

Klausur Datenbanken
Wintersemester 2022/2023
Prof. Dr. Wolfgang May
23. Februar 2023, 10:30-12:xx Uhr
Bearbeitungszeit (Papier): 90 Minuten (Ilias: 100 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 und 4 (wahlweise, Umsetzung in das relationale Modell) müssen nur noch die fehlenden Teile ergänzt werden.
- Bearbeiten Sie zuerst *entweder* Aufgabe 1 *oder* 2 (ER-Diagramm), dann *entweder* Aufgabe 3 *oder* 4 (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))	18	15
Aufgabe 2 (ER-Modell (ASCII-Art-Modus))	0	0
Aufgabe 3 (Transformation in das Rel. Modell (Upload))	14	9
Aufgabe 4 (Transformation in das Rel. Modell (ASCII))	0	0
Aufgabe 5 (Relationales Modell: CREATE TABLE)	5	3
Aufgabe 6 (Anfragen (1))	6	4
Aufgabe 7 (Anfragen (2))	4	3
Aufgabe 8 (Anfragen (3))	4	1
Aufgabe 9 (Anfragen (4))	4	3
Aufgabe 10 (Anfragen (5))	6	1
Aufgabe 11 (Anfragen (6))	5	1
Aufgabe 12 (Anfragen (wahlweise: Bäume als Upload))	0	0
Aufgabe 13 (Update an der Datenbank)	5	4
Aufgabe 14 (Eine etwas kompliziertere SQL-Anfrage)	6	2
Aufgabe 15 (Der Algebra-Ausdruck dazu)	6	2
Aufgabe 16 (Etwas Theorie)	7	1
Aufgabe 17 (Anfragen (XXXX))	0	0
Aufgabe 18 (Anfragen (XXXX))	0	0
Aufgabe 19 (Anfragen (XXXX))	0	0
Aufgabe 20 (Anfragen (XXXX))	0	0
Aufgabe 21 (Anfragen (XXXX))	0	0
Aufgabe 22 (Anfragen (XXXX))	0	0
Aufgabe 23 (Anfragen (XXXX))	0	0
Aufgabe 24 (Eine etwas kompliziertere SQL-Anfrage)	0	0
Aufgabe 25 (Entwurf)	0	0
Aufgabe 26 (Was tut dieser Algebra-Ausdruck?)	0	0
Summe	90	49

Note:

Themenstellung: EU-Lobby-/Transparenzregister

Alle Klausuraufgaben basieren auf einem gemeinsamen “Auftrag”: In der Klausur soll eine *Transparency-Datenbank* für die Erfassung der Kontakte von *EU-Politikern* mit *Interessenvertretern* aus Wirtschaft, Non-Governmental Organizations (NGOs), und anderen Staaten entwickelt werden.¹

Die Idee zur der Aufgabe stammt aus Medienberichten vom Dezember 2022, deren Hauptpersonen in der Aufgabe vorkommen.

Im ER-Diagramm und im relationalen Modell bereits vorgegebener Teil:

1. Für jede *Person* ist der Name (Annahme: Namen sind eindeutig) gespeichert, und aus welchem Land die Person kommt.
2. Für *EU-Politiker* ist gespeichert, welche *Position* in der EU sie haben, und welcher *Europaparteienfamilie* sie angehören (dies sind nicht die Parteien der einzelnen Länder, wie CDU, SPD, sondern europaweite Zusammenschlüsse der nationalen Parteien).

Dieser Begriff beinhaltet alle EU-Funktionäre, -Abgeordnete und Mitarbeiter (in der Klausur werden die englischen Begriffe dafür verwendet, die gender-unabhängig sind und damit ggf. Anfragen erleichtern).

Ursula von der Leyen ist *President of the Commission*, sie ist aus Deutschland, und gehört der “EVP”-Parteienfamilie an. *Charles Michel* ist *President of the EU-Concil*, er ist aus Belgien, und gehört der “ALDE”-Parteienfamilie an. *Paolo Gentiloni* ist *EU-Commissioner* (für Wirtschaft; EU-Commissioner sind das, was in einem normalen Staat die Minister sind;), er ist aus Italien, und gehört in der EU der “S&D”-Parteienfamilie an. *Margaritis Schinas* ist *EU-Commissioner* (für “Förderung des europäischen Lebensstils”), er ist aus Griechenland, und gehört der “EVP”-Parteienfamilie an. *Eva Kaili* ist *Member of Parliament (MEP)*, sie ist aus Griechenland, und gehört der “S&D”-Parteienfamilie an. *Francesco Giorgi* ist *Employee* der “S&D”-Fraktion, er ist aus Italien, und gehört der “S&D”-Parteienfamilie an (und er ist der Lebensgefährte von *Eva Kaili*, aber das ist nicht in der Datenbank gespeichert).

3. Für *Interessenvertreter* ist gespeichert, welche *Organisation* sie vertreten, und welche Position sie dort innehaben. Solche *Organisationen* sind z.B. Non-Governmental Organizations (NGOs), Firmen, oder Staaten:

Recep Erdogan ist aus der Türkei und ist Präsident der Türkei. *Volodymyr Selenskyj* ist aus der Ukraine und ist Präsident der Ukraine. *Tamim Al Thani* ist aus Qatar und ist dort Emir.

Gianni Infantino ist aus Italien und ist Präsident des *Weltfußballverbandes FIFA*. *Pier Antonio Panzeri* ist aus Italien und ist Direktor der NGO *Fight Impunity* (die er vor einiger Zeit zusammen mit *Francesco Giorgi* gegründet hat, was aber auch in der Datenbank nicht gespeichert ist).

¹eine solche existiert schon, beinhaltet aber bisher keine Kontakte zu ausländischen *Politikern*, siehe https://commission.europa.eu/about-european-commission/service-standards-and-principles/transparency/transparency-register_de

In der Klausur soll das ER-Diagramm und das Relationale Modell um die folgenden Daten erweitert werden:

4. Die *Länder* (es sind alle Länder, also auch Nicht-EU-Länder, gespeichert): Länder haben einen eindeutigen Landescode und einen Namen, außerdem ist für jedes Land der Mitgliedschaftsstatus in der EU und die Hauptstadt (siehe auch den nächsten Punkt) gespeichert.
5. *Städte* haben einen Namen, und liegen in einem Land. Es können mehrere Städte weltweit denselben Namen haben, aber –etwas vereinfacht– innerhalb eines Landes kommt jeder Stadtname höchstens einmal vor.

