25. Relationenalgebra, Modellierung, ER - Diagramme

25.1 Abfragen – Theorie

Aus einer Tabelle entsteht durch

Selektion

`WHERE auflage = „2006“`

Band				
band_id	ean13	titel	verfasser	auflage
120	9783540852018	Kleines 1 x 1 der Relativitätstheorie	Gottfried Beyvers	2009
121	9783827417299	Systematik-Poster: Botanik	Andreas Bresinsky	2006
122	9783134843101	Chemie : das Basiswissen der Chemie	Charles E. Mortimer	2010
123	9783133685177	Pharmakologie und Toxikologie	Heinz Lüllmann ; Klaus Mohr ; Lutz Hein	2010
124	9783827416766	Mikroskopische Technik	Maria Mulisch	2010
125	9783527315796	Chemie - einfach alles	Peter W. Atkins	2006
126	9783827418425	Exkursionsflora von Deutschland	Eckehart J. Jäger	2007
127	9783110185317	Biologie der Pflanzen	Peter H. Raven	2006
128	9783133674249	Zoologie	Rüdiger Wehner ; Walter Gehring	2007
129	9783827418432	Genome und Gene : Lehrbuch der molekularen Genetik	T. A. Brown	2007
130	9783804723429	Anatomie, Physiologie, Pathophysiologie des Menschen	Ernst Mutschler ; Hans-Georg Schaible ; Peter Vaupel	2007
131	9783827372314	Botanik	Murray W. Nabors	2007

und **Projektion**

eine **neue Tabelle**

titel	verfasser	auflage
Systematik-Poster: Botanik	Andreas Bresinsky	2006
Chemie - einfach alles	Peter W. Atkins	2006
Biologie der Pflanzen	Peter H. Raven	2006

Diese Tabelle ist nicht in der Datenbasis gespeichert; sie existiert nur so lange, wie die Abfrage geöffnet ist.

Bei einer zweiten Abfrage mit gleicher Projektion
(`WHERE titel like "Chemie*"`) entsteht eine weitere Tabelle

titel	verfasser	auflage
Chemie : das Basiswissen der Chemie	Charles E. Mortimer	2010
Chemie - einfach alles	Peter W. Atkins	2006

Die **Vereinigung** dieser beiden Tabellen

entspricht dem logischen "Oder"

```
SELECT titel, verfasser, auflage
FROM Band
WHERE (auflage =„2006") OR (titel like "Chemie*")
```

titel	verfasser	auflage
Systematik-Poster: Botanik	Andreas Bresinsky	2006
Chemie : das Basiswissen der Chemie	Charles E. Mortimer	2010
Chemie - einfach alles	Peter W. Atkins	2006
Biologie der Pflanzen	Peter H. Raven	2006

Die **Differenz** der Tabellen

entspricht logisch einem "Und Nicht"

```
SELECT titel, verfasser, auflage
FROM Band
WHERE (auflage =„2006") AND NOT (titel like "Chemie*")
```

titel	verfasser	auflage
Systematik-Poster: Botanik	Andreas Bresinsky	2006
Biologie der Pflanzen	Peter H. Raven	2006

Wenn man mit einem DBMS arbeitet, wird man wichtige Abfragen unter einem Namen Speichern, etwa die erste als „Bücher_aus_Stuttgart".

Wenn man stattdessen geschachtelte SQL – Anweisungen verwenden will, benutzt man einen Alias
(**Umbenennung**)

```
(SELECT titel, verfasser, auflage FROM Band WHERE auflage = „2006") AS Baende2006
```

25.2 Weiterführung der Bücherei - Datenbank

Modellierung bedeutet ja immer, **zu einem bestimmten Zweck** einen Ausschnitt der Realwelt zu bilden.

Welchen Zweck verfolgen wir mit einem solchen Bücherei – Programm?
Sicher nicht nur die Verwaltung der Bücher.

Wir wollen auch wissen,

- wer welches Buch gerade entliehen hat,
- wo die Entleihfrist überschritten ist,
- welche Bücher beliebt sind, welche nicht,
- ob ein Leser viele Bücher entliehen hat
- …

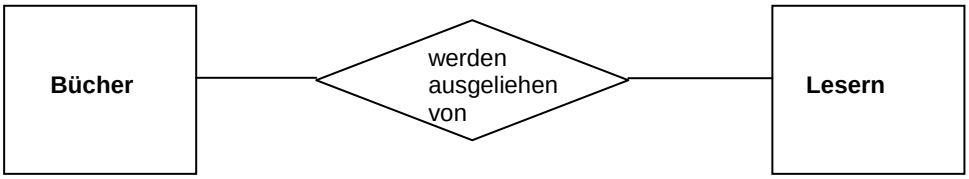
Außerdem sollten wir Mahnungen schreiben können an Leser, die die Bücher noch nicht zurückgegeben haben.

Dazu müssen wir Daten der Leser und Bücher in Beziehung setzen.

25.3 Beziehung zwischen den Tabellen

Die Bücher werden ausgeliehen, und zwar von den Lesern.

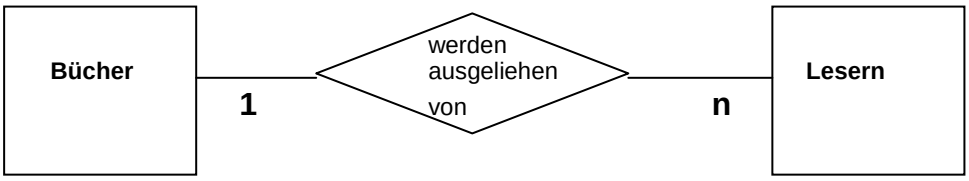
Notation dieser Beziehung als Entity – Relationship – Diagramm (ER – Diagramm):



25.4 Komplexität / Art der Beziehung

Für die Auswertung der „beliebten Bücher“ wäre es nötig, dass jedes Buch von mehreren Lesern ausgeliehen werden kann (natürlich nacheinander). In einem ER – Diagramm notiert man das folgendermaßen:

1 : n - Relation



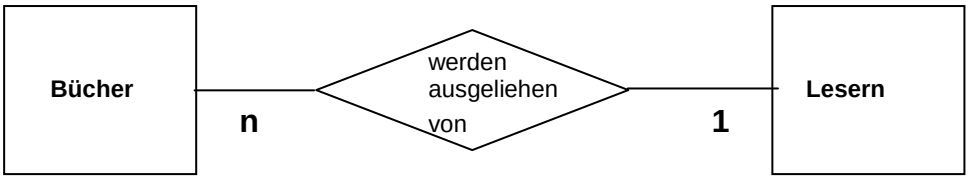
n steht dabei für eine beliebige Anzahl (Access verwendet das ∞ - Zeichen).
1 und n nennt man **Kardinalitäten**.

