

## 1 Einführung

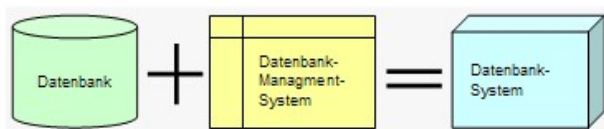
### 1.1 Was sind die Hauptaufgaben eines Datenbanksystems?

1. Daten speichern und verwalten
2. Bereitstellen von Daten in Form von Abfragen
3. Das Verändern von Daten über Manipulationsanfragen
4. Mehrbenutzersystem – Zugriffskontrolle und Sperrmechanismen

#### 1.1.1 Beispiele

- ▶ Informationssystem der Schule
  - **Daten:** Bildungsgänge, Schülerinnen und Schüler, Lehrkräfte, Räume, Prüfungen
  - **Vorgänge:** Stundenplanerstellung, Verwaltungsaufgaben, Raumplanung
- ▶ Produktionsbetrieb
  - **Daten:** Abteilungen, Mitarbeiter, Produkte, Projekte, Lieferanten
  - **Vorgänge:** Gehaltsabrechnung, Fertigung und Versand von Produkten, Lagerhaltung
- ▶ Internet-Handel
  - **Daten:** Produkte, Käufer, Verkäufer
  - **Vorgänge:** Suche nach Waren, Prüfen der Vertrauenswürdigkeit der Verkäufer, Versand der Ware

### 1.2 Datenbank, Datenbank-Management-System, Datenbanksystem



#### Datenbank

= Menge zusammengehöriger Daten

#### Datenbank-Management-System (DBMS)

= Datenverwaltungs-Programme

#### Datenbanksystem

= Datenbank + DBMS

Im Wort "Datenbank" steckt das Wort "Bank". Vieles, was wir mit diesem Wort assoziieren, trifft auf Datenbanken zu.

- ▶ Banken sind ein **sicherer** Aufbewahrungsort für unser Geld. Niemand kann dort unsere Konten einsehen, der dazu nicht befugt ist.  
*Ebenso wenig gehen Daten einer Datenbank verloren (**Datenschutz**) und sind vor unberechtigten Zugriffen geschützt. (Benutzer- und Rechteverwaltung).*
- ▶ Geld, das auf einer Bank liegt, ist **überall** schnell **verfügbar**. Es wird von der Bank professionell verwaltet. Banken erledigen die Buchhaltung für ihre Kunden, **effizient** und **korrekt**.  
*Datenbanken stellen für verschiedene Daten Verarbeitungsprozesse bereit und sind mit*

*Standard- Datenverwaltungs- Funktionen ausgestattet, die es ermöglichen die gewünschten Prozesse korrekt auszuführen.*

Unter dem Begriff Datenbank versteht man eine Menge von zusammengehörigen Daten. Wir sprechen daher besser von Datenbanksystemen. Datenbanken werden von speziellen Programmen verwaltet, den Datenbank- Management – Systemen (DBMS). DBMS und Datenbanken zusammen sind ein geschlossenes System, das man Datenbanksystem nennt.

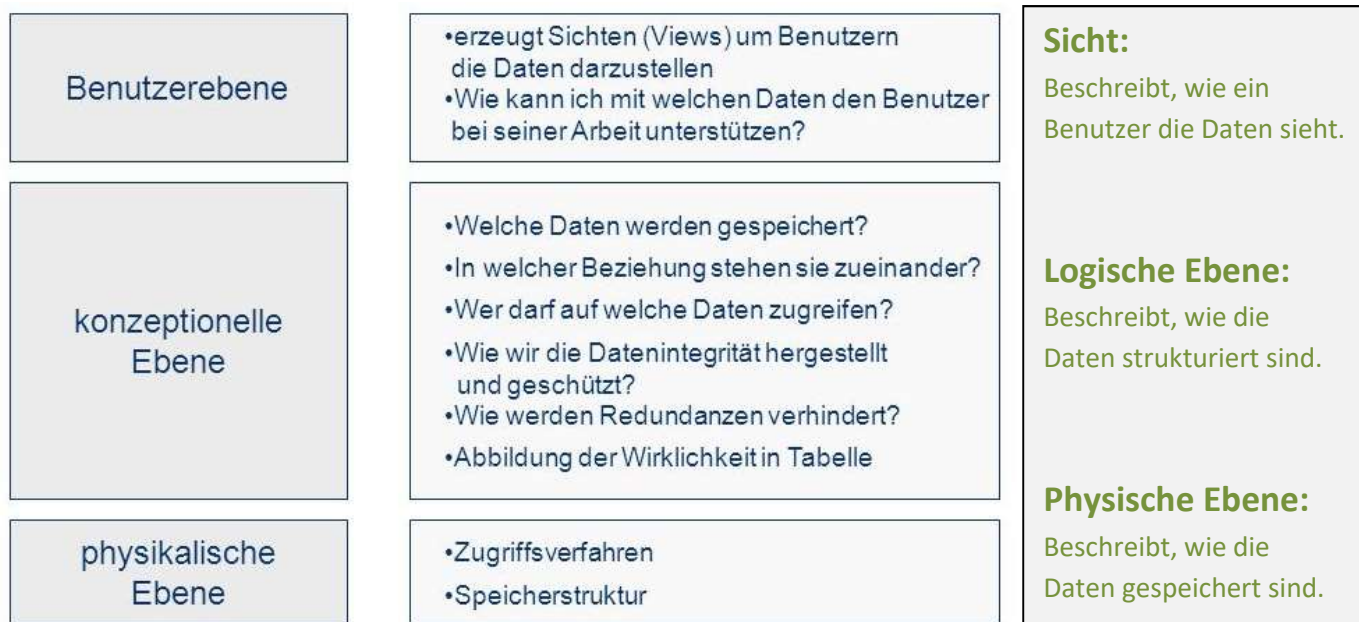
## 1.3 Arten von Daten

In einer Datenbank werden unterschiedliche Daten gespeichert.

- ▶ Stammdaten sind unveränderliche Informationen, die dauerhafte Auskunft über Objekte oder Sachverhalte geben. (Name einer Person, Bestellnummer eines Artikels, Postleitzahlen)
- ▶ Bewegungsdaten sind Informationen, die sich ständig verändern. (Saldo auf einem Konto, Körpertemperatur eines Patienten, KFZ-Kennzeichen)
- ▶ Rechendaten sind Daten, die als Berechnungsgrundlage dienen (Preise, Zinssätze, gefahrene Kilometer, Gehalt eines Mitarbeiters, Wechselkurs für eine Fremdwährung)

Anhand der Beispiele wird deutliche, dass es sich bei den gespeicherten Daten um Zahlenwerte, Texte, Daten, Bilder, Videos, usw. handeln kann.

## 1.4 Drei Schichten Architektur



### 1.4.1 Datenunabhängigkeit

Änderungen am Speichermedium, neue Zugriffspfade oder neue Platzierung der Daten haben keine Auswirkungen auf die konzeptionelle Ebene und auf die Programme, die auf die Datenbank zugreifen.

Ebenso bleiben Änderungen in der konzeptionellen Ebene (Neuzuordnung von Attributen, Datentypen) der Benutzerebene verborgen. Eventuell muss der Anwendungsadministrator die Sichten anpassen.

## 1.5 Datenintegrität

Datenintegrität ist ein Begriff für die Qualität und Zuverlässigkeit von Daten eines Datenbanksystems. Im weiteren Sinne zählt zur Integrität auch der Schutz der Datenbank vor unberechtigtem Zugriff und Veränderungen. In einer Datenbank dürfen keine widersprüchlichen Aussagen stehen.

Beispiel:

Wohnstrassen	
Strassenname	Fahrverbot
Lindenstrasse	Nein
Rosenstrasse	Ja
Gartenstrasse	19.00–7.00
...	

Strassenplanung		
Strassenname	Art	Einschränkungen
Hauptstrasse	Kantonsstrasse	keine
Gartenstrasse	Quartierstrasse	keine
Jägerweg	Quartierweg	Fussweg
...		

## 1.6 Transaktionskonzept

Eine Transaktion ist eine in sich abgeschlossene Aktivität innerhalb eines Datenbanksystems. Sie besteht aus einer festgelegten Folge von Einzeloperationen.

### 1.6.1 Beispiel

Überweisung zwischen zwei Kreditinstituten. Vom Kreditinstitut A werden 100€ abgebucht und diese 100€ sollen Kreditinstitut B gutgeschrieben werden.

*Nachdem die 100€ abgebucht werden, kommt es zu einem Systemausfall. Die Operation "gutschreiben" kann nicht mehr ausgeführt werden. !?*

### 1.6.2 ACID-Prinzip

- **A (atomicity) in ACID:**

Man spricht von atomaren Operationen, wenn eine Sequenz von Datei- Operationen entweder ganz oder gar nicht ausgeführt wird.

*Wird eine Banküberweisung an irgendeiner Stelle gestoppt, wird der Vorgang rückgängig gemacht und die Überweisung wird zurückgewiesen.*

- **C (consistency) in ACID:**

Man spricht von einer vorhandenen Datenkonsistenz, wenn nach einer Sequenz von Datei- Operationen der Datenzustand in einem konsistenten Zustand hinterlassen wird.

*Wenn zwar die Überweisung von A nach B erfolgreich verlaufen ist, aber eine Zuordnung zwischen Sender und Empfänger nicht der Wahrheit entspricht, z.B. dass die Überweisung auf ein Konto C hätte gehen müssen, so liegt eine Verletzung der Konsistenz vor. Nach jeder Transaktion muss eine Integritätsprüfung erfolgen, damit die Gültigkeit der Datenbank gewährleistet ist.*

- **I (isolation) in ACID:**

Die Isolation verhindert, dass sich parallele Ausführungen auf befindliche Datei-Operationen gegenseitig beeinflussen können.

*Eine Überweisung darf nur als eine Transaktion erfolgen. Es dürfen keine parallelen Transaktionen zum Zeitpunkt der Überweisung von A nach B stattfinden. Sollen mehrere Banküberweisungen erfolgen, so wird die Transaktion in einer Warteschlange gespeichert und Schritt für Schritt abgearbeitet.*

- **D (durability) in ACID:**

Die Dauerhaftigkeit gewährleistet, dass die Datei-Operationen dauerhaft auf einem Datenträger gesichert sind.

*Die Überweisung von A nach B muss auch noch nach Jahren oder beliebig vielen Systemabstürzen nachvollziehbar und dauerhaft gespeichert sein.*

Wurden alle Einzeloperationen erfolgreich ausgeführt, kommt es zu einem sogenannten *commit transaction*. Wird eine Transaktion dagegen vorzeitig beendet (Abbruch durch den Benutzer, Systemfehler, Integritätsverletzung) wird der Zustand vor Beginn der Transaktion wiederhergestellt (*rollback*).

---

Das Transaktionskonzept garantiert die Vollständigkeit von zusammengehörigen Operationen: Entweder werden alle Operationen einer Transaktion ausgeführt oder keine. Das Transaktionskonzept ist somit eine wesentliche Voraussetzung für die Widerspruchsfreiheit einer Datenbank.

---

## 1.7 Arten von Datenbanken

Da Daten zu vielen verschiedenen Zwecken gespeichert werden müssen, existieren unterschiedliche Arten von Datenbanken.

### ► Hierarchische Datenbanken bzw. Vernetzte Datenbanken

Die Beziehungen zwischen den Datenelementen werden mit Zeigern, Mengen und verketteten Listen aufgebaut. Nachteilig sind die starren Strukturen. Diese Art von Datenbanken wird kaum noch verwendet.

### ► Relationale Datenbanken

Hier werden Tabellen zum Ablegen von Daten verwendet. Eine relationale Datenbank besteht aus beliebig vielen Einzeltabellen, die auf vielfältige Weise miteinander verknüpft werden können. Dieser Aufbau sorgt dafür, dass die Daten in der Datenbank konsistent sind. Jede Information muss nur ein einziges Mal gespeichert werden, sodass es nicht zu Mehrdeutigkeiten kommen kann.

Die Verknüpfungen zwischen den einzelnen Tabellen werden als Beziehungen oder Relationen bezeichnet und begründen die Bezeichnung relationale Datenbank.

## 1.8 Aufgaben

- 1) In welchem Zusammenhang stehen die Begriffe Datenbank, Datenbankmanagementsystem und Datenbanksystem?
- 2) Nenne sie Beispiele für Datenbanksysteme.
- 3) Wie lauten die drei Ebenen der 3- Schichten Architektur eines DBS und welchem Zweck dienen die einzelnen Ebenen?
- 4) Im Zusammenhang mit der 3-Schichten Architektur spricht man von der "Unabhängigkeit der Daten". Was versteht man darunter?
- 5) Geben sie zu den unten genannten Vorgängen jeweils an ob in Anwendungsprogrammen, in den externen Schemata, dem konzeptionellen Schema und/oder dem physischen Schema Änderungen notwendig werden.

