

Klausur Datenbanken
Wintersemester 2020/2021
Prof. Dr. Wolfgang May
3. März 2021, 11:15-12:30 Uhr
Bearbeitungszeit: 75 Minuten
(Ilias-basierte Klausur)

Vorname:

Nachname:

Matrikelnummer:

Zwecks besserer Lesbarkeit (insbesondere auch für Nicht- $\{\text{Mut}/\text{d}/\text{Va}\}$ tersprachler*innen) wird in der Aufgabenstellung auf gegenderte Sprache verzichtet.

- Im Folgenden wird die Aufgabenstellung beschrieben. Für das ER-Diagramm, und auch für das relationale Schema ist ein Teil bereits vorgegeben. In den Aufgaben 1 und 2 (wahlweise, ER-Diagramm) und 3 (Umsetzung in das relationale Modell) müssen nur noch die fehlenden Teile ergänzt werden.
- Bearbeiten Sie zuerst *entweder* Aufgabe 1 oder 2 (ER-Diagramm), dann Aufgabe 3 (Umsetzung in das Relationale Modell), und dann die weiteren Aufgaben, die darauf aufbauen.

	Max. Punkte	Schätzung für "4"
Aufgabe 1 (ER-Modell (Foto oder PDF-Upload))	12	10
Aufgabe 2 (ER-Modell (ASCII-Art-Modus))	0	0
Aufgabe 3 (Transformation in das Rel. Modell)	4	3
Aufgabe 4 (Relationales Modell: CREATE TABLE)	4	3
Aufgabe 5 (Relationales Datenmodell: Nullwerte)	2	1
Aufgabe 6 (Anfragen (1))	6	6
Aufgabe 7 (Anfragen (2))	4	3
Aufgabe 8 (Anfragen (3))	4	2
Aufgabe 9 (Anfragen (4))	5	1
Aufgabe 10 (Anfragen (5))	8	3
Aufgabe 11 (Anfragen (wahlweise: Bäume als Upload))	0	0
Aufgabe 12 (Open Book – Open Web – Open Documentation)	4	0
Aufgabe 13 (Eine etwas kompliziertere SQL-Anfrage)	6	0
Aufgabe 14 (Geht das auch in der Algebra?)	2	1
Aufgabe 15 (Transaktionen 1: Single Choice)	1	1
Aufgabe 16 (Transaktionen 2: Transaktionalität in ILIAS)	4	2
Summe	66	36

Note:

Themenstellung: Corona-Impfung

Alle Klausuraufgaben basieren auf einem gemeinsamen “Auftrag”: In der Klausur soll eine Datenbank für die Verwaltung der Impfungen gegen Covid-19 entwickelt werden.

Die Aufgabenstellung entspricht der Situation zu Zeitpunkt der Klausur im März 2021, wo Impfungen ausschließlich über Impfzentren organisiert wurden. In Arztpraxen und von Betriebsärzten wurde erst ab Mai 2021 geimpft.

1. Die in Deutschland lebenden Personen sollen geimpft werden. Zu jeder Person ist der Name (Vorname+Nachname), Geburtsdatum, eine eindeutige ID, Adresse (Stadt, PLZ, Straße und Hausnummer) und eine Telefonnummer gespeichert. (In dieser Klausur besteht die ID zur Vereinfachung immer nur aus den Initialen der Personen.)

Elsa Müller (ID: EM) und Fritz Müller (ID: FM), geboren am 4.3.1943 bzw. am 3.4.1934 wohnen im Seniorenheim Habichtsweg 55, 37075 Göttingen.

Markus Schmidt (ID: MS), geboren am 7.7.1977, wohnt im Kreuzberggring 1 in 37075 Göttingen. Er ist Altenpfleger.

2. Für jede Person ist gespeichert, ob sie Arzt ist.

Dr. Robert Koch (ID: RK) ist Arzt, geboren am 11.12.1983, und wohnt in der Robert-Koch-Straße 1, 37073 Göttingen.

Dr. Johann Faustus (ID: JF) ist Arzt, geboren am 29.2.1980, wohnt in der Goethestrasse 42, 37073 Göttingen.

3. Es gibt verschiedene Impfstoffe. Jeder Impfstoff hat einen eindeutigen Namen. Außerdem sind der Hersteller sowie die durchschnittliche Wirksamkeit nach Studien, das Datum der Zulassung in Deutschland, und ggf. Alterseinschränkungen (von/bis) der zu impfenden Personen gespeichert. Es kann (in Zukunft) mehrere Impfstoffe desselben Herstellers geben.

Der Impfstoff BNT162B2 des Herstellers Biontech hat eine Wirksamkeit von 95%, wurde am 21.12.2020 zugelassen, und ist für Personen ab 16 Jahren empfohlen.

Der Impfstoff mRNA-1273 des Herstellers Moderna hat eine Wirksamkeit von 95%, wurde am 6.1.2021 zugelassen, und ist für Personen ab 18 Jahren empfohlen.

Der Impfstoff AZD1222 des Herstellers Astra-Zeneca wurde am 29.01.2021 zugelassen und ist für Personen im Alter von 18 bis 64 Jahren empfohlen.

4. Impfzentren: jedes Impfzentrum hat einen Namen (z.B. “Göttingen I”, “Göttingen II”, “UMG Göttingen”, “Herzberg”), eine Adresse (*Godehardstraße 11, 37081 Göttingen; Anna-Vandenhoeck-Ring 13, 37081 Göttingen; Robert-Koch-Straße 40, 37075 Göttingen; Heidestraße 10, 37412 Herzberg am Harz*), und ist einem Kreis (die oben genannten alle dem Kreis *Göttingen*) zugeordnet. Das Impfzentrum *Northeim* mit der Adresse *Wolfshof 10, 37154 Northeim* ist dem Kreis *Northeim* zugeordnet.
5. Für jeden Kreis (= deutsche Verwaltungseinheit, entspricht district, county) ist der Name, die Bevölkerungszahl, und zu welchem Bundesland er gehört, gespeichert. Der Kreis *Göttingen* hat 330000 Einwohner, der Kreis *Northeim* hat 130000 Einwohner, beide gehören zum Bundesland *Niedersachsen*.