Belgien hat den Landescode “B”, die Hauptstadt Brüssel, und ist Mitglied der EU. Die Türkei (Landescode “TR”, Hauptstadt Ankara) ist EU-Beitrittskandidat. Die Ukraine (Landescode “UA”, Hauptstadt Kiew), die Schweiz (Landescode “CH”, Hauptstadt Bern) und Qatar (Landescode “Q”, Hauptstadt Doha) haben keine Form von Mitgliedschaft in der EU.

Zürich ist eine weitere Stadt in der Schweiz.

6. Im Zentrum des Transparenzregisters steht, welche EU-Politiker sich wann und wo mit welchen Interessenvertretern getroffen haben, und welche Themen dabei besprochen wurden. Diese Daten werden immer erst eingetragen, wenn das Treffen beendet ist.
 - Jedes solche *Treffen* bekommt eine eindeutige ID (mit der es z.B. auch im EU-Terminmanager angelegt wird).
 - Zu jedem Treffen wird abgespeichert, an welchem Datum und in welcher Stadt (die nicht notwendigerweise in einem EU-Land liegen muss) und wo es stattgefunden hat (das kann eine Adresse sein, eine Firmenangabe, ein Hotel etc. – einfach eine Zeichenkette).
 - An jedem solchen Treffen können mehrere EU-Politiker und auch mehrere Interessenvertreter teilnehmen.

Beispieldaten zu den Treffen:

- Bei dem Treffen *cm987* haben sich *Charles Michel* und *Ursula von der Leyen* am 6.4.2021 in der Stadt *Ankara* (in der Türkei) im *Cankaya-Palast* mit *Recep Erdogan* getroffen². Themen waren Migration, Wirtschaft und Klimaschutz.
- Bei dem Treffen *ms246* haben sich *Margaritis Schinas*, *Gianni Infantino* und *Tamim Al Thani* am 31.3.2022 in *Zürich* in der Schweiz im *FIFA-Gebäude* getroffen und über Sport gesprochen.
- Bei dem Treffen *eva666* haben sich *Eva Kaili*, *Francesco Giorgi*, *Pier Antonio Panzeri* und *Tamim Al Thani* am 15.11.2022 in *Doha*, der Hauptstadt von Qatar, im *Emirspalast* getroffen und über Wirtschaft gesprochen.
- Bei dem Treffen *cm321* haben sich *Charles Michel*, *Ursula von der Leyen*, *Paolo Gentiloni*, *Margaritis Schinas* (und 13 weitere EU-Commissioner) am 3.2.2023 in *Kiew* (der Hauptstadt der Ukraine) im *Präsidentenamt der Ukraine* mit *Wolodymyr Selenskyj* (und weiteren) getroffen. Themen waren EU-Beitritt usw.

7. Es wird die Annahme gemacht, dass in der Datenbank *genau alle aktuell aktiven EU-Politiker* gespeichert sind, mit *allen Treffen, an denen sie jemals beteiligt waren*.

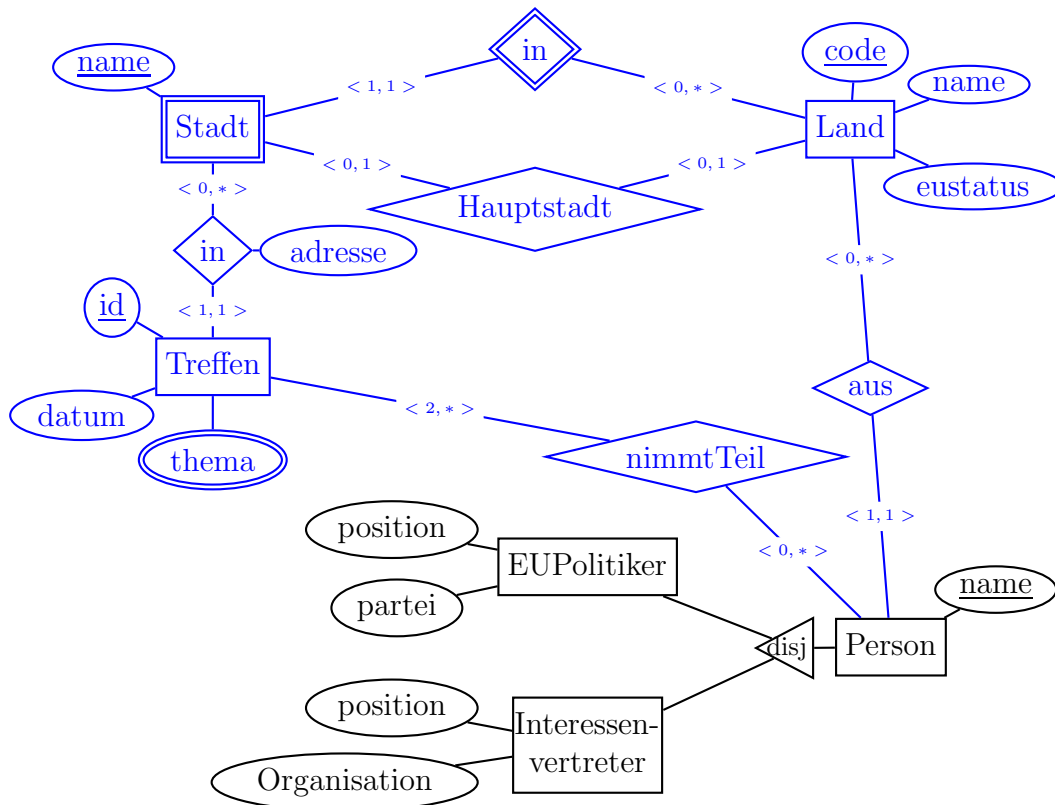
²das “Sofagate”-Event

Aufgabe 1 (ER-Modell (Foto oder PDF-Upload) [18 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.)

- 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.

