



# Spaß mit Datenbanken

1

# Kurzer Rückblick

# Datenmodellierung



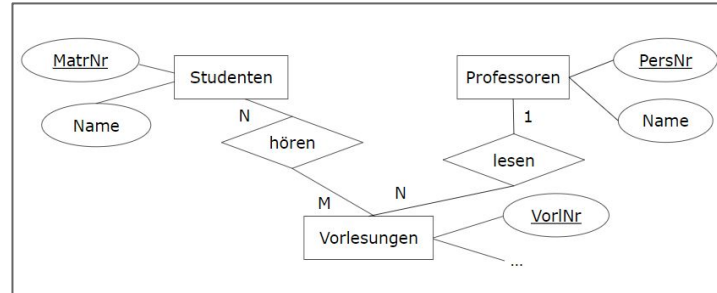
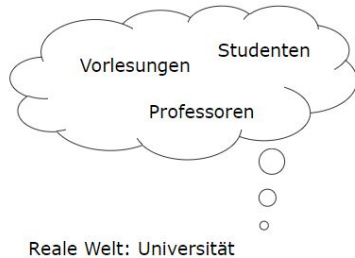
Ausschnitt



Konzeptuelles Schema



Logisches Schema



Studenten	
MatrNr	Name
26120	Fichte
25403	Jonas
...	...

hören	
MatrNr	VorlNr
25403	5022
26120	5001
...	...

Vorlesungen	
VorlNr	Titel
5001	Grundzüge
5022	Glaube und Wissen
...	...

# Datenmodellierung



Logisches Schema

Abfrage & Manipulation

Studenten	
MatrNr	Name
26120	Fichte
25403	Jonas
...	...

hören	
MatrNr	VorlNr
25403	5022
26120	5001
...	...

Vorlesungen	
VorlNr	Titel
5001	Grundzüge
5022	Glaube und Wissen
...	...

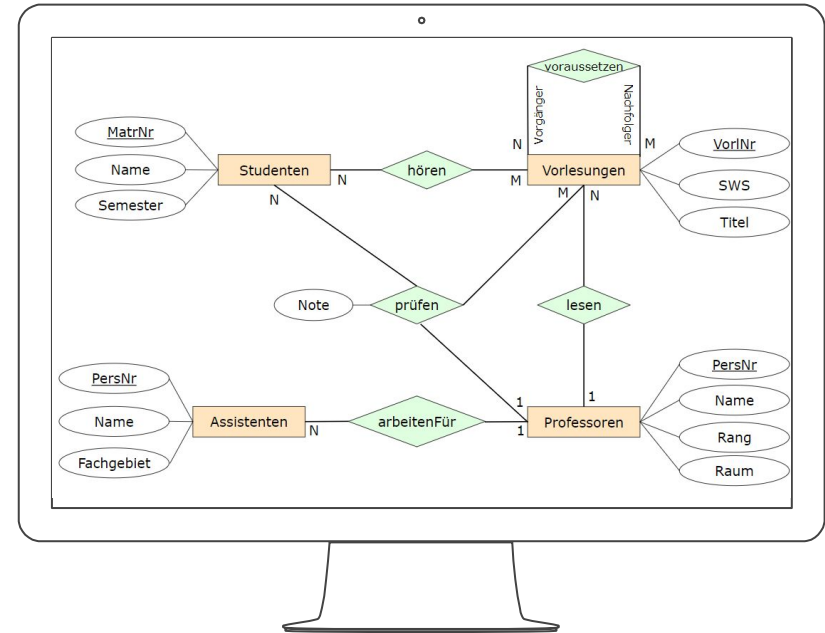
```
SELECT Name
FROM Studenten, hören, Vorlesungen
WHERE Studenten.MatrNr = hören.MatrNr AND
       hören.VorlNr = Vorlesungen.VorlNr AND
       Vorlesungen.Titel = 'Grundzüge';
```

```
UPDATE Vorlesungen
SET Titel = 'Grundzüge der Logik'
WHERE VorlNr = 5001;
```

# Entity Relationship Modell

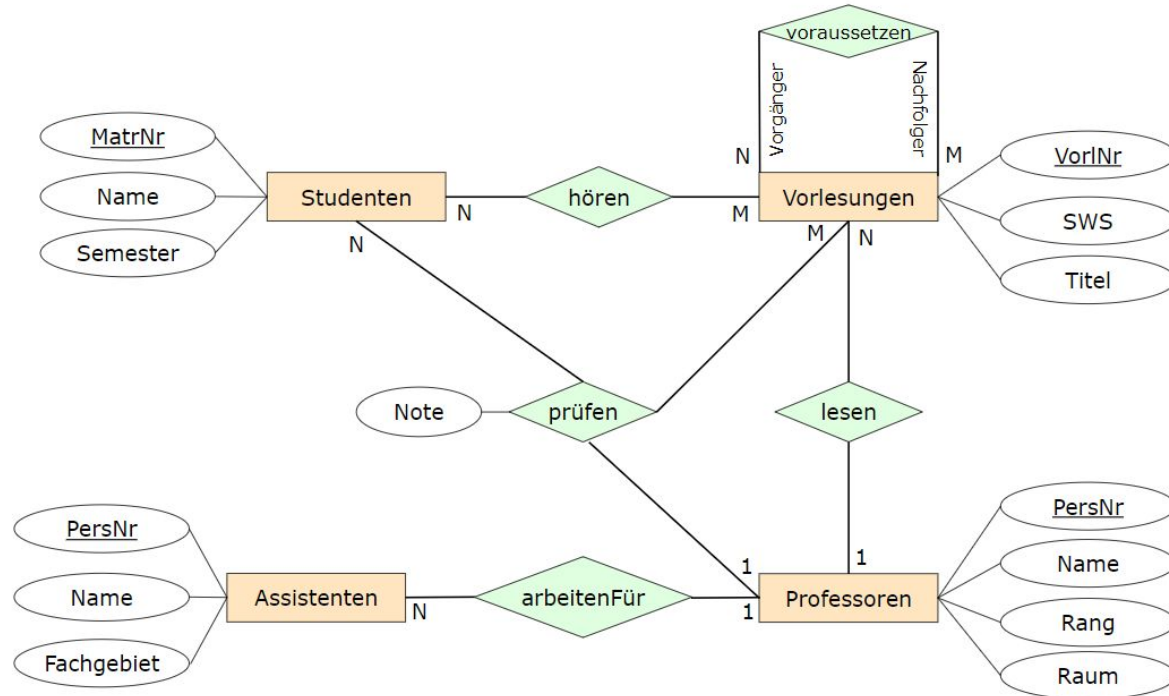
Modelliert **Gegenstände (Entities)** und die **Beziehungen (Relationships)** zwischen diesen.