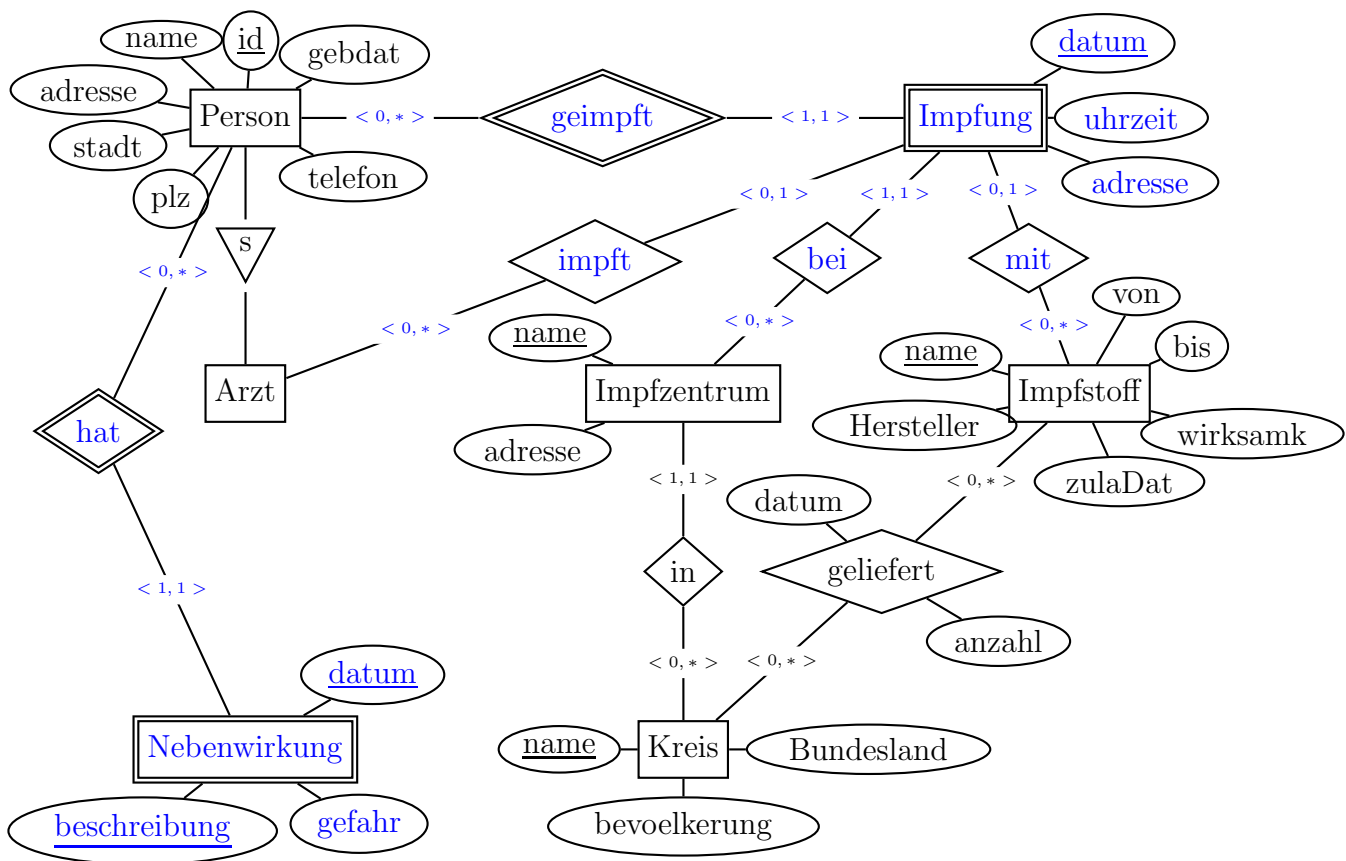
6. In der zu entwerfenden Datenbank wird gespeichert, wann wieviele Ampullen der Impfstoffe an die einzelnen Kreise geliefert wurden. Z.B. wurden am 23.12.2020 380 Ampullen *BNT162B2* an den Kreis *Göttingen* und 195 Ampullen an den Kreis *Northeim* geliefert. Am 4.2.2021 wurden noch einmal 380 Ampullen *BNT162B2* an den Kreis *Göttingen* geliefert. Am 10.2.2021 wurden 200 Ampullen des Impfstoffs *AZD1222* an den Kreis *Göttingen* geliefert.
7. Das Wesentliche an der Datenbank ist die Speicherung der einzelnen Impfungen: (das ist auch der Teil der Datenbank, der im ER-Modell in Aufgabe 1 ergänzt werden muss.)
- jede Person kann (soll) zweimal (bisher üblicherweise mit demselben Impfstoff) geimpft werden. Weitere Impfungen können in den nächsten Jahren folgen.
 - Jede einzelne Impfung ist verwaltungstechnisch (und logistisch) einem Impfzentrum zugeordnet. Die Impfungen einer Person können aber an verschiedenen Orten durchgeführt werden:
 - Pflegebedürftige Personen, die in Seniorenheimen wohnen, werden in diesen geimpft.
 - Pflegebedürftige Personen, die zuhause wohnen, können zuhause geimpft werden.
 - alle anderen werden in dem jeweiligen Impfzentrum geimpft.
 - Vor der Impfung muss immer erst ein Termin vereinbart werden, der in der Datenbank abgelegt wird (Person, wann (Datum+Uhrzeit), bei welchem Impfzentrum, ggf. abweichende Heimadresse),
 - nach der Impfung wird dazu gespeichert, mit welchem Impfstoff und von welchem Arzt geimpft wurde.
 - *Elsa Müller* (ID: EM) und *Fritz Müller* (ID: FM) wurden in ihrem Seniorenheim am 28.12.2020 um 14:30 Uhr bzw. 14:40 Uhr von *Dr. Johann Faustus* mit dem Impfstoff *BNT162B2* geimpft. Diese Impfungen sind dem Impfzentrum *Göttingen I* zugeordnet.
 - Die zweite Impfung des Ehepaars Müller erfolgte am 8.2.2021 um 10:00 Uhr bzw. 10:10 Uhr im Seniorenheim durch *Dr. Robert Koch* mit *BNT162B2*. Auch diese Impfungen sind dem Impfzentrum *Göttingen I* zugeordnet.
 - *Markus Schmidt* wurde am 15.2.2021 um 9:30 im Impfzentrum *Göttingen I* von *Dr. Johann Faustus* mit dem Impfstoff *AZD1222* geimpft.
Er hat schon einen Termin für die zweite Impfung am 10.3.2021 um 10:00, wieder im Impfzentrum *Göttingen I*.
8. Nebenwirkungen/Komplikationen werden ebenfalls erfasst. Mehr wird an dieser Stelle dazu nicht gesagt.
Die entsprechende Tabelle (mit Beispieldatensätzen) ist in Aufgabe 3 im relationalen Schema angegeben.

