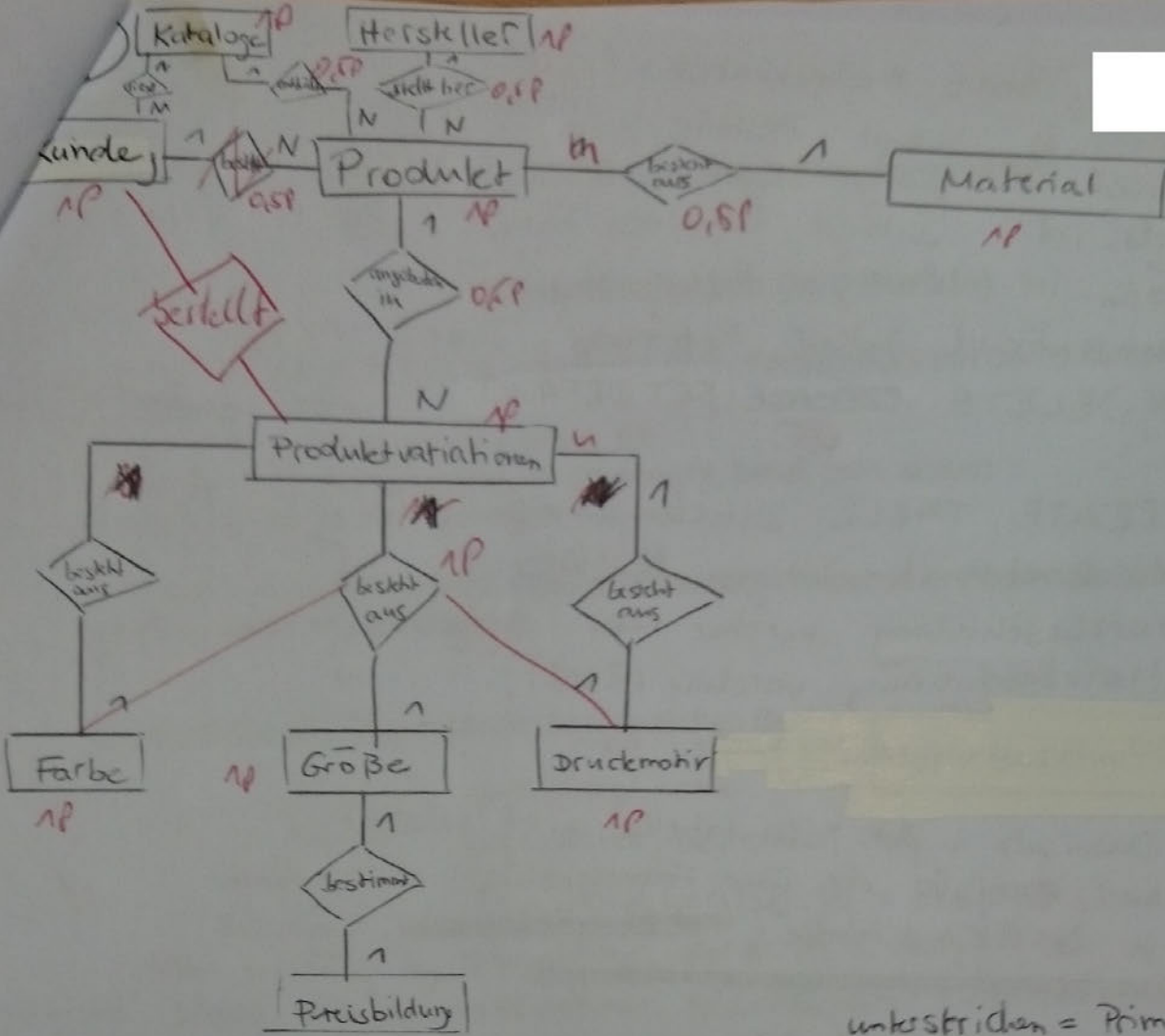
Abgebildet wird nachfolgend beispielhaft ein mögliches Basisprodukt mit einigen möglichen Produktvariationen:

Shirt (100% Baumwolle) Motiv Picasso	khaki 4045006	orange 4046055	schwarz 4046165
	wollweiß 4046085	bordeaux 4046115	
	Gr. 32/34 36/38 Preis für o.g. Gr.: 34,90	Gr. 40/42 44/46 Preis für o.g. Gr.: 42,00	

Je Druckmotiv sollen, neben dessen Kurzbezeichnung, in der Datenbank auch eine Motivbeschreibung abgelegt werden. Je Größe werden Maßangaben und Toleranzen (als Text) sowie eine Kurzbeschreibung abgespeichert. Je Farbe werden der Farbcode, ein Farbkürzel sowie eine Beschreibung verwendet.

Zur Bestellung von Produkten aus den Katalogen muss die Produktvariation abgebildet werden, wie z.B.

Kunden-Nummer	Farbcode	Produkt	Farbe	Größe	Preis
0815	4045006	Shirt (100% Baumwolle) Motiv Picasso	khaki	40/42	42,00
0815	4046115	Shirt (100% Baumwolle) Motiv Picasso	bordeaux	44/46	42,00
0816	4045006	Shirt (100% Baumwolle) Motiv Picasso	khaki	40/42	42,00



12,5/15

unterstrichen = Primärschlüssel
 - - - - - = Fremdschlüssel

- 3.2
- Kataloge (Katalog-id, Typ) ✓
 - Hersteller (Hersteller-id, name) ✓
 - Kunde (Kunden-id, name, Adresse) ✓
 - Produkt (Produkt-id, Katalog-id, Kunden-id, Hersteller-id, Typ) ✓
 - Material (Material-id, Produkt-id) ✓
 - Produktvariation (Variations-id, Produkt-id, Farb-id, Größe-id, Druckmotiv-id) ✓
 - Farbe (Farb-id, Farbcode, Farbkürzel, Beschreibung) ✓
 - Größe (Größe-id, Maßangaben, Toleranzen, Kurzbeschreibung) ✓
 - Druckmotiv (Druckmotiv-id, Kurzbezeichnung, Motivbeschreibung) ✓
 - Preisbildung (Preisbildung-id, Größe-id) ✓
 - gelesen von (Kunden-id, Katalog-id) ✓

7/7

3.3

CREATE TABLE Produktvariation (

Variations_id Integer PRIMARY KEY
 produkt_id Integer References Produkt, produkt_id ON DELETE CASCADE
 farb_id Integer References Farbe, farb_id
 gröÙe_id Integer References Größe
 druckmotiv_id Integer References Druckmotiv

ON DELETE CASCADE);

→ ~~Wenn~~ ~~Tupel~~ gelöscht

4.5/6

CREATE TABLE Druckmotiv (

druckmotiv_id Integer PRIMARY KEY
 kurzbezeichnung varchar (250)
 Motivbeschreibung varchar (250);

ON DELETE CASCADE);

Der Datensatz in der "Elterntabelle" wird gelöscht und ebenfalls die über Fremdschlüssel verbundenen in der "Kindstabelle". ~~Hier kann es sein~~. Wenn z.B.

~~Wenn es die Variation nicht mehr gibt,~~ Tupel gelöscht wird, macht die ganze Variation des Produkts keinen Sinn und so werden alle FK gelöscht.

3.4

INSERT INTO Druckmotiv (druckmotiv_id, kurzbezeichnung, Motivbeschreibung)
 VALUES (1, 'Fischer Spruch', 'Ein fischer Spruch auf dem Shirt)

2/2