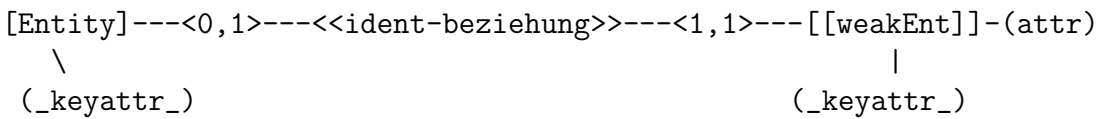
Alternativen und Hinweise:

- Man kann "nimmt Teil" auch als zwei Beziehungen zwischen "Treffen" und "EUPolitiker" bzw. "Vertreter" modellieren.
- "Adresse" als Attribut zu "treffen" anstatt zur "in"-Beziehung

- Kardinalitäten Land <1,*> bzw. <1,1> bzgl. "in" und "Hauptstadt" (mit 0-Kardinalitäten kann man Länder speichern, in denen keine Stadt für die DB relevant ist)
- Mit Mindestkardinalitäten 0 von Land bzgl. Städten und Hauptstadt muss man eine Stadt erst dann eintragen, wenn ein Treffen dort stattfindet. Länder muss man auch erst dann eintragen, wenn eine Person aus dem Land oder ein Treffen in einer Stadt in diesem Land gespeichert wird.
Entsprechende Mindestkardinalitäten von 1 würden bedeuten, dass zumindest alle Länder und Hauptstädte in der DB von Anfang an enthalten sind.

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:



Aufgabe 3 (Transformation in das Rel. Modell (Upload) [14 Punkte])

Vervollständigen Sie in dieser Aufgabe das relationale Modell. Gegeben sind die folgenden Relationen:

EUPolitiker			
Name	Land	Position	Partei
U.v.d.Leyen	D	Pres. of Commission	EVP
C.Michel	B	Pres. of Council	ALDE
P.Gentiloni	I	Commissioner	S&D
M.Schinas	GR	Commissioner	EVP
E.Kaili	GR	MEP	S&D
F.Giorgi	I	Employee	S&D
:	:	:	:

IntVertreter			
Name	Land	Organisation	Position
R.Erdogan	TR	null	Präsident
W.Selenskyj	UA	null	Präsident
T.Al Thani	Q	null	Emir
G.Infantino	I	FIFA	Präsident
P.A.Panzeri	I	FImpunity	Direktor
:	:	:	:

Geben Sie die noch fehlenden Tabellen (mit Attributen, Schlüsseln, Fremdschlüsseln etc.) für die Länder, Städte und die Informationen über die Treffen mit jeweils mindestens zwei Beispieltupeln (z.B. welche, die sich aus dem Aufgabentext ergeben) an (kein SQL CREATE TABLE-Statement, sondern grafisch als Tabellen).

Geben Sie die Fremdschlüsselreferenzen in der Form

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

an.

- Wenn Sie die Relationale-Modell-Aufgabe als pdf/png/jpg- oder Foto-Upload bearbeiten wollen, machen Sie dies HIER,
- wenn Sie die Aufgabe stattdessen als ASCII-Text im Editor bearbeiten wollen, machen Sie dies in Aufgabe 4.

Lösung

Land				Stadt	
<u>Code</u>	Name	EUStatus	Hauptstadt	<u>Name</u>	<u>Land</u>
B	Belgien	Mitglied	Brüssel	Brüssel	B
TR	Türkei	Kandidat	Ankara	Bern	CH
UA	Ukraine	null	Kiew	Zürich	CH
CH	Schweiz	null	Bern	Ankara	TR
Q	Qatar	null	Doha	Kiew	UA
:	:	:	:	:	:

Treffen					Themen	
<u>id</u>	Datum	Stadt	Land	Adresse	<u>id</u>	Thema
cm987	6.4.2021	Ankara	TR	Cankaya-Palast	cm987	Migration
ms246	31.3.2022	Zürich	CH	FIFA-Gebäude	cm987	Wirtschaft
eva666	15.11.2022	Doha	Q	Emirspalast	cm987	Klimaschutz
cm321	3.2.2023	Kiew	UA	Präsidentenamt	cm123	EU-Beitritt
:	:	:	:	:	ms246	Sport
					eva666	Wirtschaft
					:	:

TeilnEU		TeilnExt	
<u>id</u>	<u>teiln</u>	<u>id</u>	<u>teiln</u>
cm987	U.v.d.Leyen	cm987	R.Erdogan
cm987	C.Michel	cm321	W.Selenskyj
cm321	U.v.d.Leyen	ms246	T.Al Thani
cm321	C.Michel	ms246	G.Infantino
cm321	M.Schinas	eva666	P.A.Panzeri
ms246	M.Schinas	eva666	T.Al Thani
eva666	E.Kaili	:	:
eva666	F.Giorgi		
:	:		

FKs:

- Land.(Hauptstadt,Code) → Stadt(Name,Land) 1P
- Stadt.Land → Land.Name 1P
- Treffen.(Stadt,Land) → Stadt(Name,Land) 1P
- Themen.id → Treffen.id 1P
- TeilnEU.id → Treffen.id, TeilnEU.teiln → EUPolitiker.name 1+1P
- TeilnExt.id → Treffen.id, TeilnExt.teiln → IntVertreter.name (keine separaten Punkte)

Um die Speicherung der Teilnehmer durch Fremdschlüssel abzusichern muss man zwei separate Tabellen *TeilnEU* und *TeilnExt* mit unterschiedlichen Zielen der Fremdschlüsselreferenzen unterscheiden.

Wenn man stattdessen alle Teilnehmer in einer Tabelle abspeichern möchte, kann man eine Oberklassentabelle Person(Name, Land) anlegen (und Land aus EU-Politiker und EU-vertreter streichen), um einen Fremdschlüssel darauf definieren zu können.

(wegen den beiden nichttrivialen Aspekten daher "erwartete Punktzahl für 4" nur 9 Punkte)

Punkte:

Tabelle Land: 1P, PK 0.5, FK Hauptstadt 1 = 2.5

Tabelle Stadt: 1P, PK 0.5, FK Land 1 = 2.5

Tabelle Treffen: 1P, PK 0.5, FK Stadt 1 = 2.5

Tabellen TeilnEU+TeilnExt 2P, PK 0.5, 2x1P FKs = 4.5

Tabelle Themen: 0.5P, PK 0.5, FK id 1 = 2

bzw.

6 Tabellen: 5.5P ("Themen" zaehlt nur 0.5 P, s.u.)

5 PKs: 2.5P, (fuer TeilnEU und TeilExt nur 1x)

6 FKs: 6P siehe oben (jeweils 1/2 für FK und RefZiel) (fuer TeilnEU und TeilExt nur 1x)

Aufgabe 4 (Transformation in das Rel. Modell (ASCII) [0 Punkte])

Alternativ zu Aufgabe 3 können Sie dasselbe (Tabellen und Fremdschlüsselreferenzen) hier als Textfile eingeben

(Empfehlung: editieren Sie es in einer lokalen Datei und kopieren es dann ins Ilias, dann können Sie Ihre lokale Datei zur Bearbeitung der SQL-Aufgaben auch sehen).