Aufgabe 1 (ER-Modell (Foto oder PDF-Upload) [12 Punkte])

Vervollständigen Sie das angegebene ER-Modell (einschl. Kardinalitäten):

(d.h. machen Sie eine eigene Grafik (z.B. mit draw.io) in der Sie von dem vorhandenen nur die benötigten Entitätstypen (ohne deren Attribute) "neu" zeichnen und das Gesuchte dazwischenfügen, oder kopieren Sie das untenstehende ER-Diagramm in ein Zeichenprogramm und malen rein ... oder verwenden ein Blatt Papier+Smartphone.)

- Modellieren Sie die Daten zu den Impfungen wie oben beschrieben (8 P)
- Modellieren Sie die Daten zu den Nebenwirkungen wie im Text und in der Tabelle zu Aufgabe 3 beschrieben (4 P)
- Wenn Sie die ER-Modell-Aufgabe als pdf/png/jpg- oder Foto-Upload bearbeiten wollen, machen Sie dies **HIER**,
- wenn Sie die Aufgabe stattdessen als ASCII-Art im Editor bearbeiten wollen, machen Sie dies in Aufgabe 2.



Lösung

Eine naheliegende Lösung ist obenstehend eingezeichnet.

Die Nebenwirkungen sollte man nicht mit "Impfung" sondern mit der Person verbinden. Das Datum der Nebenwirkungen Impfung, liegt häufig später als das Datum der Impfung. Über die Person lässt sich dann aber herausfinden, zu welcher Impfung die Nebenwirkungen gehören.

- Punkte für Impfung:
 - als weak: 2P richtig angebunden, 4 (8 Stück) Kardinalitäten, 1 weak+id, 1 key
 - als Beziehung: 4 richtig angebunden, 4 Kardinalitäten

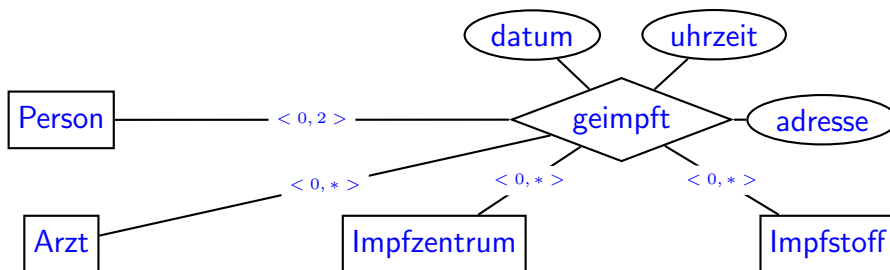
- Punkte für Nebenwirkungen:
1 richtig angebunden, 1 Kardinalitäten, 1 weak+id, 1 keys

Man kann anstatt des (schwachen) Entitätstyps "Impfung" auch eine Beziehung "geimpft" modellieren. Dann kann man allerdings keine Schlüsselbedingungen, und auch die Mindestkardinalitäten bzw. erlaubter nulls von geimpft nicht mehr angeben. Eine solche Instanz darf nur existieren, wenn Person+Impfzentrum feststehen, Person und Datum ist key, und Arzt und Impfstoff dürfen null sein.

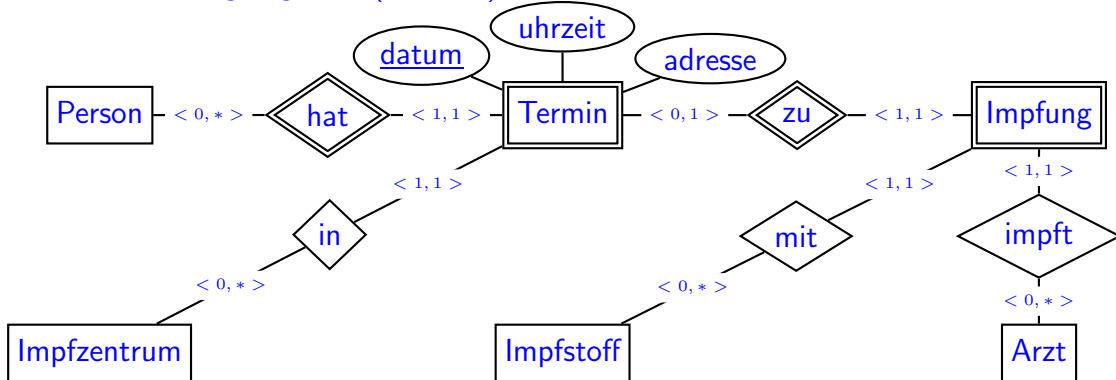
Dies ist eine Beispiel für Reifikation einer *mehrstelligen* Beziehung: Das "impfen" ist eine Beziehung zwischen einer Person, einem Arzt, einem Impfzentrum und einem Impfstoff. Daraus wird –auch in der Realwelt– der Entitätstyp "Impfung" gebildet. Umgekehrt kann man den reifizierten Entitätstyp –unter Verlust von Genauigkeit in den Kardinalitäten und Schlüssel– auch de-reifizieren und als Beziehung modellieren.

Wenn man z.B. doch die Nebenwirkungen direkt zu einer Impfung assoziieren wollte, geht das nur sauber wenn die Impfung selber ein Entitätstyp ist.

