

## Aufgabenblatt 4

### SQL

- Abgabetermin: **Mittwoch, 24.6.09, 11:00 Uhr**
- Zur Prüfungszulassung muss ein Aufgabenblatt mit mind. 25% der Punkte bewertet werden und alle weiteren Aufgabenblätter mit mindestens 50% der Punkte.
- Die Aufgaben sollen **in Zweiergruppen** bearbeitet werden.
- Abgabe:
  - per E-Mail an `dbs1-2009@hpi.uni-potsdam.de` mit Subject  
Abgabe DBS I: Aufgabenblatt <n> Namen
  - ausschließlich pdf-Dateien
  - eine Datei pro Aufgabe mit folgendem Dateinamen:  
`blatt<aufgabenblattNr>aufgabe<aufgabenNr><Nachnamen>.pdf`  
Bitte **keine Leerzeichen, Unterstriche, Umlaute, Sonderzeichen, ...** im Dateinamen!
  - **jedes Blatt beschriftet mit Namen und Matrikelnummern**
  - Wir korrigieren die Abgaben aufgabenweise. Das beschriebene Verfahren vereinfacht uns die Arbeit erheblich!

### Aufgabe 1: Daten der IMDB auf mangold importieren

Importiere die IMDB-Daten auf mangold in die Datenbank deiner Gruppe (siehe Hinweise) in das Schema imdb. Die IMDB-Daten findest du unter `R:\DBSI_naumann\imdbDaten`. Das Datenbankschema entspricht dem aus der Übung. Beachte jedoch die Reihenfolge der Attribute in den csv-Dateien (siehe erste Zeile). Die Datentypen sind wie folgt zu wählen: Attribute mit dem Namen ID oder DATE (hier eigentlich: Jahreszahl) sind vom Typ INTEGER, alle anderen Attribute vom Typ VARCHAR(255). 5 P

Hinweise:

- Jede Gruppe hat eine eigene Datenbank auf mangold in der Instanz db2 auf die nur die Gruppenmitglieder (und die Korrektoren) Zugriff haben. Der Datenbankname ergibt sich als `GROUP<no>`, wobei die jeweilige Nummer in der Datei `R:\DBSI_naumann\zuordnungDatenbankAufMangold.pdf` eingesehen werden kann.
- Zugriff auf die DB per ControlCenter (Steuerzentrale).
  - Steuerzentrale öffnen (auf den Poolrechnern: Start → Programme → IBM DB2 → DB2Copy1 → Allgemeine Verwaltungstools → Steuerzentrale)
  - in Baumansicht links: rechte Maustaste auf „Alle Systeme“ und „Hinzufügen...“; Systemname mangold; OK
  - Baum aufklappen, rechte Maustaste auf „Instanzen“ und „Hinzufügen...“; Instanzname „Aufspüren“ und „DB2“ auswählen; sinnvollen (beliebigen) Namen des Instanzknoten wählen; OK
  - Baum aufklappen; rechte Maustaste auf „Datenbanken“ und „Hinzufügen...“; Datenbankname „Aufspüren“ und die entsprechende Datenbank auswählen; OK
  - Rechte Maustaste auf die Datenbank und „Abfragen...“ liefert einen Befehlseditor, in dem beliebige SQL Anfragen abgesetzt werden können.

- Erzeugen der Relationen per „Create Table“ *im Schema imdb*.
- Import der Daten per „Import“. Gebt ein Log-File an und kontrolliert nach dem Import, ob wirklich alles geklappt hat. (Tabelle in der Objektsicht markieren, rechte Maustaste und „Importieren...“ liefert den Assistenten.)
- **Die DB2-Dokumentation findest du unter:**  
<http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp>

Es gibt je einen Punkt pro Relation mit korrekt befüllten Daten.

## Aufgabe 2: Text → SQL

Nenne für jede der folgenden natürlichsprachlichen Fragen eine geeignete SQL-Anfrage. Führe die Anfrage auf den Daten der IMDB aus und beantworte die jeweils in der Teilaufgabe gestellte Frage.

