

Lösungen der Übungsaufgaben von Kapitel 5

1. In der Charakterisierung des relationalen Modells haben wir u.a. gesagt:

- Alle Operatoren, die dem Benutzer zur Manipulation von Daten zur Verfügung stehen, operieren grundsätzlich nur auf Tabellen und ergeben auch wieder neue Tabellen.

Das bedeutet, dass wir verschiedene Operatoren hintereinander ausführen können und wieder neue Operatoren erhalten. Entscheiden Sie in den folgenden Beispielen, welche der drei Operatoren PROJECT, RESTRICT und JOIN bzw. welche Kombination und Hintereinanderschaltung Sie brauchen, um die folgenden Abfragen durchzuführen. Grundlage sind stets die Tabellen der Datenbank Allerhand.

a) Zeige alle Personen aus Bonn

Hier wird auf die Tabelle PERSON der RESTRICT-Operator angewendet: Es werden nur die Sätze aus der Tabelle Person angezeigt, bei denen **Ort** = 'Bonn' ist.

b) Zeige die Namen und Telefonnummern aller Personen aus Bonn

Hier wird auf die Tabelle PERSON zunächst der RESTRICT-Operator angewendet, der nur die Sätze aus der Tabelle Person auswählt, bei denen **Ort** = 'Bonn' ist. Anschließend wird auf diese Teiltabelle von PERSON der PROJECT-Operator angewendet, der Namen und Telefonnummern auswählt.

- c) Zeige die Artikelnummern, Artikelbezeichnungen und Artikelgruppenbezeichnungen aller Artikel

Man muss einen JOIN zwischen der Tabelle ARTIKEL und der Tabelle ARTIKELGRUPPE machen. Die Joinbedingung ist, dass der Wert des Attributs **ARTIKEL.ArtikelgruppeId** mit dem Wert des Attributs **ARTIKELGRUPPE.Id** übereinstimmen muss. Anschließend wird auf diesen Join eine Projektion angewendet, die die drei Spalten Artikelnummer, Artikelbezeichnungen und Artikelgruppenbezeichnung auswählt.

- d) Zeige die Artikelnummern, Artikelbezeichnungen und Artikelgruppenbezeichnungen aller Artikel, deren Bestandsmenge = 0 ist.

Man muss einen JOIN zwischen der Tabelle ARTIKEL und der Tabelle ARTIKELGRUPPE bilden. Die Joinbedingung ist, dass der Wert des Attributs **ARTIKEL.ArtikelgruppeId** mit dem Wert des Attributs **ARTIKELGRUPPE.Id** übereinstimmen muss. Anschließend wird auf diesen Join die Restriktion angewendet, die nur die Sätze herausfiltert, bei denen die Bestandsmenge = 0 ist. **Erst jetzt** wird die Projektion angewendet, die die drei Spalten Artikelnummer, Artikelbezeichnungen und Artikelgruppenbezeichnung auswählt.

2. Stellen Sie sich vor, die Tabelle BESTELLUNGEN sähe folgendermaßen aus: (Sie sehen die ersten sechs Sätze)

Id	Kundeld	ArtikelId	LfdNr	Artikelbezeichnung	Menge	Bestelldatum
1	13	1	1	Lampenschirme	3	03.10.24
2	23	2	1	Hilfsmotoren	7	17.09.24
3	13	2	1	Hilfsmotoren	5	12.07.24
4	13	2	2	Hilfsmotoren	12	01.02.25
5	12	2	1	Hilfsmotoren	1	01.02.25
6	3	3	1	Video-Recorder	1	07.12.24

Beachten Sie, dass zu den Attributen **KundeId** und **ArtikelId** noch eine sogenannte laufende Nummer – das Attribut **LfdNr** – mitgeführt wird, die anzeigt, die „wievielte“ Bestellung dieses Artikels von diesem Kunden hier vorliegt.

- a) Diese Tabelle hat mehrere mögliche Kandidaten für einen Primärschlüssel. Welche sind das?