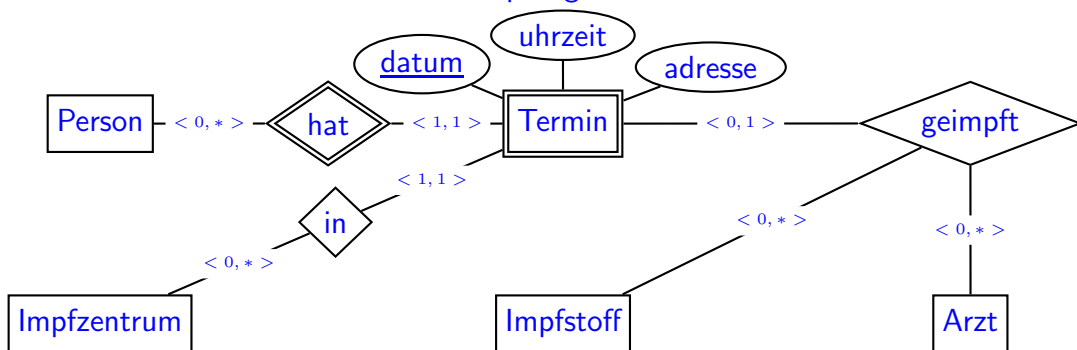
Unabhängig von der Modellierung ist die daraus abgeleitete Tabelle im Relationalen Modell (Aufgabe 3) dieselbe.



Ein Entwurf, der "Termin" und "Impfung" in zwei Entitätstypen trennt, erlaubt eine sehr exakte Modellierung bzgl. der (Mindest)kardinalitäten:



Etwas reduziert kann man auch hier die "Impfung" de-reifizieren:



Aufgabe 2 (ER-Modell (ASCII-Art-Modus) [0 Punkte])

Alternativ zu Aufgabe 1 können Sie dasselbe hier als ASCII-Art im Editor bearbeiten. So etwa so:

```
[Entity]---<0,1>---<<ident-beziehung>>---<1,1>---[[weakEnt]]-(attr)
  \                                     |
  (_key_attr)                         (_keyattr_)
```

Aufgabe 3 (Transformation in das Rel. Modell [4 Punkte])

Vervollständigen Sie in dieser Aufgabe das unten angegebene relationale Modell:

Geben Sie die Tabelle(n) (mit Attributen, Schlüsseln, Fremdschlüsseln etc.) für die Impfungen mit jeweils mindestens zwei Beispieldatensätzen (z.B. welche, die sich aus dem Aufgabentext ergeben) an.

Geben Sie hier kein SQL CREATE TABLE-Statement, sondern entsprechend "ASCII-Art" Pseudocode an. (Empfehlung: editieren Sie es in einer lokalen Datei und kopieren es am Ende ins Ilias, dann können Sie Ihre lokale Datei zur Bearbeitung der SQL-Aufgaben auch sehen).

- Die Tabellenskizze kann z.B. so aussehen:

```
tablename(_attr1_,_attr2_,attr3, attr4)
-----
      bsp11   bsp12  bsp13  bsp14
      bsp21   bsp22  bsp23  bsp24
```

oder analog

```
relname(attr1,attr2,attr3,attr4) Primary Key: (attr1,attr2)
```

- Angabe von Fremdschlüsselreferenzen z.B. so:

```
rel1(A,B) -> rel2(X,Y)
```

Person							
id	Name	GebDatum	Adresse	Stadt	PLZ	Tel.	Arzt
EM	Elsa Müller	4.3.1943	Habichtsweg 55	Göttingen	37075	...	0
FM	Fritz Müller	3.4.1934	Habichtsweg 55	Göttingen	37075	...	0
MS	Markus Schmidt	7.7.1977	Kreuzberggring 1	Göttingen	37075	...	0
RK	Robert Koch	11.12.1983	Robert-Koch-Str. 1	Göttingen	37073	...	1
JF	Johann Faustus	29.2.1980	Goethestrasse 42	Göttingen	37073	...	1
:	:	:	:	:	:	:	:

Impfstoff					
Name	Hersteller	Wirksamkeit	Zulassungsdatum	AlterVon	AlterBis
BNT162B2	BionTech/Pfizer	95	21.12.2020	16	NULL
mRNA-1273	Moderna	95	06.01.2021	18	NULL
AZD1222	Astra-Zeneca	NULL	29.01.2021	18	64
Sputnik V	Gamaleja	91	NULL	NULL	NULL
:	:	:	:	:	:

Kreis		
Name	Bevölkerung	Bundesland
Göttingen	330000	Niedersachsen
Northeim	130000	Niedersachsen
:	:	:

geliefert (Keys/Foreign Keys fehlen!)			
Impfstoff	Kreis	Datum	Anzahl
BNT162B2	Göttingen	23.12.2020	380
BNT162B2	Northeim	23.12.2020	190
BNT162B2	Göttingen	4.2.2021	380
AZD1222	Göttingen	10.2.2021	200
:	:	:	:

Impfzentrum					
Name	Strasse	PLZ	Stadt	Kreis	
Göttingen I	Godehardstraße 11	37081	Göttingen	Göttingen	
Göttingen II	Anna-Vandenhoeck-Ring 13	37081	Göttingen	Göttingen	
UMG Göttingen	Robert-Koch-Straße 40	37075	Göttingen	Göttingen	
Northeim	Wolfshof 10	37154	Northeim	Northeim	
:	:	:	:	:	

Nebenwirkungen (Keys fehlen!)			
Person	Datum	Beschreibung	Gefährlichkeit
FM	01.01.2021	Kopfschmerzen	1
MS	15.02.2021	Übelkeit	1
MS	16.02.2021	Übelkeit	1
MS	16.02.2021	Fieber 39 Grad	3
MS	16.02.2021	Schttttttelfrost	3
:	:	:	:

Lösung

Impfung								
Person	Datum	Uhrzeit	Impfzentrum	Impfstoff	Arzt	Adresse	Stadt	PLZ
EM	28.12.2020	14:30	Göttingen I	BNT162B2	JF	Habichtsweg 55	Göttingen	37075
FM	28.12.2020	14:40	Göttingen I	BNT162B2	JF	Habichtsweg 55	Göttingen	37075
EM	8.2.2021	10:00	Göttingen I	BNT162B2	RK	Habichtsweg 55	Göttingen	37075
FM	8.2.2021	10:10	Göttingen I	BNT162B2	RK	Habichtsweg 55	Göttingen	37075
MS	15.2.2021	9:30	Göttingen I	BNT162B2	JF	NULL	NULL	NULL
MS	10.3.2021	10:00	Göttingen I	NULL	NULL	NULL	NULL	NULL
:	:	:	:	:	:	:	:	:

FKs: (Person)→Person(id), (Arzt)→Person(id),

(Impfzentrum)→Impfzentrum(Name), (Impfstoff)→Impfstoff(Name)

Key: 1P 4FKs je 1/2P Rest/richtige Attribute 1P

Anmerkungen: hier waren nur die Tabellen für die Impfungen gefragt.

