

Grundlagen Datenbanken: Übung 02

Tanmay Deshpande

Gruppe 20 & 21

E-Mail: ge94vem@mytum.de



QR-Code für die Folien



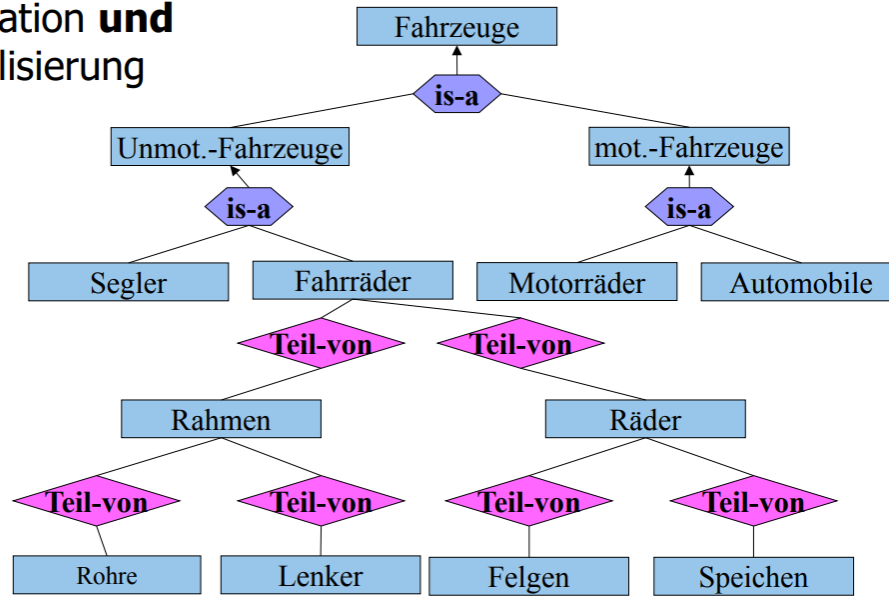
Wiederholung

Woche 02



Aggregation und Generalisierung

Aggregation **und** Generalisierung



- Generalisierung: Untergeordnete Entities bekommen alle Attribute von übergeordneter Entity + eigene Attribute
- Aggregation: Ein Entity durch eine Kombination von mehreren Entitys gebildet

Relationales Schema

- Relationale Datenbanken bestehen aus einer Menge von **Relationen** $R \subseteq D_1 \times D_2 \times \dots \times D_n$
- Relationen als Tabellen dargestellt
- Relationales Schema hilft uns, die Struktur dieser Tabelle festzulegen
- Bsp: $\text{studenten} \subseteq \text{integer}_{\text{matrnr}} \times \text{string}_{\text{name}} \times \text{integer}_{\text{semester}}$

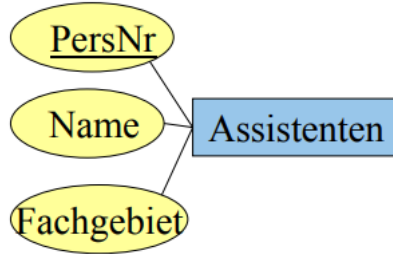
matrnr	name	semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

Schema:

Studenten: {[matrnr: integer, name: string, semester: integer]}

ER Modell zu relationalem Schema

Entity:

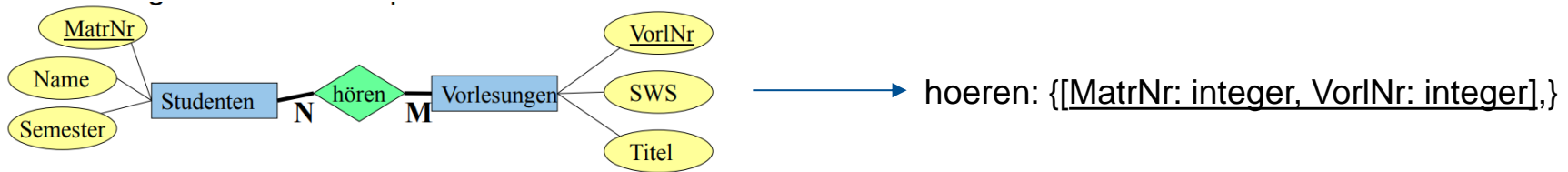


Assistenten: {[PersNr: integer, Name: string, Fachgebiet: string]}

- Attribute der Entität werden zu Attributen des Schemas
- Schlüssel bleibt erhalten