- Die Tabellenskizze kann z.B. so aussehen:

```
tabname(_attr1_,_attr2_,attr3, attr4)
```

```
-----
```

```
      bsp11  bsp12  bsp13  bsp14
      bsp21  bsp22  bsp23  bsp24
```

oder analog mit

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

- Geben Sie die Fremdschlüsselreferenzen in der Form

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

an.

Aufgabe 5 (Relationales Modell: CREATE TABLE [5 Punkte])

Geben Sie das CREATE TABLE-Statement für Ihre Tabelle, in der abgespeichert wird, wo die einzelnen Treffen stattfinden, so vollständig wie möglich an.

Lösung

```
CREATE TABLE Treffen 1.5P Basis für das Dasein aller Spalten
```

```
(id VARCHAR2(10) PRIMARY KEY, 1/2 P
```

```
 datum DATE NOT NULL, 1/2 P eine CHECK-Bedingung ist hier nicht sinnvoll nutzbar
```

```
 Stadt VARCHAR2(50) NOT NULL, 1/2P
```

```
 Land VARCHAR2(4) NOT NULL, 1/2P
```

```
 Adresse VARCHAR2(100) NOT NULL, 1/2P
```

```
 FOREIGN KEY (Stadt, Land) REFERENCES Stadt(Name, Land) 1P
```

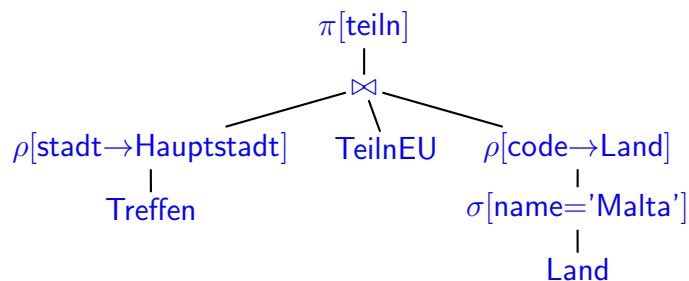
```
)
```


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

Geben Sie eine SQL-Anfrage *und* einen Ausdruck oder Baum der relationalen Algebra an, die die Namen derjenigen EU-Politiker ausgeben, die mindestens einmal an einem Treffen in der Hauptstadt des Landes *Malta* teilgenommen haben.
(Algebra-Baum entweder hier oder als Grafik in Aufg. 12)

Lösung

```
SELECT distinct e.teiln
FROM Treffen, Land, TeilnEU e
WHERE treffen.Land = Land.code
      AND treffen.Stadt = Land.Hauptstadt
      AND Land.name='Malta'
      AND Treffen.id = e.id
-- aufpassen, dass nur EU-Politiker ausgegeben werden
```



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

Geben Sie eine SQL-Anfrage an, die für jeden EU-Politiker ausgibt, wie oft er sich mit Politikern aus Qatar (Landescode "Q") getroffen hat. Es sollen nur Zeilen für Politiker ausgegeben werden, für die das mindestens einmal der Fall war.

Lösung

```
SELECT t1.teiln, COUNT(*)
FROM TeilnEU t1
WHERE EXISTS (SELECT *
              FROM TeilnExt t2, IntVertreter v
              WHERE t1.id = t2.id
                    AND t2.teiln = v.name
                    AND t2.Land = 'Q')
GROUP BY t1.teiln

SELECT t1.teiln, COUNT(DISTINCT t1.id)
FROM TeilnEU t1, TeilnExt t2, IntVertreter v
WHERE t1.id = t2.id
      AND t2.teiln = v.name
      AND t2.Land = 'Q'
      AND v.organisation is null -- sonst kein Politiker
GROUP BY t1.teiln
```

```
-- das EXIST oder DISTINCT Treffen.id ist wichtig,
-- falls bei einem Treffen mehrere Qataris dabei waren.
```

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

Geben Sie eine SQL-Anfrage (und in der *nächsten Aufgabe* einen Algebra-Ausdruck oder -Baum) an, die die Namen derjenigen EU-Politiker ausgibt, die noch nie an einem Treffen teilgenommen haben, das in der Ukraine (Landcode "UA") stattgefunden hat.

Lösung

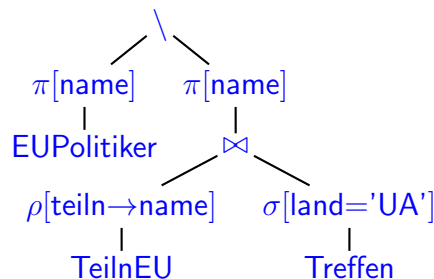
```
SELECT Name
FROM EUPolitiker p
WHERE NOT EXISTS
  (SELECT *
   FROM TeilnEU t1, Treffen t
   WHERE t1.teiln = p.name
        AND t1.id = t.id
        AND t.Land = 'UA')

(SELECT Name
 FROM EUPolitiker p)
MINUS
(SELECT t1.teiln
 FROM TeilnEU t1, Treffen t
 WHERE t1.id = t.id
        AND t.Land = 'UA')
```

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

Geben Sie einen Algebra-Ausdruck oder -Baum an, der die Anfrage aus der vorhergehenden Aufgabe beantwortet. Sie können den Ausdruck oder Baum wahlweise hier (als ASCII-Art im Editor) abgeben, oder in Aufgabe 12 als Grafik oder Foto hochladen.

Lösung



Aufgabe 10 (Anfragen (5) [6 Punkte])

Geben Sie eine SQL-Anfrage (und in der *nächsten Aufgabe* einen Algebra-Ausdruck oder -Baum) an, die die Namen derjenigen EU-Politiker ausgibt, die schon in allen Hauptstädten der EU-Mitgliedsländer an einem in der Datenbank gespeicherten Treffen teilgenommen haben.