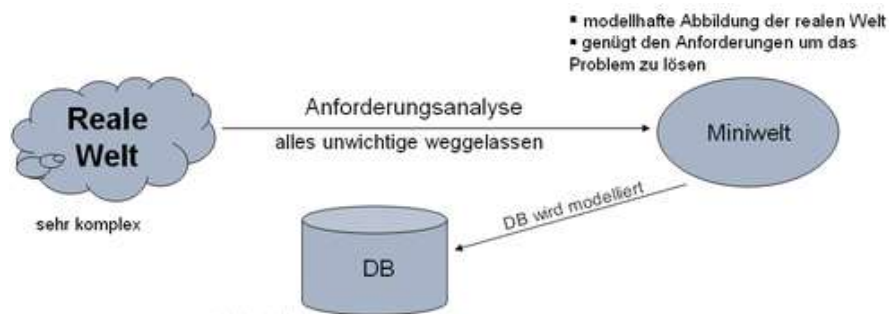
- a. Ein neues Anwendungsprogramm wird geschrieben, das bestehende Daten benutzt.
  - b. Ein Anwendungsprogramm benutzt eine geänderte Darstellung existierender Daten (z.B. SMALLINT anstatt INT)
  - c. Es werden neue Daten gespeichert oder bestehende gelöscht.
  - d. Die Daten werden auf einem anderen physischen Speichertyp gespeichert
- 6) Transaktionen werden in DBMS zur Behandlung von Änderungen benötigt. Zählen sie die wesentlichen Transaktionseigenschaften auf und beschreiben sie sie kurz.
- 7) Beurteilen sie folgende Aussagen oder Fragen unter ACID Gesichtspunkten:
- a. Mein Transaktionsprogramm wurde abgebrochen, jetzt sind alle meine Daten gelöscht.
  - b. Seit dem Abbruch meiner Transaktion sind deren Änderungen überhaupt nicht mehr vorhanden.
  - c. Eine andere Transaktion hat Änderungen meiner Transaktion überschrieben.
- 8) Worin besteht das wesentliche Merkmal relationaler Datenbanksysteme?

## 2 Datenbankentwurf

### 2.1 Von der Realität zur Datenbank

Informationssystem, z.B. zur Kontoführung bei Banken, Auskunftssystemen für Flugzeuge und Bahnen, Lagerhaltungssysteme speichern und verwalten Informationen über einen bestimmten Ausschnitt der realen. Aber selbst solche Ausschnitte sind meist so komplex, dass nicht alle Einzelheiten beschrieben werden können. Deshalb ist oft bei der Beschreibung eine Beschränkung auf die wichtigsten oder interessantesten Objekte der Realität und ihre Beziehungen untereinander sinnvoll. Wir erzeugen dadurch einen Ausschnitt der realen Welt, eine Miniwelt.

Bevor eine Datenbank erstellt werden kann muss man sich darüber klarwerden, für welchen Zweck wir die relationale Datenbank benötigen, also welche Informationen abgespeichert werden sollen. Hierzu müssen wir zunächst alle relevanten Objekte (Entitäten) und deren Beziehungen zueinander ermitteln:



Vorgehensweise zur Erzeugung eines Datenbankschemas:

- 1) Entitäten und Beziehungen identifizieren
- 2) Festlegen der Entitäts- und Beziehungstypen und deren Attribute ermitteln
- 3) Identitätsschlüssel für Entitäten bestimmen
- 4) Festlegen der Beziehungskardinalität
- 5) Wertebereiche (Domänen) der Attribute ermitteln
- 6) Entity-Relationship-Modell (ERM) zeichnen
- 7) Überführung in ein Relationenmodell und dabei die Primär- und Fremdschlüssel bestimmen
- 8) Relationen in Tabellen der relationalen Datenbank überführen
- 9) Attribute und ihre Wertebereiche in Tabellen eintragen

### 2.2 Entity Relationship Model (ERM)

Ein Entity Relationship Modell ist ein Standard für die Datenmodellierung. Es ist die grafische Beschreibung und Darstellung von Datenstrukturen, die in einer relationalen Datenbank gespeichert werden sollen.

#### 2.2.1 Entitäten, Attribute und Beziehungen

Die Grundelemente eines jeden ERM bilden Entitäten, Beziehungen und Attribute. Diese werden folgendermaßen dargestellt



- **Entität (Objekt, Gegenstand)** ist ein bestimmtes, identifizierbares Objekt der realen Welt.
- **Beziehung:** Zwischen Entitätsmengen – also Tabellen – in einer Datenbank können Beziehungen bestehen
- **Eigenschaften/Attribute:** Jede Entität besitzt eine bestimmbare Anzahl an Attributen/Eigenschaften. Um eine Entity eindeutig zu identifizieren muss ein Attribut eindeutig sein und wird als Primärschlüssel bezeichnet.

## Was könnte eine Entität sein?

- Substantive, die Dinge beschreiben, welche über verschiedene Eigenschaften verfügen
- Substantive, von denen mehrere Objekte wirklich existieren
- Beispiele für Entitäten: Kunden, Personen, Mitarbeiter, Schüler, Klassen, ...

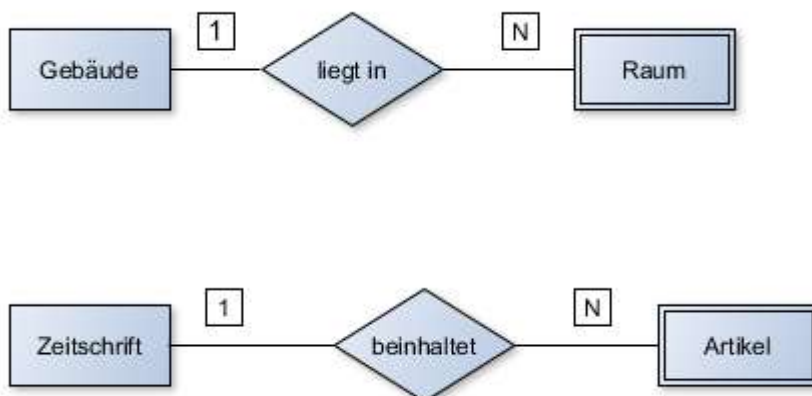
## Wie könnte ein Attribut aussehen?

- Wenn man beispielsweise eine Entität Kunden hat, könnten mögliche Attribute dieser Entität: Name, Vorname, Straße, Hausnummer sein. Damit eine Entity eindeutig identifizierbar ist, wird ihr ein sogenannter Primärschlüssel hinzugefügt.

### 2.2.2 Schwache Entitäten

In den meisten Fällen sind Entitäten eindeutig identifizierbar. Es kann aber auch vorkommen, dass Entitäten nur in Verbindung mit einer anderen Entität existieren könne. Diese Entitäten bezeichnet man als abhängig oder schwach. Eine schwache Entität wird durch ein doppelt eingrahmtes Rechteck gekennzeichnet.

Beispiele:



## 2.2.3 Wertebereich – Domäne

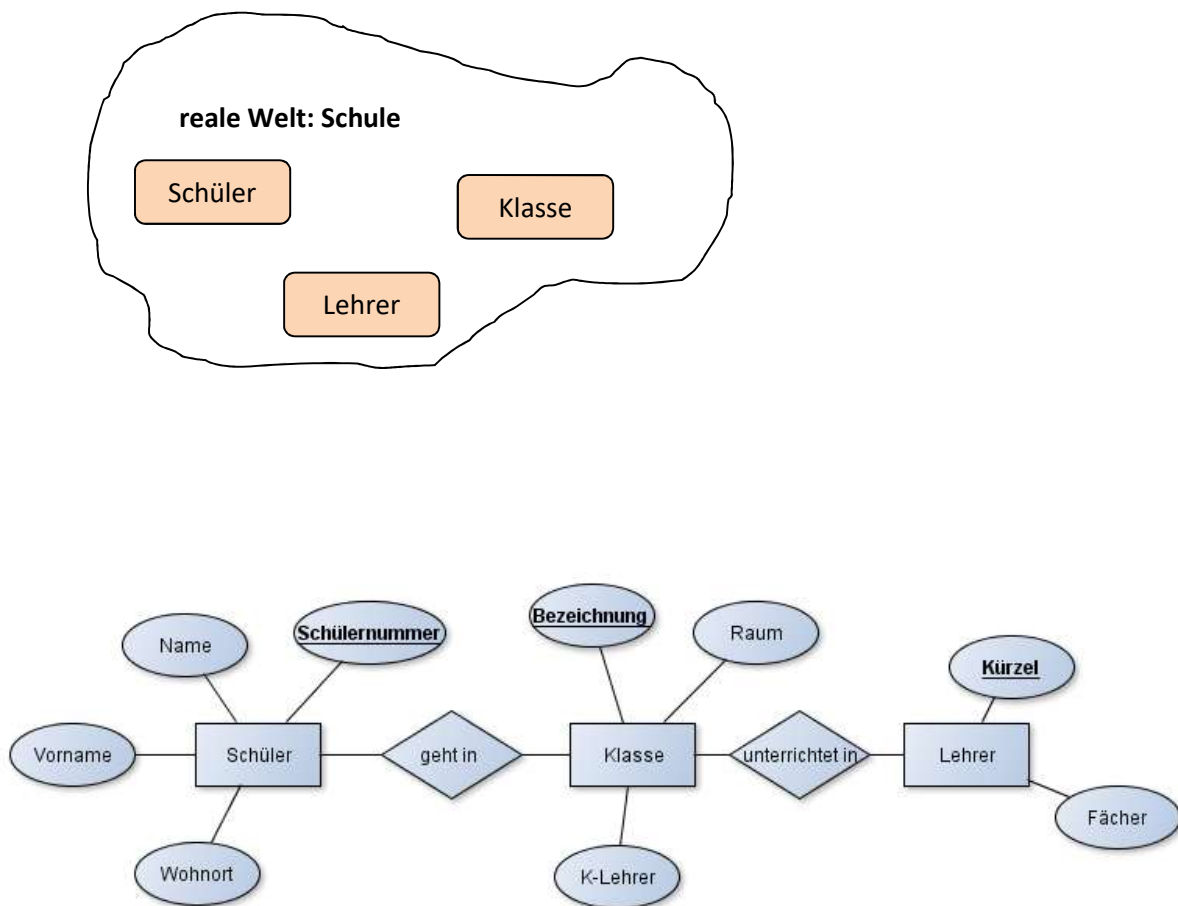
Unter einer Domäne versteht man die Menge der möglichen Werte, die ein Attribut annehmen kann. Eine Domäne besteht aus dem Namen des Attributes und einer Menge atomarer Werte.

Personalnummer = Integer

Gehalt = {g | 1000 ≤ g ≤ 10000}

Geschlecht = {männlich, weiblich}

Beispiel:



## 2.2.4 Übung:

Sie haben im Lotto gewonnen und machen sich mit einer Fitnessstudiokette selbständig. Da sie mehrere Fitnessstudios eröffnen, speichern sie zu jedem Studio die Stadt ab in der sich das Studio befindet. Ihre Mitglieder, zu denen Name und Geburtsdatum gespeichert werden sollen, sind einem Studio zugeordnet. In den Studios gibt es verschiedene Trainingsbereiche, z.B. Gerätebereich, Ausdauerbereich, Freihantelbereich.

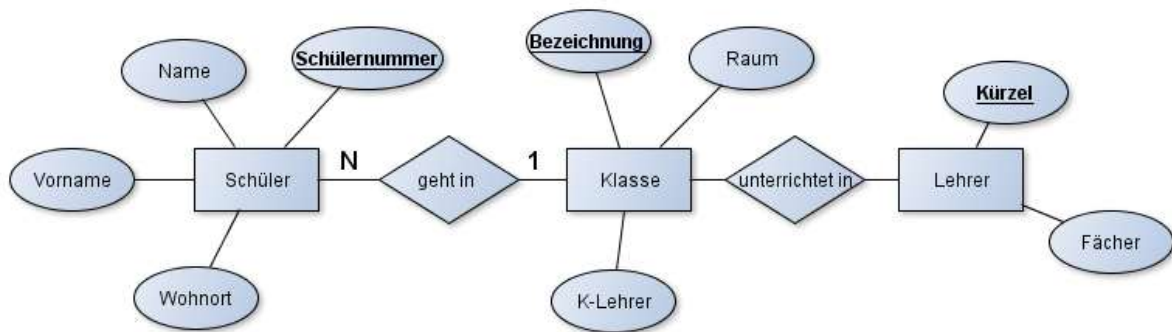
Identifizieren sie die zur Umsetzung in ein Datenbanksystem erforderlichen Entitäten, Attribute und Beziehungen und erstellen sie ein ERM.



## 2.3 Beziehungstypen

Zur Übertragung des Entity-Relationship-Modells in Datenbanktabellen ist es erforderlich Beziehungstypen festzulegen. Wir ergänzen unser bisheriges Modell Schule.

In eine Klasse können mehrere Schüler gehen. Ein Schüler geht jedoch immer in genau eine Klasse. Dieser Zusammenhang wird als Beziehungstyp bezeichnet.



Beim Beziehungstyp geht es um die Frage, ob eine Entität des Typs A mit einer oder mehreren Entitäten des Typs B in Beziehung steht.