Die in der Aufgabenstellung nicht angegebenen Schlüsselattribute für "geliefert" werden ja gleich in der nächsten Aufgabe behandelt.

Aufgabe 4 (Relationales Modell: CREATE TABLE [4 Punkte])

Geben Sie das CREATE TABLE-Statement für die oben angegebene "geliefert"-Tabelle so vollständig wie möglich an, d.h., bestimmen Sie auch die Schlüssel und Fremdschlüssel.

Lösung

```
CREATE TABLE geliefert                                     1P
(Impfstoff VARCHAR2(20) REFERENCES Impfstoff(name),      1/2 P
 Kreis VARCHAR2(20) REFERENCES Kreis(name),              1/2
 Datum DATE CHECK Datum BETWEEN 01.12.2020 AND SYSDATE, +1/2 wer dran denkt
 Anzahl NUMBER NOT NULL CHECK (Anzahl > 0),              1/2+1/2
 PRIMARY KEY (Impfstoff, Kreis, Datum))                   1 P
```

Wenn man mit der Begründung "es könnten ja an einem Tag mehrere Lieferungen desselben Impfstoffs an ein Impfzentrum geliefert werden" auch die Anzahl mit in dem PRIMARY KEY nimmt, bringt das eher wenig. Die Anzahl wird meistens die in einem oder zwei Kartons enthaltene Menge sein, also auch oft dieselbe. Dann müsste man eher noch die Uhrzeit (aber welche?) dazunehmen.

Aus der Aufgabenstellung geht hervor, dass jeweils die Zahl für den gesamten Tag gemeint ist.

Aufgabe 5 (Relationales Datenmodell: Nullwerte [2 Punkte])

Was bedeutet der Wert NULL in der 3. Spalte der Tabelle "Impfstoffe" für den Astra-Zeneca-Impfstoff?

Lösung Der dort genau hingehörende Wert für Wirksamkeit existiert (und ist ausreichend hoch, sonst wäre es nicht bekannt), aber es kann/soll kein konkreter Wert angegeben werden (es gab verschiedene Studien, die u.a. auf 62%, 70%, 82% und 90% kamen).

Die Antwort "da ist kein Wert eingetragen" ist zu platt. Es geht darum, was dieser Nullwert bedeutet: "gibt es, aber nicht eingetragen", z.B. im Gegensatz zu den Nullwerten in der 4. und 6. Spalte, "Wert existiert nicht, bedeutet bisher nicht zugelassen", "es gibt keinen Wert (Altersobergrenze)".

Aufgabe 6 (Anfragen (1) [6 Punkte])

Verwenden Sie für diese und die folgenden Aufgaben die von Ihnen vervollständigte relationale Datenbasis. Keine der Antworten soll Duplikate enthalten.

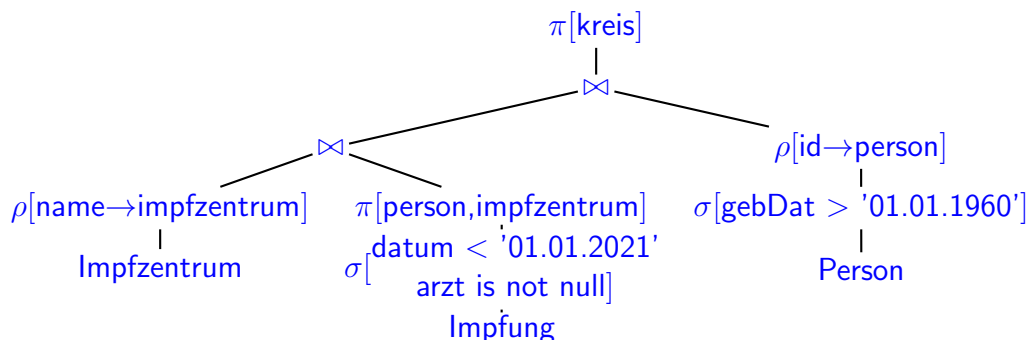
Geben Sie eine SQL-Anfrage *und* einen Ausdruck oder Baum der relationalen Algebra an, die die Namen aller Landkreise zurückgibt, in denen vor dem 1.1.2021 mindestens eine Person, die nach dem 1.1.1960 geboren wurde, geimpft wurde.

Sie können wahlweise beides zusammen im untenstehenden Texteingabefeld eingeben, oder den Algebra-Baum in Aufgabe 11 als Grafik oder Foto hochladen.

Falls Sie den Algebra-Ausdruck/Baum hier eingeben wollen, schreiben Sie pseudocode mit $\pi\dots$, $\sigma\dots$, $\text{join}(\dots, \dots)$, etc.

Lösung

```
SELECT DISTINCT kreis
FROM Person p, Impfung i, Impfzentrum j
WHERE p.id = i.person
      AND i.impfzentrum = j.name
      AND i.Datum < '01.01.2021'
      AND p.gebDat > '01.01.1960'
      AND arzt IS NOT NULL          (+1/2P wer daran dachte)
```



(Aufpassen: “Person” und “Impfzentrum” haben beide eine Spalte “Stadt”, die darf beim natürlichen Join nicht dabei sein; Impfung.Adresse ist meistens dieselbe wie Person.Adresse, oder null, das ist nur ein Problem, wenn die Person inzwischen umgezogen ist.)

```
pi[Kreis](
  natjoin(
    natjoin(rho[id->person](sigma[gebDat > '01.01.1960'](person)),
      pi[person,impfzentrum](
        sigma[Datum < '01.01.2021'](Impfung))
    rho[name->impfzentrum](Impfzentrum)))
```

Aufgabe 7 (Anfragen (2) [4 Punkte])

Geben Sie eine SQL-Anfrage an, die für jeden Kreis angibt, wieviele Impfungen bisher von den Impfzentren des Kreises durchgeführt wurden.