Lösung

```
SELECT name
FROM EUPolitiker p
WHERE NOT EXISTS
  (SELECT *
   FROM Land l
   WHERE EUstatus = 'Mitglied'
        AND NOT EXISTS
          (SELECT *
           FROM EUPolitiker p
           WHERE NOT EXISTS
             (SELECT *
              FROM Land l
              WHERE EUstatus = 'Mitglied'
                    AND (p.name, l.code, l.hauptstadt))
```



```
-- aufpassen: da es Foreign Keys auf Treffen.id gibt, kann man nicht das Treffen
-- vor den referenzierenden Tupeln löschen. 1/2P
-- Wenn man Teilnehmer-Tupel löscht, muss man später zum Löschen der Treffen
-- die WHERE-Klausel (die ja die Teilnehmer verwendet!) anpassen 1/2P
```

```
-- hier zuerst mit der kompletten WHERE-Klausel:
```

```
DELETE FROM TeilnEU te1
WHERE EXISTS
    (SELECT * FROM TeilnEU te2 WHERE te2.id = te1.id AND te2.teiln='E.Kaili') 1P
AND EXISTS (SELECT *
    FROM TeilnExt te3, IntVertreter i
    WHERE te2.id = te1.id AND te3.teiln = i.name AND i.land = 'Q'); 2P
```

```
-- es sind ja nur Treffen gespeichert, die EU-Teilnehmer haben.
```

```
-- Die zu löschenden Treffen haben jetzt keine mehr:
```

```
DELETE FROM TeilnExt x
WHERE NOT EXISTS (SELECT * FROM TeilnEU t WHERE t.id = x.id);
DELETE FROM Themen x
WHERE NOT EXISTS (SELECT * FROM TeilnEU t WHERE t.id = x.id);
-- und dann diese Treffen löschen:
DELETE FROM Treffen x
WHERE NOT EXISTS (SELECT * FROM TeilnEU t WHERE t.id = x.id);
```

Alternativen:

- man definiert erst ein View:

```
CREATE VIEW todelete(id VARCHAR2(10)) AS
SELECT id FROM Treffen t
WHERE EXISTS (SELECT * FROM TeilnEU te1 WHERE te1.id = t.id AND te1.teiln='E.Kaili')
AND EXISTS (SELECT *
    FROM TeilnExt te2, IntVertreter i
    WHERE te2.id = t1.id AND te2.teiln = i.name AND i.land = 'Q');
DELETE FROM TeilnEU WHERE id IN (SELECT id FROM todelete);
DELETE FROM TeilnExt WHERE id IN (SELECT id FROM todelete);
DELETE FROM Themen WHERE id IN (SELECT id FROM todelete);
DELETE FROM Treffen WHERE id IN (SELECT id FROM todelete);
```

- Trick mit "markieren"

```
CREATE VIEW todelete(id VARCHAR2(10)) AS
UPDATE Treffen t
SET adresse = 'todelete' -- einen unkritischen Wert in nicht-PK/FK-Spalte
WHERE EXISTS (SELECT * FROM TeilnEU te1 WHERE te1.id = t.id AND te1.teiln='E.Kaili')
AND EXISTS (SELECT *
    FROM TeilnExt te2, IntVertreter i
    WHERE te2.id = t1.id AND te2.teiln = i.name AND i.land = 'Q');
DELETE FROM TeilnEU WHERE id IN (SELECT id FROM Treffen where adresse = 'todelete');
DELETE FROM TeilnExt WHERE id IN (SELECT id FROM Treffen where adresse = 'todelete');
DELETE FROM Themen WHERE id IN (SELECT id FROM Treffen where adresse = 'todelete');
DELETE FROM Treffen WHERE adresse = 'todelete';
```

- man argumentiert, dass die FKs mit ON DELETE CASCADE spezifiziert sind, und man nur die Treffen löschen muss.

Aufgabe 14 (Eine etwas kompliziertere SQL-Anfrage [6 Punkte])

Geben Sie eine SQL-Anfrage (und in der nächsten Aufgabe einen Algebra-Baum) an, der die Namen derjenigen EU-Politiker ausgibt, die in jeder Hauptstadt eines EU-Landes (Vollmitglieds) mit jedem der aktuellen EU-Commissioner irgendwann einmal gemeinsam auf einem Treffen waren (nicht notwendigerweise alle bei demselben Treffen in einer Hauptstadt).

Lösung (Diese Aufgabe korrespondiert zu der darauffolgenden Theorieaufgabe)

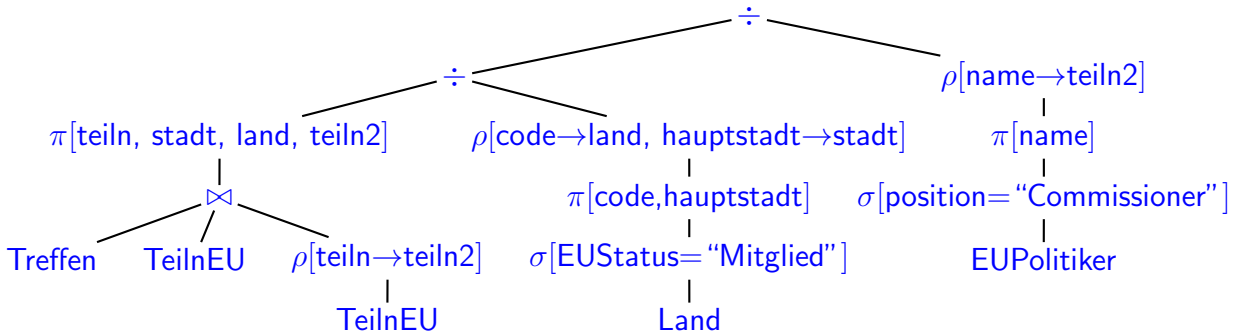
```
SELECT res.teiln
FROM TeilnEU res
WHERE NOT EXISTS      -- eine Hauptstadt ...
  (SELECT *
   FROM land l
   WHERE eustatus = 'Mitglied'
   AND EXISTS
     -- zu der es einen Commissioner gibt, mit dem sich res.teiln dort
     -- noch nicht getroffen hat
     (SELECT *
      FROM EUPolitiker comm
      WHERE position = 'Commissioner'
      AND NOT EXISTS
        (SELECT *
         FROM treffen t, teilnEU t1, teilnEU t2
         WHERE t.land = l.code
              AND t.stadt = l.hauptstadt
              AND t.id = t1.id
              AND t.id = t2.id
              AND t1.teiln = res.teiln
              AND t2.teiln = comm)))
```

```
SELECT res.teiln
FROM TeilnEU res
WHERE NOT EXISTS      -- eine Hauptstadt und ein Commissioner ...
  (SELECT *
   FROM land l, EUPolitiker comm
   WHERE l.eustatus = 'Mitglied'
   AND comm.position = 'Commissioner'
   -- ... die die Bedingung verletzen:
   AND NOT EXISTS
     (SELECT *
      FROM treffen t, teilnEU t1, teilnEU t2
      WHERE t.land = l.code
            AND t.stadt = l.hauptstadt
            AND t.id = t1.id
            AND t.id = t2.id
            AND t1.teiln = res.teiln
            AND t2.teiln = comm)))
```