Wir unterscheiden drei verschiedene Beziehungstypen:

- 1 : 1 Beziehung
- 1 : N Beziehung
- N : M Beziehung

### 2.3.1 1:1 Beziehung

In einer „eins zu eins“-Beziehung in relationalen Datenbanken ist jeder Datensatz in Tabelle A genau einem Datensatz in Tabelle B zugeordnet und umgekehrt. Diese Art von Beziehung sollte in der Modellierung vermieden werden, weil die meisten Informationen, die auf diese Weise in Beziehung stehen, sich in einer Tabelle befinden können. Eine 1:1-Beziehung verwendet man nur, um eine Tabelle aufgrund ihrer Komplexität zu teilen oder um einen Teil der Tabelle aus Gründen der Zugriffsrechte zu isolieren.

Eine Ehefrau ist mit genau einem Ehemann verheiratet und umgekehrt.



**Übung:** Ein Abteilungsleiter leitet genau eine Abteilung. Eine Abteilung wird von einem Abteilungsleiter geleitet.

## 2.3.2 1:N Beziehung

Eine „eins zu viele“-Beziehung relationalen Datenbanken ist der häufigste Beziehungstyp in einer Datenbank. In einer 1:n-Beziehung können einem Datensatz in Tabelle A mehrere passende Datensätze in Tabelle B zugeordnet sein, aber einem Datensatz in Tabelle B ist nie mehr als ein Datensatz in Tabelle A zugeordnet.

In einem Museum werden mehrere Kunstwerke ausgestellt. Ein Kunstwerk kann nur in einem Museum stehen.



**Übung:** Ein Mitarbeiter gehört zu einer Abteilung. In einer Abteilung arbeiten mehrere Mitarbeiter.

## 2.3.3 N:M Beziehung

Bei „viele zu viele“-Beziehung in relationalen Datenbanken können jedem Datensatz in Tabelle A mehrere passende Datensätze in Tabelle B zugeordnet sein und umgekehrt. Diese Beziehungen können nur über eine dritte Tabelle, eine Verbindungstabelle C, realisiert werden.

Ein Kunde bestellt mehrere Artikel. Ein Artikel kann von mehreren Kunden bestellt werden.



**Übung:** Ein Student nimmt an mehreren Vorlesungen teil. An jeder Vorlesung nehmen mehrere Studenten teil.

## 2.4 Aufgaben

### 1. Aufgabe

Ergänzen Sie zu den jeweiligen Beziehungen die entsprechenden Beziehungstypen.

Fügen Sie eine Beschreibung, wie im Beispiel gezeigt hinzu.

Jede Mutter hat mindestens eins oder mehrere Kinder geboren. Jedes Kind wurde von genau einer Mutter geboren.

## 2. Aufgabe

In Büroräumen (charakterisiert durch eine Zimmernummer) sitzen seit einem Zeitpunkt Mitarbeiter (Personalnummer, Name, Titel, Status) an einem bestimmten Platz. In den Zimmern sind Telefone (besitzen eine eindeutige Telefonnummer) aufgestellt, die als Hausapparat oder Amtsapparat geschaltet sind.

## 3. Aufgabe

Eine Fluggesellschaft (eindeutige Bezeichnung, Land, Hauptsitz) besitzt seit einem bestimmten Datum Flugzeuge (eindeutige Flugzeug-Nummer, Datum der letzten Kontrolle) unterschiedlichen Typs (Typbezeichnung, Anzahl Sitze, Geschwindigkeit). Jede Fluggesellschaft beschäftigt seit einem bestimmten Datum eine Menge von Piloten (Personal-Nummer, Name, Geburtsdatum, Qualifikation, Flugstunden). Passagiere (Kundennummer, Name, Adresse, Geburtsdatum) haben Flüge (Flug-Nummer, Datum, Abflugsort, Zielflughafen, Dauer) gebucht (Anzahl Sitze, Klasse, Preis). Ein Flug wird von mehreren Piloten und einem bestimmten Flugzeug realisiert.

## 4. Aufgabe

Stellen Sie sich vor, Sie arbeiten als Berater bei einem Automobilzulieferer und sollen Informationen zur Erstellung eines Kundenmanagementsystems ermitteln.

Hierzu sammeln Sie folgende Informationen:

- Für einen Kunden können mehrere Angebote erstellt worden sein
- Ein Kunde kann mehrere Aufträge erteilen
- Für einen Kunden wird die Kundennummer, Name, Adresse, Ansprechpartner, Telefonnummer und Branche erfasst
- Ein Auftrag besteht aus einer oder mehreren Auftragspositionen
- Zu einem Auftrag gehören Auftragsnummer Wunschliefertermin und die Lieferadresse
- Einer Auftragsposition ist genau ein Produkt zugeordnet
- Mit Hilfe einer Produktnummer wird ein Produkt eindeutig beschrieben. Produktname, Beschreibung, Gewicht und Dimension sind weitere Eigenschaften eines Produktes
- Die Rechnung an einen Kunden bezieht sich auf eine oder mehrere Auftragspositionen
- Ein Kunde kann mehrere Reklamationen haben
- Eine Reklamation bezieht sich immer auf eine Rechnung und ein Produkt
- Die Reklamation umfasst den Reklamationsgrund, die Kurzbeschreibung des Problems und einen Regulierungswunsch des Kunden

## 5. Aufgabe

Sie finanzieren ihr Studium, indem Sie in einer EDV-Beratungsfirma jobben. Diese Beratungsfirma erhält den Auftrag, für das IOC (International Olympic Committee) eine Datenbank für die nächsten Olympischen Spiele zu erstellen, die folgenden Wirklichkeitsausschnitt enthalten soll.

Die einzelnen Wettkämpfe der Olympischen Spiele sind durch den Namen der Sportart, den Termin und die Sportstätte gekennzeichnet. An jedem Wettkampf nehmen mehrere Sportler teil, die durch eine Startnummer identifiziert werden und außerdem natürlich, wie jede Person, einen Namen besitzen. Jeder Wettkampf wird von einem Schiedsrichter geleitet, dem für diese Spiele eine eindeutige Personalnummer zugeordnet wurde. Die Schiedsrichter werden bei einem Wettkampf von verschiedenen Helfern unterstützt, die ebenfalls eine eindeutige Personalnummer erhalten haben. Die Sportler und Schiedsrichter gehören jeweils einer Nation an, zu der der Name des Mannschaftsleiters und eine Telefonnummer für Rückfragen abgespeichert werden. Dies gilt zwar ebenfalls für die Helfer, soll jedoch hier nicht berücksichtigt werden.

## 6. Aufgabe

Mehrere in ganz Deutschland verteilte Zoos sollen mittels einer Datenbank verwaltet werden. Jeder Zoo (eindeutige Identifikationsnummer, Name und Ort) hat mehrere Tierarten (bezeichnet durch ihren Namen). Diese Tierarten werden von mehreren Pflegern (Personalnummer, Name, Geburtsdatum, Gehalt) gepflegt und befinden sich in jeweils einem ihnen zugeteilten Raum (Raum-Nummer, Fläche). Als Raum werden hier auch Außengehege bezeichnet. Von jeder Tierart gibt es einige Exemplare (eindeutige Tier-Nummer, Alter, Geschlecht). Jedes Tier bekommt eine spezielle Futtermischung (eindeutige Nummer, Bezeichnung), welche aus verschiedenen Futtermitteln (eindeutige Nummer, Name) hergestellt wird. Die Futtermittel werden von Lieferanten (eindeutige Lieferanten-Nummer, Name, Adresse) geliefert und in Lagern (eindeutige Lager-Nummer, Kapazität) aufbewahrt.

## 2.5 Kardinalitäten

Mit Hilfe von Kardinalitäten kann man Beziehungstypen genauer beschreiben. Für jeden Beziehungstyp gibt man an, in wie vielen Beziehungen eine Entity mindestens (min) stehen muss und maximal (max) stehen kann.

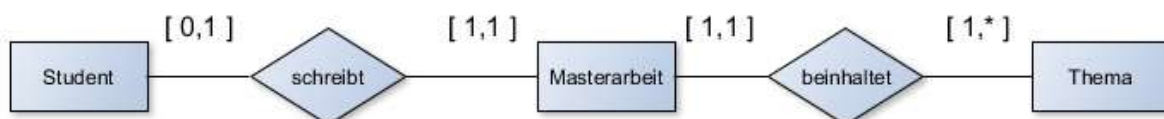
Beispiel 1:

Ein Kunde erteilt einen oder mehrere Aufträge und ein Auftrag stammt von genau einem Kunde.



Beispiel 2:

Studenten dürfen genau eine oder keine Masterarbeit schreiben, wobei jede Masterarbeit von genau einem Studenten geschrieben wird. Jede Masterarbeit ist genau einem Thema zugeordnet, aber ein Themengebiet kann in mehreren Masterarbeiten bearbeitet werden.



## 1. Aufgabe

Erstellen sie ein ER-Modell für den folgenden Sachverhalt. Geben sie die Kardinalitäten sinnvoll an.

- Es gibt Tierarten. Jede Tierart hat einen Namen, gehört zu einer Gattung und erreicht ein bestimmtes Gewicht. Z.B. Löwen gehören zur Gattung Raubkatzen und werden bis zu 250kg schwer, Elche sind Huftiere und bis zu 800kg schwer.
- Es gibt Länder mit Namen, diese liegen auf Kontinenten, die auch mit einem Namen bezeichnet werden.
- Der weltweite, freilebende Bestand der verschiedenen Tierarten verteilt sich auf viele Länder, oft auch auf bestimmte Gebiete in diesen Ländern. Z.B. leben 300000 Elche in Schweden und 200000 Elche in Norwegen.
- Außerdem gibt es Zoos, die insbesondere bei der Erhaltung bedrohter Tierarten, wie z.B. dem Panda eine wichtige Rolle spielen. Zoos liegen in Städten, die wiederum in Ländern liegen. In jedem Zoo werden mindestens 3 Tierarten gehalten.

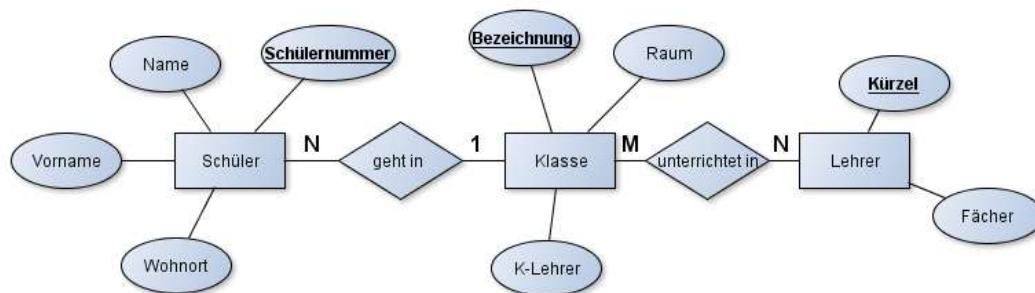
## 2.6 Überführung in das Relationenmodell

Als zweiter Schritt beim Entwurf einer Datenbank steht nach der Modellierung die Umsetzung des Modells in ein Relationenschema an. Dazu wird jede Entität in ein eigenes Relationenschema (Tabelle) abgebildet.

Die Tabelle enthält Tupel auch Datensätze genannt, diese entsprechen den Zeilen der Tabelle. Jedes Tupel enthält Werte für eine Reihe von Attributen (Spalten), die Eigenschaften der Entität beschreiben.

Die Menge der möglichen Werte eines Attributes wird als Domäne bezeichnet.

Beispiel:



### 1. Schritt

Jede Entität wird mit ihren Attributen als Tabelle dargestellt.



In jeder Tabelle muss ein eindeutiger Wert vorkommen, den wir Primärschlüssel nennen. Dieser Wert wird immer unterstrichen dargestellt.