Realisierung der 1:n – Beziehung mit den vorhanden Tabellen:

Die Lesertabelle bekommt ein neues Feld: „Ausgeliehenes Buch“; die Buchnummer notieren wir bei den Lesern, die dieses Buch ausgeliehen haben.

Bei einer solchen 1 . n - Relation kann aber **jeder** Leser **nur ein** Buch entleihen.

Wenn wir den Aspekt „Jeder Leser kann mehrere Bücher entleihen“ modellieren wollen, müssen wir Kardinalitäten umdrehen:



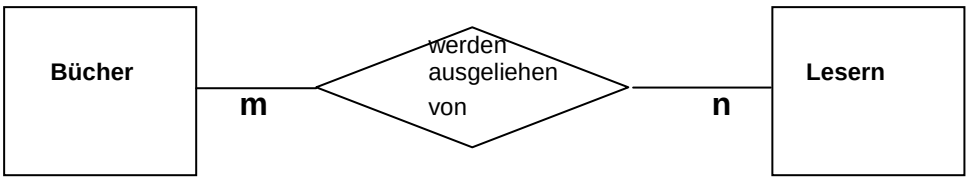
Realisierung der 1:n – Beziehung:

Die Büchertabelle bekommt ein neues Feld: „Leser“; die Lesernummer notieren wir bei den Büchern, die dieser Leser ausgeliehen hat.

Jetzt kann aber jedes Buch nur max. einen Leser haben …

Die Lösung liegt in einer Kombination der beiden Beziehungsarten, einer m.n - Relation

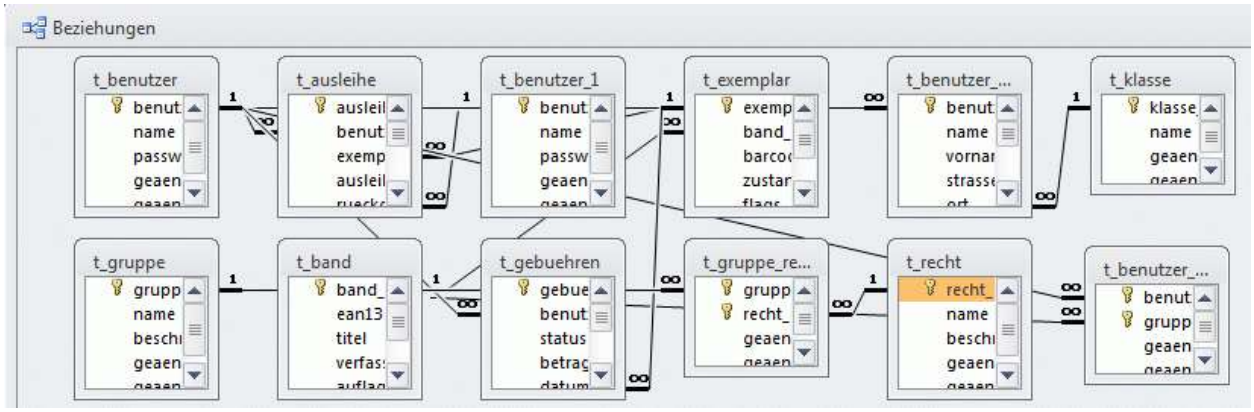
m : n - Relation



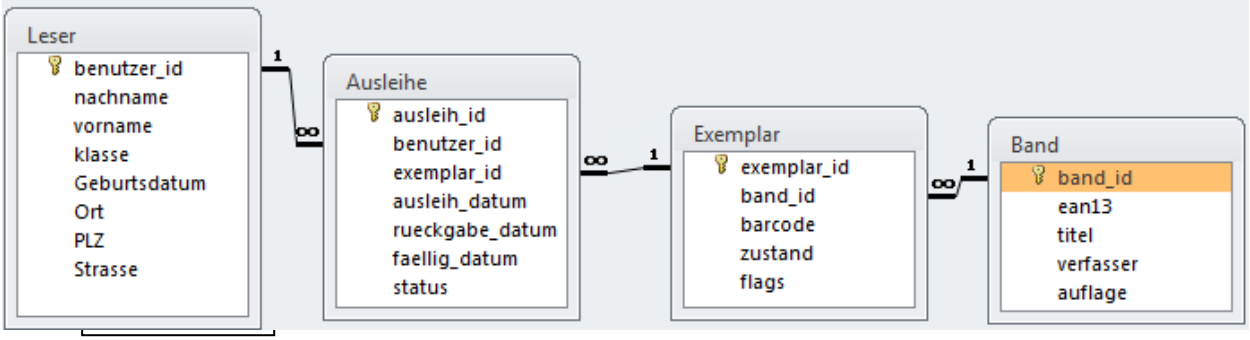
Eine Realisierung durch einfaches Hinzufügen eines Feldes in der Leser- oder Büchertabelle ist jetzt nicht mehr möglich.

m:n – Beziehungen werden durch Beziehungstabellen realisiert.

Das zugrunde liegende Beispiel einer Schulbuchausleihe verwendet folgende Tabellen und Beziehungen:

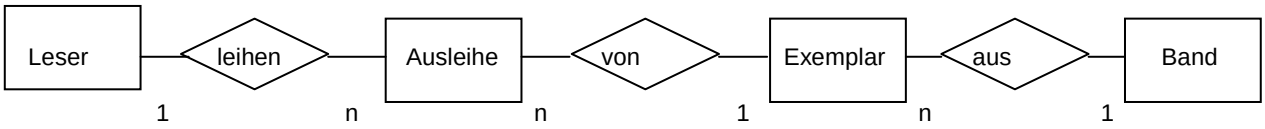


Wir betrachten hier die vereinfachte Form:



Weil es von den Schulbüchern Klassensätze gibt, muss man noch zwischen dem einzelnen „Exemplar“ und dem Buch (hier „Band“) unterscheiden.