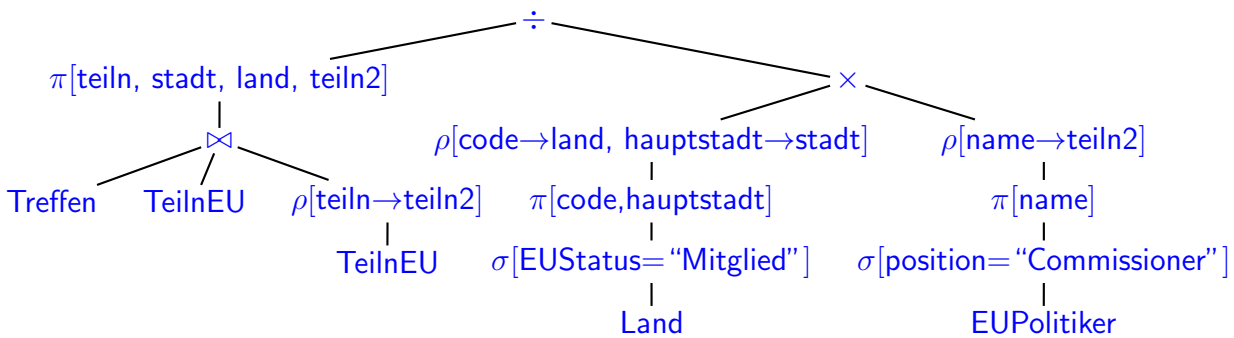
Aufgabe 15 (Der Algebra-Ausdruck dazu [6 Punkte])

Geben Sie einen Algebra-Ausdruck oder -Baum an, der die Anfrage aus der vorhergehenden Aufgabe beantwortet. Sie können den Ausdruck oder Baum wahlweise hier (als ASCII-Art im Editor) abgeben, oder in Aufgabe 12 als Grafik oder Foto hochladen.

Lösung



- linker Unterbaum: alle Tripel teiln1, teiln2, (stadt,land), so dass sich (EU-)teiln1 mit (EU-)teiln2 in (stadt,land) getroffen haben.
(Bsp: Uschi war einmal in Kiew gemeinsam mit Commissioner Margaritis Schinas).
- Division durch (euland+hauptstadt) ergibt alle (teiln1,teiln2)-Paare, die sich in jeder EU-Hauptstadt getroffen haben,
- Division durch teiln2 ergibt alle teiln1, für die das mit jedem teiln2=Commissioner der Fall ist.
- Man kann die Reihenfolge der beiden *divs* vertauschen.
- ODER: man kann für denselben linken Unterbaum durch alle Paare (teiln2, (euland+hauptstadt)) dividieren, und alle teiln1 bekommen, die jede Kombination (teiln2, (euland+hauptstadt)) erfüllen:



Aufgabe 16 (Etwas Theorie [7 Punkte])

Seien $R(A, B, C)$, $S(B)$ und $T(C)$ Relationen.

Begründen oder widerlegen Sie, ob

$$(R \div S) \div T = R \div (T \times S)$$

gilt.

Geben Sie dazu zuerst die Signaturen (d.h., welche Spaltennamen haben die Ergebnisse

der Ausdrücke) der beiden Seiten der Gleichung an (1P), und führen Sie dann den Beweis/Begründung (6P).

(Ilias: Schreiben Sie ggf. “ \div ” als “:” oder einfach als “div”.)

Lösung

Signaturen:

- Die Signatur von $R(A, B, C) \div S(B)$ ist $[A, C]$. Damit hat $(R \div S) \div T$ nur noch eine Spalte A .
 $T \times S$ hat Spalten B und C (die Reihenfolge, ist egal!). Damit hat $R \div (T \times S)$ auch nur noch eine Spalte A .

Es gilt Gleichheit. Auch hier verhält sich die Division so, wie man es von natürlichen Zahlen kennt. Z.B. $24 : 3 : 2 = 24 : (2 \cdot 3) = 4$.

- Am kürzesten ist der formale Beweis in der mathematischen Tupelschreibweise: $R \div S = \{\mu \in \text{Tup}(A, C) \mid \mu \times S \subseteq R\}$ und

$$\begin{aligned} (R \div S) \div T &= \{\nu \in \text{Tup}(A) \mid \nu \times T \subseteq R \div S\} \\ &= \{\nu \in \text{Tup}(A) \mid \nu \times T \subseteq \{\mu \in \text{Tup}(A, C) \mid \mu \times S \subseteq R\}\} \\ &= \{\nu \in \text{Tup}(A) \mid \nu \times T \times S \subseteq R\} \end{aligned}$$

und mit der Kommutativität des kartesischen Produkts:

$$\begin{aligned} &= \{\nu \in \text{Tup}(A) \mid \nu \times S \times T \subseteq R\} \\ &= R \div (S \times T) . \end{aligned}$$

- semiformal ist es einfacher zu lesen, aber auch etwas länger:

$R \div S$ ist eine Tabelle über Attributen (A, C) :

$$R \div S = \{(a, c) \in \text{Tup}(A, C) \mid \forall b \in S : (a, b, c) \in R\}.$$

Diese kann man durch T teilen und erhält

$$\begin{aligned} (R \div S) \div T &= \{(a) \in \text{Tup}(A) \mid \forall c \in T : (a, c) \in R \div S\} \\ &= \{(a) \in \text{Tup}(A) \mid \forall c \in T : (a, c) \in \{(a, c) \in \text{Tup}(A, C) \mid \forall b \in S : (a, b, c) \in R\}\} \\ &= \{(a) \in \text{Tup}(A) \mid \forall c \in T, \forall b \in S : (a, b, c) \in R\} \end{aligned}$$

was wiederum symmetrisch $= R \div (S \times T)$ ist.

- Intuitiv: $R \div S$ sind alle (a, c) die mit jedem b gemeinsam in R auftreten. $(R \div S) \div T$ sind wiederum alle (a) davon, die auch mit jedem c in $(R \div S)$ auftreten. Also genau alle (a) die mit allen Kombinationen von (b, c) in R auftreten, was $= R \div (S \times T)$ und aufgrund der Kommutativität von \times auch $= R \div (T \times S)$ ist.

Man kann es auch anhand von Aufgabe 14 veranschaulichen: R : alle Quadrupel (EU-Politiker, Land+Stadt, Commissioner), so dass sich die beiden in dieser Stadt getroffen haben.

(Bsp: Uschi war einmal in Kiew gemeinsam mit Commissioner Margaritis Schinas).