ER Modell zu relationalem Schema

Beziehung:



- Schlüsselattribute der beteiligten Entitäten + eigene Attribute der Beziehung werden zur Attribute des Schemas
- Was wird zum Schlüssel?:
 1:N/N:1 Beziehung => Schlüsselattribute von "N-Entity" werden zu Schlüssel
 1:1 Beziehung => Schlüsselattribute von einer der beiden Entitäten als Schlüssel gewählt
 N:M Beziehung => Schlüsselattribute von beiden formen gemeinsam den Schlüssel
- Daumregel: Entitäts gegenüber die "1" bestimmen Schlüssel!

Verfeinerung des relationalen Schemas

- **Vorlesungen** : {[VorlNr, Titel, SWS]}
 - **Professoren** : {[PersNr, Name, Rang, Raum]}
 - **lesen**: {[VorlNr, PersNr]}
-
- **Vorlesungen** : {[VorlNr, Titel, SWS, **gelesenVon**]}
 - **Professoren** : {[PersNr, Name, Rang, Raum]}
- Verfeinerung durch eliminieren von Beziehungen und zusammenfassen von Relationen
 - Nur Relationen mit gleichem Schlüssel zusammenfassen!
 - **1:N / N:1 Beziehungen:**
 - Beziehungsrelation verschwindet
 - Attribute der Beziehung + Fremdschlüssel der '1-Entity' in 'N-Entity' hinzugefügt
 - **1:1 Beziehungen:**
 - Beziehungsrelation + Eine der Entitys verschwindet,
 - Attribute der Beziehung + Attribute der entfernten Entity in verbleibende Entity übernommen

Null-Werte

- Entstehen, wenn es in einer Relation nicht zu jedem Tupel Einträge zu allen Attribute gibt
- Sollte man möglichst vermeiden
- Eine Methode wäre die Relation in Teilrelationen aufzuteilen

ID	Titel	Jahr	GOTY ¹
1	God of War	2018	2018
2	Final Fantasy VII Remake	2020	-
3	Ghost of Tsushima	2020	-
4	It Takes Two	2021	2021

ID	Titel	Jahr
1	God of War	2018
2	Final Fantasy VII Remake	2020
3	Ghost of Tsushima	2020
4	It Takes Two	2021

Jahr	GOTY
2018	1
2021	4

Relationale Algebra

- Menge von Operationen, die es ermöglichen, Relationen zu manipulieren
- Abgeschlossen (Ergebnis einer Operation ist immer eine Relation, auf die weitere Operationen geführt werden können)
- Tool zum Üben: <https://dbis-uibk.github.io/relax/calc/local/uibk/local/1>

Selektion σ

- Filtriert die Zeilen von der Tabelle heraus, die die Selektionsbedingung erfüllen

Selektion

$\sigma_{\text{Semester} > 10}$ (Studenten)

$\sigma_{\text{Semester} > 10}$ (Studenten)		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12

Projektion π

- Wähle die erwähnten Spalten aus der Relation aus

$\Pi_{\text{Rang}}(\text{Professoren})$

$\Pi_{\text{Rang}}(\text{Professoren})$	
Rang	
C4	
C3	

Kartesisches Produkt x

- Alle mögliche Kombinationen von TüpeIn aus den beiden Relationen

Kartesisches Produkt Professoren x hören

Professoren				hören	
PersNr	Name	Rang	Raum	MatrNr	VorlNr
2125	Sokrates	C4	226	26120	5001
...
2125	Sokrates	C4	226	29555	5001
...
2137	Kant	C4	7	29555	5001