- **Entity**
  - **Objekt**, über welches Informationen zu speichern sind (bspw. Vorlesung, Professor, Prüfung).
- **Relationship**
  - **Beziehung** zwischen Entities. z.B. Professor liest Vorlesung, Professor prüft Student.
- **Attribut**
  - **Eigenschaft** von Entities oder Beziehungen. z.B. Name von Professor, Titel einer Vorlesung.





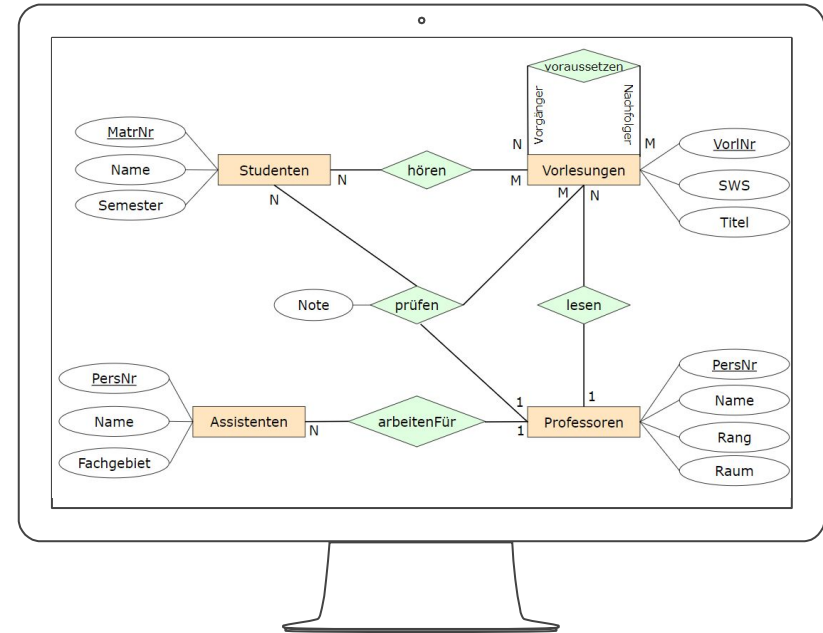
# ER-Modell - Chen Notation





# Entity Relationship Modell

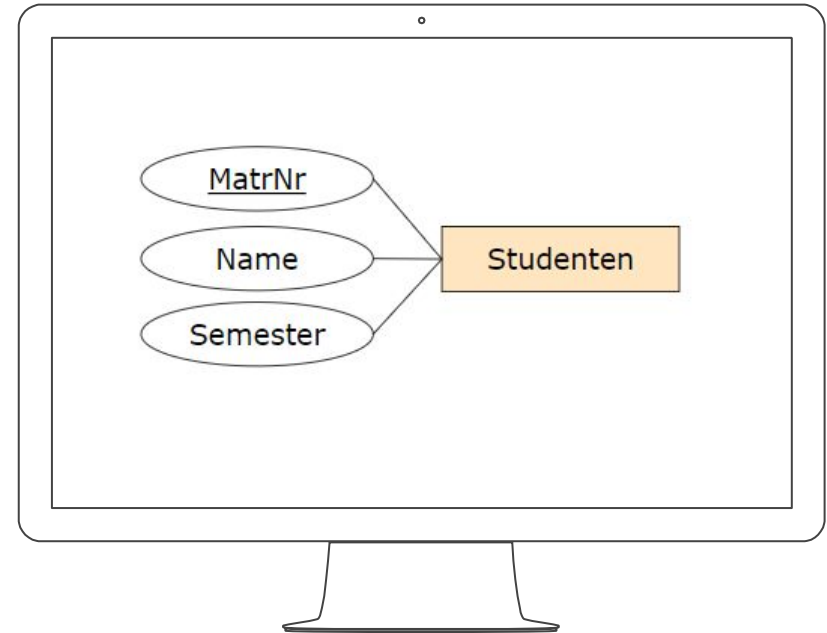
- **Werte**
  - Primitive Datenelemente, die direkt darstellbar sind
  - Werden durch **Datentypen** beschrieben
- **Datentypen**
  - Vorgegebene Standard-Datentypen, wie etwa ganze Zahlen (**int**), Zeichenketten (**string**), Datumswerte (**date**) etc.





# Attribute

- Modellieren Eigenschaften von **Entities oder Beziehungen**
- **Alle Entities** eines Entity-Typen (hier bspw. "Studenten") **haben dieselben** Arten von **Eigenschaften**.
- Attribute werden somit für **Entity-Typen** deklariert
- textuelle Notation:  
E (A1 : D1 , ..., Ax : Dx )
- Attributen ist ein **Datentyp** zuzuweisen