Das ER – Diagramm (ohne Attribute) dazu:



In der **Ausleihtabelle als Beziehungstabelle** notieren wir eigentlich nur die Daten der Beziehung:
Welcher Leser hat welches Buch ausgeliehen, also benutzer_id und exemplar_id.

benutzer_id	exemplar_id
1079	77
1085	91
871	1862
734	1975
750	77
754	91
730	76
751	89
742	83
752	78
741	98
753	94

In unserem speziellen Fall der Ausleihtabelle sind weitere Angaben sinnvoll.

Es handelt sich dann nicht mehr um eine reine Beziehungstabelle, sondern um eine weitere Entität, die den Ausleihvorgang modelliert mit den zusätzlichen (sinnvollen) Attributen „ausleih_datum“, „rueckgabe_datum“ und „faellig_datum“.

Ausleihe						
ausleih_id	benutzer_id	exemplar_id	ausleih_datum	rueckgabe_datum	faellig_datum	status
1554	1079	77	20110314090216		20120313000000	1
1555	1085	91	20110314090241		20120313000000	1
1556	871	1862	20110317094724		20120316000000	1
1557	734	1975	20110321085159		20120320000000	1
1558	750	77	20110321085224		20120320000000	1
1559	754	91	20110321085234		20120320000000	1
1560	730	76	20110321085244		20120320000000	1
1561	751	89	20110321085258		20120320000000	1
1562	742	83	20110321085309		20120320000000	1
1563	752	78	20110321085321		20120320000000	1
1564	741	98	20110321085331		20120320000000	1
1565	753	94	20110321085341		20120320000000	1

25.5 Theorie der Beziehungen

In einer relationalen Datenbank werden (theoretisch) nicht nur die Entities, sondern auch **alle Beziehungen** zwischen Entities als **Tabellen** (Relationen) gesehen.

Ohne Berücksichtigung der Besonderheiten der Ausleihe und der Kardinalitäten besteht eine Relation „Ausleihe“ aus Schlüsselpaaren wie im Beispiel nebenan:

benutzer_id	exemplar_id
1079	77
1085	91
871	1862
734	1975
750	77
754	91
730	76
751	89
742	83
752	78
741	98
753	94

kartesisches Produkt

Bei einer SQL – Abfrage über zwei Tabellen ohne einschränkende Bedingung steht **jede** Entity der ersten Tabelle mit **jeder** Entity der zweiten Tabelle in Beziehung.

```
SELECT Leser.nachname, Ausleihe.ausleih_datum
FROM Leser, Ausleihe
```

Wie viele Datensätze sind das dann?

Für jeden Lesern mit jedem Ausleihvorgang eine eigene Zeile …

Das ergibt 1439 * 2888 = 4 155 832 Zeilen

Das sind über 4 Millionen, bei noch größeren Datenmengen übersteigt das alle physischen Speichermöglichkeiten.

Und es ist sinnlos!

Bei „Schmitz“ werden nicht nur die Daten zu den eigenen Entleihvorgängen angezeigt, sondern sein Name wird auch mit den Daten aller anderen Entleihvorgänge „gemischt“.

Es sollten hier nur die Entleihdatensätze und Leserdatensätze berücksichtigt werden, bei denen die benutzer_id übereinstimmt.!

Dies sind dann wieder etwa 3000 (= Anzahl der Entleihdatensätze), weil zu jedem Entleihvorgang ein Leser gehört (aber nicht umgekehrt).

Nachname	**ausgeliehen**
Schmitz	05.01.1999
Schmitz	05.01.1999
Schmitz	05.01.1999
Schmitz	07.01.1999
Schmitz	07.01.1999
Schmitz	07.01.1999
Schmitz	10.01.1999
Schmitz	11.01.1999
Schmitz	11.01.1999
Schmitz	11.01.1999
……..	…weitere 300 Zeilen mit Schmitz
Müller	05.01.1999
Müller	05.01.1999
Müller	05.01.1999
Müller	07.01.1999
Müller	07.01.1999
Müller	07.01.1999
Müller	10.01.1999
Müller	11.01.1999
Müller	11.01.1999
Müller	11.01.1999
…….	…weitere 300 Zeilen mit Müller
	… und das selbe mit allen weiteren Lesern

(Inner) Join

Die Lösung:
Es wird von vorneherein mit Hilfe einer Bedingung gesucht – meistens Übereinstimmung von Feldinhalten (Äqui – Join) – die den Umfang der Ergebnisrelation in Grenzen hält.

```
SELECT Leser.Nachname, Ausleihe.ausgeliehen
FROM Leser INNER JOIN Ausleihe ON Leser.LeserNr = Ausleihe.LeserNr
```

Die SQL-Anweisung

```
SELECT Leser.Nachname, Ausleihe.ausgeliehen
FROM Leser, Ausleihe
WHERE Leser.LeserNr = Ausleihe.LeserNr
```

wird übrigens von den DBMS genauso umgesetzt wie der JOIN, ist also in der Praxis gleichwertig.

Weil die m:n – Beziehungen durch eine (hier notwendige) Beziehungstabelle in zwei 1:n – Beziehungen aufgeteilt werden, sind die praktisch realisierten Beziehungen fast alle vom Typ 1:n. Beim Join über eine solche 1:n – Beziehung entscheidet die Seite, auf der das n steht, über die Zahl der Datensätze im Join; diese Tabelle wird auch Detailtabelle genannt. Auf der 1 – Seite ist das verwendete Attribut hingegen durchweg der Primärschlüssel.

(LEFT / RIGHT) JOIN / auch: Outer Join

Wenn man feststellen will, welche Bücher noch nie ausgeliehen wurden, hilft der Join über die Entleihvorgänge nicht.