Umbenennung ρ

Umbenennung von Relationen

Beispiel: Ermittlung indirekter Vorgänger 2. Stufe der Vorlesung 5216

$$\Pi_{V1.Vorgänger(\sigma_{V2.Nachfolger=5216 \wedge V1.Nachfolger = V2.Vorgänger}(\rho_{V1}(voraussetzen) \times \rho_{V2}(voraussetzen)))}$$

Umbenennung von Attributen

$$\rho_{Voraussetzung} \leftarrow Vorgänger(voraussetzen)$$

Natural Join

- Ähnlich wie kartesisches Produkt, aber:
 - Kombiniert nur die Einträge, in denen alle Attribute mit gleichem Namen den gleichen Wert haben
 - Doppelte Spalten werden wegprojiziert

CID	Name	CEO
1	Twitter	Elon Musk
2	Facebook	Mark Zuckerberg
3	Apple	Tim Cook



CID	Headquarters
1	San Francisco
2	Menlo Park
3	Cupertino
4	Mountain View

=

CID	Name	CEO	Headquarters
1	Twitter	Elon Musk	San Francisco
2	Facebook	Mark Zuckerberg	Menlo Park
3	Apple	Tim Cook	Cupertino

Angenommen, die linke Tabelle heißt L und die rechte R.
Was hier eigentlich passiert:

$$\pi_{L.CID, L.Name, L.CEO, R.Headquarters} (\sigma_{L.CID = R.CID} (L \times R))$$

Theta Join \bowtie_{θ}

- Ähnlich wie natural Join, aber es gilt:
 - Kombiniert nur die Einträge, in denen die Bedingung θ gilt
 - Doppelte Spalten werden nicht wegprojiziert
 - **Equi-Join:** Sonderform der Theta-Join, wo die Bedingung der Form $a = b$ ist

Beispiel: $\rho_L(\text{Länder}) \bowtie_{L.LID=R.LID \wedge L.LID=1} \rho_R(\text{Regierungschefs})$

Länder	
LID	LName
1	USA
2	Deutschland
3	VR China

Regierungschefs		
RID	RName	LID
1	Olaf Scholz	2
2	Xi Jinping	3
3	Joe Biden	1



$L \bowtie_{L.LID=R.LID \wedge L.LID=1} R$				
LID	LName	RID	RName	LID
1	USA	3	Joe Biden	1

Was hier eigentlich passiert:

$$\sigma_{L.LID=R.LID \wedge L.LID=1}(\rho_L(\text{Länder}) \times \rho_R(\text{Regierungschefs}))$$

Aufgaben

Woche 02

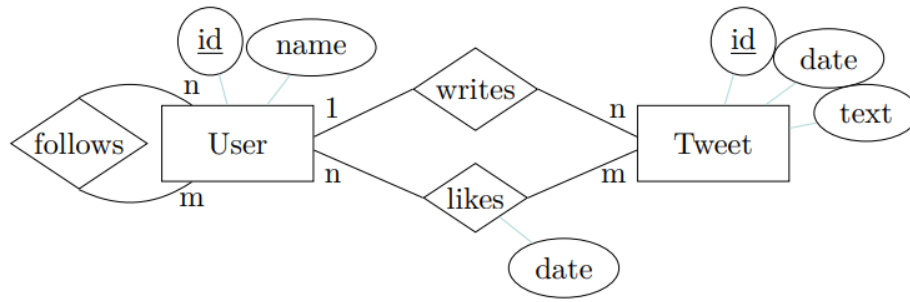


Aufgabe 1

- Erstellen Sie ein ER-Modell womit sich kausale Zusammenhänge darstellen lassen (Prinzip von Ursache und Wirkung). Nehmen Sie an, dass eine Ursache mehrere Wirkungen haben kann, und dass eine Wirkung auf maximal eine Ursache zurückzuführen ist.
- Geben Sie die Funktionalitäten an. Verwenden Sie die (min,max)-Notation.
- Geben Sie außerdem eine (beispielhafte) mögliche Ausprägung Ihres ER-Modells an.

19

Aufgabe 2



- a) Übertragen Sie das ER-Modell in ein relationales Schema.
- b) Verfeinern Sie das relationale Schema durch Elimination von Relationen.
- c) Kann das Attribut *date* des Entity-Typs **Tweet** stattdessen der Relationship **writes** zugeordnet werden? Kann das Attribut *date* der Relationship **likes** dem Entity-Typen **Tweet** zugeordnet werden?

