

DATENBANKEN

Tabellen

Schweler			
S#	Name	Klasse	Freifächer
1	Franz	4C	Ch, Ma
2	Erna	6A	Ma, In
3	Hans	6B	Ch

↑
Schlüssel

↑
nicht
atomar

Schweler		
S#	Name	Klasse
1	Franz	4C
2	Erna	6A
3	Hans	6B

→ voll
funktional
abhängig

↑
zweite Normalform

Freifachwahl	
S#	Freifach
1	Ch
1	Ma
2	Ma
2	In
3	Ch

↖ zwei Schlüssel
= voll funktional
abhängig

Schweler		
S#	Name	Klasse
1	Franz	4C
2	Erna	6A
3	Hans	6A

Klassen	
Klasse	Lehrer
4C	Bach
6A	Felder

↑
3. Normalform

erste Normalform:

alle Werte der Merkmale atomar

Schweler			
S#	Freifächer	Name	Klasse
1	Ch	Franz	4C
1	Ma	Franz	4C
2	Ma	Erna	6A
2	In	Erna	6A
3	Ch	Hans	6B

↑
atomar
1. Normalform

↑
redundante
Informationen

dritte Normalform:

Schweler			
S#	Name	Klasse	Lehrer
1	Franz	4C	Bach
2	Erna	6A	Felder
3	Hans	6A	Felder

↖ nicht 3. Normalform

↑
redundant

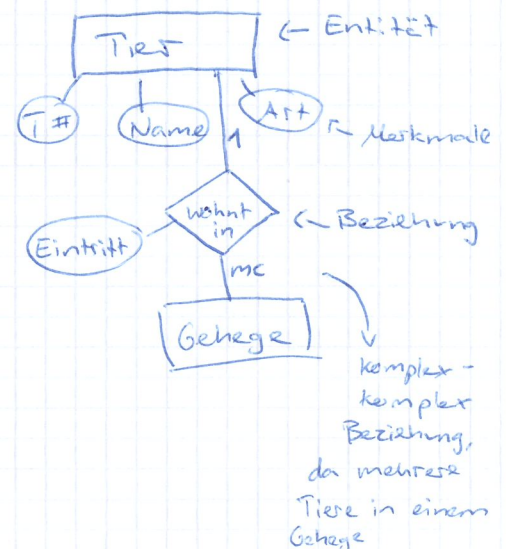
Entitäten - Beziehungsmodell

Entität: ein eindeutig bestimmtes Objekt.

Beziehung: zwischen zwei Entitäten

Assoziationstyp:

- 1 → genau ein
- c → kein oder ein
- m → ein oder mehrere
- mc → kein, ein oder mehrere



SQL

SELECT Merkmale | * \rightarrow alle Merkmale einer Tabelle | DISTINCT keine Replikate
FROM Tabelle
WHERE Bedingungen
ORDER BY sortieren | ASC \leftarrow aufsteigend DESC \leftarrow absteigend

Komplex-komplex: GROUP BY gruppieren

Beispiele

- o SELECT nachname FROM autor WHERE nachname IN ('Maier', 'Meier', 'Müller');
Autoren Nachnamen
- o SELECT MIN(jahr) FROM buch
älteste Buch
- o SELECT COUNT(*) FROM verlag
Anzahl Verlage
- o SELECT COUNT(*) FROM autor WHERE zusatz IS NOT NULL;
Autoren mit Zusatz
- o SELECT MAX(preis) FROM buch WHERE titel LIKE '%Algorithmen%';
teuerstes Buch über Algorithmen

komplexere Anfragen:

- o SELECT titel, preis AS "Durchschnittspreis" FROM buch GROUP BY preis, titel, waehrung HAVING titel LIKE '%Algorithmen%' AND preis > (SELECT AVG(preis) FROM buch WHERE waehrung IS NULL) AND waehrung IS NULL
Anzahl Algorithmen-Bücher über Durchschnittspreis
- o SELECT name, jahr, COUNT(buchid) FROM verlag v, buch b WHERE b.verlagsid = v.verlagsid AND jahr IS NOT NULL GROUP BY name, jahr ORDER BY COUNT(buchid) DESC
Anzahl Bücher pro Verlag pro Jahr