

## Übungsblatt 2: Datenbank-Anfragen mit SQL

Wir nutzen als Testsystem Oracle 11g, auf das mithilfe des SQL Developer zugegriffen werden kann. Starten Sie den SQL Developer, loggen Sie sich in die Datenbank ein und machen Sie sich mit dem Programm vertraut.

Informationen zur Datenbank:

Host: orcist.imn.htwk-leipzig.de

Port: 1521

Service-Name: dblv13.imn.htwk-leipzig.de

Zugrunde liegt folgendes Relationenschema:

```
zutat (zid, zname, alkoholgehalt)
lokal (lid, lname, plz, stadt)
glas (gid, gname)
cocktail (cid, cname, alkoholisch, gid)    //alkoholisch={y,n}
person (pid, name, geburtsdatum)
zutat_cocktail (zid, cid, menge)
cocktail_person (cid, pid1, pid2)
cocktail_lokal (cid, lid)
```

Nutzen Sie zur Erstellung einer Beispiel-Datenbank ein SQL-Script, das Sie von folgender URL beziehen können:

<http://www.imn.htwk-leipzig.de/~kudrass/Downloads/CreateCocktail.sql>

1. In einem Trinkspiel soll jeder gegen jeden antreten. Geben Sie die zugehörige Liste der Spielpaare (Name, Name) aus.
2. Geben Sie die Namen der Gläser und Cocktails in einer einspaltigen Liste/Tabelle aus.
3. Für welche Cocktails gibt es noch kein Rezept in der Datenbank?
4. In welchen Lokalen wird kein Knieweich serviert?
5. Gesucht ist die Summe der Einheiten der einzelnen Cocktails.
6. Gesucht ist die Summe der Einheiten der einzelnen Cocktails, die Alkohol enthalten.
7. Gesucht sind alle Cocktails mit mehr als 2 Zutaten und die Anzahl der Zutaten pro Cocktail.
8. Der reale Alkoholgehalt eines Cocktails berechnet sich aus der Summe über alle (Alkoholgehalt der Zutat \* Einheiten der Zutat) geteilt durch die Summe aller Einheiten. Führen Sie eine geeignete Umbenennung der Attribute der Lösungsrelation durch.
9. Geben Sie für alkoholische und nicht alkoholische Getränke jeweils die Getränke aus, die mehr als 4 Zutaten haben.
10. Legen Sie eine Sicht `Cocktail_Alkoholgehalt` mit den Attributen `CID` und `Alkoholgehalt` an.
11. Gesucht sind jeweils von jedem Lokal der minimale und maximale ausgeschenkte Alkoholgehalt. Benutzen Sie die dazu erstellte Sicht.
12. Legen Sie eine Tabelle `Updatewatcher` an mit den Attributen (`tabellenname` `char(20)`, `anzahl` `number`) und machen Sie ein Insert mit `values('cocktail', 0)`.

13. Erzeugen Sie einen Trigger, der bei jedem Update auf der Tabelle `cocktail` ein entsprechendes Update auf der Tabelle `Updatewatcher` ausführt - also das Attribut `anzahl` beim Attribut `tabellenname='cocktail'` um 1 hoch setzt.
14. Erzeugen Sie einen Trigger der bei einem Insert auf `cocktail` diesen Cocktail dem Lokal hinzufügt (ein Insert in Tabelle `cocktail_lokal`), welches die wenigsten Cocktails anbietet.