Lösungsvorschlag 2

- a)

User : { [id, name] }

Tweet : { [id, date, text] }

follows : { [follower_id, follows_id] }

writes : { [tweet_id, user_id] }

likes : { [user_id, tweet_id, date] }

→

b) User : { [id, name] }

Tweet : { [id, user_id, date, text] }

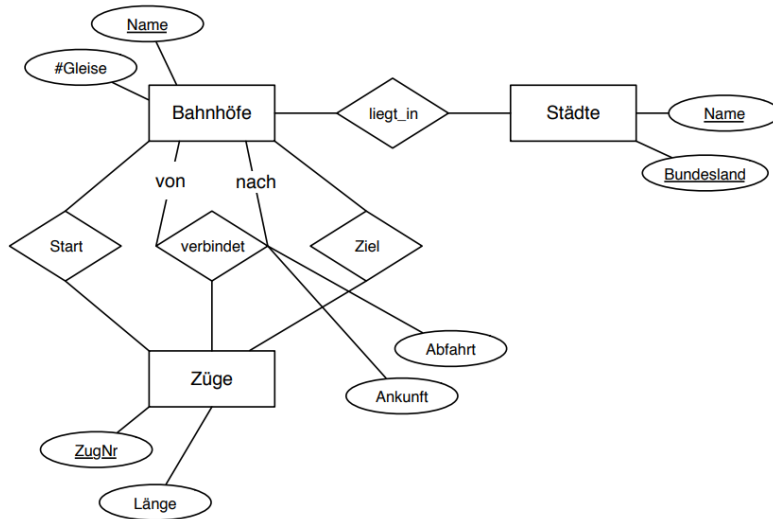
follows : { [follower_id, follows_id] }

likes : { [user_id, tweet_id, date] }

c) *date* (aus *Tweet*) kann auch *writes* zugeordnet werden. Nach der Verfeinerung sind alle Attribute von *writes* und von *Tweet* in einer Relation zusammengefasst. Für das relationale Schema ist es also egal, ob das Attribut ursprünglich zu *writes* oder zu *Tweet* gehört hat.