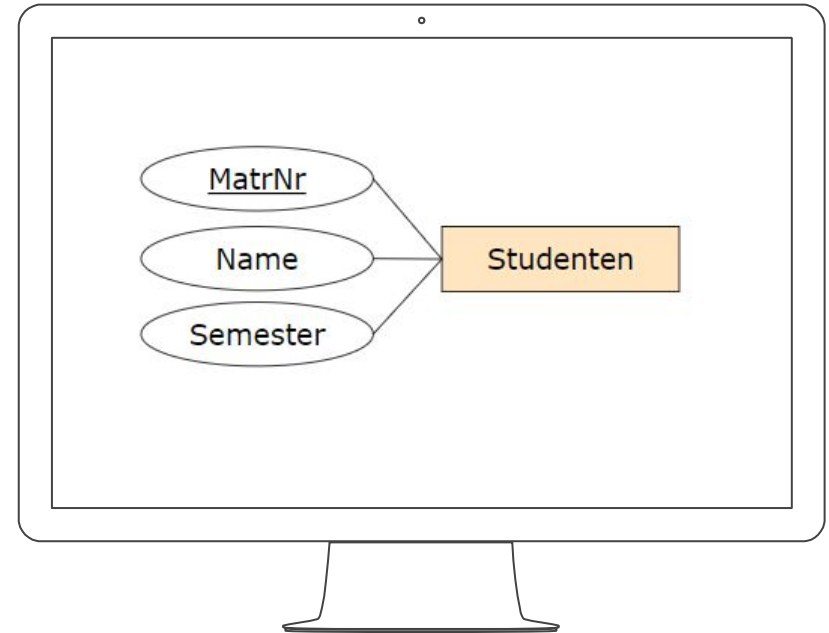






# Schlüssel

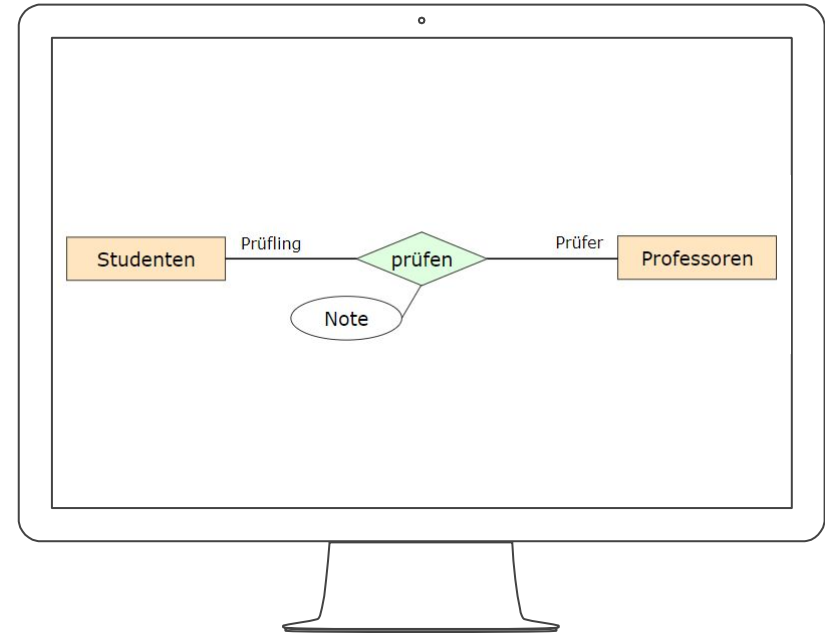
- Die Werte der Schlüsselattribute identifizieren Entities eindeutig
- Bei mehreren Schlüsselkandidaten ist ein **Primärschlüssel** zu wählen (im Modell unterstreichen)
- Beispiel rechts: Die Matrikelnummer kennzeichnet einen Studenten eindeutig. Zu jeder Matrikelnummer gibt es genau einen Studenten.





# Beziehungen

- Beziehungen zwischen Entities werden zu **Beziehungstypen** zusammengefasst.
- Beziehungen können ebenfalls **Attribute** besitzen (Note im Beispiel)



# Beziehungen

## Unterscheidung in Grade

Am häufigsten: **binär**

Zwei beteiligte Entitäten

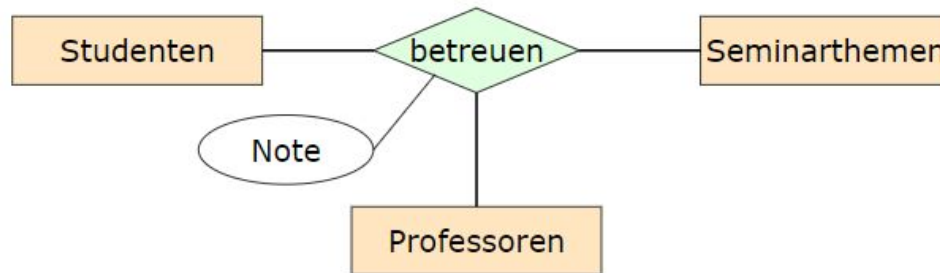


# Beziehungen

## Unterscheidung in Grade

Weiters: **ternär**

Drei beteiligte Entitäten

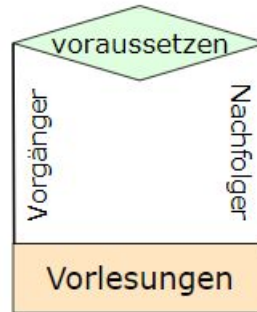


# Beziehungen

## Unterscheidung in Grade

Und dann noch: **unär**

Eine beteiligte Entität



# Beziehungen

## Unterscheidung Kardinalitäten

Beschreiben die Beziehung genauer.