Lösung

```

SELECT Kreis, COUNT(*)
FROM Impfung i, Impfzentrum j
WHERE i.impfzentrum = j.name
      AND Arzt IS NOT NULL      -- Impfung wurde durchgefuehrt
      -- DATUM < SYSDATE kann eine nicht erledigte abgesagte Karteileiche sein!
GROUP BY Kreis

```

(wenn jemand daran dachte, (kreis,0) dazuzunehmen, wenn dort noch nicht geimpft wurde +1)

Aufgabe 8 (Anfragen (3) [4 Punkte])

Geben Sie eine SQL-Anfrage an, die die Namen aller Impfzentren ausgibt, von denen noch keine Person mit dem Impfstoff "mRNA-1273" geimpft wurde.

Lösung

```

SELECT name
FROM Impfzentrum i
WHERE NOT EXISTS
  (SELECT *
   FROM Impfung j
   WHERE j.Impfzentrum = i.name
        AND Impfstoff = 'mRNA-1273')

```

(Der Aspekt "schon geimpft" wurde hier nicht nochmal bewertet, da er beim GROUP BY schon explizit dabei war, und die Aufgabenstellung davon ausgeht, dass Impfstoff nur ungleich NULL ist, wenn schon geimpft wurde)

Aufgabe 9 (Anfragen (4) [5 Punkte])

Geben Sie einen Algebra-Ausdruck oder Baum an, der die Anfrage aus der vorhergehenden Aufgabe beantwortet.

Sie können den Baum wahlweise hier (als ASCII-Art im Editor) abgeben, oder in Aufgabe 11 als Grafik oder Foto hochladen.

Lösung

```

              minus
            /      \
rho[name->impfzentrum] pi[impfzentrum]
 pi[name]              sigma[Impfstoff = 'mRNA-1273']
 Impfzentrum           Impfung

```

Aufgabe 10 (Anfragen (5) [8 Punkte])

Welche der folgenden Algebra-Bäume und SQL-Anfragen beantworten die Anfrage "die Namen aller Kreise, an die alle bisher zugelassenen Impfstoffe mindestens einmal geliefert wurden" korrekt?

(es können mehrere Bäume und SQL-Anfragen richtig sein)

1.

```

SELECT name
FROM kreis k
WHERE ALL
  (SELECT impfstoff FROM Impfstoff i
   WHERE zugelassen IS NOT NULL)
  IN (SELECT impfstoff, kreis FROM geliefert)

```

2.

```

SELECT name
FROM kreis k
WHERE NOT EXISTS
  (SELECT *
   FROM Impstoff i
   WHERE zugelassen IS NOT NULL
   AND (i.name, k.name) NOT IN
        (SELECT impfstoff, kreis FROM geliefert))

```

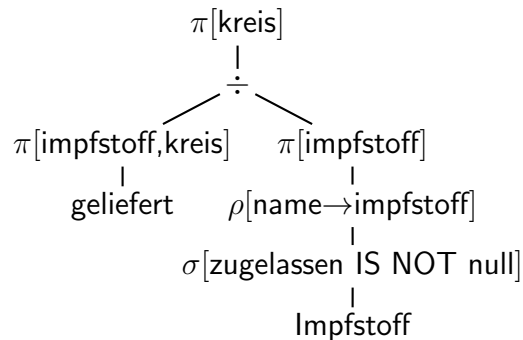
3.

```

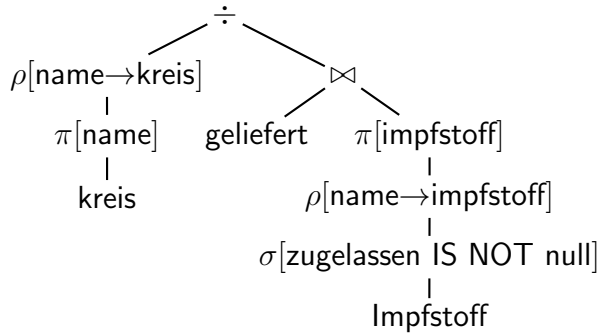
SELECT name
FROM kreis k
WHERE name NOT IN
  (SELECT k.name
   FROM ((SELECT impfstoff, kreis
          FROM geliefert)
        MINUS
        (SELECT i.name, k.name
         FROM impfstoff i, kreis k
         WHERE i.zugelassen IS NOT NULL)))

```

4.



5.



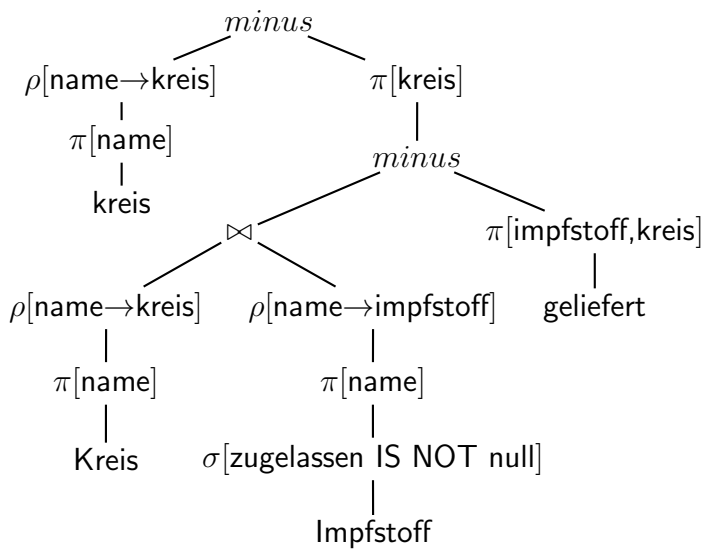
6.

```

SELECT name
FROM kreis
WHERE EXISTS
  (SELECT *
   FROM impfstoff i, kreis k
   WHERE i.zugelassen IS NOT NULL)
AND name NOT IN
  (SELECT * FROM geliefert WHERE impfstoff != i)

```

7.