Bei einem **Left Join** werden aus der zuerst genannten Tabelle (hier Bücher) **alle** Datensätze berücksichtig

```
SELECT [Band].titel, [Band].band_id, Exemplar.zustand, Exemplar.exemplar_id,
Ausleihe.ausleih_id, Ausleihe.ausleih_datum

FROM [Band] INNER JOIN (Exemplar LEFT JOIN Ausleihe ON Exemplar.exemplar_id =
Ausleihe.exemplar_id) ON [Band].band_id = Exemplar.band_id;
```

Das Ergebnis zeigt in den entsprechenden Zeilen keine Ausleihdaten an.

titel	band_id	zustand	exemplar_id	ausleih_id	ausleih_datum
Biologie - Lehrbuch für die Oberstufe	19	0	2315	2851	20110927132920
Biologie - Lehrbuch für die Oberstufe	19	0	2316	2346	20110914132955
Biologie - Lehrbuch für die Oberstufe	19	0	2317		
Biologie - Lehrbuch für die Oberstufe	19	0	2344		
Biologie - Lehrbuch für die Oberstufe	19	0	2346		
Biologie - Lehrbuch für die Oberstufe	19	0	2396	2402	20110914134255
Biologie - Lehrbuch für die Oberstufe	19	0	2946	2718	20110923114539
Biologie - Lehrbuch für die Oberstufe	19	0	2947		
Pons Kompaktwörterbuch Spanisch	20	0	220		
Pons Kompaktwörterbuch Spanisch	20	0	221		
Pons Kompaktwörterbuch Spanisch	20	0	222		

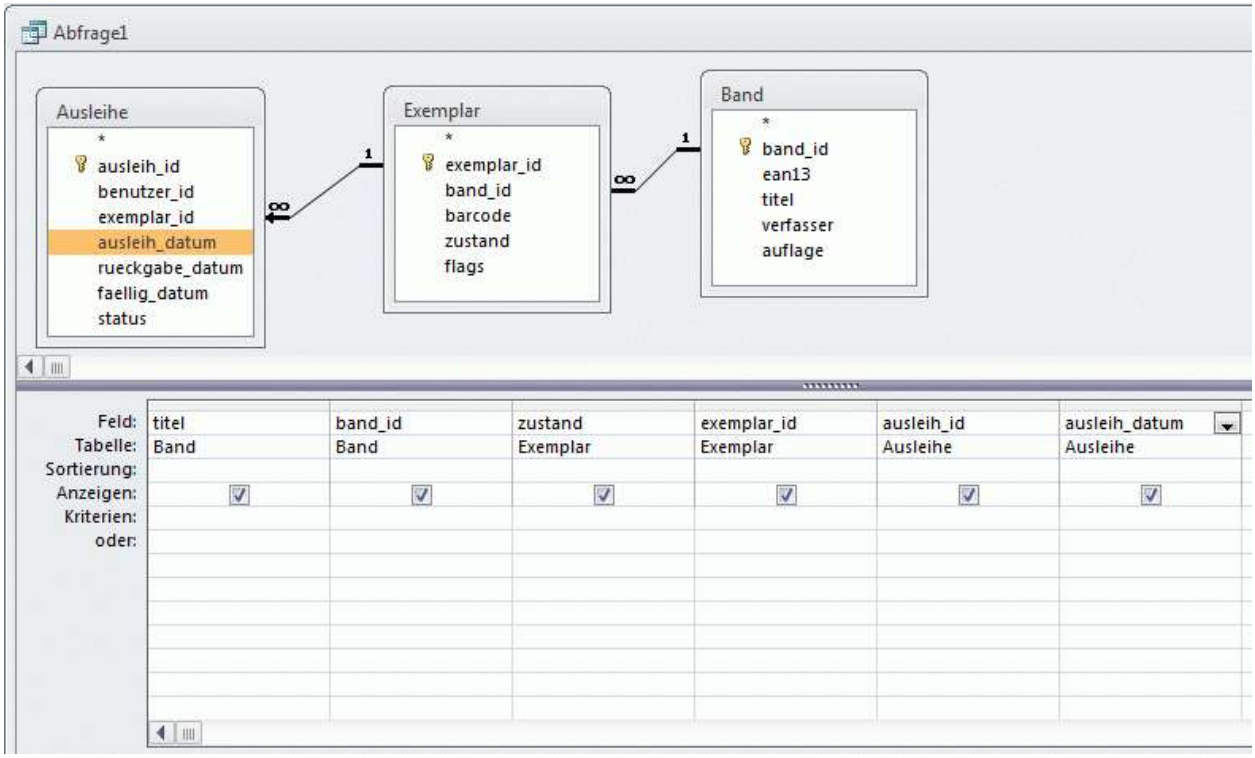
Mit dem Zusatz
```
WHERE Ausleihe.ausleih_id Is Null;
```

findet man dann die noch nicht ausgeliehenen Exemplare.

25.6 Abfragen über mehrere (verbundene) Tabellen (Handling)

Die zwischen den Tabellen bestehenden Beziehungen werden automatisch berücksichtigt, wenn wir eine Abfrage mit Daten aus mehreren Tabellen erstellen
Hier für den „Left Join“ von oben

(Datenbankfenster → Abfragen → Neu → Entwurfsansicht)



Abfrage: Ausgeliehene Exemplare

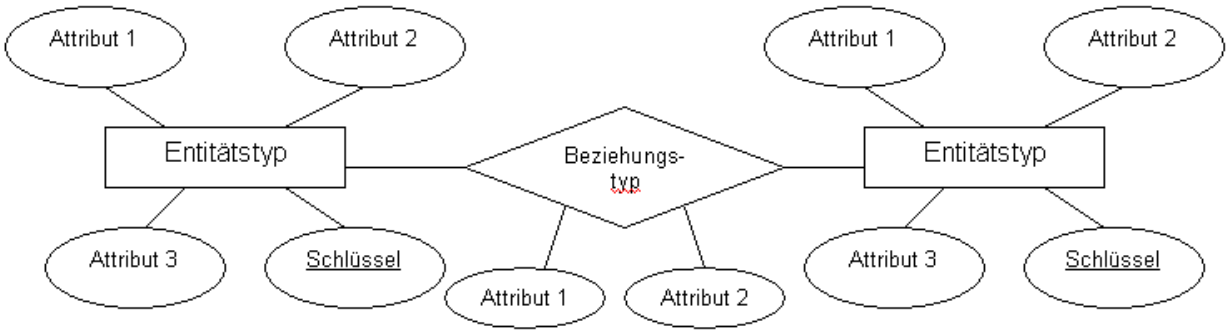
Die angezeigte Abfrage „Ausgeliehene Bücher“ zeigt alle Buchexemplare an mit dem Titel aus dem zugehörigen (gleiche band_id) „Band“. Wenn ein Ausleihe – Datensatz mit gleicher „exemplar_id“ existiert, werden auch die Ausleihdaten angegeben.