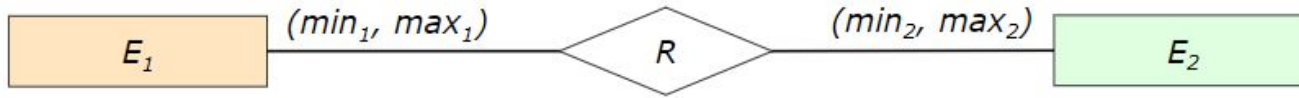
Formen: 1:1, 1:N, N:1, M:N

Müssen immer eingehalten werden

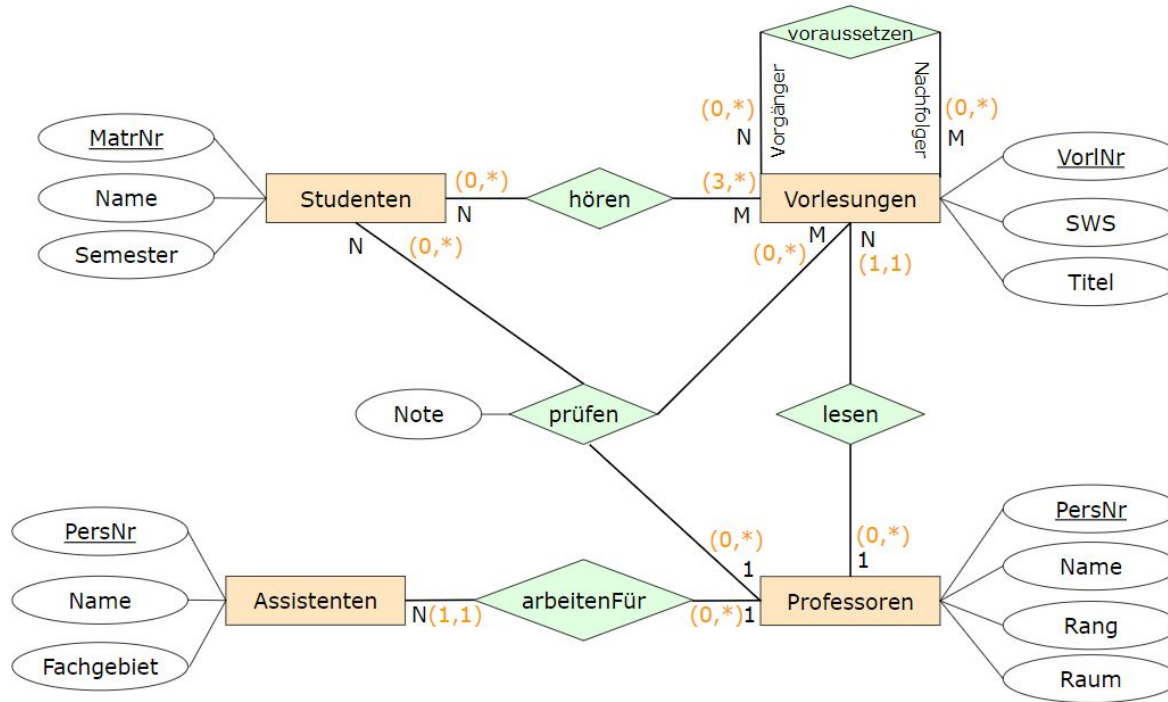


# (min, max)-Notation

- Verleiht einer Beziehung noch mehr Information
- Schränkt Teilnahme von Instanzen durch Vorgabe von minimal- und maximalwert.



# Beispiel (min, max)-Notation

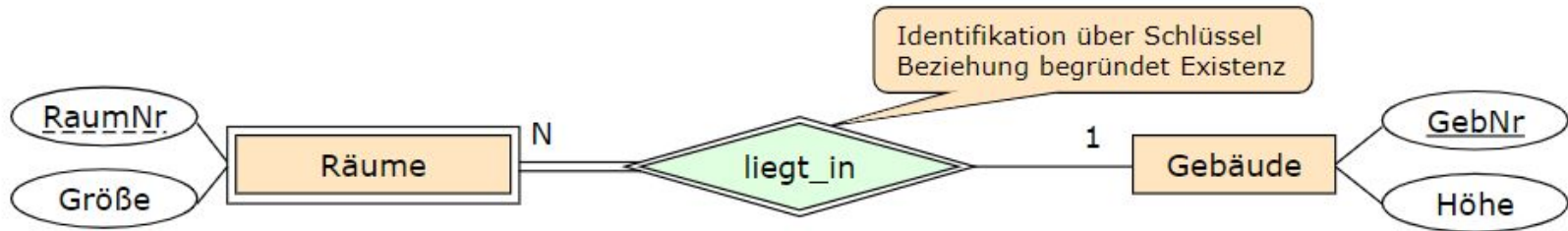




# Existenzabhängigkeit

## Starke und Schwache Entitäten

- Schwache Entities sind in ihrer Existenz von einer anderen (starken) abhängig
- Sind nur in Kombination mit dem Schlüssel der starken Entity eindeutig identifizierbar.



# Relationenmodell

- Konzeptuell ist die Datenbank eine **Menge von Tabellen**
- Eine Tabelle = "**Relation**"
- Eine Zeile = "**Tupel**"

**Professoren**

<u>PersNr</u>	<u>Name</u>	<u>Rang</u>	<u>Raum</u>
2125	Sokrates	C4	226
2126	Russel	C4	232
2127	Kopernikus	C3	310
2133	Popper	C3	52
2134	Augustinus	C3	309
2136	Curie	C4	36
2137	Kant	C4	7

**Vorlesungen**

<u>VorlNr</u>	<u>Titel</u>	<u>SWS</u>	<u>GelesenVon</u>
5001	Grundzüge	4	2137
5041	Ethik	4	2125
5043	Erkenntnistheorie	3	2126
5049	Mäeutik	2	2125
4052	Logik	4	2125
5052	Wissenschaftstheorie	3	2126
5216	Bioethik	2	2126
5259	Der Wiener Kreis	2	2133
5022	Glaube und Wissen	2	2134
4630	Die 3 Kritiken	4	2137

# Relationenmodell

## Schlüssel

- Attribute, deren Werte ein Tupel **eindeutig** identifizieren sind **Schlüssel**.
- Schlüssel soll **minimal** sein, also aus so wenig Attributen wie möglich bestehen.
- Schlüsselwert darf **niemals** NULL sein!