Schüler				
Primärschlüssel	Schüler	Entität		
	<u>Schülernummer</u>	Name	Vorname	Wohnort
	1000	Kugel	Fritz	Landau
Tupel, Datensatz	1001	Berg	Lucy	Neustadt
	1002	Berg	Moni	Landau

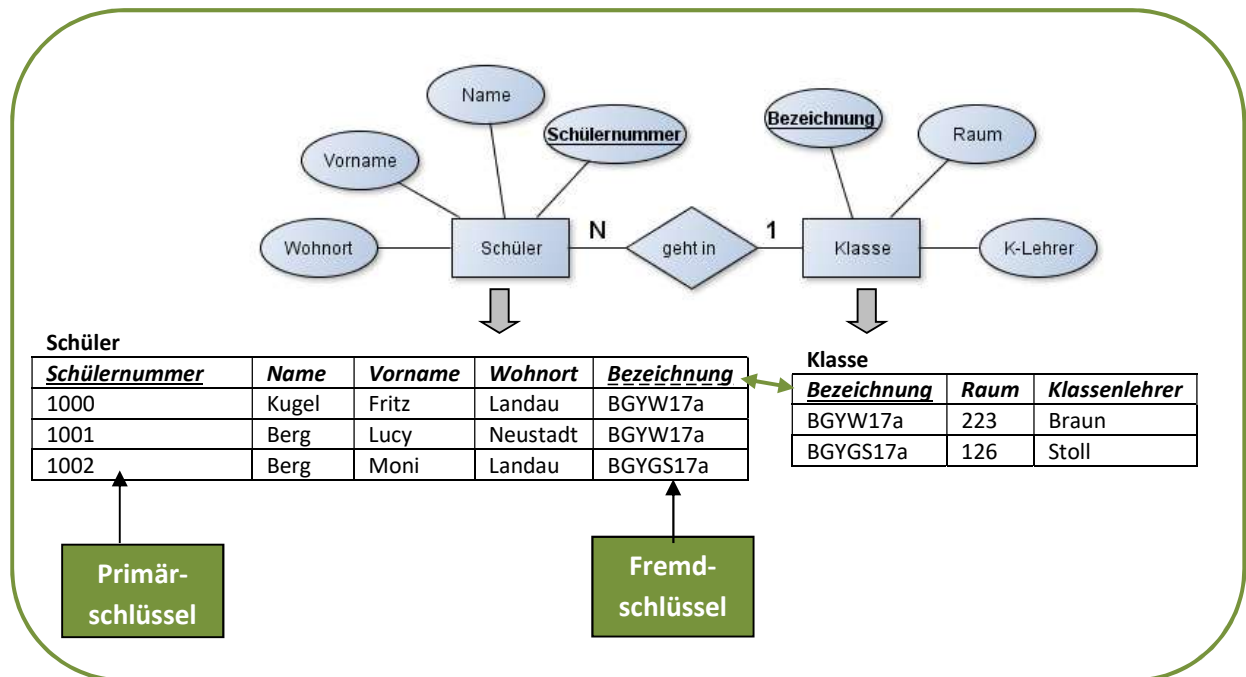
Klasse		
<u>Bezeichnung</u>	Raum	Klassenlehrer
BGYW17a	223	Braun
BGYGS17a	126	Stoll

Lehrer	
<u>Kürzel</u>	Fächer
BRA	Ehtik, GK
GHZ	BWL, Mathematik
STL	Biologie, Chemie
KLR	BWL, Französisch
RNR	BWL, Englisch

## 2. Schritt

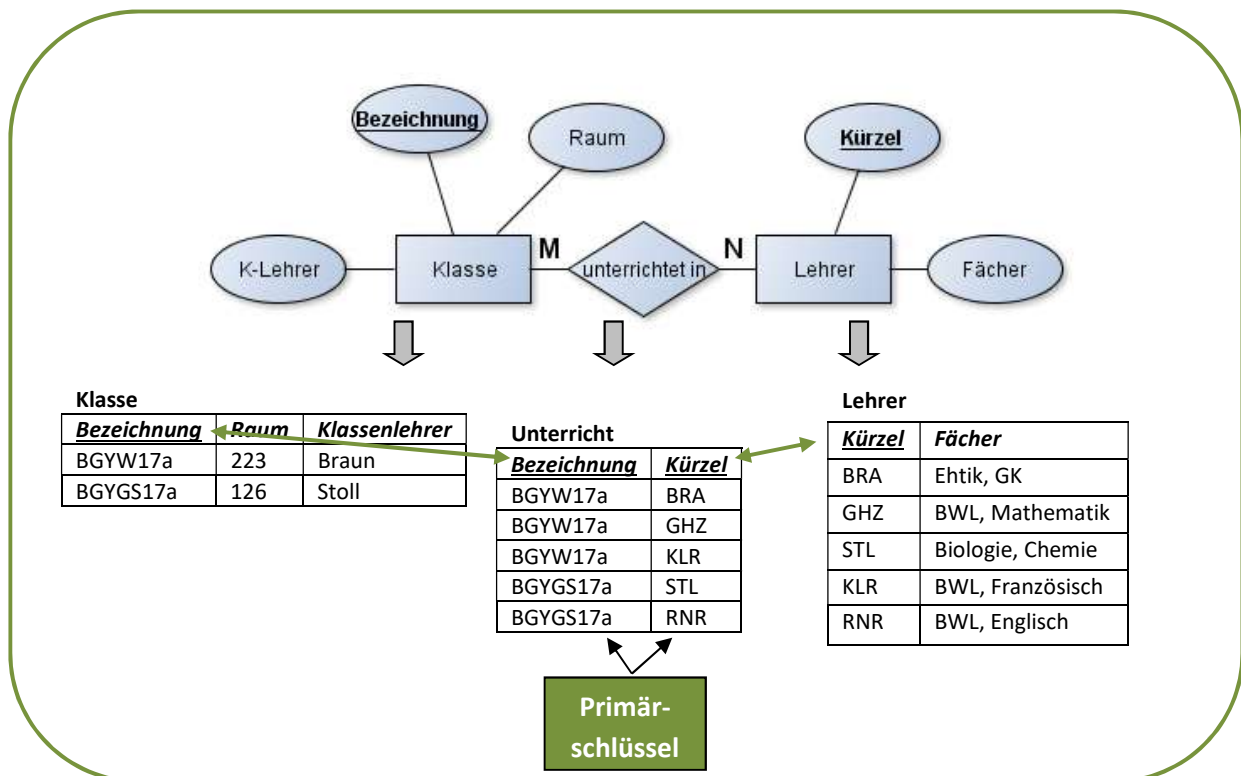
Beziehungen zwischen den Tabellen herstellen.

### → Umsetzung einer 1:N Beziehung



Der Primärschlüssel der Mastertabelle ("1") wird als Fremdschlüssel in die Detailtabelle ("N") aufgenommen. Der Fremdschlüssel wird gestrichelt unterstrichen dargestellt.

### → Umsetzungen einer N:M Beziehung





Eine N:M Beziehung wird durch eine eigene Tabelle abgebildet, in der die Primärschlüssel der zugehörigen Entitäten als Fremdschlüssel eingetragen werden. Diese beiden Fremdschlüssel bilden entweder einen zusammengesetzten Primärschlüssel oder es wird ein neuer Schlüssel definiert.

Damit ergeben sich für unser Beispiel die folgenden Tabellen in Kurzschreibweise:

Schüler (Schülernummer, Name, Vorname, Wohnort, Bezeichnung)

Klasse (Bezeichnung, Raum, Klassenlehrer)

Lehrer (Kürzel, Fach)

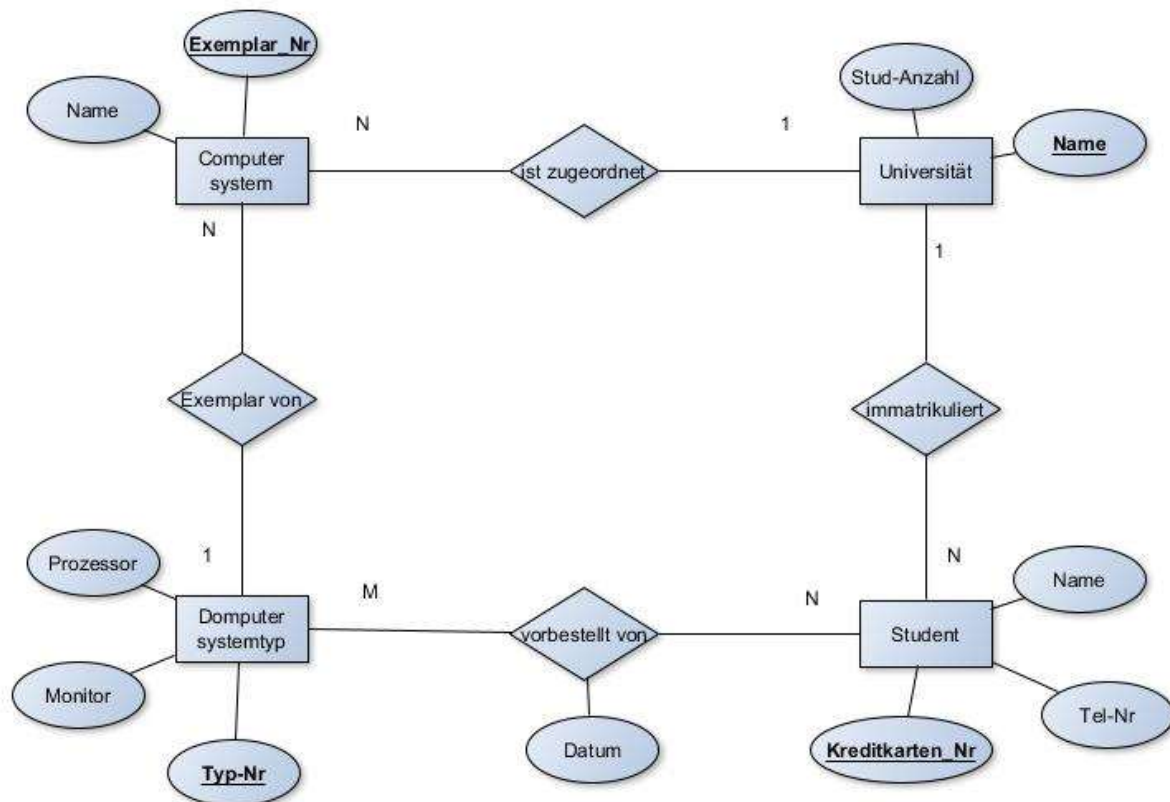
Unterricht (Bezeichnung, Kürzel)

## → Umsetzungen einer 1:1 Beziehung

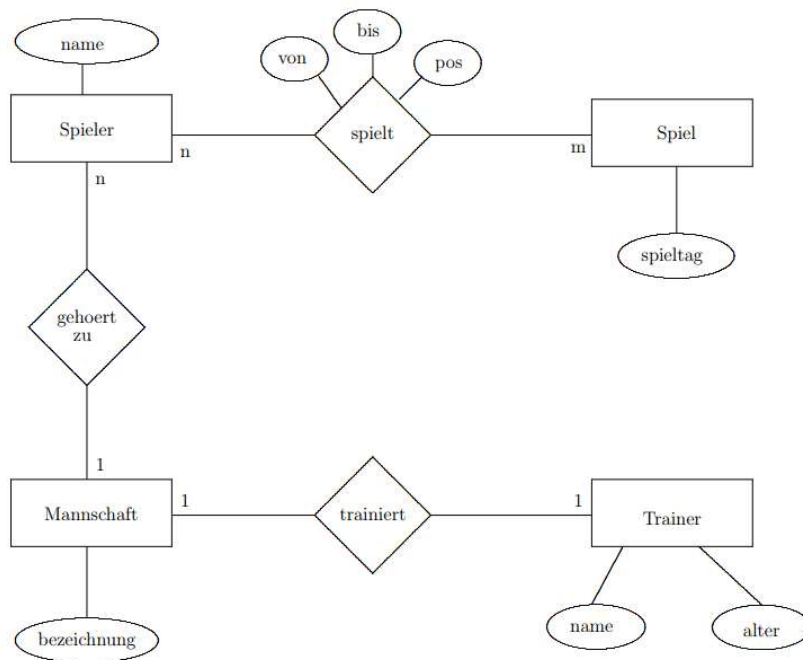
Bei einer 1:1 Beziehung kann frei ausgewählt werden, in welche Tabelle der Primärschlüssel als Fremdschlüssel übernommen werden soll.

## 2.7 Aufgabe

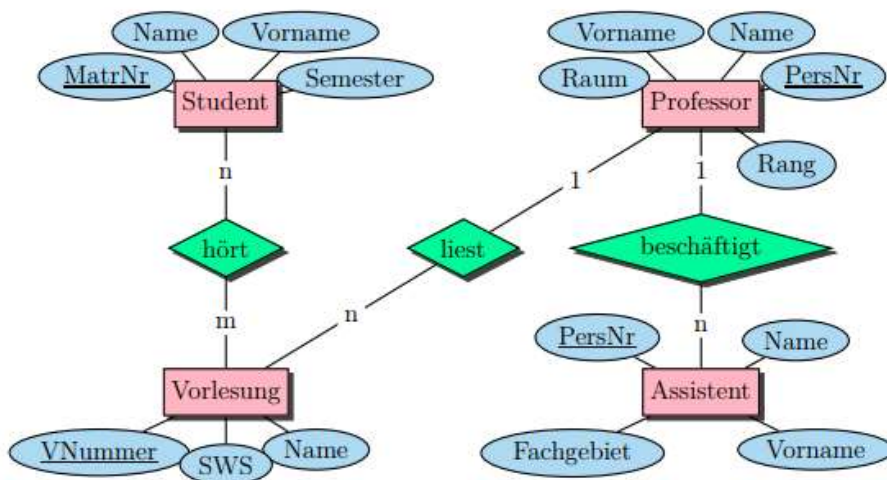
**1. Aufgabe:** Überführen sie das abgebildete ERM in ein Relationenmodell.



**2. Aufgabe:** Ergänzen sie bei den Entitäten die notwendigen Primärschlüssel und überführen sie das ERM in ein Relationenmodell. Die Attribute bei der Relation "spielt" werden in die neu zu bildende Tabelle mit aufgenommen.



**3. Aufgabe:** Transformieren sie das ERM in ein Relationenmodell



## 2.8 Normalisierung

### Optimierung von Datenstrukturen

Von einer brauchbaren Datenbank fordert man auf alle Fälle die Widerspruchsfreiheit. Abfragen zu bestimmten Aspekten dürfen nicht zu sich widersprechenden Ergebnissen führen. Solche Widersprüche (Anomalien) können sich ergeben, wenn beim Ändern, Einfügen oder Löschen von Daten nicht alle erforderlichen Stellen in den entsprechenden Tabellen bearbeitet werden.