In SQL:

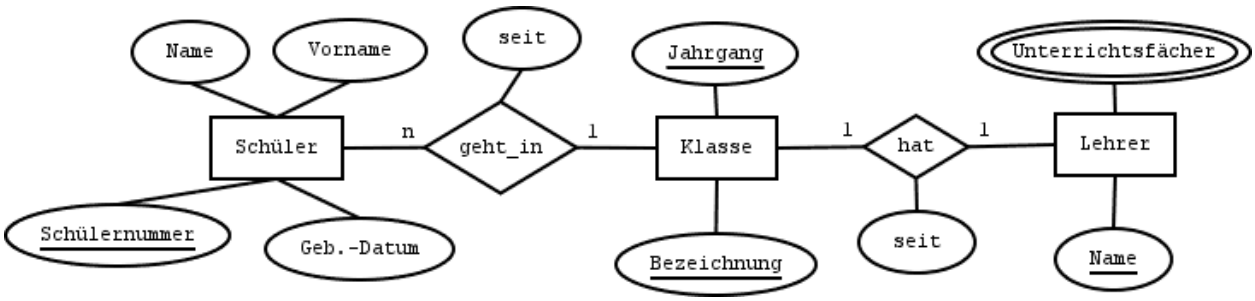
```
SELECT [Band].titel, [Band].band_id, Exemplar.zustand, Exemplar.exemplar_id,
Ausleihe.ausleih_id, Ausleihe.ausleih_datum

FROM [Band] INNER JOIN (Exemplar LEFT JOIN Ausleihe ON Exemplar.exemplar_id =
Ausleihe.exemplar_id) ON [Band].band_id = Exemplar.band_id;
```

25.7 ER - Diagramme

allgemeine Form:



Aufgabe:
Hier ein ER – Diagramm für ein nahe liegendes Beispiel (Schule)



Was wird hier implizit vorausgesetzt ?
Geben Sie ein sinnvolleres ER – Modell an!

Aufgabe:

Die „Ceylon – Direkthilfe“ verkauft Tee und Grußkarten.

Erstellen Sie zu den Daten auf der nächsten Seite

- ein ER – Diagramm und
- ein Access – Relationendiagramm

unter Berücksichtigung der Normalformen.

Name	Vorname	Straße	PLZ	Ort	Nr	Versandkosten	Artikelbez	EPreis	Anzahl	GPreis
Thebud	Marlies	Nymphenweg 31	D-42719	Solingen	1	7,00 €	besondere Artikel	12,00 €	4	48,00 €
Thebud	Marlies	Nymphenweg 31	D-42719	Solingen	1	7,00 €	Satz handgefertigter Ceylon - Grußkarten	30,00 €	2	60,00 €
Thebud	Marlies	Nymphenweg 31	D-42719	Solingen	1	7,00 €	Packung Ceylon - Tee à 100g	7,00 €	7	49,00 €
Thebud	Marlies	Nymphenweg 31	D-42719	Solingen	1	7,00 €	Packung Ceylon - Tee à 100g	6,00 €	5	30,00 €
Beier	Christian	Katernberge 215	D-07819	Pillingsdorf	2	5,00 €	Satz handgefertigter Ceylon - Grußkarten	30,00 €	1	30,00 €
Beier	Christian	Katernberge 215	D-07819	Pillingsdorf	2	5,00 €	Packung Ceylon - Tee à 100g	6,00 €	2	12,00 €
Grabitz	Ulf	Lohmühler Weg 55	D-07819	Pillingsdorf	3	0,00 €	besondere Artikel	7,00 €	2	14,00 €
Weber	Astrid + Harald	Kiefernweg 1	D-80335	München	7	4,00 €	Satz handgefertigter Ceylon - Grußkarten	30,00 €	2	60,00 €
Thebud	Marlies	Nymphenweg 31	D-42719	Solingen	8	6,00 €	Packung Bio - Tee im Holzkästchen	8,00 €	2	16,00 €
Thebud	Marlies	Nymphenweg 31	D-42719	Solingen	8	6,00 €	Satz handgefertigter Ceylon - Grußkarten	30,00 €	3	90,00 €
Hellberg	Dr. Olaf	Wittkuller Str. 159	D-80335	München	9	7,00 €	Packung Ceylon - Tee à 100 g	6,00 €	2	12,00 €
Hellberg	Dr. Olaf	Wittkuller Str. 159	D-80335	München	9	7,00 €	Packung Bio - Tee im Holzkästchen	8,00 €	2	16,00 €

26 Normalisierung von Datenbanken

Eine sehr guter Lehrtext steht im Netz unter
www.tinohempel.de/info/info/datenbank/normalisierung.htm

27 Aufgabenbeispiele

Aufgabe 1:

In einem Krankenhaus werden Daten über Patienten, deren An- und Abmeldedatum in Stationen des Krankenhauses und die einzelnen Behandlungen abgespeichert.

1) Patienten haben Namen, ein Geschlecht, ein Geburtsdatum, eine Krankenkasse und eine Versicherungsnummer,
2) Eine Station hat eine Bezeichnung und einen Chefarzt und verfügt über eine Anzahl von Betten, die durchnumeriert sind.
3) Ein Patient hat immer ein An- und Abmeldedatum,
4) Behandlungen sind durch ihren Namen, ihre Dauer und den Patientennamen charakterisiert.

Erstellen Sie ein ER-Diagramm in dem Sie auch die Komplexitäten der Beziehungen angeben.

Aufgabe 2:

Eine Datenbank eines Tanzsportvereins enthält folgende Daten:

1) Tanzkurse

Kurs-Nr	Tanzkurs	Kursleiter	Kursstunden
A1	Standard 1	Müller	20
A2	Standard 2	Müller	20
B1	Step	Grimm	10
C1	Moderne Tänze	Sommer	15
C2	Rock'n Roll	Sauer	15
D1	Bewegungstherapie	Grimm	20

2) Mitglieder

Mitglieds-Nr	Name	Vorname	Gruppe
1	Kaufmann	Patricia	A
2	Schmidt	Barbara	A
3	Braun	Oliver	C
4	Kiefer	Bernd	D
5	Schmidt	Barbara	B
6	Hansen	Gertrud	A
7	Blume	Simon	B
8	Kraus	Margit	C

3) Kursbelegung

Kurs-Nr	Mitglieds-Nr	Tanzleistung
A1	3	gut
A1	4	sehr gut
A1	5	sehr gut
A1	8	befriedigend
A2	1	gut
A2	3	sehr gut
A2	7	gut
B1	2	sehr gut
B1	3	sehr gut
B1	6	befriedigend
C1	4	befriedigend
C1	5	gut
C2	1	gut
C2	2	gut
C2	3	befriedigend
C2	4	gut
C2	5	sehr gut