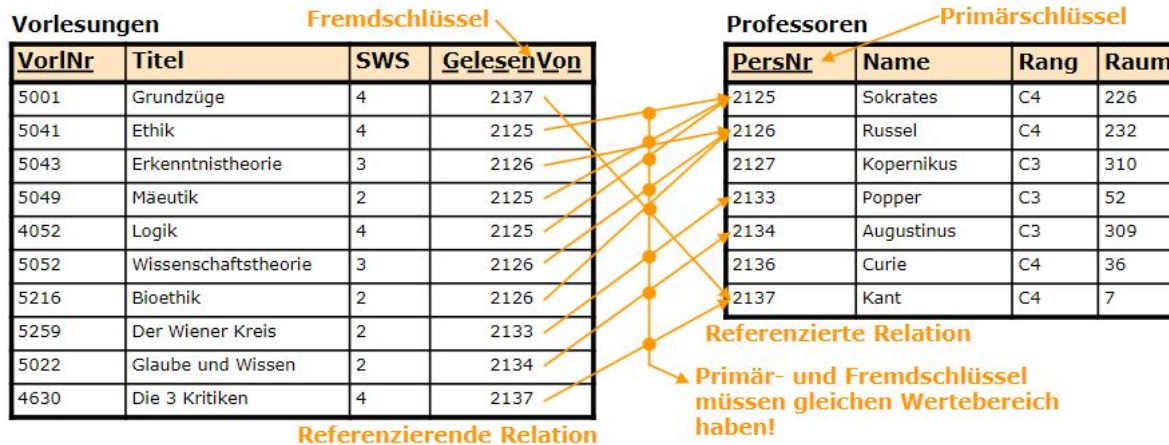
Professoren

<u>PersNr</u>	Name	Rang	Raum
2125	Sokrates	C4	226
2126	Russel	C4	232
2127	Kopernikus	C3	310
2133	Popper	C3	52
2134	Augustinus	C3	309
2136	Curie	C4	36
2137	Kant	C4	7

# Relationenmodell

## Fremdschlüssel

- Schlüssel einer Tabelle können in einer anderen (oder derselben) Tabelle als **eindeutige Verweise** genutzt werden.



# Relationale Darstellung (Uni)

Logischer Entwurf:

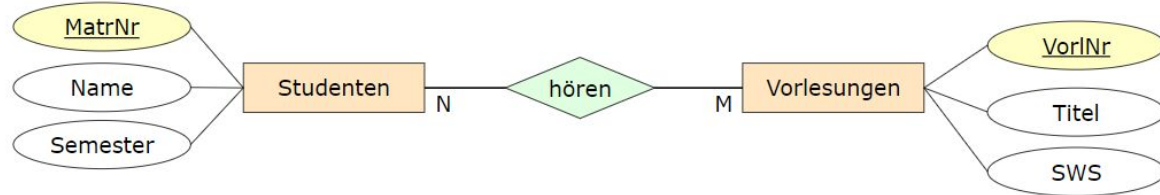
Studenten: {[MatrNr:integer, Name:string, Semester:integer]}

Vorlesungen: {[VorlNr:integer, Titel:string, SWS:integer]}

Professoren: {[PersNr:integer, Name:string, Rang:string, Raum:integer]}

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

# Kreuztabellen für N:M Beziehung



Studenten

<u>MatrNr</u>	Name
24002	Xenokrates
25403	Jonas
26120	Fichte
26830	Aristoxenos
28106	Carnap
29555	Feuerbach
...	...

hören

<u>MatrNr</u>	<u>VorlNr</u>
26120	5001
24002	5001
24002	4052
...	...

Vorlesungen

<u>VorlNr</u>	Titel	SWS
5001	Grundzüge	4
5041	Ethik	4
5043	Erkenntnistheorie	3
5049	Mäeutik	2
4052	Logik	4
5052	Wissenschaftstheorie	3
5216	Bioethik	2
5259	Der Wiener Kreis	2
5022	Glaube und Wissen	2
4630	Die 3 Kritiken	4

Logischer Entwurf:

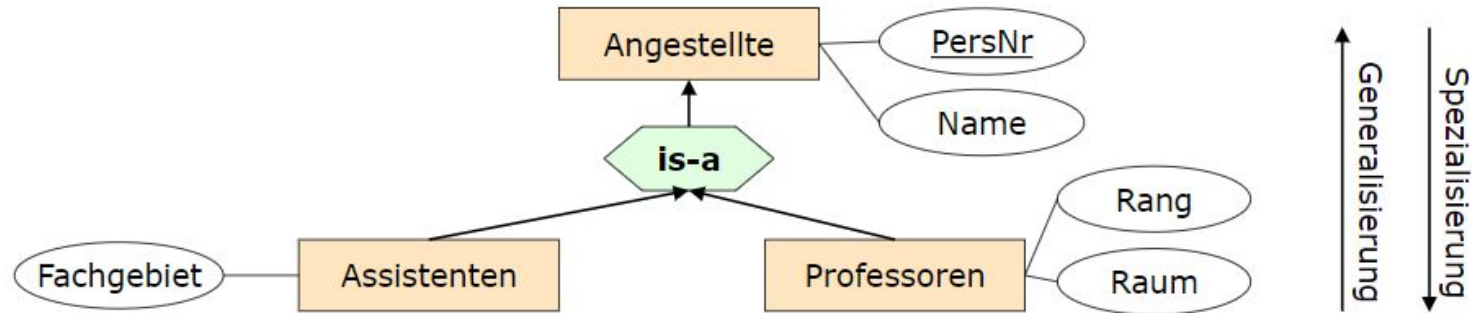
hören: {[MatrNr:integer, VorlNr:integer]} (**N:M**)

