

## 1 Hinweise

Für die Bearbeitung des Praktikums können Sie das DBS des Lehrstuhls nutzen. Die nötigen Accountdaten wurden Ihnen per E-Mail mitgeteilt. Sie können sich mithilfe des SQL-Developers mit dem Lehrstuhl-DBS verbinden. Genauere Informationen zur Vorgehensweise erhalten Sie im Moodle-System.

Die relevanten Tabellen der Cocktaildatenbank, welche Sie zur Lösung benötigen, heißen wie folgt: COCKTAIL, GLAS, LOKAL, ZUTAT, COCKTAIL\_LOKAL, ZUTAT\_COCKTAIL.

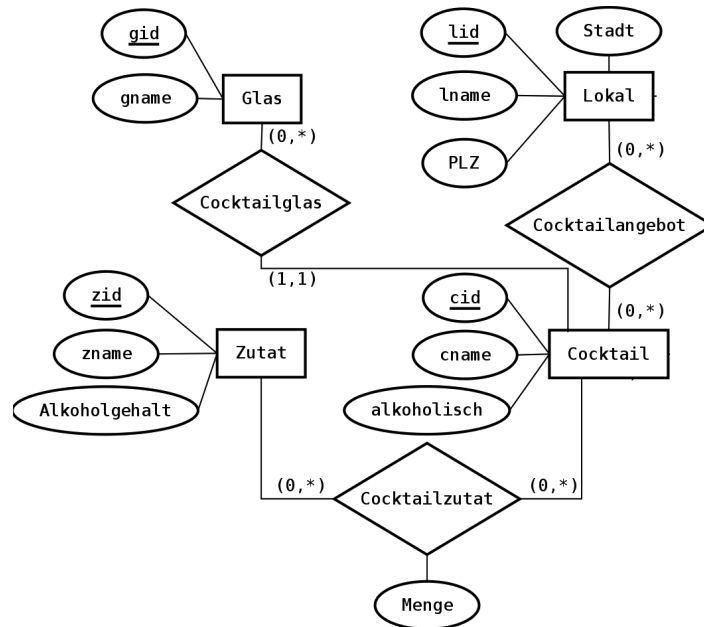
Das Skript zur Erzeugung der Cocktaildatenbank finden Sie im Moodle-System (*Praktikum.sql*). Dieses Skript legt im Falle einer *lokalen DB-Installation* (auf Ihrem eigenen Rechner) alle nötigen Tabellen an, wenn Sie es ausführen. Im DBS des Lehrstuhls sind alle Tabellen aber bereits für Sie angelegt worden. Wenn Sie das DBS des Lehrstuhls nutzen, müssen Sie das Skript also *NICHT* nochmal im SQL-Developer ausführen!

Beachten Sie bitte, dass es sich bei den angegebenen Tabellennamen lediglich um *Synonyme* für die tatsächlichen Tabellennamen handelt. Diese sollen die Arbeit vereinfachen. Vermeiden Sie daher selbst Tabellen mit einem dieser Namen anzulegen, da die Synonyme sonst nicht mehr nutzbar sind und Sie auf die eigentlichen Tabellen keinen direkten Zugriff haben. Sollten Sie es dennoch tun, müssen Sie Ihre jeweilige Tabelle zuerst wieder entfernen (*DROP*), bevor Sie das Synonym wieder nutzen können.

Nutzen Sie SQL-Befehle zur Beantwortung der Aufgaben und formulieren Sie diese so, dass sie ohne Kenntnisse der konkreten Daten funktionieren. Das bedeutet insbesondere, dass Sie für alle Anfragen nur genau die Daten verwenden können, die in der Aufgabenstellung vorgegeben sind und nicht von Hand zusätzliche Daten aus Tabellen herauslesen dürfen.

Die Daten in der Datenbank sind nur beispielhaft und teilweise unvollständig. Die Angabe der konkreten Ergebnisse der Anfragen sind für die Lösung der Aufgaben nicht nötig.