8.

```

SELECT name
FROM kreis
WHERE name NOT IN
  (SELECT kreis
   FROM ((SELECT i.name as impfstoff, k.name as kreis
          FROM impfstoff i, kreis k
          WHERE i.zugelassen IS NOT NULL)
        MINUS

```

```
(SELECT impfstoff, kreis
FROM geliefert))
```

Lösung

1. Falsch: das keyword "ALL" in SQL kann man nicht so verwenden. Selbst wenn es (ähnlich wie "every" in XQuery) in diesem Stil erlaubt wäre, würde (impfstoff) IN (impfstoff, kreis) nicht passen.
Hinweis: wenn dieses ALL so erlaubt wäre, wäre es in der Vorlesung sicher drangekommen.
2. Richtig.
3. Das Minus ist falsch herum: Es ergäbe Kreise, in die nicht-zugelassene Impfstoffe geliefert wurden. Durch das "NOT IN" außen würde die Anfrage also (hoffentlich) alle Kreise ergeben.
4. Richtig. Die äußere Projektion ist aber redundant.
5. Falsch. Der linke Ast einer Division darf nicht einspaltig sein, und der rechte darf nicht mehr Attribute als der linke haben.
6. Falsch. Die Anfrage ist auch syntaktisch kaputt. Das "impfstoff != i" hat erstens keine Komponente (z.B. "i.name?"), und könnte auf die Aliasvariable i nicht zugreifen, da sie nicht in einer äußeren Anfrage definiert wird.
7. Richtig. Das ist der zur Definition der Division äquivalente Algebra-Ausdruck.
8. Richtig. Das ist die SQL-Anfrage, die den zur Definition der Division äquivalenten Algebra-Ausdruck darstellt.

Bewertung (in Ilias automatisch): +1 Punkt für jedes richtige Kreuz, -1 für jedes falsche. Die Empfehlung von E-Learning/E-Klausuren für Ilias ist allgemein, mehr "geschlossene Fragen", also Fragen die eine automatische Korrektur erlauben, zu verwenden.

Aufgabe 11 (Anfragen (wahlweise: Bäume als Upload) [0 Punkte])

Laden Sie hier wahlweise die Algebra-Bäume für Aufgabe 6 ("Anfragen (1)") und Aufgabe 9 ("Anfragen (4)") als Grafik/Fotos hoch (beide in einer Datei oder einzeln).

Aufgabe 12 (Open Book – Open Web – Open Documentation [4 Punkte])

Ein externer Arzt, Dr. Sommer (bereits mit Benutzername `drsommer` angelegt), soll die Tabellen, in denen die Impfungen und die Nebenwirkungen gespeichert sind, lesen können. Er soll somit keinen Zugriff auf personenbezogene Daten haben.

Wie lautet der SQL-Befehl/die SQL-Befehle, mit dem der Datenbankadministrator Herrn Dr. Sommer die entsprechenden Zugriffsrechte geben kann, um die Tabellen lesen, aber nicht ändern zu können?

Benutzen sie eine Suchmaschine Ihrer Wahl und Dokumentation im Web (Hinweis: in den Folien der Datenbankvorlesung ist es nicht enthalten).

Die Syntax ist für alle Datenbanksysteme (Oracle, Postgres, ...) dieselbe.

Lösung

```
GRANT SELECT ON Impfungen TO drsommer;
GRANT SELECT ON Nebenwirkungen TO drsommer;
```

Nicht "GRANT READ", und auch nicht in einem gemeinsamen Statement möglich.
In einigen Lösungen wurde ein Join-View

```
CREATE VIEW as drsommersview
SELECT *
FROM Impfungen i, Komplikationen k
WHERE i.person = k.person
```

erzeugt. Das ist grundsätzlich keine schlechte Idee, löst aber das Problem nicht: Dr. Sommer könnte dann ggf. das View *und* alle Tabellen lesen; das Problem der Zugriffsrechte bliebe offen, und müsste durch

```
GRANT SELECT ON drsommersview TO drsommer;
```

gelöst werden. Das Inner-Join View hätte außerdem dem Nachteil, dass Impfungen, zu deren Personen keine Nebenwirkungen gespeichert sind, darin nicht enthalten wären – siehe nächste Aufgabe (das war so nicht beabsichtigt, passt aber sehr gut).

Aufgabe 13 (Eine etwas kompliziertere SQL-Anfrage [6 Punkte])

Welche Anfrage muss Dr. Sommer (siehe auch vorhergehende Aufgabe) stellen, um für jeden Impfstoff herauszufinden, wieviel Prozent der Geimpften über Nebenwirkungen mit Gefährlichkeitsstufe 3 oder höher berichtet haben?

Lösung

```
SELECT impfstoff, count(distinct k.person)/count(distinct i.person) * 100
FROM Impfungen i
     LEFT OUTER JOIN
     Komplikationen k
WHERE i.person = k.person and k.gefaehrlichkeit >2
GROUP BY impfstoff
```

-- man kann die Nebenwirkungen auch in einer
-- korrelierten select-subquery zusammensuchen:

```
SELECT impfstoff, (SELECT count(distinct i2.person)
                   FROM Impfungen i2, Nebenwirkungen n
                   WHERE i2.person=n.person
                   AND i2.impfstoff = i.impfstoff)
                /count(distinct i.person) * 100 -- alle damit geimpften
FROM Impfungen i
GROUP BY impfstoff
```

-- wenn man die Nebenwirkungen "einfach" in einem zweiten FROM berechnet,
-- muss man puzzeln, um nicht diejenigen Impfstoffe zu verlieren,
-- bei denen keine schwerwiegenden Nebenwirkungen registriert wurden:

```
SELECT alle.impfstoff, komp.anzahl/alle.anzahl * 100
```

```

FROM
  (SELECT count(distinct i.person) as anzahl, impfstoff
   FROM Impfungen
   GROUP BY impfstoff
  ) alle,
  ((SELECT impfstoff, count(distinct i.person) as anzahl
   FROM Impfungen i, Komplikationen k
   WHERE i.person = k.person and k.gefaehrlichkeit >2
   GROUP BY impfstoff)
  UNION
  (SELECT impfstoff, 0
   FROM impfstoff
   WHERE impfstoff NOT IN (SELECT impfstoff
                           FROM Impfungen i, Komplikationen k
                           WHERE i.person = k.person and k.gefaehrlichkeit >2))
  ) komp,
WHERE komp.impfstoff = alle.impfstoff

```

Aufgabe 14 (Geht das auch in der Algebra? [2 Punkte])

Diese Frage kann man durchaus auch beantworten, falls man die SQL-Anfrage in der vorhergehenden Aufgabe nicht angeben konnte: Gibt es einen Ausdruck der relationalen Algebra, der die SQL-Anfrage aus der vorhergehenden Aufgabe ausdrückt? (mit Begründung)

Lösung Man muss gruppieren und aggregieren (zählen). Diese Operationen sind in der relationalen Algebra nicht ausdrückbar.