- (i) die **Id**
- (ii) die Attributkombination (**KundeId**, **ArtikelId**, **LfdNr**)

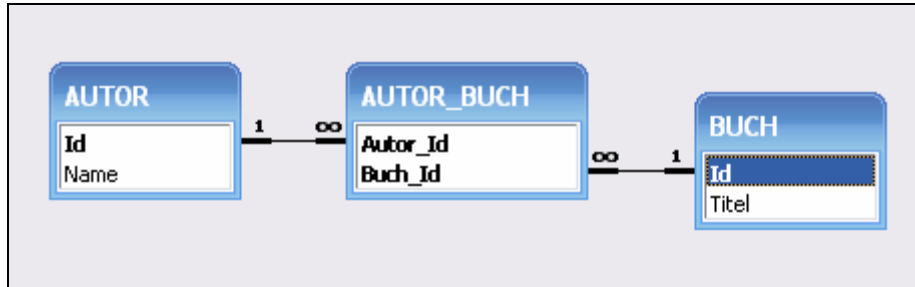
- b) Für welchen würden Sie sich entscheiden? Begründen Sie Ihre Entscheidung.

Beide Kandidaten sind möglich und vernünftig, für die **Id** spricht lediglich die einfachere Struktur.

3. Sie wollen eine Datenbank zur Verwaltung Ihrer Bibliothek einrichten. Unter anderem wollen Sie für die Buchtitel die Autoren speichern. Nun gibt es zu den meisten Buchtiteln nur einen Autor. Das verleitet Sie vielleicht zu einem falschen Tabellenentwurf. Je länger Sie darüber nachdenken, desto klarer wird Ihnen, dass Sie für diese Informationen drei Tabellen brauchen. Wie sehen diese Tabellen aus und warum ist das die beste Entscheidung?

Sie sehen hier einen möglichen Tabellenentwurf, der mehrere Vorteile hat:

- (i) Es gibt keine Redundanzen
- (ii) Ein Buch kann, ohne dass NULL-Werte auftreten, keinen Autor haben (wie z.B. die Bibel oder jedes Telefonbuch)
- (iii) Ein Buch kann einen oder auch mehrere (beliebig viele) Autoren haben, ohne dass man an der Tabelle BUCH irgendetwas ändern muss.



4. Formulieren Sie die vollständige Menge der Constraints – der Bedingungen – für die Korrektheit eines Satzes der Tabelle BESTELLUNGEN. Machen Sie dasselbe für die Tabelle ERLEDIGTEBESTELLUNGEN.

Die Constraints für BESTELLUNGEN könnten lauten:

- (i) **Id** ist eindeutig für jeden Satz und ungleich NULL
- (ii) **KundeId** hat stets einen Wert, der einem Wert von **Id** in der Tabelle KUNDE entspricht.
- (iii) **ArtikelId** hat stets einen Wert, der einem Wert von **Id** in der Tabelle ARTIKEL entspricht.
- (iv) **Menge** muss stets eine Zahl > 0 sein.
- (v) **Bestelldatum** muss ein gültiges Datum sein.

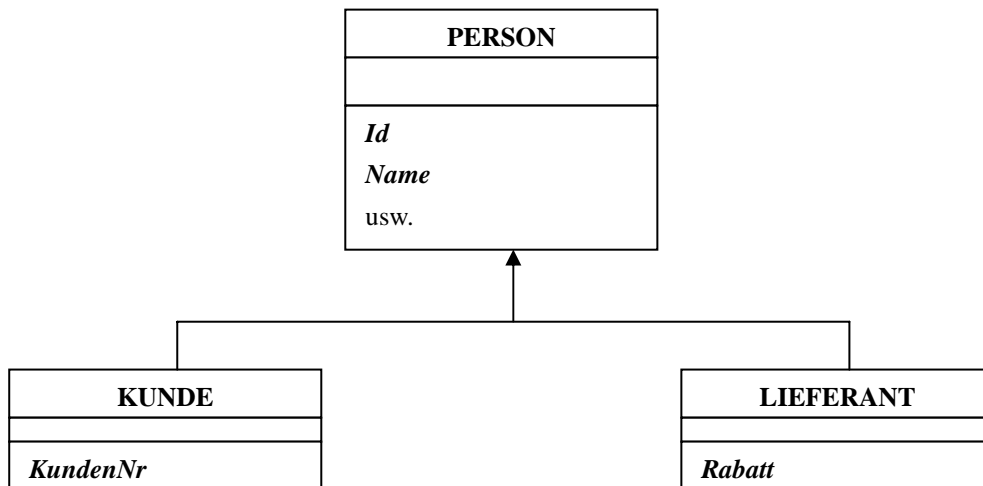
Die Constraints für ERLEDIGTEBESTELLUNGEN könnten lauten:

- (i) **Id** ist eindeutig für jeden Satz und ungleich NULL. Außerdem muss es einen Satz in der Tabelle BESTELLUNGEN mit derselben **Id** geben.
- (ii) **Commitdatum** muss ein gültiges Datum sein. Außerdem darf das **Bestelldatum** des Satzes in der Tabelle BESTELLUNGEN, der dieselbe **Id** hat, nicht hinter diesem **Commitdatum** liegen.

5. Nehmen Sie an, wir hätten in unserer Datenbank sämtliche DELETE-Optionen auf CASCADE gesetzt. Nun löschen wir aus der Tabelle ARTIKELGRUPPE den **einen** Satz mit der **Id 1** und der **Bezeichnung Kleinteile**. Wie viele Sätze werden dann insgesamt aus der Datenbank gelöscht? Welche Tabellen sind alle betroffen?

Es werden insgesamt **16 Sätze** gelöscht. Betroffen sind die Tabellen:
ARTIKELGRUPPE, ARTIKEL, BESTELLUNGEN und
ERLEDIGTEBESTELLUNGEN

6. Betrachten Sie noch einmal unsere zwei „Spezifikationen“ unserer Personentabelle:



Sie haben gelernt: Wenn wir diese Beziehung (Ein Kunde „ist eine“ Person, ein Lieferant „ist eine“ Person) relational abbilden wollen, müssen wir die jeweiligen Primärschlüssel aufeinander beziehen. Die Art dieses Bezugs muss folgenden Anforderungen genügen:

- Es darf nicht möglich sein, eine Person zu löschen, solange es Kunden oder Lieferanten mit demselben Primärschlüssel gibt.
- Man kann aber andererseits zulassen, dass es Personen mit Primärschlüsseln gibt, die weder in der Tabelle KUNDE noch in der Tabelle LIEFERANT vorkommen. Mit anderen Worten: Ich will zulassen, dass es Personen gibt, die we-

der Lieferant noch Kunde sind. (Wer weiß, wozu man die noch einmal brauchen kann). Ich brauche also keine Löschoptionen von Kunde/Lieferant in Richtung Person festzulegen.

Welche der drei folgenden Möglichkeiten ist die richtige Festlegung

- a) KUNDE.**Id** ist Fremdschlüssel mit Referenz zu PERSON.**Id** , aber PERSON.**Id** ist **nicht** Fremdschlüssel mit Referenz zu KUNDE.**Id**
- b) KUNDE.**Id** ist **nicht** Fremdschlüssel mit Referenz zu PERSON.**Id** , aber PERSON.**Id** ist Fremdschlüssel mit Referenz zu KUNDE.**Id**
- c) KUNDE.**Id** ist Fremdschlüssel mit Referenz zu PERSON.**Id** und PERSON.**Id** ist Fremdschlüssel mit Referenz zu KUNDE.**Id**

Nur eine der drei Möglichkeiten ist korrekt. Hinweis: Überprüfen Sie die referentielle Integrität.

Nur Möglichkeit a) ist korrekt. Nur so kann es Personen geben, die keine Kunden sind, während es hingegen unmöglich ist, dass es Kunden gibt, die keine Personen sind.

Bei Möglichkeiten b) und c) müsste jede Person auch ein Kunde sein.

Datenbanken

Theorie, Entwurf und Programmierung relationaler

Datenbanken

Schubert, M.

2007, XII, 344 S. Mit Online-Extras., Softcover

ISBN: 978-3-8351-0163-0