Dann ist es egal, ob man zuerst durch die Commissioners teilt, und die (Person, Stadt) bekommt, wo die Person schon mit jedem der Commissioner gemeinsam war, und dann schaut, für welche Person das für alle EU-Hauptstädte erfüllt ist,

oder ob man erst durch alle EU-Hauptstädte teilt, und diejenigen (Person, Commissioner) bekommt, die schon gemeinsam in jeder Hauptstadt waren, und dann schaut, ob das mit allen Commissionern einmal der Fall war (siehe zweiter Baum als Lösung der vorhergehenden Aufgabe).

Weitere Übungsaufgaben mit dieser Datenbank

Aufgabe 17 (Anfragen (XXXX) [0 Punkte])

Geben Sie eine SQL-Anfrage *und* einen Ausdruck oder Baum der relationalen Algebra an, die alle Paare von EU-Politikern und Interessenvertretern ausgibt, die aus demselben Land stammen und sich mindestens einmal in der Hauptstadt ihres gemeinsamen Heimatlandes getroffen haben.

(Algebra-Baum entweder hier oder als Grafik in Aufg. 12)

Lösung

```
SELECT distinct e.name, v.name
FROM EUPolitiker e, IntVertreter v, TeilnEU t1, TeilnExt t2, Treffen, Land
WHERE e.Land = v. Land
      AND e.Land = treffen.Land
      AND e.Land = Land.code
      AND Treffen.id = t1.id
      AND Treffen.Land = Land.code
      AND Treffen.Stadt = Land.Hauptstadt
      AND t1.id = t2.id
      AND t1.teiln = e.name
      AND t2.teiln = v.name
```

Das ist ein Paradebeispiel eines breiten Joins, wo man die Funktionalität des natürlichen Joins über gleichnamige Attribute ausnutzen kann.

Man muss daran denken, Attribut "Position" aus EUPolitiker und IntVertreter mindestens an einer Stelle zu eliminieren.

(6 Relationen, 9 Bedingungen, 1 Risiko, 1 Distinct = 17/2 Punkte; 4 P SQL, 5 P Algebra)

$$\pi[\text{euname, extname}]$$
$$(\rho[\text{name} \rightarrow \text{euname}, \text{position} \rightarrow \text{bla}](\text{EUPolitiker}) \bowtie$$
$$\rho[\text{name} \rightarrow \text{extname}](\text{IntVertreter}) \bowtie$$
$$\rho[\text{stadt} \rightarrow \text{Hauptstadt}](\text{Treffen}) \bowtie$$
$$\rho[\text{teiln} \rightarrow \text{euname}](\text{TeilnEU}) \bowtie$$
$$\rho[\text{teiln} \rightarrow \text{extname}](\text{TeilnExt}) \bowtie$$
$$\rho[\text{code} \rightarrow \text{Land}](\text{Land}))$$

Aufgabe 18 (Anfragen (XXXX) [0 Punkte])

Geben Sie eine SQL-Anfrage (und in der *nächsten Aufgabe* einen Algebra-Ausdruck oder Baum) an, die die Namen derjenigen EU-Politiker ausgibt, die sich noch *nie* mit einem Politiker aus Qatar (Landescode "Q") getroffen haben.

Lösung

```
SELECT Name
FROM EUPolitiker
WHERE name NOT IN (Ergebnis von Aufgabe Anfragen-3)
--- das ist die faule Tour.
```



```

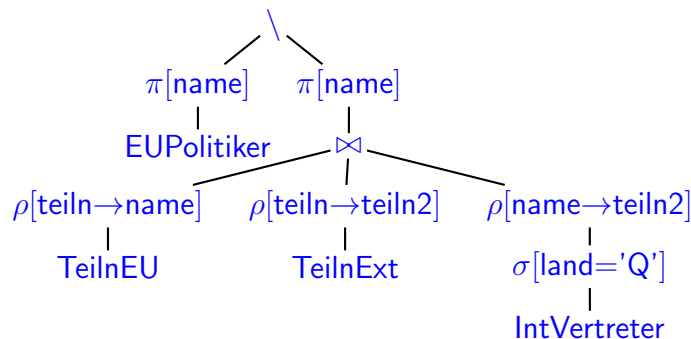
SELECT Name
FROM EUPolitiker p
WHERE NOT EXISTS -- ähnlich wie in der vorigen Aufgabe:
  (SELECT *
   FROM TeilnEU t1, TeilnExt t2, IntVertreter v
   WHERE t1.teiln = p.name
        AND t1.id = t2.id
        AND t2.teiln = v.name
        AND t2.Land = 'Q')

```

Aufgabe 19 (Anfragen (XXXX) [0 Punkte])

Geben Sie einen Algebra-Ausdruck oder -Baum an, der die Anfrage aus der vorhergehenden Aufgabe beantwortet. Sie können den Ausdruck oder Baum wahlweise hier (als ASCII-Art im Editor) abgeben, oder in Aufgabe 12 als Grafik oder Foto hochladen.

Lösung



Aufgabe 20 (Anfragen (XXXX) [0 Punkte])

Geben Sie eine SQL-Anfrage *und* einen Ausdruck oder Baum der relationalen Algebra an, die die Namen derjenigen EU-Politiker ausgibt, die noch nicht in allen Hauptstädten der (derzeitigen) EU-Mitgliedsländer an einem in der Datenbank gespeicherten Treffen teilgenommen haben.

(Hinweis: das ist keine DIV wegen “nicht jedem”)

Aufgabe 21 (Anfragen (XXXX) [0 Punkte])

Geben Sie eine SQL-Anfrage (und in der *nächsten Aufgabe* einen Algebra-Ausdruck oder -Baum) an, die die Namen derjenigen Personen (EU-Politiker und Interessenvertreter) ausgeben, mit denen sich *Ursula von der Leyen* in deren Landeshauptstadt getroffen hat.