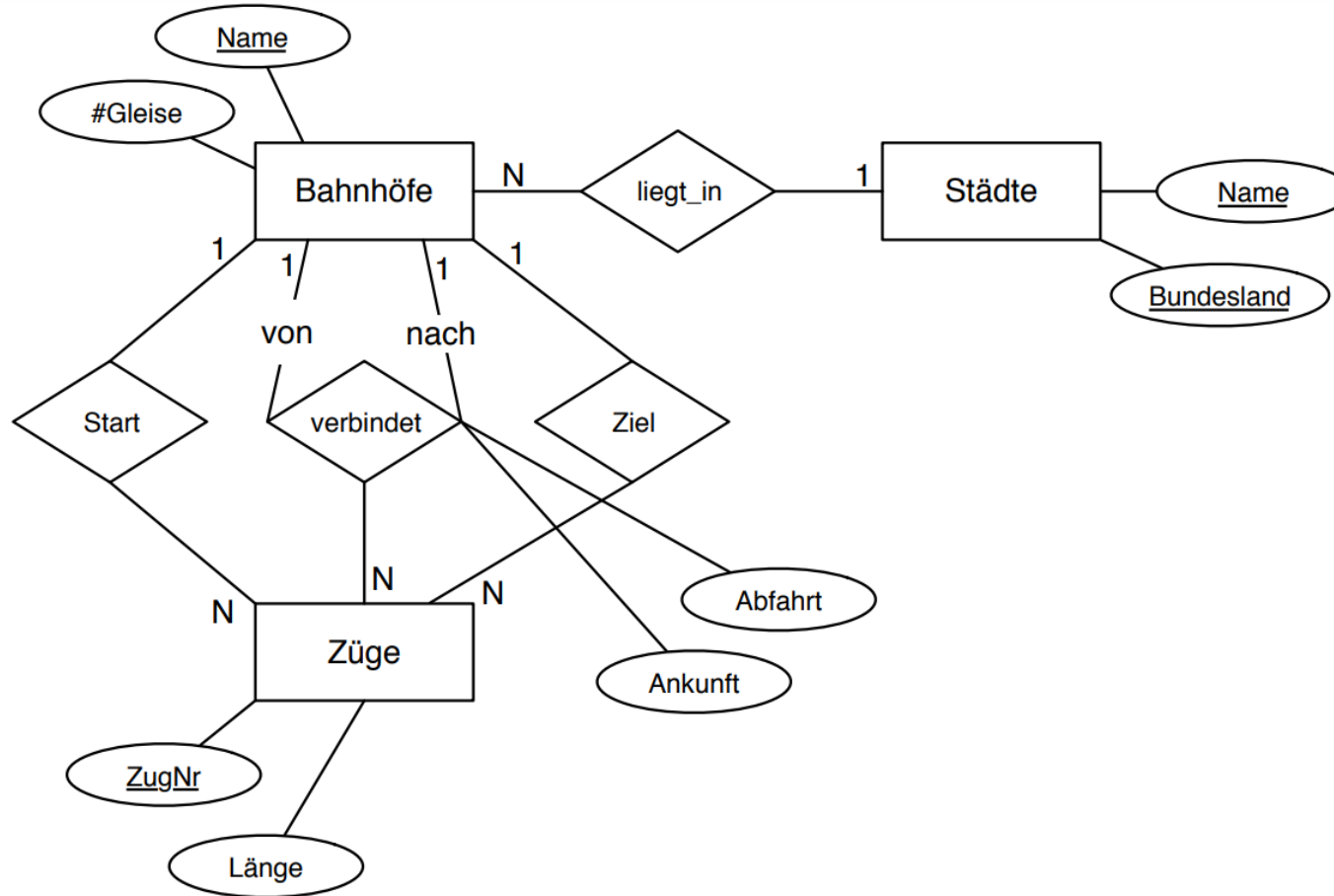
date (aus *likes*) kann keinem anderen Entity-Typen zugeordnet werden, da weder User noch Tweet wegen ihrer n-zu-m-Beziehung ein eindeutiges „like_date“-Attribut haben können.

Aufgabe 3



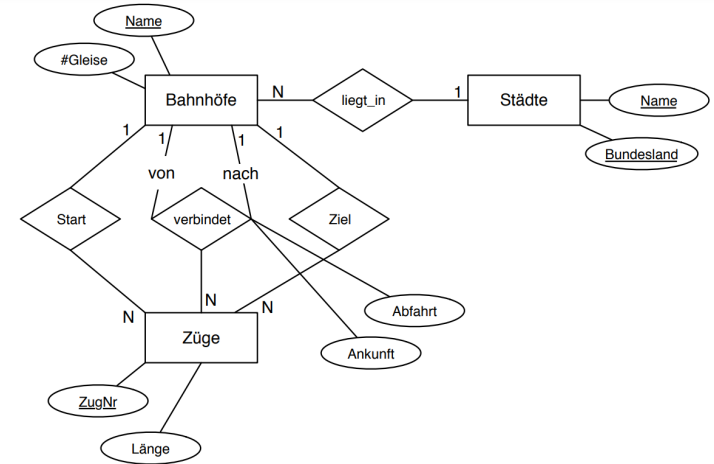
- a) Fügen Sie bei den Beziehungen Funktionalitätsangaben hinzu.
- b) Übertragen Sie das ER-Modell in ein relationales Schema.
- c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen

Bemerkung: *verbindet* modelliert ein Teilstück einer Verbindung. Bsp.: Auf der Strecke München → Hamburg gibt es einen Eintrag für die Teilstrecke von München nach Nürnberg, einen Eintrag für Nürnberg nach Würzburg, einen Eintrag für die Teilstrecke Würzburg nach Göttingen und einen Eintrag von Göttingen nach Hamburg.