D1	7	gut
D1	8	gut

a) Geben sie ein Datenbankschema an

b) Beschreiben Sie in Worten das Ergebnis folgender Abfragen und geben Sie das Ergebnis der Abfragen an:

b1)
```
SELECT Tanzkurs, Kursstunden
FROM Tanzkurse
WHERE Kursleiter = „Grimm“;
```

b2)
```
SELECT Mitglieder.Name, Mitglieder.Vorname
FROM Tanzkurse INNER JOIN (Mitglieder INNER JOIN Kursbelegung ON Mitglieder.MitgliedsNr =
Kursbelegung.MitgliedsNr) ON TanzKurse.KursNr = Kursbelegung.KursNr)
WHERE Tanzkurse.Tanzkurs = „Step“
ORDER BY Mitglieder.Name;
```

b3)
```
SELECT KursNr, Tanzleistung, COUNT(Tanzleistung) AS AnzahlNote
FROM Kursbelegung
GROUP BY Tanzleistung;
```

c) Erstellen Sie die folgenden Abfragen in SQL:

c1) Geben Sie Name und Vorname aller Mitglieder aus, die in Gruppe A sind.
c2) Geben Sie die Nummern aller Mitglieder aus, die am Kurs B1 teilnehmen und deren Tanzleistung „sehr gut“ ist.
c3) Geben Sie Bezeichnung und Kursleiter der Tanzkurse an, an denen Barbara Schmidt teilnimmt.
c4) Geben Sie Namen und Vornamen aller „Rock’n Roller“ an.

Aufgabe 3:

In einem Büro sind folgende Daten zu speichern:

- Artikel mit Artikelnummern, Lieferantennummer, Mindestbestand, Bestellmenge und Preis pro Stück,
- Eingänge mit Nummer, Artikelnummer, Anzahl der Einheiten, und Bemerkungen und Eingangsdatum
- Lieferanten mit Lieferantennummer, Lieferantenname, Straße, Postleitzahl, Stadt und Lieferzeit
- Ausgänge mit Ausgangsnummer, Artikelnummer, Anzahl der Einheiten, Ausgangsdatum und betreuender Mitarbeiter

a) Erstellen Sie ein ER-Diagramm in dem Sie auch die Komplexitäten angeben.

b) Die Tabelle Artikel hat folgendes Aussehen.

Artikel-Nr	Artikelname	Lieferanten-Nr	Mindestbestand	Bestellmenge	Preis pro Stück
ALG-001	Bleistifte	1	20	100	0,20 EUR
ALG-002	Briefumschläge (10 Stk)	2	10	50	5,00 EUR
ALG-003	DIN A4 Papier (500 Blatt)	3	10	50	23,85 EUR
ALG-004	Post-It Notes 654	1	10	60	9,80 EUR
ALG-005	Post-It Notes 656	1	10	60	10,40 EUR
ALG-006	Rotstifte	1	20	100	0,50 EUR
ALG-007	Büroklammern (100 St.)	1	30	100	0,10 EUR
EDV-001	Bildtrommeln	3	3	5	249,00 EUR
EDV-002	Toner	3	10	20	85,90 EUR
EDV-003	Disketten (3,5)	3	100	1000	1,20 EUR
EDV-004	Laser Labels 6590 (25 Blatt)	3	5	10	35,90 EUR
EDV-006	Drucker-Tinte	3	4	10	37,40 EUR

Erstellen Sie folgende Abfragen:

b1) Es sollen alle Artikel des Lieferanten mit der Nummer 1 ausgegeben werden.

b2) Gesucht sind alle Artikel mit einem Preis von unter 10 EUR und einer Bestellmenge von mindestens 100 Stück.

c) Die Tabelle Personal enthält folgende Daten:

Personal Nr	Name	Vorname	Tätigkeit	Abteilung	Gehalt	Eintrittsdatum
1	Lorenz	Sophia	Vorbereiter	Gruppe A	2.300,00 EUR	01.01.90
2	Hohl-Kohl	Tatjana	Zubereiter	Gruppe C	4.305,50 EUR	01.03.94
3	Willschrein	Theodor	Vorbereiter	Gruppe D	2.455,55 EUR	15.04.92
4	Richter	Hans-Otto	Gruppenleiter	Gruppe E	6.300,00 EUR	01.10.94
5	Wiesenland	Brunshilde	Gruppenleiter	Gruppe C	6.980,00 EUR	01.01.95
6	Morgenrot	Bettina	Vorbereiter	Gruppe E	2.299,90 EUR	01.01.95
7	Tapfer	Karl-Heinz	Fahrer	Transport	3.700,00 EUR	08.08.93
8	Zottelhagen	Friedhelm	Personalreferent	Personal	6.800,00 EUR	24.07.90
9	Siebental	Wilfried	Zubereiter	Gruppe A	3.980,40 EUR	01.01.91
10	Sappel-Schmalz	Claudia-Maria	Gruppenleiter	Gruppe A	6.145,00 EUR	20.06.92
11	Hölzer	Richard	Vorbereiter	Gruppe C	2.890,10 EUR	01.01.95
12	Kampheimer	Klaus-Dieter	Gruppenleiter	Gruppe B	7.100,40 EUR	01.12.94
13	Schlotterbeck	Rafaela	Vorbereiter	Gruppe A	2.120,20 EUR	15.06.94
14	Collmar-Schmidt	Nadine	Personalreferent	Personal	7.200,00 EUR	15.04.90
15	Humpe	Sybille	Zubereiter	Gruppe D	4.506,60 EUR	01.11.93
16	Hellmeister	Sepp	Vorbereiter	Gruppe B	2.140,20 EUR	20.09.92
17	Braunbart	Helge	Zubereiter	Gruppe E	4.310,00 EUR	13.02.91
18	Kaufmann	Sonja	Einkäufer	Einkauf	3.800,00 EUR	01.08.91
19	Hoffmann	Theresa	Gruppenleiter	Gruppe D	6.533,30 EUR	01.02.95
20	Klinker	Ralf-Rüdiger	Fahrer	Transport	3.750,00 EUR	07.02.95
21	Saubermann	Konrad	Zubereiter	Gruppe B	4.688,30 EUR	03.08.90
22	Morgenrot	Anna-Marie	Buchhalter	Rechnungswesen	4.100,00 EUR	20.11.90
23	Schnarrenberger	Ludwig	Einkäufer	Einkauf	3.500,50 EUR	01.09.92
24	Lauterbach	Wilma	Zubereiter	Gruppe A	4.820,00 EUR	01.05.91
25	Knochenbrech	Heinz-Heribert	Hausmeister	Allgemeine Verwaltung	2.560,00 EUR	01.05.95
32	Hausmann-Lüdenscheid	Klara Sophie	Putzhilfe	Allgemeine Verwaltung	2.140,00 EUR	01.05.99

Was ergeben folgende Abfragen? Beschreiben Sie das Ergebnis in Worten und geben das Ergebnis in Tabellenform an.