(Ausserdem muss man mit Zahlen rechnen, d.h. dividieren und multiplizieren, auch das ist in der relationalen Algebra nicht möglich).

Aufgabe 15 (Transaktionen 1: Single Choice [1 Punkte])

Welche der folgenden Abkürzungen steht für die wesentlichen Eigenschaften von Transaktionen in der Informatik?

- AHA+L
- ACID
- mRNA
- GWG

Lösung

- ACID: Atomicity, Consistency, Isolation, Durability steht für die Eigenschaften von Transaktionen.
- AHA+L: Abstand halten, Hygiene, Alltagsmasken + Lüften steht für die aktuellen Corona-Verhaltensregeln (naja, hat an sich immerhin mit Multiuser-Betrieb und virtueller Isolation zu tun)

- GWG: Geldwäschegesetz. In dessen Paragraph 46 geht es um Finanztransaktionen. (GWG steht auch für *Gesellschaft für Wirtschaftsförderung und Stadtentwicklung Göttingen*, aber das ist hier auch nicht gefragt).
- mRNA steht für *messenger ribonucleic acid*, eine einzelsträngige Ribonukleinsäure, die genetische Information für den Aufbau eines Proteins trägt und eines der Wirkungsverfahren von Corona-Impfungen ist. ACID kommt dabei immerhin auch vor.

Aufgabe 16 (Transaktionen 2: Transaktionalität in ILIAS [4 Punkte])

Beschreiben Sie (kurz) für jede der durch das in der vorhergehenden Aufgabe gefundene Akronym beschriebenen Eigenschaften von Transaktionen, ob und wie sie *für die Klausurbearbeitung durch die Studierenden* bei einem Online-Klausursystem wie etwa dem ILIAS, das sie gerade verwenden, relevant ist.

Lösung Wichtig ist bei der Aufgabe, zwischen der Anwendungssemantik (von der das darunterliegende Datenhaltungssystem nichts weiss) und eben dem Datenhaltungssystem zu unterscheiden. In der Anwendungssemantik "Klausur" besteht eine "Isolation" zwischen den Teilnehmern, aber genau diese (jeder Teilnehmer arbeitet ausschliesslich auf der eigenen Klausur) hat zur Folge, dass transaktionale virtuelle Isolation (von gleichzeitigen Zugriffen auf dasselbe Datenitem) keine Rolle spielt, weil es diese in der Anwendung nicht geben kann.

- Atomicity: Jein. Jede einzelne Aktion "Speichern" ist an sich sowieso schon ein atomares INSERT oder UPDATE. Man kann argumentieren "der ganze eingegebene Text/Grafik oder garnichts soll gespeichert werden", bzw. dass bei einer Multiple-Choice-Aufgabe alle "Teile" der Abgabe gespeichert werden.
Hinweis: Bezüglich der Gesamtklausur hat man es nicht mit einer "lange andauernden Transaktion" zu tun, sondern einer Interaktion, die aus einer Folge einzelner Transaktionen besteht.
- Consistency: Jein. Es gibt nur wenige echte Integritätsbedingungen. Das System nimmt auch Schrotteingaben (z.B. bei den SQL-Aufgaben) an.
Es nimmt aber leider auch Grafik-Uploads bei Quellcode-Aufgaben an, die dann nie wieder zu dekodieren sind. Das wäre ein Consistency-Aspekt, der eigentlich einen Fehler (=Ablehnung der Transaktion "Upload" auslösen müsste).
Die maximal zulässige Größe einer Datei ist eine (willkürliche) Integritätsbedingung.
Rein bezüglich der Datenbank unter Ilias kann man argumentieren, dass eine von der ILIAS-Anwendungsebene übermittelte Speicherung sich auf eine Frage (mit einer Frage-ID) beziehen muss, die es auch in der DB gibt (würde Programmierfehler abfangen).
Punktezuordnung je nach Argumentation.
- Isolation: Systemseitig nicht relevant: jeder Benutzer arbeitet ja bei dieser Anwendung nur auf den "eigenen" Datenitems. Es können sich damit keine zwei Transaktionen "ins Handwerk pfuschen".
Damit muss eine (virtuelle) Isolation von Transaktionen während der Klausur nicht zusätzlich durch eine Transaktionsverwaltung gewährleistet werden.
Das Arbeiten in zwei Browser-Tabs ist hier auch nicht relevant (Zwei Aufgaben → verschiedene Datenitems, und selbst wenn man in beiden Tabs dieselbe Aufgabe bearbeiten würde, könnte man nicht in beiden im selben Moment eine Transaktion auslösen).
(Bei der Korrektur mit Ilias sollte man allerdings sehr vorsichtig sein, was alles gespeichert wird, wenn man aufgabenweise oder teilnehmerweise korrigiert ... schon ein unvorsichtiges "reload" kann Korrekturanmerkungen löschen.)

- Durability: absolut wichtig. Jede Eingabe sollte bereits sofort abgespeichert sein, falls z.B. die Software/Verbindung des Teilnehmers zusammenbricht oder die ILIAS-Webservice-Software. Ausserdem sollte sie nach "Speichern" sofort dauerhaft (also am besten extern redundant) abgelegt sein (falls der ILIAS-Server abstürzt), oder (z.B. nach einigen Tagen) die Festplatte stirbt.