



## 1. Pflichtübung

### Einführung in die Datenbanksysteme Datenbanken für die Bioinformatik

*Annika Hinze, Manuel Scholz*

#### Bemerkungen

- Dies ist die erste der drei Pflichtübungen im Rahmen der Vorlesung. Für den Scheinerwerb ist ein Mittelwert von mehr als 60% notwendig.
- Die Übung muss bis spätestens Freitag den 2.6.06 um 12.00 Uhr im Fach des jeweiligen Tutors (bzw. im Büro) abgegeben werden.
- Bitte alle Namen der beteiligten Personen der Gruppe auf die Lösungszettel schreiben. Die Abgabe erfolgt in den Projektgruppen.
- Das Kopieren von Lösungen wird bei allen beteiligten Gruppen als Täuschungsversuch gewertet.

#### 1. Aufgabe (20 Punkte)

Betrachten Sie folgende Situationsbeschreibung:

“Der Weihnachtsmann ärgert sich schon seit Jahrhunderten über die schlechte Organisation seines Betriebes und möchte jetzt alles in einer Datenbank verwalten.

Im Betrieb des Weihnachtsmannes gibt es zwei Arten von Helfern: Gnome und Elfen. Beide haben einen eindeutigen Namen und eine wöchentliche Arbeitszeit. Gnome haben zusätzlich einen Stundenlohn - Elfen nicht, sie arbeiten weil es ihnen Spaß macht, jedoch haben sie dafür in der Arbeitswoche einen freien Tag (Mo – Fr) und einen Spitznamen, bei dem sie der Weihnachtsmann ruft. Gnome und Elfen helfen beim Herstellen und Verpacken der Geschenke, dabei wird genau vermerkt, wie viel Minuten ein Helfer an einem Geschenk gearbeitet hat. An einem Geschenk arbeiten zum Teil auch mehrere Helfer. Wenn ein Geschenk fertig ist wird es mit einer Geschenk-Nr. und dem Namen des beschenkten Kindes versehen. Die Geschenk-Nr. ist nur eindeutig bezüglich des beschenkten Kindes. Für jedes Geschenk wird der Inhalt gespeichert. Die Geschenke werden dann in Säcke gepackt. In einem Sack befinden sich Geschenke für mehrere Kinder, welche dem Weihnachtsmann mit Namen, Adresse und Alter bekannt sind. Die Kinder sind durch ihren Namen und ihre Adresse eindeutig bestimmbar. Außerdem weiß der Weihnachtsmann bei jedem Kind ob es artig war. Ein Kind kann beliebig viele Geschenke bekommen, jedoch auch keines (wenn es unartig war). Ein Sack hat eine eindeutige Sack-Nr. und wird auf einen bestimmten Schlitten gepackt. Ein Schlitten kann mit maximal 1000 Säcken beladen werden und hat eine eindeutige Fahrgestellnummer. Aus der riesigen Rentierherde in der alle Rentiere dem Weihnachtsmann mit eindeutigem Namen bekannt sind werden je nach Schlitten 2 bis 4 vor einen Schlitten gespannt.“

- a) Entwerfen Sie ein ER-Diagramm für den beschriebenen Sachverhalt. Identifizieren Sie alle notwendigen Entitäten und Beziehungen. Tragen Sie die Kardinalitäten in Min-Max Notation ein. Kennzeichnen Sie die Schlüssel der Entitäten. Fügen Sie keine künstlichen Schlüssel ein.
- b) Geben sie das Relationsschema zu Ihrem Entwurf an, fassen sie die Relationen so weit wie möglich zusammen.
- c) Definieren sie das Relationsschema in SQL.

## 2. Aufgabe (15 Punkte)

Gegeben sei eine Datenbank mit folgenden Relationen:

Sachbearbeiter(SBNr, SBName, Geschlecht, Wochenstunden)

Kunde(Kundennr, Name, Adresse, SBNr)

Bestellung(Nr, Kundennr, Artikel, Preis)

a) Beschreiben sie die Funktion der folgenden SQL Anfrage:

```
SELECT k.name, COUNT(*)
FROM Kunde k, Bestellung b
WHERE b.Kundennr = k.Kundennr
GROUP BY k.name
HAVING COUNT(*) = ( select MAX(COUNT(*))
                    FROM Kunde k, Bestellung b
                    WHERE b.Kundennr = k.Kundennr
                    GROUP BY k.name);
```

b) Formulieren Sie die folgenden Anfragen in SQL:

- 1) Alle Namen der Sachbearbeiter und die Anzahl der von ihnen betreuten Kunden von Sachbearbeitern, die mehr als 20 Kunden betreuen.
- 2) Alle Kunden (Namen) mit dem Gesamtpreis aller bisher von ihnen bestellten Artikel.
- 3) Alle Daten der Sachbearbeiter die einen Kunden betreuen, dessen Name mit "M" anfängt.
- 4) Durchschnittliche Wochenstunden der weiblichen und männlichen Mitarbeiter.
- 5) Alle Namen der Sachbearbeiter, die alle Kunden betreuen.
- 6) Alle Namen der Kunden, die mindestens einen Artikel haben, der 20% teurer ist als der Durchschnittspreis aller bisherigen Bestellung (von allen Kunden).

c) Betrachten sie folgende SQL Anweisung und formulieren Sie eine andere Anweisung mit EXISTS, welche die gleiche Ergebnismenge besitzt und ohne die DISTINCT Anweisung auskommt.

```
SELECT DISTINCT s.SBName
FROM Sachbearbeiter s, Kunde k, Bestellung b
WHERE k.SBNr = s.SBNr
AND k.Kundennr = b.Kundennr
AND b.Artikel = 'Harry Potter 5';
```