*Hinweise:*

- Benenne aggregierte Spalten so um, dass sinnvolle Spaltennamen ausgegeben werden.
  - Einige der Tabellen enthalten eine Spalte mit dem Namen ID. Hierbei handelt es sich jedoch um Zeilennummern und nicht um Werte, die etwa einen Film oder eine Person eindeutig identifizieren müssen.
  - Treffe *soweit nötig* geeignete Annahmen bezüglich des DB-Schemas.
- a) Gib alle Schauspieler an, die mit „Thalbach, Anna“ zusammen in einem Film gespielt haben. **2 P**
  - b) Wieviele Schauspieler (weibliche und männliche) gibt es? **2 P**
  - c) Gib alle Filmtitel und die dabei vertretene Rolle des Producers „Branco, Paulo (I)“ an. **2 P**
  - d) Gib alle Schauspielerinnen aus, die keine zugehörigen Einträge in der Filmtabelle haben! Jede Schauspielerin soll dabei nur einmal ausgegeben werden und die Ausgabe soll nach den Namen sortiert werden. Gib in der Abgabe nur die ersten drei Schauspielerinnen an. **3 P**
  - e) Gib alle männlichen Schauspieler an, die nach 1990 in mindestens zwei Filmen gespielt haben. **3 P**

## Aufgabe 3: Text → SQL (cont.)

Die Aufgabenstellung und die Hinweise aus Aufgabe 2 gelten weiter.

- a) Gib alle Schauspieler an, die *auch* in Filmen des Genres „Action“ spielen und deren Name mit „T“ beginnt. **3 P**
- b) Gib alle Schauspieler an, die *nur* in Filmen des Genres „Action“ spielen und deren Name mit „T“ beginnt. **4 P**
- c) Gib die Namen aller Producer an, die 2001 Filme in beliebten Genres gedreht haben. Ein „beliebtes Genre“ sei ein Genre in dem mindestens 200 Filme gedreht wurden. **5 P**
- d) Erstelle eine Top-3 Liste der Filme mit den meisten Schauspielern und Schauspielerinnen (Name)! Sortiere entsprechend. Hinweis: Recherchiere hierzu die `FETCH FIRST` Klausel. **4 P**

## Aufgabe 4: SQL → Text

Gib natürlichsprachlich wieder, wonach die folgenden SQL-Queries suchen.

```
a) WITH ProdAct AS
(
  SELECT Prod.Name AS PName,
         Act.Name AS AName,
         COUNT(Act.MOVIE_ID) AS CountM
  FROM (
         SELECT NAME, MOVIE_ID FROM ACTOR
         UNION ALL
         SELECT NAME, MOVIE_ID FROM ACTRESS
       ) AS Act
  INNER JOIN MOVIE AS Mov ON Act.MOVIE_ID = Mov.MID
  INNER JOIN PRODUCER AS Prod ON Mov.MID = Prod.MOVIE_ID
  GROUP BY Prod.Name, Act.Name
  ORDER BY Prod.Name, Act.Name
)

SELECT ProdAct.AName, ProdAct.PName, CountM
FROM ProdAct,
(
  SELECT AName, MAX(CountM) AS maxValue
  FROM ProdAct GROUP BY AName
) AS maxCount
WHERE ProdAct.AName = maxCount.AName AND
      ProdAct.CountM = maxCount.maxValue
```

5 P

b) Hinweis: Der Operator TABLESAMPLE BERNOULLI (prozent) ist IBM UDB-spezifisch und gibt zufällig ausgewählte Tupel der Ergebnismenge zurück.

```
SELECT *
FROM ACTOR AS a WHERE a.MOVIE_ID IN
(
  SELECT DISTINCT m.MID
  FROM MOVIE AS m
  TABLESAMPLE BERNOULLI (0.10)
  INNER JOIN GENRE AS g ON g.MOVIE_ID = m.MID
  WHERE g.GENRE NOT LIKE '%Adult%'
)
UNION
  SELECT a.MOVIE_ID AS mid
  FROM ACTOR AS a
  TABLESAMPLE BERNOULLI (0.01)
  INNER JOIN PRODUCER AS p ON p.NAME = a.NAME
  WHERE a.MOVIE_ID = p.MOVIE_ID
)
UNION
  SELECT *
  FROM ACTOR AS a
  TABLESAMPLE BERNOULLI (2)
```

Gib zusätzlich an ob Tupel doppelt ausgegeben werden und begründe deine Antwort! 6 P

## Aufgabe 5: Sichten

a) Erstelle auf Basis der IMDB-Relationen einen View mit den folgenden Informationen:

- die Namen aller Schauspieler und Schauspielerinnen,
- deren Geschlecht ('m'/'f'),
- deren Filme (movie\_id) und
- deren Rolle

Das Schema würde also wie folgt aussehen: ActorMW(name, sex, movie\_id, role). Erstelle die View in deiner Datenbank auf mangold und gib das Statement in deiner Abgabe an. **2 P**

b) Formuliere die Anfrage aus Aufgabe 4a) mit Hilfe des Views. **4 P**