#### Änderungs-, Einfüge- oder Lösch Anomalien.

Anomalien treten bevorzugt auf, wenn gleiche Informationen an verschiedenen Stellen auftreten, obwohl es ausreichen würde, sie nur an einer Stelle festzuhalten. Man spricht dann von **Redundanz** im Datenbestand.

Beispiel:

Ein Mitarbeiter gehört immer genau zu einer Abteilung und mehrere Mitarbeiter können in verschiedenen Projekten arbeiten.

**Projektmitarbeit**

MiNr	Name	AbtNr	Abteilung	Projekte
101	Egon	42	DB	Infra, Portal
102	Erna	42	DB	Frame, Infra
103	Uwe	43	GUI	Frame, Portal

#### Operationen auf Datenbanken:

**Löschen:** Das Projekt Frame soll aus der Tabelle gelöscht werden.

**Ändern:** Das Projekt Infra soll in Infrastruktur umbenannt werden.

**Einfügen:** Am neuen Projekt Markt sollen die Mitarbeiter Egon und Uwe mitarbeiten.

Eine solche Tabelle trifft man in der Praxis häufig an, wenn Tabellenkalkulationsprogramme zur Verwaltung von Daten genutzt werden.

Zur Vermeidung von Anomalien und Redundanzen in Datenbanken unterzieht man den Datenbestand einem Normalisierungsprozess über die Normalformen.

#### Die wichtigsten Gründe für das Normalisieren einer Datenbank sind:

1. Vermeidung von Redundanz (doppelte Einträge)
2. Vermeidung von Anomalien (widersprüchliche Daten)
3. Konsistenz (Vollständigkeit und Richtigkeit)
4. Vereinfachung der Wartung

Die wichtigsten Normalformen sind die 1 NF, 2 NF und 3 NF

## 2.8.1 Die 1. Normalform

Eine Tabelle befindet sich in der **1. Normalform** (1. NF), wenn ihre Attribute nur einfache (atomare) Attributswerte aufweisen und keine Wiederholfelder vorkommen und ein eindeutiger Primärschlüssel gegeben ist.

Beispiel:

**Projektmitarbeit**

<u>MiNr</u>	Name	AbtNr	Abteilung	Projekte
101	Egon	42	DB	Infra, Portal
102	Erna	42	DB	Frame, Infra
103	Uwe	43	GUI	Frame, Portal

Das Attribut Projekte ist kein einfaches Attribut, weil es Mehrfacheinträge gibt. Diese werden auf mehrere Tupel aufgeteilt:

**Projektmitarbeit**

<u>MiNr</u>	Name	AbtNr	Abteilung	<u>ProjektNr</u>	Projekt
101	Egon	42	DB	1	Infra
101	Egon	42	DB	2	Portal
102	Erna	42	DB	3	Frame
102	Erna	42	DB	1	Infra
103	Uwe	43	GUI	3	Frame
103	Uwe	43	GUI	2	Portal

Man sieht nun sofort, dass die Tabelle Redundanzen enthält. Der Name ist mit der Abteilungsnummer gekoppelt und müsste nicht für jedes Projekt wiederholt werden. Wenn der Name geändert werden müsste, müsste dies an mehreren Stellen geschehen. → Fehleranfällig.

In der Tabelle werden verschiedene Sachverhalte abgebildet, die voneinander unabhängig sind. Diese verschiedenen Sachverhalte sollen auch mit verschiedenen Tabellen dargestellt werden.

## 2.8.2 Die 2. Normalform

Eine Tabelle befindet sich in der **2. Normalform (2. NF)**, wenn sie in der 1. Normalform ist und jedes Nichtschlüsselattribut voll funktional abhängig vom Primärschlüssel ist.

Die oben dargestellte Tabelle befindet sich nicht in der 2. NF, weil das Attribut Projekt nicht voll funktional vom Primärschlüssel abhängig ist, sondern schon durch den Teilschlüssel ProjektNr eindeutig identifiziert werden kann. Für die 2. NF werden die Tabellen aufgespalten:

**Personen**

<u>MiNr</u>	Name	AbtNr	Abteilung
101	Egon	42	DB
102	Erna	42	DB
103	Uwe	43	GUI

**Proiekt**

<u>ProjektNr</u>	Projekt
1	Infra
2	Portal
3	Frame

**PersonenProiekt**

<u>MiNr</u>	<u>ProjektNr</u>
101	1
101	2
102	3
102	1
103	3
103	2

Sind nun alle Redundanzen beseitigt?

Die Tabellen Projekte und PersonenProjekte enthalten keine Redundanzen mehr. Aber in der Tabelle Personen gibt es noch Redundanz. für jede Person ist der Abteilungsname gespeichert, obwohl dieser Name mit der Abteilungsnummer gekoppelt ist. Der Abteilungsname ist also transitiv vom Primärschlüssel abhängig.

## 2.8.3 Die 3. Normalform

Eine Tabelle befindet sich in der **3. Normalform** (3. NF), wenn sie in der 2. Normalform ist und jedes Nichtschlüsselattribut nicht transitiv vom Primärschlüssel abhängig ist, bzw. kein Nichtschlüsselattribut von einem anderen Nichtschlüsselattribut abhängig ist.

Wir lagern also auch noch die transitiv abhängigen Attribute in eigene Tabellen aus:

**Personen**

<u>MiNr</u>	Name	AbtNr
101	Egon	42
102	Erna	42
103	Uwe	43

**Abteilung**

<u>AbtNr</u>	Abteilung
42	DB
43	GUI

**Projekt**

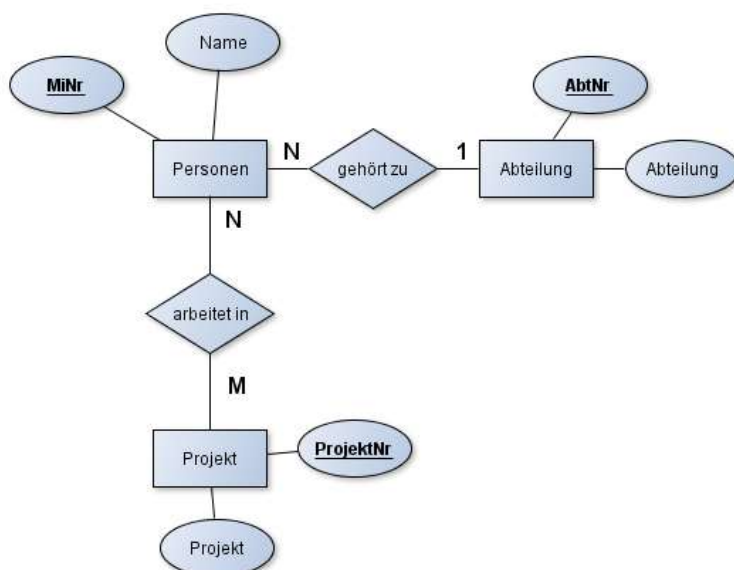
<u>ProjektNr</u>	Projekt
1	Infra
2	Portal
3	Frame

**PersonenProjekt**

<u>MiNr</u>	<u>ProjektNr</u>
101	1
101	2
102	3
102	1
103	3
103	2

Betrachten wir uns einmal des ERM zur Aufgabenstellung:

Ein Mitarbeiter gehört immer genau zu einer Abteilung und mehrere Mitarbeiter können in verschiedenen Projekten arbeiten.



## 2.9 Aufgaben

### 1. Aufgabe

Die untenstehende Tabelle zur Verwaltung von Fahrzeugdaten soll in die 3. Normalform überführt werden. Transformieren sie die Tabelle schrittweise in die 1., 2. und 3. Normalform. Erstellen sie zur Kontrolle das ERM Modell.

Fahrzeug							
Hersteller	Modell	Typ	Anzahl Räder	PS	Adresse	Land	Klima-anlage
VW	Käfer	PKW	4	45	Volkswagenstr. 10, 34444Wolfsburg	Deutschland	Ja
Hanomag	II	LKW	4	220	Musterstr. 1, 22222 Hamburg	Deutschland	Nein
Vespa	Vespa Light	Motorroller	2	4	La Strada 2, 122 Rom	Italien	Nein
Suzuki	II	Motorrad	2	8	Koniciva 3, Yokohama	Japan	Nein
Ford	Focus	PKW	4	65	Fritzweg 2, 54455 Köln	Deutschland	Ja
VW	Golf	PKW	4	72	Volkswagenstr. 10, 34444Wolfsburg	Deutschland	Nein
VW	L 80	LKW	4	140	Volkswagenstr. 10, 34444Wolfsburg	Deutschland	Nein
VW	Transporter	PKW	4	72	Volkswagenstr. 10, 34444Wolfsburg	Deutschland	Ja

### 2. Aufgabe

Um Bestelldaten angemessen verwalten zu können, soll die Tabelle in die 3. Normalform gebracht werden. Transformieren sie die Tabelle schrittweise in die 1., 2. und 3. Normalform. Erstellen sie zur Kontrolle das ERM Modell.

Vorname	Name	Straße	Ort	Datum	ArtNr	Artikel	M	RK	RS
Frank	Meier	Birkenstr 17	Köln	15.03.04	25	Papier A4	2	3	0
Frank	Meier	Birkenstr 17	Köln	15.03.04	28	Patrone	1	5	0
Frank	Meier	Birkenstr 17	Köln	15.03.04	35	Stifte	3	2	0
Jens	Braun	Maystr. 51	Köln	15.03.04	45	Folie	10	3	10
Jens	Braun	Maystr. 51	Köln	15.03.04	41	Ordner 10	5	10	10
Jens	Braun	Maystr. 51	Köln	15.03.04	13	Papier A3	10	3	10
Georg	Wegner	Waldweg 3	Solm	15.03.04	15	Hefter	20	10	10
Georg	Wegner	Waldweg 3	Solm	15.03.04	28	Patrone	5	5	10
Georg	Wegner	Waldweg 3	Solm	15.03.04	42	Hüllen	3	3	10
Jenny	Müller	Burgstr. 4	Ulm	16.03.04	13	Papier A3	6	3	0
Jenny	Müller	Burgstr. 4	Ulm	16.03.04	28	Patrone	5	5	0
Jenny	Müller	Burgstr. 4	Ulm	16.03.04	35	Stifte	10	2	0
Carola	Schmidt	Am Ufer 2	Trier	16.03.04	15	Hefter	20	10	5
Carola	Schmidt	Am Ufer 2	Trier	16.03.04	41	Ordner 10	5	10	5

(KNr: Kundennummer, ArtNr.: Artikelnummer, M: Menge, RK: Rubrik, RS: Rabattsatz)

1. Normalform



## 3. Aufgabe

Büromaterial

Material-nummer	Material-bezeichnung	Lieferanten-nummer	Lieferanten-bezeichnung	Lager-bestand	Preis
101	CD-Rohling	2	InnoPaper	61	1.99
221	Kugelschreiber	2	InnoPaper	156	0.99
101	CD-Rohling	31	Computer-Mendel	61	1.89
134	Papier A4	2	InnoPaper	702	8.99
002	Toner schwarz	31	Computer-Mendel	8	39.50
007	Heftzwecken	26	Muster-Büroartikel	81	0.79
026	Schnellhefter	2	InnoPaper	81	0.89

## 4. Aufgabe

Spielmittelverleih (unnormalisiert)

Kunden-nr	Kundenname	Kundenanschrift	Verleih-nr	Verleih-datum	Spielnr	Spielbezeichnung	Rückgabe am
101	Müller, Fritz	12345 Berlin, Mustergasse 4	5	01.08.2008	21, 15	Sitzball, Bausteine	03.08.2008
101	Müller, Fritz	12345 Berlin, Mustergasse 4	5	01.08.2008	37	Traktor	
102	Meier, Frieda	23456 Hamburg, Webergasse 1	1	05.07.2008	15	Bausteine	15.07.2008
102	Meier, Frieda	23456 Hamburg, Webergasse 1	6	04.08.2008	21	Sitzball	
103	Lehmann, Max	12345 Berlin, Fallweg 1	3	10.07.2008	21, 37	Sitzball, Traktor	20.07.2008

## 5. Aufgabe

0. Normalenform										
KNR	KNAM	KTEIL	KPR	DNR	DNAM	STD	DTL	RAUM	RPZ	RTYP
1	Excel-Einführung	16	800,00 €	1010	Meier	16	57321	E1	16	PC-Labor
2	Excel-Aufbau	13	1.200,00 €	1010, 1230	Meier, Schulz	12, 12	57321, 63491	E1	16	PC-Labor
3	UNIX-Einführung	9	1.200,00 €	1132	Weber	24	44362	E4	12	UNIX-Labor
4	Access-Einführung	13	800,00 €	1230	Schulz	16	63491	W2	16	PC-Labor
5	Access-Aufbau	10	1.200,00 €	1230, 1010	Schulz, Meier	8, 16	63491, 57321	W2	16	PC-Labor



## 3 Datenbankentwicklung

### 3.1 Anlegen der Datenbankstruktur

Um mit Tabellen arbeiten zu können, also Datensätze anzeigen, verändern oder bearbeiten zu können, benötigen wie zuerst eine Tabelle.