1)
```
SELECT Personal.Abteilung, SUM(Personal.Gehalt) AS SummevonGehalt, MAX(Personal.Gehalt) AS
MaxvonGehalt, MIN(Personal.Gehalt) AS MinvonGehalt
FROM Personal
GROUP BY Personal.Abteilung;
```

2)
```
SELECT Personal.PersonalNr, Personal.Name, Personal.Vorname, Personal.Abteilung
FROM Personal
WHERE (Personal.Tätigkeit="Gruppenleiter") AND (Personal.Abteilung) Like "Gruppe *")
ORDER BY Personal.Abteilung;
```

3)
```
SELECT Personal.PersonalNr, Personal.Name, Personal.Vorname
FROM Personal
WHERE Personal.Gehalt>5000;
```

Aufgabe 4:

Modellieren Sie die folgenden Situationen in ER – Diagrammen. Geben Sie jeweils die zusätzlichen Bedingungen an, die Sie in ihrem Modell voraussetzen. Notieren Sie die wichtigen Attribute und die Kardinalitäten.

a) Schüler - Klasse - Lehrer - Fach
b) Musiker - CD - Song
c) Fan - Fußballverein
d) Spieler - Mannschaft(Liga) - Verein
e) Arzt - Patient
f) Abgeordneter - Partei
g) Schauspieler - Film - Filmverleih

Aufgabe 5:

Noch ein. Aufgabe zum Thema Schule.
Wir betrachten zunächst die Situation in einer Schule der SI, und zwar statisch, also ohne Zeitfaktor.

Schüler gehen in Klassen; diese werden von Lehrern unterrichtet.

a) Entwickeln sie dazu ein ER – Diagramm mit Kardinalitäten und wichtigen Attributen.
b) Berücksichtigen sie bei der Übersetzung in ein Datenbankschema, dass 1:n – Beziehungen durch einen Fremdschlüssel realisiert werden, n:m – Beziehungen hingegen durch eine Beziehungstabelle. Kennzeichnen Sie Primärschlüssel und Fremdschlüssel!
c) Beziehen Sie die Relation Fach (Fachkuerzel, Fachname) so ein, dass die Unterrichtsverteilung ablesbar wird. (ER – Diagramm und Datenbankschema ergänzen).

Nun betrachten wir eine Stufe der SII: Schüler werden in Kursen von Lehrern unterichtet.

d) Zeichnen Sie ein ER – Diagramm (mit Attributen und Kardinalitäten) und geben Sie ein Datenbankschema an!
e) Ergänzen Sie die Datenbank so, dass Noten (Halbjahres – Endnoten) und Fehlstunden erfasst werden können (nur ein Halbjahr).
f) Wie muss man die Datenbank aus e) ergänzen, damit mehrere Halbjahre erfasst werden können?
g) Geben Sie zu jeder Relation (Tabelle) aus e) eine Tabelle mit drei Datensätzen aus dem Datenbestand Ihrer Schule an!

Aufgabe 6:

Schulorchester

Aufgabe 7:

Die Autovermietung „Dornbusch“ mit Zentrale in Solingen – Schlebusch verwaltet die Entleihdaten in einer Excel – Tabelle. Ein Auszug:

Name	Adresse	Abfahrt	Fahrzeug	Auto marke	Tarif	Rückgabe datum	Rückgabeort	Tage
Fritz Schmitz	42777 Solingen Bartstr. 7	23.11.2011	SG-DB 23	Benz	4	25.11.2011	Solingen	2
Chantal Meyer	43601 Haan Windhövel 1	23.11.2011	SG-DB 77	Wind	2	24.11.2011	Vohwinkel	1
Fritz Schmitz	42777 Solingen BartStr. 7	27.11.2011	SG-DB 23	Benz	4	29.11.2011	Solingen	2
Jacqueline Pauls	42699 Ohligs Baustr. 44	26.11.2011	SG-DB 123	Baguette	3	29.11.2011	Solingen	3
Jacqueline Pauls	43601 Haan Sudetenstr. 1	27.11.2011	SG-DB 77	Wind	2	27.11.2011	Haan	1

Es soll eine Datenbank erstellt werden.
Erstellen Sie zunächst (in mehreren Schritten) ein Datenbankschema.

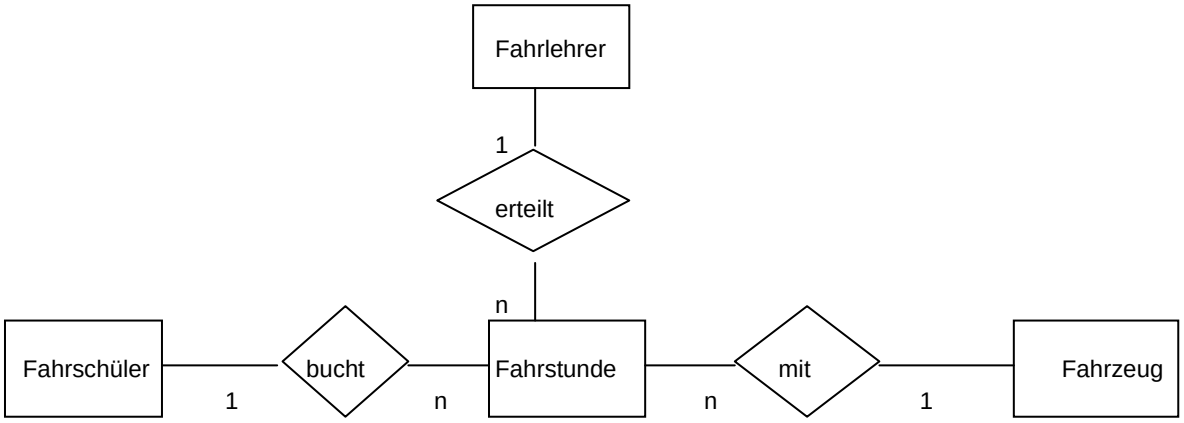
a) Der Primärschlüssel soll aus Name und Abfahrt bestehen. Erläutere, weshalb der Name hier ungeeignet ist, ergänze eine weitere Spalte, um die Person eindeutig zu machen und definiere dann einen neuen Primärschlüssel in Kombination mit Abfahrt.
b) Bringen Sie die Tabelle in die erste Normalform!
c) Wo verletzt die Tabelle die zweite Normalform? Ändern Sie das Datenbankschema, so dass die zweite Normalform erfüllt ist.
d) Dasselbe mit der dritten Normalform.
e) Bei der Mietdauer handelt es sich nicht um Grunddaten. Welche Abhängigkeit besteht?

