

Aufgabe 1

a) Vorteile & Nachteile von DBS die mit/ohne die Mengen Operationen INTERSECT & EXCEPT:

- Mit den Mengen Operationen kann eine Tabelle als Mathematisches Objekt betrachtet. (Vorteil)
- INTERSECT verändert manchmal die Reihenfolge der Tabelle – man kann es mit ORDER BY Lösen (Nachteil)
- Mengen Operationen anders als JOIN brauchen keine vergleich Operationen. (Vorteil)
- EXCEPT Operationen anders als NOT IN gibt keine Duplikaten zurück - nur DISTINCT Ergebnisse (Nachteil wenn man die Duplikaten braucht)

Bsp. Mit EXCEPT: (Tabelle Kunde mit attribute Kund_nr)

```
SELECT kund_nr FROM Kunde
EXCEPT SELECT kund_nr FROM Verkauf;
```

Simulation – Mit NOT IN :

```
SELECT kund_nr FROM Kunde
WHERE kund_nr NOT IN (SELECT kund_nr FROM Verkauf);
```

Bsp. Mit INTERSECT:

(2 Einfache Tabellen Hiking und Traveling Mit SerialNummer und Name – siehe unten)

```
SELECT * FROM Hiking INTERSECT SELECT * FROM Traveling;
```

Simulation – Mit NATURAL JOIN

```
SELECT * FROM Hiking NATURAL JOIN Traveling;
```

SSN	FullName
683809234	Francis Forster
238802932	Catherine Cross
730990231	David Drew
488300922	Priscilla Pinkleton
388392883	Howard Hutchings
578389982	Jacqueline Jackson

Table named Hiking

SSN	FullName
683809234	Francis Forster
339500391	Thomas Trump
730990231	David Drew
578389982	Jacqueline Jackson
979385894	Christopher Cummings
988233885	Andrea Atkins
447743821	Yvonne York

SSN	FullName
683809234	Francis Forster
339500391	Thomas Trump
730990231	David Drew
578389982	Jacqueline Jackson
979385894	Christopher Cummings
988233885	Andrea Atkins
447743821	Yvonne York
343438829	William Winston
399038820	Vincent Volt
238802930	Catherine Cross

Table named Travelling

b)i) in Worten:

Wir suchen ein Produkt b, dessen **Preis größer 209** ist. Falls dieser b eine **kleinere Größe** und **kleineres Gewicht** als ein beliebiges anderes Produkt (c) dann muss auch gelten, dass der **Preis kleiner** ist jedoch die **Menge größer** ist.

ii) Wir wissen von der Mengenlehre dass $A \Rightarrow B$ ist äquivalent zu $\neg A \vee B$

In dem Fall definieren wir:

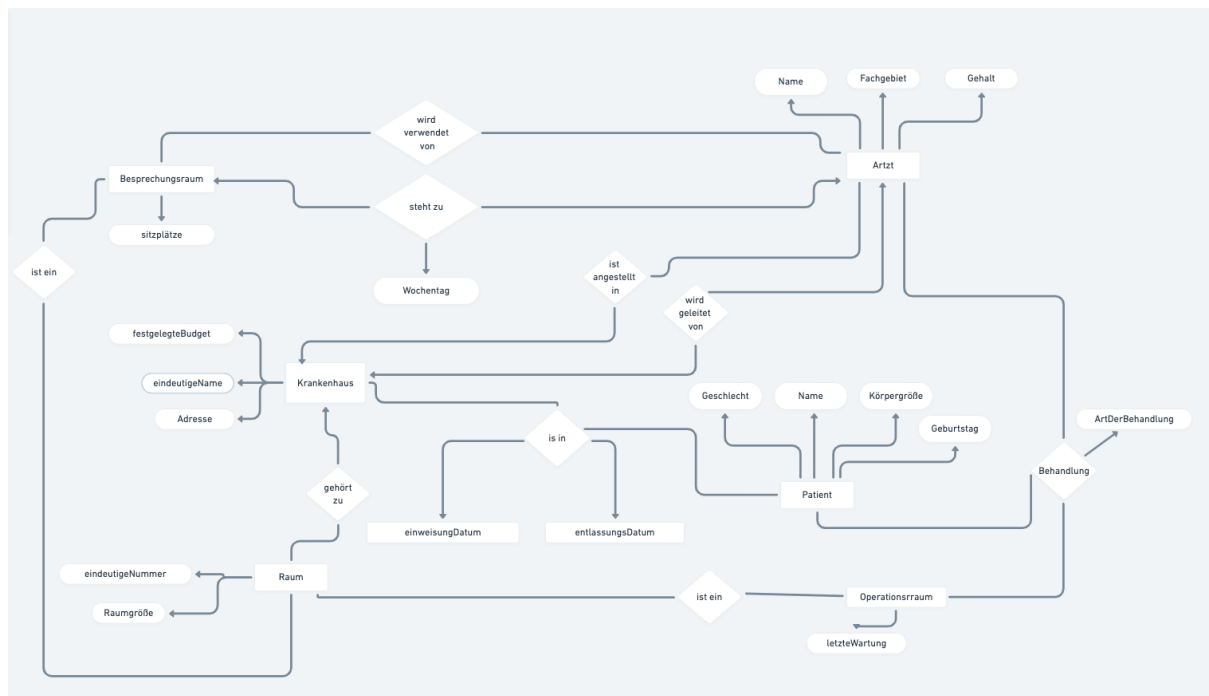
- $A = (c.\text{Größe} > b.\text{Größe}) \wedge (c.\text{Gewicht} > b.\text{Gewicht})$

- $\neg A = \neg ((c.Größe > b.Größe) \wedge (c.Gewicht > b. Gewicht)) =$
 $= \neg(c.Größe > b.Größe) \vee \neg (c.Gewicht > b. Gewicht)$
- $B = (c.Preis > b. Preis) \wedge (c.Menge > b. Menge)$

Daraus erfolgt $\neg A \vee B =$

$(\neg(c.Größe > b.Größe) \vee \neg (c.Gewicht > b. Gewicht)) \wedge ((c.Preis > b. Preis) \wedge (c.Menge > b. Menge))$

Aufgabe 2: ER Diagramm



Aufgabe 3:

a)

```
CREATE TABLE Teilnehmer(  
    TNr INTEGER PRIMARY KEY,  
    Vorname VARCHAR(100),  
    Nachname VARCHAR(100) NOT NULL,  
    Geburtsdatum CHAR(10),  
    Wohnort VARCHAR(100)  
);  
  
CREATE TABLE Gruppe(  
    GNr INTEGER PRIMARY KEY,  
    Gruppenname VARCHAR(100) NOT NULL,  
    Leiter INTEGER REFERENCES Teilnehmer(TNr);  
);  
  
CREATE TABLE Mitglied (  
    Teilnehmer INTEGER REFERENCES Teilnehmer (TNr),  
    Gruppe INTEGER REFERENCES Gruppe (GNr)  
);
```

b) Da TNr eine PrimaryKey ist, können wir die Einträge nicht ändern.

c)

```
SELECT GNr, Gruppenname  
FROM Gruppe NATURAL JOIN Hackathon  
WHERE Hackathon.Ort = "Hannover" AND Hackathon.Kapazität > 15  
  
-- Zusätzlich Namen der entsprechenden Hackathon  
  
SELECT Name  
FROM Hackathon  
WHERE Hackathon.Ort = "Hannover" AND Hackathon.Kapazität > 15
```

d)

```
CREATE VIEW JAD1 AS  
SELECT Datum, COUNT(DISTINCT HNr) AS anzahlHack  
FROM Hackathon  
GROUP BY Datum;  
  
SELECT Datum FROM JAD1  
WHERE anzahlHack = (SELECT MAX(anzahlHack) FROM JAD1);
```

e)

```
Create VIEW Jad2 AS  
SELECT Gruppe, SUM(Punkte) AS allPunkte  
FROM Teilnahme
```

```
GROUP BY Gruppe;

SELECT Gruppe, IF(allPunkte = NULL, 0, allPunkte) FROM Jad2
```

f) Rationale Algebra:

$\pi_{\text{GNr, Gruppenname}}(\sigma_{\text{Teilnahme.Hackathon} = 17 \vee \text{Teilnahme.Hackathon} = 94}(\text{Teilnahme} \bowtie \text{Gruppe}))$

Tupelkalkül:

Schema(gr) = Schema(Gruppe);
 $\{ [\text{gr.GNr}, \text{gr.Gruppenname}] \mid \text{Gruppe}(\text{gr}) \wedge (\exists t \in \text{Teilnahme}) \wedge (\text{gr.GNr} = t.\text{Gruppe}) \wedge (t.\text{Hackathon} = 17 \vee t.\text{Hackathon} = 94) \}$

Aufgabe 4:

S (A, B, C, D, E, F, G, H)

- $G \rightarrow H$ • $F, G \rightarrow E, C$ • $F, G \rightarrow D$
- $B, A \rightarrow C$ • $E \rightarrow A, H$ • $F, G, A \rightarrow B$

a) {F, G} einzige Schlüsselkandidat!

S 2.ter NormalForm?

Die Bedingungen der zweiten Normalform sind:

- Jedes Attribut ist prim oder
- voll funktional abhängig von jedem Schlüssel Kandidaten!

Wir betrachte die Attribute in S die unter verdacht sind: (A,B,C,D,E)

- A voll funktional abhängig von E ($E \rightarrow A, H$) aber E von F,G ($F, G \rightarrow E, C$) \rightarrow auch von (F,G)
- B voll funktional abhängig von F,G ($F, G, A \rightarrow B$)
- C voll funktional abhängig von B,A (BA sind abhängig von F,G)
- D voll funktional abhängig von F,G ($F, G \rightarrow D$)
- E voll funktional abhängig von F,G ($F, G \rightarrow E, C$)
- F Prim
- G Prim

Da Alle Fälle entweder Prim oder voll funktional abhängig von F,G sind, erfüllt S die Eigenschaften von dem 2ter Normalform.

b) Relationsschema R(A, B, C, D, E, F, G, H, I)

- $C, B \rightarrow A$ • $E \rightarrow C, F.$ • $B, F \rightarrow I$
- $H \rightarrow C.$ • $C, F \rightarrow B.$ • $F \rightarrow D, G$ • $B, I \rightarrow E$

Nur auf der linken Seite?

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

H bestimmt ganz R \rightarrow H ist ein Superschlüssel

Noch zu beweisen wäre, dass dieser Minimal ist.

- Sei $K \subseteq R$, und K ein Schlüsselkandidat für R wenn $K \rightarrow R$ also $R \subseteq K^+$ und K Minimal ist, also $\forall \alpha \in K: \alpha \rightarrow R$ (Keine Teilmenge von K bestimmt R)

In unserem Fall es existiert überhaupt keine Teilmenge von $\{H\}$ „eine Menge mit einem einzigen Element H“!!

\rightarrow Damit ist $\{H\}$ minimal aber auch eindeutig (+ es gibt keinen anderen) $\odot \rightarrow$ Es handelt sich hier um einen Schlüsselkandidat!

Aufgabe 5:

- a) Das einzige Objekt, auf dem 2 verschiedenen Transaktionen geschieht ist w.
 $r2(w) - r1(w) - w2(w)$ Keine Anomalie!

b)

$S2 = (w1(w), r2(v), r1(y), w1(v), r1(v), w2(v), w1(y), r2(w), r2(y), r1(v), w1(y), w2(w))$

- $R2(v) - w1(v) - w2(v)$ – Lost Update
- $R1(v) - w2(v) - r1(v)$. – Non Repeatable Read
- $w1(y), r2(y), w1(y)$ Dirty Read

c)

$S3 = (r2(y), r2(x), w1(x), r1(x), w1(y), r2(w), w2(x), r1(x), r2(y), w2(y), w2(w), w2(w))$

- $r2(y), w1(y), r2(y)$ – NON Repeatable Read