Tabellen legt man mit dem Befehl CREATE TABLE an.

**Klasse**

<u>Bezeichnung</u>	<u>Raum</u>	<u>Klassenlehrer</u>
BGYW17a	223	Braun
BGYGS17a	126	Stoll

```
CREATE TABLE klasse
(
    Bezeichnung VARCHAR (10) NOT NULL,
    Raum INTEGER,
    Klassenlehrer VARCHAR (20),
    CONSTRAINT pk_Bezeichnung
    PRIMARY KEY (Bezeichnung)
)
```

NOT NULL bedeutet, dass der Eintrag nicht leer bleiben darf.  
CONSTRAINT legt die Integritätsbedingungen fest. In diesem Fall den Primärschlüssel

#### 3.1.1 Syntax und Optionen

<pre>CREATE TABLE tabellenname (     feldname1 datatype1 [NOT NULL]     [AUTO_INCREMENT],     feldname2 datatype2,     feldname3 datatype3,...     CONSTRAINT feldnamex     PRIMARY KEY (feldnamex) )</pre>	<ul style="list-style-type: none"> <li>• tabellenname = Name der Tabelle</li> <li>• feldname = "Spalte" einer Tabelle (Attribut aus ERM)</li> <li>• datatype = INTEGER, FLOAT, VARCHAR, TIME, DATE</li> <li>• NOT NULL = Feld darf nicht LEER sein</li> <li>• AUTO_INCREMENT = Zahl wird bei jedem neuen Datensatz automatisch um eins erhöht</li> <li>• PRIMARY KEY = Feld wird zum Primärschlüssel und referenziert damit jeden Datensatz eindeutig</li> <li>• CONSTRAINT = Bedingungen (Angabe von Primär- und Fremdschlüssel, Wertebereichseinschränkungen, Angabe eines Default Wertes)</li> </ul>
---	---

#### 3.1.2 Tabelle mit Fremdschlüssel anlegen

**Schüler**

<u>Schülernummer</u>	<u>Name</u>	<u>Vorname</u>	<u>Wohnort</u>	<u>Bezeichnung</u>
1000	Kugel	Fritz	Landau	BGYW17a
1001	Berg	Lucy	Neustadt	BGYW17a
1002	Berg	Moni	Landau	BGYGS17a

```
CREATE TABLE schueler
```

```
(
  Schülernummer INTEGER NOT NULL AUTO_INCREMENT,
  Name VARCHAR (30),
  Vorname VARCHAR (30),
  Wohnort VARCHAR (50),
  Bezeichnung VARCHAR (10),
  CONSTRAINT pk_Schülernummer
  PRIMARY KEY (Schülernummer),
  CONSTRAINT fk_Bezeichnung
  FOREIGN KEY (Bezeichnung) REFERENCES klasse (Bezeichnung)
)
```

Fremdschlüssel Bezeichnung als  
Tabellenspalte aufnehmen.  
Datentyp muss identisch sein, zu  
Primärschlüssel in der Mastertabelle.

Die Bedingung Fremdschlüssel  
verweist auf die Mastertabelle

## Lehrer

<u>Kürzel</u>	<u>Fach 1</u>	<u>Fach 2</u>
BRA	Ehtik	GK
GHZ	BWL	Mathematik
STL	Biologie	Chemie
KLR	BWL	Französisch
RNR	BWL	Englisch

```
CREATE TABLE lehrer
```

```
(
  Kürzel VARCHAR (3) NOT NULL,
  Fach1 VARCHAR (10) NOT NULL,
  Fach2 VARCHAR (10) NOT NULL,
  CONSTRAINT pk_Kürzel
  PRIMARY KEY (Kürzel)
)
```

Jede Lehrkraft muss zwei Fächer  
unterrichten können.

## Unterricht

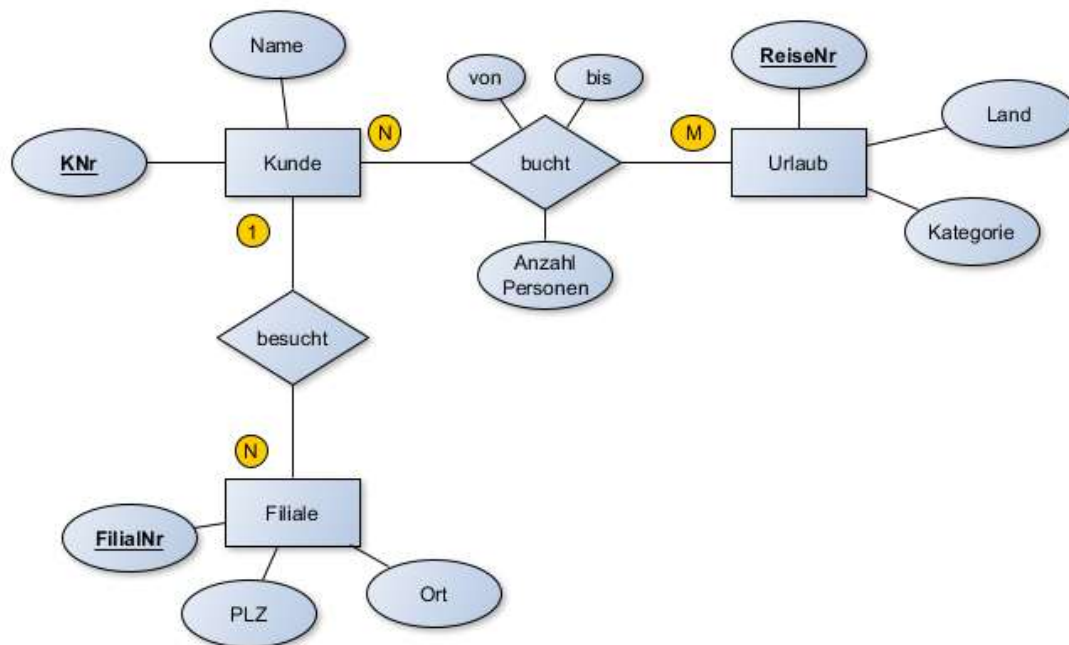
<u>Bezeichnung</u>	<u>Kürzel</u>
BGYW17a	BRA
BGYW17a	GHZ
BGYW17a	KLR
BGYGS17a	STL
BGYGS17a	RNR

```
Create Table unterricht
```

```
(
  Bezeichnung varchar (8) NOT NULL,
  Kürzel varchar (3) NOT NULL,
  constraint pk_Bezeichnung primary key (Bezeichnung, Kürzel),
  constraint fk_Bez foreign key(Bezeichnung) references klasse (Bezeichnung),
  constraint fk_Kürzel foreign key (Kürzel) references lehrer (Kürzel)
)
```

## 3.1.3 Aufgabe:

1. Erstellen sie zu dem untenstehenden ERM das dazugehörige relationale Tabellenschema.
2. Erzeugen sie die Tabellen mithilfe von SQL-Befehlen in MySQL.
3. Ergänzen sie in jeder Tabelle mindestens drei ausgedachte Datensätze.



## 3.2 Referentielle Integrität – Beziehungen zwischen Tabellen

Eine Beziehung wird immer durch die Verknüpfung von Fremd- und Primärschlüssel realisiert. Dabei wird der Primärschlüssel als zusätzliche Spalte in der zu verknüpfenden Tabelle hinzugefügt. Diese Spalte beinhaltet dieselben Werte, wie die Primärschlüsselspalte.

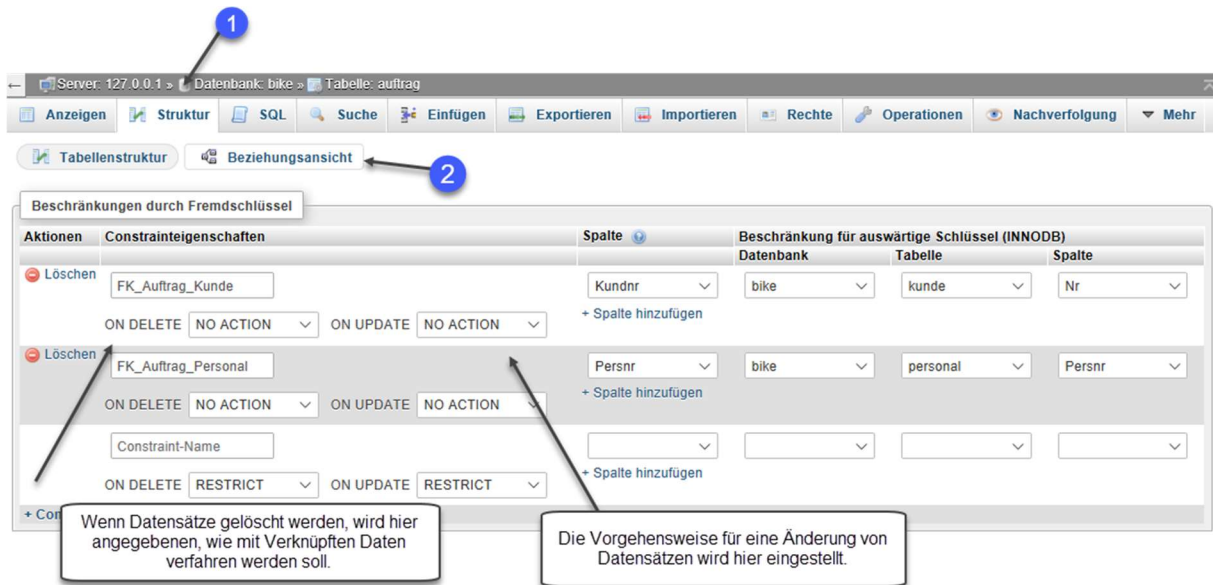
Voraussetzung dafür ist, dass Datentyp und Feldgröße der zu verknüpfenden Attribute gleich sind.

Die Namen der Attribute können unterschiedlich sein. Sollen Datensätze aus der Mastertabelle gelöscht werden, die über einen Fremdschlüssel mit einer Detailtabelle verknüpft sind, müssen Regeln erstellt werden, damit nicht versehentlich noch benötigte Einträge mit gelöscht werden. Das Gleiche gilt für die Änderung von Primärschlüsselwerten.

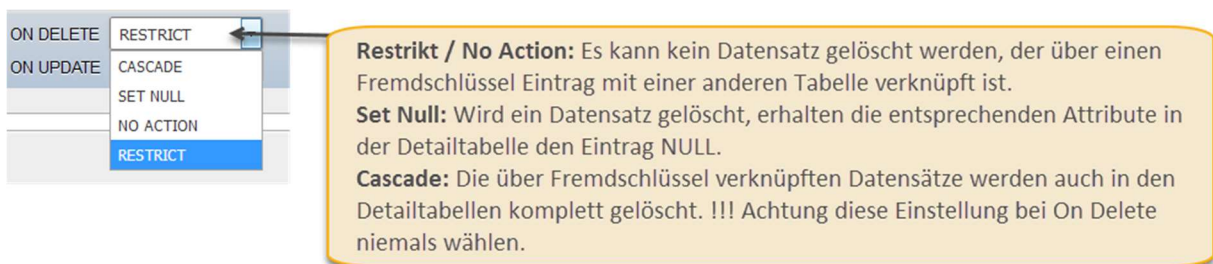
Die referentielle Integrität kann entweder über den SQL Befehl Constraint eingerichtet werden oder über die MySQL Oberfläche.

Dazu wird die entsprechende Tabelle aufgerufen und unter Struktur und Beziehungsübersicht erscheint untenstehende Abbildung (oder ähnlich).

Bei der Darstellung handelt es sich um Tabelle *auftrag* aus der Datenbank *bike*. Falls noch nicht vorhanden, eine Datenbank mit dem Namen *bike* erstellen und die Datei *bike.sql* importieren.



Hier sind zwei Fremdschlüsselbeziehungen eingetragen. Für diese können unterschiedliche Aktionen ausgewählt werden:



Ändern Sie alle Einstellungen auf NO ACTION ab.

## 1. Aufgabe:

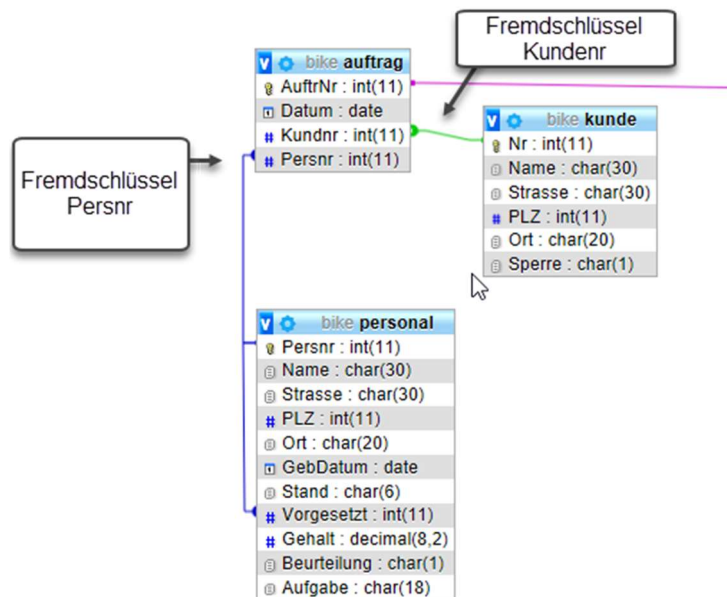
Öffnen sie die Tabelle Kunde.