## 2 Vorgegebenes ER-Diagramm



## 3 QL

1. Geben Sie die Namen aller Cocktails aus, die in der Datenbank existieren (Tabelle COCKTAIL)!
2. Finden Sie alle Informationen über Lokale, welche die Postleitzahl 39108 haben (Tabelle LOKAL)!
3. Geben Sie die Postleitzahlen (PLZ) aller Lokale aus und vermeiden Sie dabei Duplikate (Tabellen LOKAL).
4. Welche Zutaten haben einen Alkoholgehalt zwischen 0 und 50 (Tabelle ZUTAT)?
5. Geben Sie die Namen (CNAME) aller Cocktails aus, die ein 'i' als letzten Buchstaben ihres Namens enthalten und aus mindestens drei Buchstaben bestehen (Tabelle COCKTAIL)!
6. Geben Sie die Namen aller Gläser und aller Cocktails in einer einspaltigen Tabelle aus. Nutzen Sie dazu eine geeignete Mengenoperation. (Tabellen GLAS und COCKTAIL)
7. Für welche Cocktails (CID ausgeben) gibt es noch kein Rezept in der Datenbank, d.h. welche Cocktails werden nicht in ZUTAT\_COCKTAIL erwähnt?
8. In welchen Lokalen (LNAME ausgeben) wird der Cocktail *Zombie* (CNAME) nicht serviert?
9. Was ist der durchschnittliche Alkoholgehalt der Zutaten (Tabelle ZUTAT)?
10. Bestimmen Sie mit einer Anfrage die Anzahl der Zutaten sowie das Minimum, das Maximum und die Summe aller Mengeneinheiten pro Cocktail (Tabelle ZUTAT\_COCKTAIL).

11. Bestimmen Sie den Alkoholgehalt aller Cocktails. Der Alkoholgehalt berechnet sich aus der Summe über alle (Alkoholgehalt der Zutat \* Mengeneinheiten der Zutat) geteilt durch die Summe aller Mengeneinheiten. (Tabelle ZUTAT\_COCKTAIL)
12. Geben Sie die Namen der Gläser aus, die für mehr als 2 Cocktails verwendet werden (Tabellen COCKTAIL und GLAS).

## 4 DDL und DML

1. Erstellen Sie eine Tabelle MEINE\_COCKTAILS, die den gleichen Aufbau wie COCKTAIL hat. Fügen Sie außerdem den Inhalt der Tabelle COCKTAIL in MEINE\_COCKTAILS ein. Denken Sie auch an die Vergabe der Integritätsbedingungen!
2. Fügen Sie in Tabelle MEINE\_COCKTAILS einen neuen Cocktail „Lila Kuh“ ein. Der Cocktail ist alkoholisch, wird in einem Schwenker serviert und hat die ID 18.
3. Der Cocktail „Lila Kuh“ heißt eigentlich „Blaue Kuh“. Korrigieren Sie diesen Fehler in Tabelle MEINE\_COCKTAILS.
4. Löschen Sie alle Cocktails aus der Tabelle MEINE\_COCKTAILS, die als Zutat „Campari“ enthalten.
5. Legen Sie eine Sicht Cocktail\_Alkoholgehalt mit den Attributen cid und alkoholgehalt an. Der Alkoholgehalt berechnet sich wie in Aufgabe 11. (Tabellen ZUTAT, COCKTAIL und ZUTAT\_COCKTAIL)
6. Vergeben Sie an einen beliebigen Nutzer alle Rechte für Tabelle MEINE\_COCKTAILS, einschließlich der Möglichkeit Zeilen zu löschen.
7. Entziehen Sie dem Nutzer das Recht zum Löschen in Tabelle MEINE\_COCKTAILS wieder.
8. OPTIONAL: Legen Sie eine Tabelle UPDATEWATCHER an mit den Attributen (tabellenname char(20), anzahl number) und machen Sie ein Insert mit VALUES('meine\_cocktails', 0). Erzeugen Sie anschließend einen *Trigger*, der bei jedem Update auf der Tabelle MEINE\_COCKTAILS ein entsprechendes Update auf der Tabelle UPDATEWATCHER ausführt. (Also das Attribut „Anzahl“ für tabellenname='meine\_cocktails' um 1 hochsetzt.)

