

**Übung zur Vorlesung *Datenbanksysteme I* im WS 20**

Gabriel Haas ([gabriel.haas@uni-jena.de](mailto:gabriel.haas@uni-jena.de))

[dbis1.github.io/courses/ws20/db1/](https://dbis1.github.io/courses/ws20/db1/)

**Blatt Nr. 08**

*Abgabe auf [moodle.uni-jena.de](https://moodle.uni-jena.de).*

**Hausaufgabe 1**

*Gegeben sei eine Relation*

$$R : \{[A : \text{integer}, B : \text{integer}, C : \text{integer}, D : \text{integer}, E : \text{integer}]\},$$

*die schon sehr viele Daten enthält (Millionen Tupel). Sie „vermuten“, dass folgendes gilt:*

- a)  $AB$  ist ein Superschlüssel der Relation*
- b)  $DE \rightarrow B$*

*Formulieren Sie SQL-Anfragen, die Ihre Vermutungen bestätigen oder widerlegen.*

**Hausaufgabe 2**

*Betrachten Sie das Relationenschema*

*Fahrplan:  $\{[Linie, Verbund, von, nach, von\ GPS, nach\ GPS, Preis, \#Fahrzeuge, Modus]\}$*

*mit der folgenden beispielhaften Ausprägung:*

<i>Linie</i>	<i>Verbund</i>	<i>von</i>	<i>nach</i>	<i>von GPS</i>	<i>nach GPS</i>	<i>Preis</i>	<i>\#Fahrzeuge</i>	<i>Modus</i>
<i>U6</i>	<i>MVV</i>	<i>GF</i>	<i>G</i>	<i>0N 0W</i>	<i>1S 0W</i>	<i>1€</i>	<i>20</i>	<i>U-Bahn</i>
<i>U6</i>	<i>MVV</i>	<i>G</i>	<i>GH</i>	<i>1S 0W</i>	<i>2S 0W</i>	<i>1€</i>	<i>20</i>	<i>U-Bahn</i>
<i>U6</i>	<i>MVV</i>	<i>GH</i>	<i>FR</i>	<i>2S 0W</i>	<i>5S 0W</i>	<i>3€</i>	<i>20</i>	<i>U-Bahn</i>
<i>U3</i>	<i>MVV</i>	<i>MF</i>	<i>GI</i>	<i>8S 0W</i>	<i>9S 0W</i>	<i>1€</i>	<i>16</i>	<i>U-Bahn</i>
<i>690</i>	<i>MVV</i>	<i>GF</i>	<i>DI</i>	<i>0N 0W</i>	<i>1N 0W</i>	<i>1€</i>	<i>5</i>	<i>Bus</i>
<i>690</i>	<i>MVV</i>	<i>DI</i>	<i>NF</i>	<i>1N 0W</i>	<i>3N 1W</i>	<i>2€</i>	<i>5</i>	<i>Bus</i>
<i>690</i>	<i>MVV</i>	<i>NF</i>	<i>EH</i>	<i>3N 1W</i>	<i>5N 2W</i>	<i>2€</i>	<i>5</i>	<i>Bus</i>
<i>S1</i>	<i>MVV</i>	<i>NF</i>	<i>EH</i>	<i>3N 1W</i>	<i>5N 2W</i>	<i>3€</i>	<i>8</i>	<i>S-Bahn</i>

- a) Bestimmen Sie die geltenden FDs.*
- b) Bestimmen Sie die Kandidatenschlüssel.*

### Hausaufgabe 3

Gegeben sei die durch folgende SQL-Statements definierte Ausprägung einer Relation.

```
create table kinder_fahrraeder (  
  person varchar(100) not null,  
  kind_name varchar(100) not null,  
  kind_alter integer not null,  
  fahrrad_typ varchar(100) not null,  
  fahrrad_farbe varchar(100) not null  
);  
insert into kinder_fahrraeder values  
  ('Thomas', 'Markus', 10, 'Trekking-Fahrrad', 'schwarz'),  
  ('Thomas', 'Markus', 10, 'Mountainbike', 'rot'),  
  ('Thomas', 'Johanna', 5, 'Trekking-Fahrrad', 'schwarz'),  
  ('Thomas', 'Johanna', 5, 'Mountainbike', 'rot');
```

Es gelten die beiden komplementären MVD

1.  $person \twoheadrightarrow \{kind\_name, kind\_alter\}$  und
2.  $person \twoheadrightarrow \{fahrrad\_typ, fahrrad\_farbe\}$

sowie die FD

3.  $kind\_name \rightarrow kind\_alter$ .

- a) Laura, das dritte Kind von Thomas, wird geboren. Fügen Sie Laura per SQL-Insert-Statement hinzu und beachten Sie dabei die MVDs. Formulieren Sie Ihr Statement so, dass es auch ohne Kenntnis der Fahrräder von Thomas funktioniert (d.h. nicht `insert ... 'Mountainbike', 'rot'`);).
- b) Allgemein gesprochen: In eine Relation  $R : \{[A, B, C]\}$  mit den MVDs  $A \twoheadrightarrow B$  und  $A \twoheadrightarrow C$  soll für ein bestimmtes  $a$  in Spalte  $A$  ein neuer Wert  $b$  in Spalte  $B$  eingefügt werden. Wie viele Tupel müssen hinzugefügt werden, damit die MVDs weiterhin gelten?
- c) Was passiert, wenn Thomas seine beiden Fahrräder verkauft?
- d) Überlegen Sie wie die Relation `kinder_fahrraeder` mittels Zerlegung in eine höhere Normalform gebracht werden kann. Versuchen Sie die Aufgabe intuitiv, also ohne Synthese- oder Dekompositionsalgorithmus zu lösen.
- e) Schreiben Sie ein SQL-Statement um zu prüfen ob die MVDs der Relation `kinder_fahrraeder` erfüllt sind.