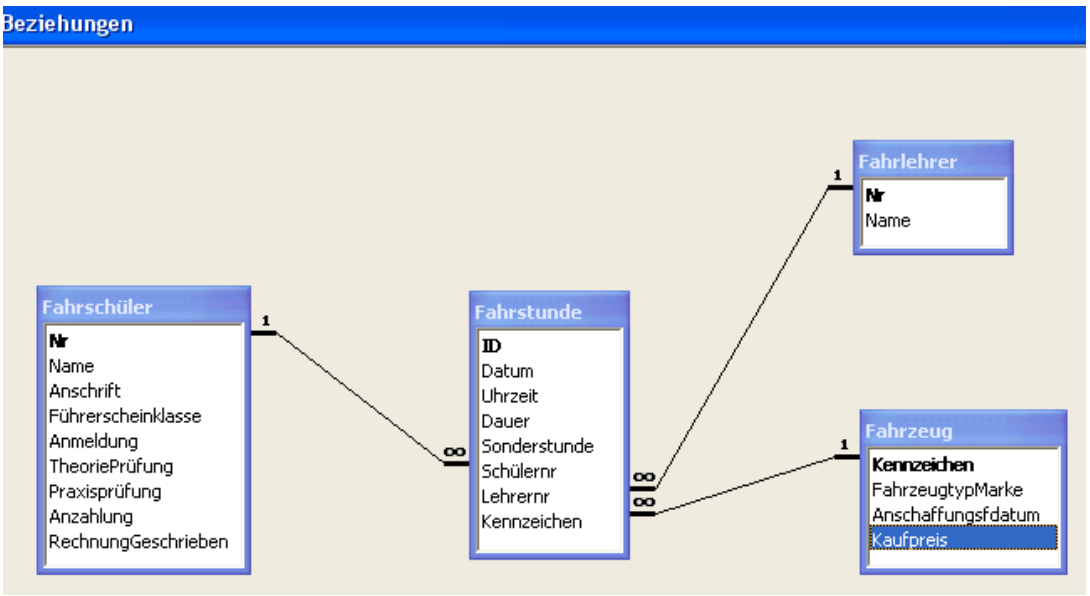
Aufgabe 8:

Durch den Zusammenschluss von fünf Fahrschulen in Solingen ist ein Unternehmen mit 11 Fahrlehrern und 15 Fahrzeugen entstanden. Um organisatorisch den Überblick über das Unternehmen zu behalten, soll die Firma Itter – Datensysteme eine Datenbank erstellen, die folgende Anforderung erfüllt:

> Zur **Terminplanung** und **Abrechnung** soll festgehalten werden, wann ein Fahrschüler eine Fahrstunde bei einem Fahrlehrer hat und welches Fahrzeug verwendet wird.

Itter – Datensysteme liefert einen Rohentwurf in Form eines ER – Diagramms und eines Beziehungsdiagramms ab.





a) Es soll zusätzlich erfasst werden, wann und über welches Autohaus die Fahrzeuge beschafft worden sind; zur Kostenkontrolle sollen die Daten der einzelnen Betankungsvorgänge (auch der Fahrlehrer, der getankt

hat) gespeichert werden.
Ergänze beide Diagramme, so dass diese Anforderungen erfüllt werden können.

b) Erläutere umgangssprachlich die folgenden SQL – Abfragen ! (Wie sind sie aufgebaut, was ist das Ergebnis ?)

```
SELECT Fahrzeug.Kennzeichen, Fahrzeug.Anschaffungsdatum, Fahrzeug.Kaufpreis
FROM Fahrzeug
WHERE (Fahrzeug.Anschaffungsfdatum)<(Date()-2*365);
```

```
SELECT Fahrlehrer.Name, Sum(Fahrstunde.Dauer) AS [Summe von Dauer]
FROM Fahrlehrer INNER JOIN Fahrstunde ON Fahrlehrer.Nr = Fahrstunde.Lehrernr
WHERE (((Fahrstunde.Datum)>Date()-30))
GROUP BY Fahrlehrer.Name;
```

```
SELECT Fahrschüler.Name, Fahrlehrer.Name, Fahrstunde.Datum, Fahrstunde.Uhrzeit,
Fahrzeug.Kennzeichen, Fahrzeug.FahrzeugtypMarke
FROM Fahrzeug INNER JOIN (Fahrschüler INNER JOIN (Fahrlehrer INNER JOIN
Fahrstunde ON Fahrlehrer.Nr = Fahrstunde.Lehrernr) ON Fahrschüler.Nr =
Fahrstunde.Schülernr) ON Fahrzeug.Kennzeichen = Fahrstunde.Kennzeichen
WHERE (((Fahrschüler.Name)="Albert"))
ORDER BY Fahrstunde.Datum, Fahrstunde.Uhrzeit;
```

c) Entwirf Abfragen, die folgende Daten liefern:

Für die Morgenbesprechung der Fahrlehrer soll eine Übersicht über die heute geplanten Fahrstunden erstellt werden (mit Uhrzeit, Name des Fahrschülers und des Fahrlehrers).

Für jeden Fahrschüler soll angezeigt werden, wie viele Fahrstunden er/sie schon genommen hat.

Welche Fahrschüler sind schon über ein Jahr angemeldet und haben noch keine Prüfung abgelegt ?

d) Die Tabelle Fahrschüler in Anlage 2 entspricht nicht der 1. und 3. Normalform. Erläutere, wieso die Normalformen verletzt sind, und stelle dar, wie man Abhilfe schaffen kann !