# Datenbankentwurf

Weitere Konzepte

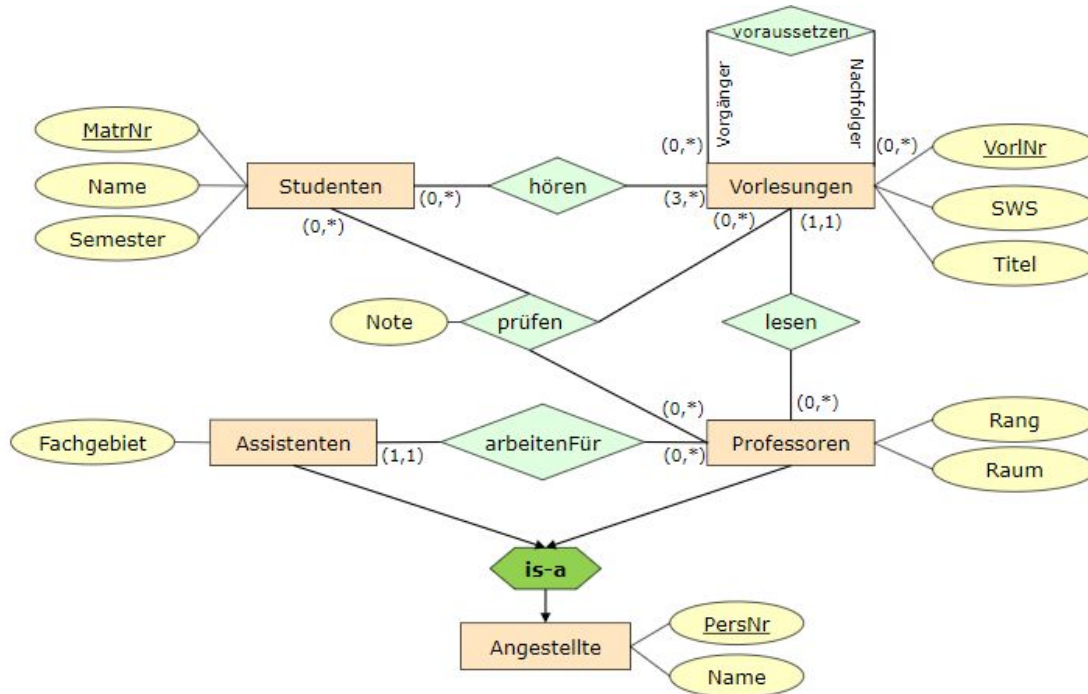
# Generalisierung / Spezialisierung

- Untertypen werden von Obertypen abgeleitet.
- Untertypen haben neben den vererbten Attributen auch eigene Attribute.
- Ober- und Untertyp besitzen den gleichen **Primärschlüssel**



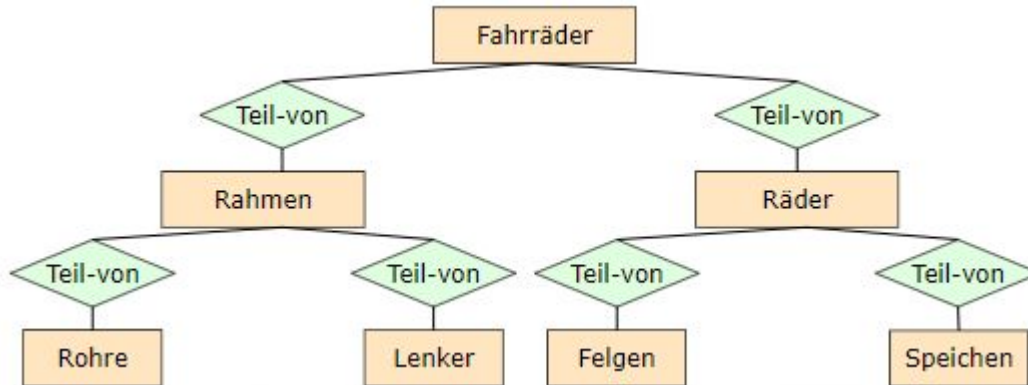


# Generalisierung / Spezialisierung



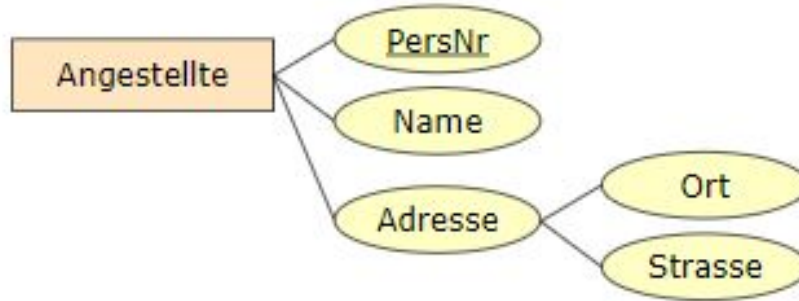
# Aggregation

- Entitäten, die gemeinsam einen komplexen Entitäten-Typen bilden, werden einander zugeordnet.
- Es gibt den übergeordneten Typen (Aggregat), dem ein oder mehrere Typen (Teile) untergeordnet sind.
- Wird als “**Teil-von**” oder “**part-of**” Relationship bezeichnet.



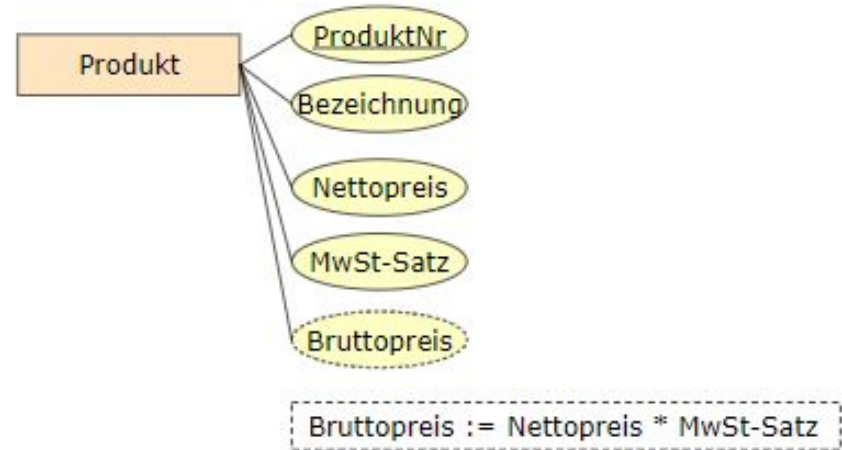
# Strukturierte Attribute

- Attributgruppen, die Gemeinsamkeiten in ihrer Verwendung haben.



# Abgeleitete Attribute

- Attribute, deren Werte nicht gespeichert werden.
- Werte können durch eine Anfrage an die Datenbank bestimmt werden.



# Surrogate Keys - Künstlicher Schlüssel

- Zusätzliches Schlüsselattribut, ohne Anwendung in der realen Welt
- In der Regel Datentyp: NUMBER
- Dient zur eindeutigen Identifizierung der Entität
- Ersetzen aus mehreren Attributen zusammengesetzten Primärschlüssel
- einfacherer Index-Aufbau
- schnellere Suche..