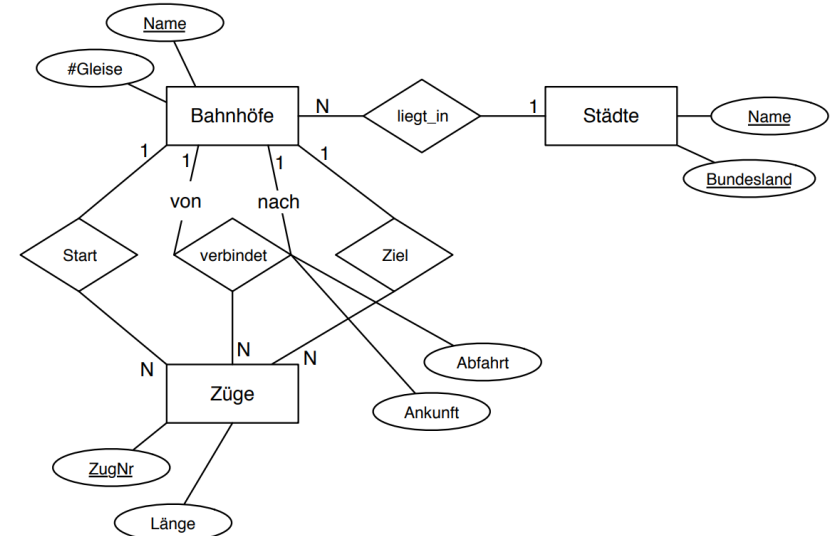
Lösungsvorschlag 3b

Städte : {[Name : string, Bundesland : string]}
 Bahnhöfe : {[Name : string, #Gleise : integer]}
 Züge : {[ZugNr : integer, Länge : integer]}
 liegt_in : {[BName : string, SName : string, Bundesland : string]}
 Start : {[ZugNr : integer, BName : string]}
 Ziel : {[ZugNr : integer, BName : string]}
 verbindet : {[VonBahnhof : string, NachBahnhof : string,
 ZugNr : integer, Abfahrt : date, Ankunft : date]}



Lösungsvorschlag 3c

Städte : {[Name : string, Bundesland : string]}
 Bahnhöfe : {[Name : string, #Gleise : integer,
 SName : string, Bundesland : string]}
 Züge : {[ZugNr : integer, Länge : integer,
 StartBahnhof : string, ZielBahnhof : string]}
 verbindet : {[VonBahnhof : string, NachBahnhof : string,
 ZugNr : integer, Abfahrt : date, Ankunft : date]}