Lösung

```

SELECT DISTINCT x.name
FROM Treffen t, TeilnEU vdl,
  ((SELECT id, name, land
   FROM TeilnEU, EUPolitiker
   WHERE teiln = name)
 UNION
 (SELECT id, name, land
  FROM TeilnExt, IntVertreter

```

```

WHERE teiln = name)
) x,
Land
WHERE t.id = vdl.id
AND vdl.teiln = 'U.v.d.Leyen'
AND t.id = x.id
AND x.land = land.code
AND t.stadt = land.hauptstadt

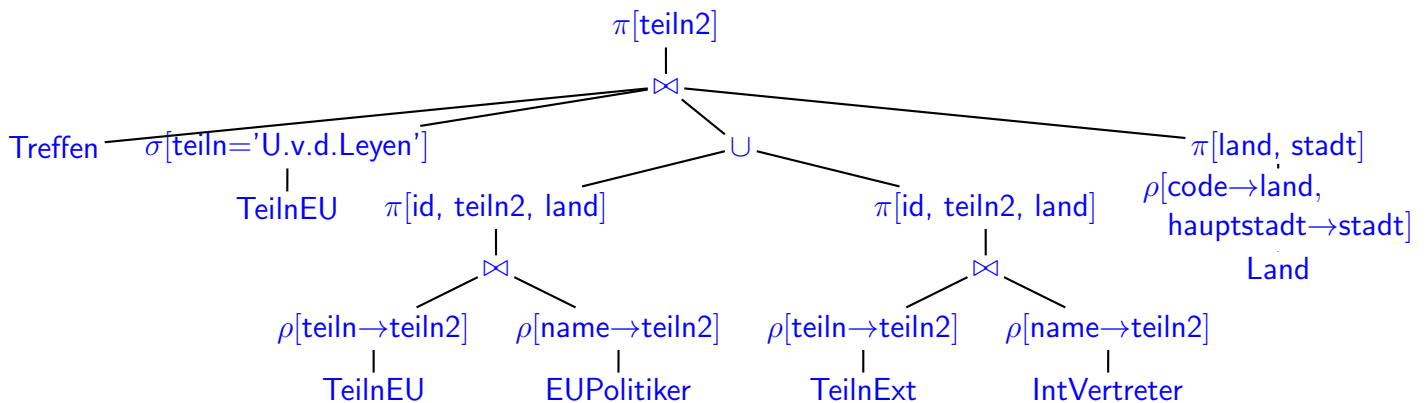
```

-- Wenn man die Tabellen TeilnEU und TeilnExt in einer einzelnen Tabelle hat
-- braucht man trotzdem ein UNION.

Aufgabe 22 (Anfragen (XXXX) [0 Punkte])

Geben Sie einen Algebra-Ausdruck oder -Baum an, der die Anfrage aus der vorhergehenden Aufgabe beantwortet. Sie können den Ausdruck oder Baum wahlweise hier (als ASCII-Art im Editor) abgeben, oder in Aufgabe 12 als Grafik oder Foto hochladen.

Lösung



(Vorsicht: Land hat auch eine Spalte "name")

Aufgabe 23 (Anfragen (XXXX) [0 Punkte])

Geben Sie eine SQL-Anfrage *und* einen Ausdruck oder Baum der relationalen Algebra an, die die IDs derjenigen Treffen ausgibt, bei denen aus jedem EU-Mitgliedsland mindestens ein Teilnehmer anwesend war.

Aufgabe 24 (Eine etwas kompliziertere SQL-Anfrage [0 Punkte])

Geben Sie eine SQL-Anfrage an, die die Namen derjenigen Städte in Nicht-EU-Staaten ausgibt, in denen ein Treffen stattfand, an dem *mindestens die Hälfte aller (aktuellen) EU-Commissioner* (auch wenn diese zum Zeitpunkt des Treffens evtl. keine Commissioner waren) teilgenommen haben.

Lösung

```

SELECT DISTINCT t.stadt          -- count(*) kann man auch ausgeben ...
FROM Treffen t, TeilnEU x, EUPolitiker pol, Land
WHERE t.id = x.id
AND x.teiln = pol.name

```

```

AND pol.Position = 'Commissioner'
AND t.land = land.code
AND land.eustatus = null
GROUP BY ID, stadt
HAVING count(*) >
    0.5 * (SELECT count(*)
           FROM EUPolitiker
           WHERE pol.Position = 'Commissioner')

SELECT DISTINCT tt.stadt
FROM
    (SELECT t.stadt, count(*) as commissionerTeilg
     FROM Treffen t, TeilnEU x, EUPolitiker pol, Land
     WHERE t.id = x.id
          AND x.teiln = pol.name
          AND pol.Position = 'Commissioner'
          AND t.land = land.code
          AND land.eustatus = null
     GROUP BY ID, stadt ) tt,
    (SELECT count(*) as anzahlCommissioner
     FROM EUPolitiker
     WHERE pol.Position = 'Commissioner')
WHERE commissionerTeilg > 0.5 * anzahlCommissioner

```

Das Treffen am 4.2.2023 in Kiew war das erste und bisher einzige solche. Dort waren 16 Mitglieder der EU-Kommission (von 27) anwesend.

% <https://www.zdf.de/nachrichten/politik/eu-kommission-reise-kiew-unterstuetzung>

% Teilnehmerliste: <https://audiovisual.ec.europa.eu/en/photo/P-060090~2F00-17>

Aufgabe 25 (Entwurf [0 Punkte])

In der in Aufgabe 3 vorgegebenen Tabelle “IntVertreter” ist für Nicht-EU-Politiker (wie R.Erdogan und W.Selenskyj) in der Spalte “Organisation” *null* eingetragen.

Welche Vorteile bzw. Risiken hätte es, wenn man dort noch einmal das Land (das ja in der Spalte “Land” schon enthalten ist), eintragen würde?

Lösung Vorteil wäre: Mit SELECT name, organisation könnte man sich einfach einen Überblick ausgeben lassen. Es stünde halt “alles” in dieser Spalte. Intuitiv würde das “gut aussehen”.

Nachteil wäre: so kann man mit where organisation is null einfach herausfinden, welche dieser Personen Politiker sind.

Nachteil wäre: wenn ein Landescode zufällig mit dem Kürzel einer Organisation oder eines Unternehmens übereinstimmt (“C” für den “(British) Commonwealth (of Nations)” und für Cuba), bekommt man eine falsche Semantik.

Weitere Aspekte:

Kein Vorteil: man könnte trotzdem keinen Fremdschlüssel auf Land.code definieren, da ja die Kürzel der Organisationen dort nicht eingetragen sind. So viel bringt es also nicht

Kein Vorteil: Da an ExtTeilnehmer.Organisation nichts drangejoint werden kann (es gibt ja keine Tabelle für die Organisationen), würde man die Spalte sowieso nur bei Politikern benutzen

können, und dann kann man auch gleich Land.code nehmen, wo der Fremdschlüssel definiert ist (und damit ein Index in der Datenbank existiert).
Selbst wenn eine Tabelle "Organisation" zum dranjoinen existieren würde, müsste man in einer Join-Anfrage zwischen Politikern und Organisationsvertretern unterscheiden, um die jeweils passende Tabelle auszuwählen.