# Festigen

# Existenzabhängigkeit

Warum können Beziehungen zwischen einer starken und einer schwachen Entität, keine **N:M** Beziehungen sein?

# Beziehungen

Student

Kind

Kunde

Mitarbeiter

Patient

Schüler

Ehemann

Mitarbeiter

Veranstaltung

Mutter

Artikel

Vorgesetzter

Blinddarm

Klasse

Ehefrau

Fähigkeiten



# Kino Management

**ERM**

**interpretieren**

# Datenbankentwurf

Relationenmodell

# Relationenmodell

## Ziele

- Verbesserung (Feinabstimmung) des logischen Entwurfs **ohne** dabei semantische Informationen zu verlieren
- Redundanzvermeidung durch **Normalformen**
- Vermeidung von Anomalien / Inkonsistenzen

# Relationenmodell - Redundanzen

## Schlechtes Relationenschema

Professoren

PersNr	Name	Rang	Raum	<u>VorlNr</u>	Titel	SWS
2125	Sokrates	C4	226	5041	Ethik	4
2125	Sokrates	C4	226	5049	Mäeutik	2
2125	Sokrates	C4	226	4052	Logik	4
...	...	...	...	...	...	...
2133	Popper	C3	52	5295	Der Wiener Kreis	2
2137	Kant	C4	7	4630	Die 3 Kritiken	4

# Ziel: Redundanz vermeiden

- Redundante Informationen benötigen unnötigen Speicherplatz
- Änderungen sind schwer korrekt umsetzbar, weil die Änderungen an allen Stellen vorgenommen werden müsste.

# Ziel: Anomalien vermeiden

- **Änderungsanomalie:** Beim Ändern eines Wertes müssen viele andere Tupel ebenfalls geändert werden
- **Einfügeanomalie:** Beim Einfügen eines Tupels können bestimmte Werte nicht angegeben werden, da sie noch nicht bekannt sind. Wenn bspw. Schlüsselwerte fehlen, kann Tupel nicht einmal eingefügt werden
- **Löschanomalie:** Beim Löschen geht mehr Information verloren, als beabsichtigt.

# Ziel: Anomalien vermeiden

Professoren

PersNr	Name	Rang	Raum	<u>VorlNr</u>	Titel	SWS
2125	Sokrates	C4	226	5041	Ethik	4
2125	Sokrates	C4	226	5049	Mäeutik	2
2125	Sokrates	C4	226	4052	Logik	4
...	...	...	...	...	...	...
2133	Popper	C3	52	5295	Der Wiener Kreis	2
2137	Kant	C4	7	4630	Die 3 Kritiken	4

- **Änderungsanomalie:** Bsp. Sokrates zieht um
- **Einfügeanomalie:** Bsp. Curie ist neu und liest noch keine Vorlesung
- **Löschanomalie:** Bsp. "Die 3 Kritiken" fällt weg.



# Ziel: Anomalien vermeiden

- Anomalien entstehen, wenn nicht zusammenpassende Informationen zusammen gespeichert werden.
- Lösung bietet die Zerlegung des Schemas in Teilschemata (alle Informationen zu Professoren werden in einer Relation "Professoren" gespeichert. Alle Informationen zu Vorlesungen werden in einer Relation "Vorlesungen gespeichert usw.

# Normalformen

- Legen **Eigenschaften** von Relationsschemata fest
- **Verbieten** bestimmte **Kombinationen** in Relationen
- Sollen Redundanzen und Anomalien vermeiden

# Normalformen

## Erste Normalform

- Erlaubt nur **atomare Attribute** in den Relationsschemata. D.h. Attributwerte sind Elemente von **Standard-Datentypen** wie *integer* oder *string*, aber keine Mengenwerte wie *array* oder *set*

Nicht in 1NF:

Eltern		
Vater	Mutter	Kinder
Johann	Martha	{Else, Lucie}
Heinz	Martha	{Cleo}
...	...	...

in 1NF (= flache Relation)

Eltern		
Vater	Mutter	Kinder
Johann	Martha	Else
Johann	Martha	Lucie
Heinz	Martha	Cleo

# Normalformen

## Zweite Normalform

- **Partielle Abhängigkeit** liegt vor, wenn ein Attribut funktional nur von einem Teil des Schlüssel abhängt.
- Verstoß gegen 2NF deutet darauf hin, dass in der Relation Informationen über mehr als ein Konzept modelliert werden.
- Zweite Normalform eliminiert **partielle Abhängigkeiten** bei Nichtschlüsselattributen.

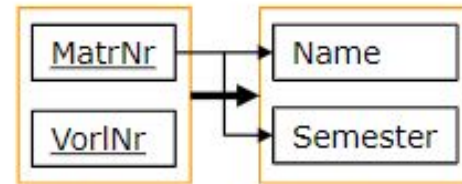
# Normalformen

## Zweite Normalform

StudentenBelegung

MatrNr	VorlNr	Name	Semester
26120	5001	Fichte	10
27550	5001	Schopenhauer	6
27550	4052	Schopenhauer	6
28106	5041	Carnap	3
28106	5052	Carnap	3
28106	5216	Carnap	3
28106	5259	Carnap	3
...	...		...

- $\{\text{MatrNr}\} \rightarrow \{\text{Name}\}$  und
- $\{\text{MatrNr}\} \rightarrow \{\text{Semster}\}$



