

## DAB 2 – Bewertetes Praktikum 2, FS2019

Vorname, Name: Pascal Brunner Punkte: \_\_\_\_\_

**Abgabe:** Die Abgabe der folgenden Aufgaben muss *in Papierform* erfolgen. Elektronisch abgegebene Lösungen erhalten 1 Punkt abzug.

**Abgabetermin:** gemäss Terminplan. Verspätete Abgaben werden *nicht akzeptiert*.

**Punkte:** Insgesamt gibt es 6 Punkte. Fehlende Abgaben werden mit 0 Punkten bewertet.

## Aufgabe 1 (2 Punkte): Vereinfachung Relationenformate

Gegeben seien folgende zwei Relationenformate:  $R(a, b, c)$  und  $S(b, c, d, e)$ . Alle Domänen seien ganze Zahlen. Der Operator  $\bowtie$  repräsentiert den Natürlichen Verbund.

Formen Sie die beiden folgenden relationalen Ausdrücke so um, dass daraus möglichst „kostengünstige“ Abfragepläne entstehen, also mögliche Selektionen und Projektionen so früh wie möglich. Die Resultate sind wieder als relationale Ausdrücke hinzuschreiben (je 1 Punkt).

- a)  $\sigma_{c>10}(\sigma_{b=3 \wedge d=4}(R \bowtie S))$   
ist äquivalent zu:

$$R \bowtie (\sigma_{c>10 \wedge b=3 \wedge d=4}(S))$$

- b)  $\sigma_{d>10}(\pi_{a,d}(R \bowtie S))$   
ist äquivalent zu:

$$\pi_{a,d}(R \bowtie (\sigma_{d>10}(S)))$$

## Aufgabe 2 (2 Punkte): Entschachtelung SQL-Anweisungen

Entschachteln Sie die beiden folgenden SQL-Anweisungen, d.h. schreiben Sie die entschachtelte Form (d.h. ohne Subquery) als SQL-Anweisung hin.

- a) Alle Flüge welche in Myakuri Aero oder Mogadishu starten:

```
SELECT * FROM Flug
WHERE Von IN (SELECT Flughafen_ID FROM Flughafen
WHERE Name = 'HYAKURI AERO' OR Name = 'MOGADISHU');
```

```
SELECT *
FROM Flug f
INNER JOIN Flughafen fh
ON f.von = fh.Flyhafen_ID
WHERE fh.Name = 'HYAKURI AERO'
OR fh.Name = 'MOGADISHU';
```

## DAB 2 – Bewertetes Praktikum 2, FS2019

b) Alle Flughäfen von denen kein Flug startet

```
SELECT * FROM Flughafen
WHERE NOT EXISTS (SELECT TOP 1 1 FROM Flug WHERE Flug.Von = Flughafen.Flughafen_ID)
```

SELECT \*  
FROM Flughafen fh  
LEFT JOIN Flug f  
ON fh.Flughafen\_ID = f.von  
WHERE f.von IS NULL;

### Aufgabe 3 (2 Punkte)

Die beiden Benutzer Meier und Huber arbeiten mit einer relationalen Datenbank mit der Tabelle T(A, B), die zu Beginn zwei Sätze (10, 20) und (30, 40) enthält (alle Domänen integer).

Benutzer	Operation	Ergebnis
Meier	BEGIN TRANSACTION; UPDATE T SET B=50 WHERE A=10;	(10,20) wird zu (10,50)
Huber	BEGIN TRANSACTION; SELECT B FROM T;	Als Ergebnis werden die beiden Werte 50 und 40 geliefert
Meier	COMMIT;	Transaktion des Benutzers Meier ist abgeschlossen
Huber	COMMIT;	Transaktion des Benutzers Huber ist abgeschlossen

Markieren Sie, welche Isolationslevel (SQL-Server) zu dem beschriebenen Ablauf passen (es sind mehrere Antworten erlaubt, pro Fehler ein Punkt abzug):

- ☒ READ UNCOMMITTED  
☐ READ COMMITTED  
☐ REPEATABLE READ  
☐ SNAPSHOT  
☐ SERIALIZABLE

Isolationsebene	Dirty Read	Non-Repeatable Read	Phantom Read	Lost Update
READ UNCOMMITTED	möglich	möglich	möglich	möglich
READ COMMITTED <small>Häufig in der Praxis</small>	verhindert	möglich	möglich	verhindert
REPEATABLE READ	verhindert	verhindert	möglich	verhindert
SERIALIZABLE	verhindert	verhindert	verhindert	verhindert