	Nr	Name	Strasse	PLZ	Ort	Sperre
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	1	Fahrrad Shop	Obere Regenstr. 4	93059	Regensburg	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	2	Zweirad-Center Staller	Kirschweg 20	44267	Dortmund	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	3	Maier Ingrid	Universitätsstr. 33	93055	Regensburg	1
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	4	Rafa - Seger KG	Liebigstr. 10	10247	Berlin	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	5	Biker Ecke	Lessingstr. 37	22087	Hamburg	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	6	Fahrräder Hammerl	Schindlerplatz 7	81739	München	0

Die Firma Biker Ecke stellt ihr Sortiment um, kommt daher für sie als Kunde nicht mehr in Frage. Sie möchten den Datensatz gerne löschen. Allerdings ist der Primärschlüssel Kundenr mit der Tabelle Auftrag über den Fremdschlüssel Kundenr verknüpft und es befindet sich dort auch ein Auftrag mit der Nr.6.

	AuftrNr	Datum	Kundnr	Persnr
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	1	2008-08-04	1	2
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	2	2008-09-06	3	5
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	3	2008-10-07	4	2
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	4	2008-10-18	6	5
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	5	2008-11-03	1	2
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	6	2018-12-13	5	9

Biker Ecke



Versuchen Sie in der Tabelle Kunde den Datensatz mit der Kundennr 5 zu löschen.

	Nr	Name	Strasse	PLZ	Ort	Sperre
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	1	Fahrrad Shop	Obere Regenstr. 4	93059	Regensburg	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	2	Zweirad-Center Staller	Kirschweg 20	44267	Dortmund	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	3	Maier Ingrid	Universitätsstr. 33	93055	Regensburg	1
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	4	Rafa - Seger KG	Liebigstr. 10	10247	Berlin	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	5	Biker Ecke	Lessingstr. 37	22087	Hamburg	0
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	6	Fahrräder Hammerl	Schindlerplatz 7	81739	München	0

Es erscheint die folgende Fehlermeldung:

**#1451 - Kann Eltern-Zeile nicht löschen oder aktualisieren: eine Fremdschlüsselbedingung schlägt fehl ('bike`.`auftrag`, CONSTRAINT 'FK\_Auftrag\_Kunde' FOREIGN KEY ('Kundnr') REFERENCES 'kunde' ('Nr') ON DELETE NO ACTION ON UPDATE CASCADE)**



Die Einstellung ON DELETE NO ACTION sorgt dafür, dass keine Datensätze versehentlich gelöscht werden können, die mit weiteren eventuell noch benötigten Datensätzen verknüpft sind.

Sollen Einträge in Datensätzen geändert werden, wird dies bei RESTRICT /NO ACTION ebenso verweigert.

Die Einstellung ON UPDATE Cascade ermöglicht eine Änderung. Alle weiteren Einträge werden in der Folge angepasst.

### 3. Aufgabe

Ernst Pach mit der Personalnummer 9 hat den Auftrag mit der AuftrNr 6 bearbeitet. Jetzt soll seine Personalnummer von 9 auf 10 geändert werden.

AuftrNr	Datum	Kundnr	Persnr
1	2008-08-04	1	2
2	2008-09-06	3	5
3	2008-10-07	4	2
4	2008-10-18	6	5
5	2008-11-03	1	2
6	2018-12-13	5	9

Ändern sie die Personalnummer von Ernst Pach von 9 auf 10 ab.

	Persnr	Name	Strasse	PLZ
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	1	Maria Forster	Ebertstr. 28	93051
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	2	Anna Kraus	Kramgasse 5	93047
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	3	Ursula Rank	Dreieichstr. 12	60594
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	4	Heinz Rolle	In der Au 5	90455
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	5	Johanna Köster	Wachtelstr. 7	90427
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	6	Marianne Lambert	Fraunhofer Str 3	92224
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	7	Thomas Noster	Mahlergasse 10	93047
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	8	Renate Wolters	Lessingstr. 9	86159
<input type="checkbox"/> Bearbeiten  Kopieren  Löschen	9	Ernst Pach	Olgastr. 99	70180

Hier kann ein Datensatz verändert werden.

AuftrNr	Datum	Kundnr	Persnr
1	2008-08-04	1	2
2	2008-09-06	3	5
3	2008-10-07	4	2
4	2008-10-18	6	5
5	2008-11-03	1	2
6	2018-12-13	5	10

Über Cascade werden die verknüpften Werte mit verändert.

## 4 Sprachkonzepte der Abfragesprache SQL

### 4.1 Grundstruktur

Allgemeiner Aufbau einer SELECT- Anweisung:

Die Teile in den eckigen Klammern sind optional.

SELECT Spaltenliste

FROM Tabellenliste

[WHERE Bedingungsausdruck]

[GROUP BY Spaltenliste]

[HAVING Bedingungsausdruck]

[ORDER BY Spaltenliste]

Dabei sind nur die SELECT und FROM Anweisung zwingend.

SELECT: Welche Spalten sollen ausgegeben werden?

FROM: In welcher Tabelle befinden sich die Spalten?

WHERE: Welche Zeilen sollen in der Ergebnistabelle ausgegeben werden?

**Beispiel:**

```
SELECT Name, Strasse, Ort
FROM kunde
WHERE Ort = 'Regensburg'
```

Name	Strasse	Ort
Fahrrad Shop	Obere Regenstr. 4	Regensburg
Maier Ingrid	Universitätsstr. 33	Regensburg

**Select:** Welche Spalten sollen angezeigt werden?

**From:** Aus welcher Tabelle?

**Where:** Welche Bedingung muss erfüllt sein?

### 4.2 Projektion

Unter Projektion versteht man eine Abfrage, die sich auf eine Auswahl von Spalten beschränkt und keine weitere Bedingung beinhaltet.

```
SELECT *
FROM kunde
```

Nr	Name	Strasse	PLZ	Ort	Sperre
1	Fahrrad Shop	Obere Regenstr. 4	93059	Regensburg	0
2	Zweirad-Center Staller	Kirschweg 20	44267	Dortmund	0
3	Maier Ingrid	Universitätsstr. 33	93055	Regensburg	1
4	Rafa - Seger KG	Liebigstr. 10	10247	Berlin	0
5	Biker Ecke	Lessingstr. 37	22087	Hamburg	0
6	Fahrräder Hammerl	Schindlerplatz 7	81739	München	0

Mit Select \* werden alle Spalten angezeigt.

```
SELECT Name, Strasse, PLZ, Ort
FROM kunde
```

Name	Strasse	PLZ	Ort
Fahrrad Shop	Obere Regenstr. 4	93059	Regensburg
Zweirad-Center Staller	Kirschweg 20	44267	Dortmund
Maier Ingrid	Universitätsstr. 33	93055	Regensburg
Rafa - Seger KG	Liebigstr. 10	10247	Berlin
Biker Ecke	Lessingstr. 37	22087	Hamburg
Fahrräder Hammerl	Schindlerplatz 7	81739	München

Speichern sie alle Abfragen z.B. mit 01\_Aufgabe, usw. ab.

**1. Aufgabe:** Aus der Tabelle Personal soll nur der Name und das Gehalt ausgegeben werden.

## 4.2.1 Sortieren von Abfrageergebnissen

Das Ergebnis einer Abfrage kann nach unterschiedlichen Kriterien sortiert ausgegeben werden. Dies wird durch die ORDER-BY Anweisung erreicht.

**Beispiel:** Die Tabelle Artikel soll aufsteigend nach dem Preis sortiert werden.

```
SELECT *
FROM artikel
ORDER BY Preis
```

Order By Preis: Die Tabelle ist aufsteigend nach dem Preis sortiert.

Anr	Bezeichnung	Netto	Steuer	Preis	Farbe	Mass	Einheit	Typ
500011	Rohr 34CrMo4 2.1mm	3.36	0.64	4.00	NULL	2,1mm	CM	F
500012	Rohr 34CrMo3 2.4mm	3.61	0.69	4.30	NULL	2,4mm	CM	F
500009	Sattelstuetze	4.62	0.88	5.50	NULL	NULL	ST	F
500004	Gruppe Deore XT	5.04	0.96	6.00	NULL	XT	ST	F
500003	Gruppe Deore LX	5.88	1.12	7.00	NULL	LX	ST	F
500001	Rohr 25CrMo4 9mm	6.30	1.20	7.50	NULL	9mm	CM	F
500015	Schlauch	6.72	1.28	8.00	NULL	26.7mm	ST	F

**Beispiel:** Die Tabelle Auftragsposten soll absteigend nach dem Gesamtpreis sortiert werden

Order By Gesamtpreis desc: Gesamtpreis absteigend sortiert.

```
SELECT *
FROM auftragsposten
ORDER BY Gesamtpreis DESC
```

PosNr	AuftrNr	ArtNr	Anzahl	Gesamtpreis
201	2	100002	3	1950.00
101	1	200002	2	800.00
401	4	100001	1	700.00
301	3	100001	1	700.00
202	2	200001	1	400.00



**2. Aufgabe:** Sortieren sie die Tabelle Artikel absteigend nach dem Preis.

**3. Aufgabe:** Sortieren sie die Tabelle Personal aufsteigend nach dem Gehalt

**Beispiel:** Die Tabelle Auftrag soll zuerst nach der Kundennummer sortiert werden und bei gleicher Kundennummer nach dem Datum

```
SELECT *
FROM auftrag
ORDER BY Kundnr, Datum
```

AuftrNr	Datum	Kundnr	Persnr
1	2008-08-04	1	2
5	2008-11-03	1	2
2	2008-09-06	3	5
3	2008-10-07	4	2
6	2018-12-13	5	10
4	2008-10-18	6	5

innerhalb der Kundnr nach Datum sortiert.

Aufsteigend sortiert nach Kundnr

## 4.2.2 Übung

Legen sie eine neue Datenbank *comic* an und importieren sie die Datenbank *comic.sql*.

**4. Aufgabe:** Sortieren sie die Namen in der Tabelle Autoren aufsteigend.

id	name
5	Franquin
2	Goscinny
3	Hergé
4	Kuijpers
1	Uderzo

**5. Aufgabe:** Erstellen sie eine Liste aus der Tabelle Alben, die nur die Spalten Reihe und Titel enthält.

**6. Aufgabe:** Erstellen Sie eine Liste aus der Tabelle Alben, die die Spalten Reihe, Titel und Preis enthält und die absteigend nach dem Preis sortiert ist.

reihe	titel	preis
Franka	Das Kriminalmuseum	8.80
Franka	Das Meisterwerk	8.80
Asterix	Der große Graben	5.00
Asterix	Die Trabantentstadt	3.80
Asterix	Asterix als Legionär	3.00
Asterix	Asterix der Gallier	2.80
Asterix	Asterix und Kleopatra	2.80
Gespenster Geschichten	Gespenster Geschichten	1.20
Tim und Struppi	Der geheimnisvolle Stern	NULL
Tim und Struppi	Tim und der Haifischsee	NULL

**7. Aufgabe:** Erstellen Sie eine Liste aus der Tabelle Alben, die zuerst nach der Reihe und dann nach dem Titel sortiert ist

reihe	titel	band	preis	jahr
Asterix	Asterix als Legionär	10	3.00	NULL
Asterix	Asterix der Gallier	1	2.80	1968
Asterix	Asterix und Kleopatra	2	2.80	1968
Asterix	Der große Graben	25	5.00	1980
Asterix	Die Trabantenstadt	17	3.80	1974
Franka	Das Kriminalmuseum	1	8.80	1985
Franka	Das Meisterwerk	2	8.80	1986
Gespenster Geschichten	Gespenster Geschichten	1	1.20	1974
Tim und Struppi	Der geheimnisvolle Stern	1	NULL	1972
Tim und Struppi	Tim und der Haifischsee	23	NULL	1973

## 4.3 Selektion

Bei einer Selektion werden nur die Datensätze angezeigt, die eine bestimmte Bedingung erfüllen. Bedingungen werden in SQL mit Hilfe der WHERE-Klausel definiert.