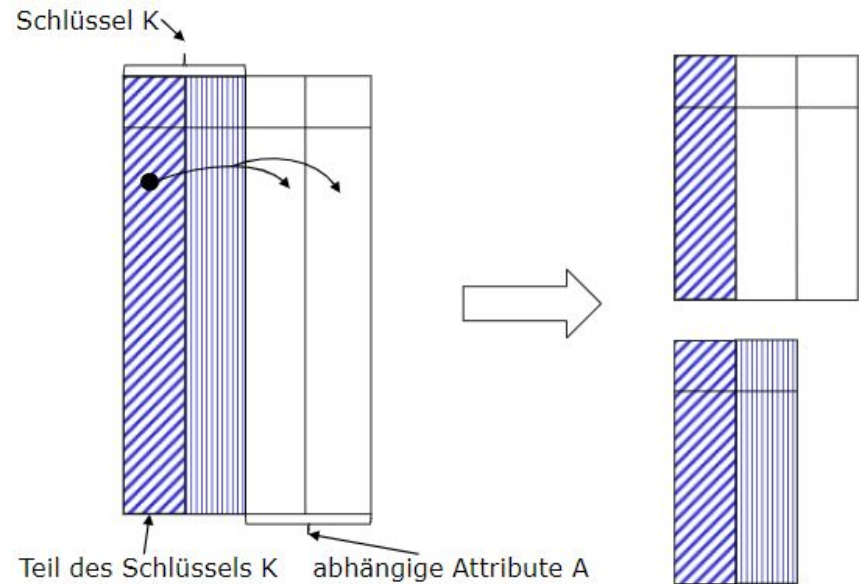
→ Schlüssel

→ zusätzliche  
funktionale  
Abhängigkeiten

# Normalformen

## Zweite Normalform

- Eliminierung partieller Abhängigkeiten



# Normalformen

## Zweite Normalform

- Eliminierung partieller Abhängigkeiten

Relation in 2NF:

StudentenBelegung: {MatrNr, VorlNr, Name, Semester}



hören: {MatrNr, VorlNr}

Studenten: {MatrNr, Name, Semester}

# Normalformen

## Dritte Normalform

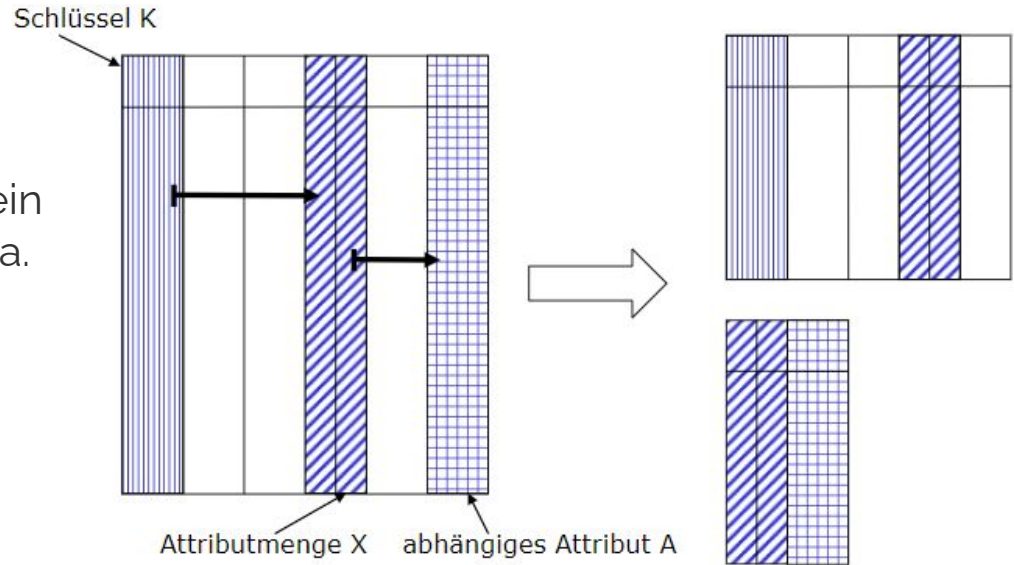
- Eliminiert (zusätzlich) transitive Abhängigkeiten
- Beispiel:
  - $R = \{\underline{\text{PersNr}}, \text{Name}, \text{Raum}, \text{Rang}, \text{PLZ}, \text{Ort}, \text{Straße}\}$
  - $\{\text{PersNr}\} \rightarrow \{\text{PLZ}\}$  und  $\{\text{PLZ}\} \rightarrow \{\text{Ort}\}$
- Man beachte: 3.NF betrachtet **nur** Nichtschlüsselattribute als Endpunkt transitiver Abhängigkeiten.



# Normalformen

## Dritte Normalform

- Eliminierung transitiver Abhängigkeiten durch Verschiebung transitiv abhängiger Attribute in ein neues Relationenschema.



# Normalformen

## Dritte Normalform

- Elminierung transitiver Abhängigkeiten

### Relation in 3NF:

Professoren: {ProfNr, Name, Raum, Rang, PLZ, Ort, Straße}



Professoren: {ProfNr, Name, Raum, Rang, PLZ, Straße}

Orte: {PLZ, Ort}

# Normalformen

- **1NF:** Ein Relationenschema ist in 1. Normalform, wenn dessen Wertebereiche atomar sind.
- **2NF:** Ein Relationenschema ist in 2. Normalform, wenn es in 1. Normalform ist und jedes Nichtschlüsselattribut voll funktional vom Primärschlüssel abhängig ist.
- **3NF:** Ein Relationenschema ist in 3. Normalform, wenn es sich in 2. Normalform befindet, und kein Nichtschlüsselattribut vom Primärschlüssel transitiv abhängig ist.

# Zur Wiederholung

- Jedes Nicht-Schlüssel-Attribut ist in einer Relation  $R$  voll funktional abhängig vom Primärschlüssel  $S$ . Welche Normalform ist gegeben?
- In einer Relation sei kein Sekundärattribut (Nicht-Schlüsselattribut) transitiv abhängig von einem Schlüsselattribut.

# Zur Wiederholung

- Jedes Nicht-Schlüssel-Attribut ist in einer Relation R voll funktional abhängig vom Primärschlüssel S. Welche Normalform ist gegeben?  
→ 1.NF und 2.NF
- In einer Relation sei kein Sekundärattribut (Nicht-Schlüsselattribut) transitiv abhängig von einem Schlüsselattribut.

# Zur Wiederholung

- Jedes Nicht-Schlüssel-Attribut ist in einer Relation R voll funktional abhängig vom Primärschlüssel S. Welche Normalform ist gegeben?  
→ 1.NF und 2.NF
- In einer Relation sei kein Sekundärattribut (Nicht-Schlüsselattribut) transitiv abhängig von einem Schlüsselattribut.  
→ 3.NF