

1 Aufgabe D1 - Multiple Choice

	Richtig	Falsch
Die Komplexität des Nested-Loop Joins ist stets höher als die des Merge-Joins.	<input type="checkbox"/>	<input type="checkbox"/>
Wenn die Daten vorab in sinnvoller Weise sortiert wurden, kann dies die Ausführung des Nested Loops beschleunigen.	<input type="checkbox"/>	<input type="checkbox"/>
Die Berechnung der Intervall-Grenzen ist beim Equi-Depth Histogramm aufwendiger als beim Equi-Width Histogramm.	<input type="checkbox"/>	<input type="checkbox"/>
Eine zustandsbehaftete Ausführung hat die Eigenschaft, dass der Zustand explizit erfasst und in einer Datenbank gespeichert wird.	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Beim asynchronen Zugriff wird die Kontrolle an den Aufrufer zurückgegeben, sobald die letzte Kopie des Datenobjekts geschrieben wurde.	<input type="checkbox"/>	<input type="checkbox"/>
Eine sinnvolle Möglichkeit der Auflösung von Inkonsistenzen von mehreren Versionen des Einkaufswagens ist, ihre Schnittmenge zu berechnen.	<input type="checkbox"/>	<input type="checkbox"/>
Der Kommunikationsaufwand in strukturierten P2P-Systemen ...	<input type="checkbox"/>	<input type="checkbox"/>
Vector Clocks sind Listen ...	<input type="checkbox"/>	<input type="checkbox"/>
PIQL ...	<input type="checkbox"/>	<input type="checkbox"/>
PIQL ...	<input type="checkbox"/>	<input type="checkbox"/>
Im PIQL ...	<input type="checkbox"/>	<input type="checkbox"/>
Der DataStop-Operator ...	<input type="checkbox"/>	<input type="checkbox"/>

2 Aufgabe D2 - Normalformen

2.1 Teilaufgabe a)

$\{ D, B \}$ und $\{ D, C \}$

2.2 Teilaufgabe b)

- $D^+ = \{ A, D, E, F, G \}$
- $B^+ = C^+ = \{ A, B, C, E, F, G \}$

TODO: Was kann ich daraus auf die NF folgern?

2.3 Teilaufgabe c)

TODO

3 D3 - SQL

3.1 Teilaufgabe a)

```
1 CREATE VIEW GoalsPerPlayer AS (  
2     SELECT player_id, name, team, SUM(goals) AS sum_goals  
3     FROM Player  
4     JOIN Participation ON Participation.player_id = Player.player_id  
5     GROUP BY player_id  
6 )
```

3.2 Teilaufgabe b)

```
1 CREATE VIEW AlwaysParticipating AS (  
2     SELECT player_id, name  
3     FROM Player  
4     JOIN Participation ON Player.player_id = Participation.player_id  
5     HAVING SUM(Participation.cup_id) =  
6         (  
7             SELECT SUM(cup_id)  
8             FROM cup_id  
9         )  
10 )
```

TODO: Geht das schöner?

3.3 Teilaufgabe c)

TODO: Keine Ahnung, was das soll. Das WHERE verwirrt mich. Werden hier nur Teams angeschaut, die weniger Punkte haben also alle Spieler ohne Mannschaft zusammen?

4 D4 - Transaktionen und Histories

TODO: Transaktionen

4.1 Teilaufgabe a)

TODO: Serialisierbarkeitsgraph

4.2 Teilaufgabe b)

TODO: Serialisierbarkeitsgraph