## 5 Datenbankentwurf

Führen Sie anhand der folgenden Beschreibung einen Datenbankentwurf durch. Das Ergebnis dieser Aufgabe soll folgende Punkte beinhalten: (1) ein erweitertes ER-Diagramm, (2) die zugehörige Transformation des ER-Diagramms in das relationale Modell und (3) die daraus resultierenden DDL-Befehle zur Erzeugung des Datenbankschemas (einschließlich Integritätsbedingungen).

## 5.1 Beschreibung

Die Cocktaildatenbank soll um zusätzliche Informationen zu Personen und Veranstaltungen erweitert werden. Eine Person (Tabelle PERSON) besitzt eine eindeutige Personennummer (PID), einen Vornamen (VORNAME) und einen Nachnamen (NACHNAME). Für jede Person soll gespeichert werden, wie häufig sie einen bestimmten Cocktail gekauft hat (ANZAHL). Weiterhin kann eine Person genau einen Lieblingscocktail (LIEBLINGSCOCKTAIL) haben, muss es aber nicht. Eine Veranstaltung (Tabelle VERANSTALTUNG) hat eine eindeutige Veranstaltungsnummer (VID), einen Titel (VTITEL) und optional ein Datum (VDATUM). Jede Veranstaltung findet in genau einem Lokal statt. Veranstaltungen können von beliebig vielen Personen besucht werden und eine Person kann beliebige viele Veranstaltungen besuchen. Wenn eine Person aus der Datenbank gelöscht wird, sollen die Informationen über die von dieser Person gekauften Cocktails und besuchten Veranstaltungen ebenfalls automatisch entfernt werden.

## 5.2 Entwurf

- Erweitern Sie das existierende ER-Diagramm um die in der Beschreibung erwähnten Aspekte. Machen Sie dabei bitte aus Gründen der Übersichtlichkeit kenntlich, welche Teile vorgegeben sind und welche Teile von Ihnen stammen.
- Transformieren Sie das ER-Diagramm in das relationale Modell. Nutzen Sie die textuelle Notation zur Angabe der Relationenschemata.
- Leiten Sie aus den definierten Relationenschemata passende DDL-Befehle zur Erstellung der zugehörigen Tabellen ab. Achten Sie auf die Vergabe geeigneter Integritätsbedingungen.

## 5.3 Anfragen

Testen Sie Ihren Entwurf, indem Sie die die Tabellen im DBS des Lehrstuhls mithilfe der DDL-Befehle erzeugen. Fügen Sie anschließend eine kleine Menge von Testdaten in jede Tabelle ein und prüfen Sie die Ergebnisse der folgenden Anfragen auf Korrektheit.

1. Ermitteln Sie die Anzahl der Personen, die einen Lieblingscocktail haben.
2. Geben Sie die Nachnamen (NACHNAME) der Personen aus, die bisher keine Veranstaltung besucht haben.
3. Ermitteln Sie, wieviele Cocktails jede Person insgesamt gekauft hat. Geben Sie jeweils die Personennummer (PID), den Nachnamen (NACHNAME) der Person und die ermittelte Summe aus.
4. Geben Sie alle Paare von Personen (p1.NACHNAME, p2.NACHNAME) aus, welche den gleichen Lieblingscocktail haben und mindestens einmal die gleiche Veranstaltung besucht haben.
5. OPTIONAL: Bestimmen Sie alle Veranstaltungen, die in diesem Monat stattfinden (VDATUM).