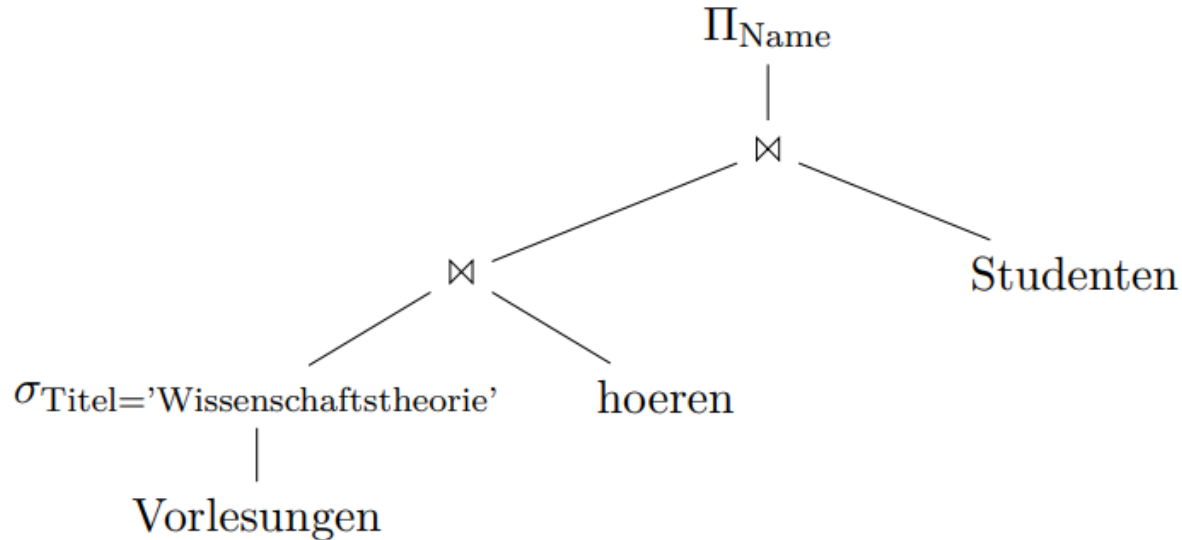


Aufgabe 4

- Formulieren Sie die folgenden Anfragen auf dem bekannten Universitätsschema in Relationenalgebra. Geben Sie die Lösungen in der Operatorbaum-Darstellung an.
- a) Geben Sie alle Studenten an, die die Vorlesung Wissenschaftstheorie gehört haben.
- b) Geben Sie die Titel der Vorlesungen an, die die Vorlesung Wissenschaftstheorie direkt voraussetzen.
- c) Geben Sie Paare von Studenten(-Namen) an, die sich aus der Vorlesung Grundzüge kennen

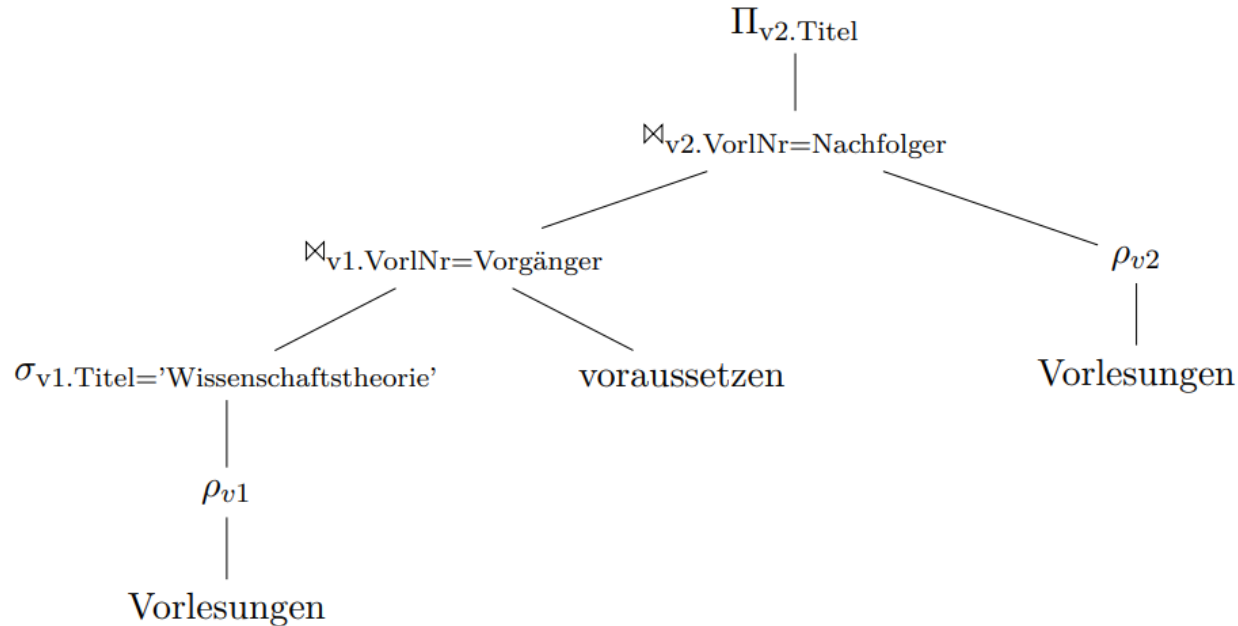
Lösungsvorschlag 4a

Die Namen aller Studenten, die die Vorlesungen *Wissenschaftstheorie* gehört haben



Lösungsvorschlag 4b

Die Titel aller Vorlesungen, die die Vorlesung *Wissenschaftstheorie* direkt voraussetzen



Lösungsvorschlag 4c

Paare von Studenten(-Namen) an, die sich aus der Vorlesung Grundzüge kennen

