

Einführung in Datenbanken

— Übungsblatt 11 (Mengenoperationen, Join) —

Ihre Lösungen laden Sie bitte in die Übungsplattform in StudIP hoch ([StudIP-Eintrag der Vorlesung], Reiter „Übungsplattform“, dann auf „Weiter zu ... Übungsplattform“).

Einsendeschluss ist Montag, der 31.01.2022, 18⁰⁰.

Hausaufgabe 11 (25 Punkte)

Teil a) bis c) beziehen sich auf die Beispiel-Datenbank von Folie 14-76:

- `Resultate_1(STUDENT, H, Z, E)`
- `Resultate_2(STUDENT, ATYP, PROZENT)`

Die Ergebnisse von Hausaufgaben (H), Zwischenklausur (Z) und Endklausur (E) sind hier zusammengefasst. Es ist jeweils der erreichte Anteil der Gesamtpunktzahl gespeichert (in Prozent). Dabei sind zwei verschiedene Darstellungen möglich:

Resultate_1			
STUDENT	H	Z	E
Ann Lloyd	80	90	95
Jim Ford	95	60	75

Resultate_2		
STUDENT	ATYP	PROZENT
Ann Lloyd	H	80
Ann Lloyd	Z	90
Ann Lloyd	E	95
Jim Ford	H	95
Jim Ford	Z	60
Jim Ford	E	75

Wählen Sie im Adminer das Schema „`resultate_public`“.

- a) (5 Punkte) Schreiben Sie eine Anfrage in SQL, die nur die Tabelle `Resultate_1` verwendet, und als Ergebnis den Inhalt der Tabelle `Resultate_2` liefert. Das erwartete Ergebnis ist also die Tabelle `Resultate_2` (ohne Tabellenüberschrift). Die Groß-/Kleinschreibung der Spaltennamen ist nicht wichtig. Eine spezielle Sortierung der Zeilen ist auch nicht verlangt.

Mit Ihrer Anfrage könnte man also `Resultate_2` als Sicht (virtuelle Tabelle) definieren, wenn `Resultate_1` als gespeicherte Tabelle vorliegt (oder als vorher definierte Sicht). In dieser Aufgabe brauchen Sie selbstverständlich keine Sicht zu definieren, da Sie das in unserer Adminer-Installation nicht testen können.

- b) **(5 Punkte)** Schreiben Sie nun eine Anfrage in SQL, die die umgekehrte Umstrukturierung vornimmt: Diese Anfrage soll nur die Tabelle `Resultate_2` verwenden, und die Tabelle `Resultate_1` als Ergebnis liefern (natürlich wieder mit beliebiger Groß-/Kleinschreibung der Spaltennamen und ohne vorgegebene Sortierung).
- c) **(5 Punkte)** Schreiben Sie nun eine Anfrage, um das Ergebnis von Lösungen zu Aufgabe a) zu testen. Dazu sei folgende „Lösung“ gegeben:

```
WITH LOESUNG(STUDENT, ATYP, PROZENT)
AS (SELECT DISTINCT STUDENT, 'H' AS ATYP, 80 AS PROZENT
    FROM Resultate_1)
...
```

Das Ergebnis ist natürlich falsch:

STUDENT	ATYP	PROZENT
Ann Lloyd	H	80
Jim Ford	H	80

Ihre Aufgabe ist nun, die „symmetrische Differenz“ mit dem korrekten Ergebnis, also `Resultate_2` zu berechnen. Das Ergebnis soll also folgende Zeilen enthalten:

- Zeilen von `LOESUNG`, die nicht in `Resultate_2` vorkommen („falsche Zeilen“) und
- Zeilen von `Resultate_2`, die nicht in `LOESUNG` vorkommen („fehlende Zeilen“).

In einer zusätzlichen Spalte `DIFF` sollen die falschen Zeilen mit „-“ und die fehlenden Zeilen mit „+“ markiert werden:

STUDENT	ATYP	PROZENT	DIFF
Jim Ford	H	80	-
Ann Lloyd	Z	90	+
Ann Lloyd	E	95	+
Jim Ford	H	95	+
Jim Ford	Z	60	+
Jim Ford	E	75	+

Zur Vereinfachung können Sie davon ausgehen, dass die zu testende Anfrage keine Duplikate liefert (und auch das korrekte Ergebnis keine enthält).

Sortierung und die Groß-/Kleinschreibung der Spaltenüberschriften sind egal.

Die folgende Anfrage ist in relationaler Algebra zu schreiben. Sie bezieht sich auf die gleiche Datenbank wie oben (plus die Tabelle `LOESUNG`). Unter folgender Webadresse können Sie Relax mit dieser Datenbank öffnen:

[<http://dbis-uibk.github.io/relax/calc/gist/e0a76d711d449d67914a2b188b7ea137>]

- d) Schreiben Sie die symmetrische Differenz (plus die zusätzliche Spalte `DIFF`) in relationaler Algebra. Das Ergebnis sollte also das gleiche sein wie bei c). Die Spaltennamen können sich etwas unterscheiden (Relax gibt immer einen Relationennamen mit aus).

Wenn Sie zusätzlich üben wollen, können Sie auch a) und b) in relationaler Algebra formulieren. Es gibt dafür aber keine Punkte.

Die letzte Aufgabe bezieht sich auf die bekannte Datenbank mit Informationen über Nahrungsergänzungsmittel (insbesondere Multivitamin-Tabletten):

- `Stoff_Kategorie(Kat, Bezeichnung, Sort_Nr)`
- `Stoff(Vit, Einheit, Tagesdosis°, Kat→Stoff_Kategorie)`
- `Praeparat(Pid, Name, Hersteller, PZN°, Anz°, Einheit°, Tagesdosis°, Gewicht°, Preis°, glutenfrei°, lactosefrei°)`
- `Inhalt(Pid → Praeparat, Vit → Stoff, Menge, Prozent°, Anmerkung°)`
- `Zutat(Pid → Praeparat, Seq, Name, Anmerkung°)`

Im Adminer ist es das Schema „vit_public“:

[https://dbs.informatik.uni-halle.de/edb?pgsql=db&username=student_gast&db=postgres&ns=vit_public]

- e) (**5 Punkte**) Geben Sie für alle Nahrungsergänzungsmittel den Gehalt an Calcium und an Vitamin D (`Vit='D'`) aus, inklusive dem Wert 0, wenn der jeweilige Vitalstoff nicht enthalten ist. Das Ergebnis sollte so aussehen:

Pid	Name	Calcium	Vitamin D
1	A-Z Vital	199.0	5.0
2	Centrum	162.0	5.0
3	A-Z Komplett	120.0	5.0
4	A-Z Complete	137.0	5.0
5	Kardiodrink	0	20.0
6	Mineraldrink	0	5.0
7	Magnesium Verla 300	0	0

Die Ausgabespalten sollen so heißen wie hier gezeigt (das schließt immer auch die Groß-/Kleinschreibung ein).