Hinweis: In Oracle können Sie den aktuellen Monat z.B. mit folgendem Befehl bestimmen: `SELECT EXTRACT(MONTH FROM sysdate) FROM dual;`

6. OPTIONAL: Bestimmen Sie das Lokal (LNAME ausgeben), welches am häufigsten für Veranstaltungen benutzt wird.

Hinweis: In Oracle 11c können Sie die erste Zeile einer sortierten Tabelle mit der Bedingung `WHERE rownum = 1` ausgeben. Beachten Sie dabei aber die Abarbeitungsreihenfolge der einzelnen Teile einer Anfrage: `FROM`, `WHERE`, `GROUP BY`, `HAVING`, `ORDER BY`, `SELECT`.

## 6 Java

### 6.1 Aufgabenstellung

In Abschnitt 5 wurde die Cocktaildatenbank um Informationen zu Personen und Veranstaltungen erweitert. Ziel der Java-Aufgabe ist es nun, ein Anwendungsprogramm zu implementieren, welches die Verwaltung von Personen auf einem Client-Rechner ermöglicht. Das Programm soll über folgende Funktionalitäten verfügen:

- Tabelle `PERSON` erstellen
- eine neue Person mit PID, Vorname, Nachname und optionalem Lieblingscocktail hinzufügen
- alle Attribute aller vorhandenen Personen anzeigen
- alle Personen mit einem bestimmten Vornamen anzeigen (\*)
- alle Personen löschen
- alle Personen mit einem bestimmten Vornamen löschen
- Tabelle `PERSON` entfernen (\*)

Die Vorgabe besteht aus zwei Java-Dateien: `PersonManager.java` und `Persons.java`. Schauen Sie sich diese Dateien genau an, bevor Sie mit der Umsetzung beginnen.

Die Klasse `PersonManager` ist für die Verbindung mit dem DBS und die Menüführung zuständig. In dieser Klasse müssen Sie lediglich Ihre Verbindungsinformationen eintragen (Serveradresse, Benutzername und Passwort). Alle weiteren Änderungen führen Sie in der Klasse `Persons` durch.

Die Klasse `Persons` setzt die oben genannten Funktionalitäten um. Mit (\*) markierte Funktionalitäten sind dabei bereits Teil der Vorgabe und müssen nicht mehr implementiert werden. Sie können diese Funktionen als Beispiel/Vorlage für die Implementierung der verbleibenden Funktionalitäten nutzen.

Geben Sie Ihre angepassten Quelltextdateien `PersonManager.java` und `Persons.java` in einem ZIP-Archiv ab. Eine entsprechende Möglichkeit zur Abgabe wird rechtzeitig im Moodle-System freigeschaltet.

## 6.2 Kompilieren und Starten der Anwendung

Der Quellcode lässt sich direkt von der Kommandozeile übersetzen und ausführen. Der passende JDBC-Treiber liegt dem Archiv bei (`ojdbc14-1.jar`) und muss beim Starten zum Classpath hinzugefügt werden (Parameter `-cp`).

- Kompilieren (alle Plattformen):

```
javac *.java
```

- Start der Anwendung von der Kommandozeile (Windows):

```
java -cp ojdbc14-1.jar;. PersonManager
```

- Start der Anwendung von der Kommandozeile (Un\*x, Linux, MacOS X):

```
java -cp ojdbc14-1.jar:. PersonManager
```

Zur Kompilierung benötigen Sie eine Java-SE-SDK-Installation (JDK) (z.B. Version 1.7, neuere/ältere Versionen gehen sicherlich auch). Sollten Sie noch nicht mit Java programmiert haben, haben Sie in der Regel nur ein Java-Runtime-Environment (JRE) installiert. Sie benötigen das JDK dann zusätzlich zum JRE.

- Windows-Nutzer finden das Java-SE-SDK (JDK) hier:  
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- Linux-Nutzer haben wahrscheinlich bereits alles installiert oder beziehen das JDK über die Repositories der entsprechenden Distribution.
- Apple liefert Java 1.6 für MacOS X bis zur Version „Snow Leopard“ entweder per Update oder auf der „XCode/Developer-Tools“-CD-ROM. Ab MacOS X Version „Lion“ muss JDK 1.7 manuell installiert werden, siehe obiger Link.