**Beispiel:** Welche Alben kosten weniger als 5 € ?

```
SELECT *  
FROM `alben`  
WHERE preis < 5
```

titel	preis
Asterix der Gallier	2.80
Asterix und Kleopatra	2.80
Asterix als Legionär	3.00
Die Trabantenstadt	3.80
Gespenster Geschichten	1.20

**Beispiel:** Welche Titel sind aus der Reihe Asterix ?

```
SELECT titel  
FROM `alben`  
WHERE reihe='Asterix'
```

titel
Asterix der Gallier
Asterix und Kleopatra
Asterix als Legionär
Die Trabantenstadt
Der große Graben

**Achtung!!! Text und Datum in Hochkomma**

### 4.3.1 Zusammengesetzte Bedingungen AND, OR

**Beispiel:** Welche Titel aus der Reihe Asterix kosten 2,80€?

```
SELECT *  
FROM `alben`  
WHERE reihe='Asterix' AND preis=2.8
```

titel
Asterix der Gallier
Asterix und Kleopatra

**Achtung!!! Dezimalpunkt, kein Komma**

**Beispiel:** Welche Titel sind aus der Reihe Asterix oder nach 1980 erschienen?

```
SELECT *  
FROM `alben`  
WHERE reihe = 'Asterix' OR jahr > 1980
```

reihe	titel	jahr
Asterix	Asterix der Gallier	1968
Asterix	Asterix und Kleopatra	1968
Asterix	Asterix als Legionär	NULL
Asterix	Die Trabantenstadt	1974
Asterix	Der große Graben	1980
Franka	Das Kriminalmuseum	1985
Franka	Das Meisterwerk	1986

**Beispiel:** Welche Titel kosten weniger als 3,00€ ? In der Ausgabe sollen auch die Datensätze enthalten sein, bei denen das Attribut preis keinen Eintrag enthält.

```
SELECT *  
FROM `alben`  
WHERE preis < 3 OR preis IS NULL
```

reihe	titel	band	preis	jahr
Asterix	Asterix der Gallier	1	2.80	1968
Asterix	Asterix und Kleopatra	2	2.80	1968
Gespenster Geschichten	Gespenster Geschichten	1	1.20	1974
Tim und Struppi	Der geheimnisvolle Stern	1	NULL	1972
Tim und Struppi	Tim und der Haifischsee	23	NULL	1973

## 4.3.2 Verneinte Bedingung (NOT)

**Beispiel:** Welche Titel sind nicht aus der Reihe Asterix?

```
SELECT *  
FROM `alben`  
WHERE NOT reihe = 'Asterix'
```

reihe	titel	band	preis	jahr
Franka	Das Kriminalmuseum	1	8.80	1985
Franka	Das Meisterwerk	2	8.80	1986
Gespenster Geschichten	Gespenster Geschichten	1	1.20	1974
Tim und Struppi	Der geheimnisvolle Stern	1	NULL	1972
Tim und Struppi	Tim und der Haifischsee	23	NULL	1973

## 4.3.3 Bereichseinschränkung (BETWEEN)

**Beispiel:** Welche Titel kosten zwischen drei und fünf Euro?

```
SELECT *
FROM `alben`
WHERE preis BETWEEN 3 AND 5
```

reihe	titel	band	preis	jahr
Asterix	Asterix als Legionär	10	3.00	NULL
Asterix	Die Trabantstadt	17	3.80	1974
Asterix	Der große Graben	25	5.00	1980

## 4.3.4 Einschränkungen auf Listenwerte (IN)

**Beispiel:** Welche Titel sind im Jahr 1968, 1986 oder 1972 erschienen?

```
SELECT *
FROM `alben`
WHERE jahr IN (1968, 1986, 1972)
```

reihe	titel	band	preis	jahr
Asterix	Asterix der Gallier	1	2.80	1968
Asterix	Asterix und Kleopatra	2	2.80	1968
Franka	Das Meisterwerk	2	8.80	1986
Tim und Struppi	Der geheimnisvolle Stern	1	NULL	1972

## 4.3.5 Datumsfunktionen

Funktion	Beschreibung
SECOND()	Gibt die Sekunden zurück.
MINUTE()	Gibt die Minuten zurück.
HOURL()	Gibt die Stunden zurück.
DAY()	Gibt den Tag zurück.
DAYOFWEEK()	Gibt den Wochentag (1=Sonntag, 7=Samstag) zurück.
MONAT()	Gibt den Monat (als Zahl) zurück.
YEAR()	Gibt das Jahr zurück.
NOW()	Gibt das aktuelle Datum und die aktuelle Zeit zurück ('YYYY-MM-DD HH:MM:SS').
DATE_SUB()	Zieht vom ersten Argument eine Zeitspanne ab.
DATE_ADD()	Addiert auf das erste Argument eine Zeitspanne.

## 4.3.6 Aufgaben:

**8. Aufgabe:** Welche Titel gehören zu einem Band mit der Nr. 1?

KNr	Name	FilialNr
1	Müller	2
2	Meier	2

**9. Aufgabe:** Welche Reihe ist im Jahr 1972 erschienen?

```
reihe
Tim und Struppi
```

Die weiteren Aufgaben sind mit der Datenbank eurodata zu bearbeiten. Achten sie auf eine sinnvolle Auswahl der angezeigten Spalten. Sie können sich an den vorgegebenen Ausgabewerten orientieren.

**10. Aufgabe:** Bei welchen Projekten beträgt der Auftragswert mehr als 25.000,00€?

Bezeichnung	Auftragswert
Umzug Stein AG	50000.00
Aufbau Intranet, Druckerei Wolf	31500.00
Vernetzung Bau AG	105000.00
Vernetzung, Großmann	25200.00
DV-Labor, BBS III	42000.00
Umzug Stadtanzeiger	189000.00
Win-Server, Lacke Hansen	31500.00
Office-Schulung, BATIX	26250.00

**11. Aufgabe:** Welche Mitarbeiter wurden vor 1998 eingestellt?

Name	Vorname	Einstellungsdatum
Meyer	Inge	1995-01-01
Schneider	Ute	1995-01-01
Berg	Karin	1995-09-01
Wasser	Ute	1995-12-01
Huber	Sepp	1996-02-01
Schmitz	Hans	1996-05-01
Schmidt	Jürgen	1996-05-01
Klein	Kurt	1996-09-15
Huber	Anke	1996-12-01
Schmidt	Beate	1997-04-01
Bauer	Klaus	1997-04-01
Meier	Anton	1997-06-01

**12. Aufgabe:** Welche Projekte mit einem Auftragswert unter 30.000,00€ wurden nicht storniert?

Bezeichnung	Auftragswert
Einrichtung Apache-Web-Server, Moberg Gm	8400.00
Vernetzung, Großmann	25200.00
Linux Server Stadtverwaltung Brühl	15750.00
Office-Schulung, BATIX	26250.00
Umstellung Windows- auf Linux-Server, Zu	9975.00
Schulung, Naumann & Co.	5250.00

**13. Aufgabe:** Welche Projekte unter 30.000,00€ oder ab 100.000,00€ wurden storniert?

Bezeichnung	Auftragswert
Vernetzung Bau AG	105000.00
Umrüstung Neumann OHG	12600.00
Adabas SQL-Server, MTech	21000.00

**14. Aufgabe:** Welche Projekte haben einen Auftragswert von 20.000,00€ bis einschließlich 50.000,00€?

Bezeichnung	Auftragswert
Umzug Stein AG	50000.00
Aufbau Intranet, Druckerei Wolf	31500.00
Vernetzung, Großmann	25200.00
DV-Labor, BBS III	42000.00
Win-Server, Lacke Hansen	31500.00
Adabas SQL-Server, MTech	21000.00
Office-Schulung, BATIX	26250.00

**15. Aufgabe:** Bei welchen Projekten wurde noch keine Zahlung geleistet?

PNR	Bezeichnung	Auftragswert
78	Einrichtung Apache-Web-Server, Moberg Gm	8400.00
80	Vernetzung Bau AG	105000.00
82	DV-Labor, BBS III	42000.00
85	Umrüstung Neumann OHG	12600.00
86	Linux Server Stadtverwaltung Brühl	15750.00
87	Win-Server, Lacke Hansen	31500.00
89	Office-Schulung, BATIX	26250.00
90	Umstellung Windows- auf Linux-Server, Zu	9975.00
91	Schulung, Naumann & Co.	5250.00

**16. Aufgabe:** Welche Mitarbeiter wohnen nicht in Köln?

Name	Vorname	Ort
Meyer	Inge	Bonn
Berg	Karin	Wiesbaden
Wasser	Ute	Aachen
Huber	Sepp	München
Schmidt	Jürgen	Hamburg
Kammer	Ulrike	Würselen
Klein	Kurt	Karlsruhe
Huber	Anke	Augsburg
Schmidt	Beate	Düsseldorf
Bauer	Klaus	Hannover
Meier	Anton	Freiburg
Meier	Doris	Suhl
Schmidt	Barbara	Chemnitz
Walczak	Hildegard	Dresden
Lausen	Werner	Euskirchen
Görner	Heidrun	Köln-Porz
Preuß	Karl	Eschweiler
Messner	Bärbel	Pulheim

**17. Aufgabe:** Welche Mitarbeiter wohnen in Chemnitz, Dresden oder Suhl?

Name	Vorname	Ort
Meier	Doris	Suhl
Schmidt	Barbara	Chemnitz
Walczak	Hildegard	Dresden



## 4.3.7 Mustervergleiche

**Beispiel:** Welche Titel beginnen mit Asterix ?

```
SELECT *
FROM `alben`
WHERE titel LIKE 'Asterix%'
```

% steht für beliebig viele Zeichen.

reihe	titel	band	preis	jahr
Asterix	Asterix der Gallier	1	2.80	1968
Asterix	Asterix und Kleopatra	2	2.80	1968
Asterix	Asterix als Legionär	10	3.00	NULL

**Beispiel:** Es wird eine Reihe gesucht, die mit *Fran* beginnt und mit *a* endet. Es ist nicht bekannt, ob die Reihe mit k oder c geschrieben wird.

```
SELECT *
FROM `alben`
WHERE reihe LIKE 'Fran_a'
```

Ist genau ein Zeichen nicht bekannt wird der Unterstrich\_ verwendet

reihe	titel	band	preis	jahr
Franka	Das Kriminalmuseum	1	8.80	1985
Franka	Das Meisterwerk	2	8.80	1986

**Beispiel:** Es werden Autoren gesucht, deren Namen mit einem Buchstaben zwischen 'G' und 'K' beginnen.

```
SELECT *
FROM `autoren`
WHERE name RLIKE '^[G-K]'
```

id	name
2	Goscinny
3	Hergé
4	Kuijpers

**Beispiel:** Es werden Reihen gesucht, die so ähnlich wie Aschteriks klingen.

```
SELECT reihe, titel
FROM `alben`
WHERE SOUNDEX (reihe) = SOUNDEX('Aschteriks')
```

reihe	titel
Asterix	Asterix der Gallier
Asterix	Asterix und Kleopatra
Asterix	Asterix als Legionär
Asterix	Die Trabantenstadt
Asterix	Der große Graben

## 4.3.8 Aufgaben

**19. Aufgabe:** Bei welchen Mitarbeitern beginnt der Name mit 'Sch' ?

Name	Vorname
Schneider	Ute
Schmitz	Hans
Schmidt	Jürgen
Schmidt	Beate
Schmidt	Barbara
Schmidt	Klaus

**20. Aufgabe:** Welche Projekte haben etwas mit Servereinrichtung oder –installation zu tun?

PNr PS	Bezeichnung	Auftragswert	Anzahlung	Beginn	Ende	Storno	P_Leiter FS (MNr.Mitarbeiter)
78	Einrichtung Apache-Web-Server, Moberg Gm	8400.00	NULL	2006-01-01	2006-07-15	n	810
86	Linux Server Stadtverwaltung Brühl	15750.00	0.00	2006-08-10	0000-00-00	n	809
87	Win-Server, Lacke Hansen	31500.00	0.00	2006-06-01	2006-06-10	j	803
88	Adabas SQL-Server, MTech	21000.00	20000.00	2006-09-01	0000-00-00	j	804
90	Umstellung Windows- auf Linux-Server, Zu	9975.00	0.00	2006-10-08	0000-00-00	n	200

**21. Aufgabe:** Eine Telefonnotiz ist kaum noch lesbar. Es ist zu erkennen, dass die Mitarbeiternummer des Leiters mit '80' beginnt. Aus dem Text ist zudem ersichtlich, dass noch keine Zahlung geleistet wurde. Für welche Projekte trifft dies zu?

PNr PS	Bezeichnung	Auftragswert	Anzahlung	Beginn	Ende	Storno	P_Leiter FS (MNr.Mitarbeiter)
80	Vernetzung Bau AG	105000.00	NULL	2006-07-01	2006-07-31	j	800
86	Linux Server Stadtverwaltung Brühl	15750.00	0.00	2006-08-10	0000-00-00	n	809
87	Win-Server, Lacke Hansen	31500.00	0.00	2006-06-01	2006-06-10	j	803
89	Office-Schulung, BATIX	26250.00	0.00	2006-10-01	0000-00-00	n	803

**22. Aufgabe:** Welche Mitarbeiter wohnen nicht im Postleitzahlbereich 5?

**23. Aufgabe:** Welche Mitarbeiter wohnen im Postleitzahlbereich 7 oder 8?

**24. Aufgabe:** Welche Mitarbeiter heißen Schmitt oder so ähnlich?

MNr PS	Name	Vorname	Strasse
209	Schmidt	Jürgen	Osdorfer Weg 7
710	Schmidt	Beate	Klosterstr. 1
803	Schmidt	Barbara	Rößlerstr. 33
805	Schmidt	Klaus	Vogelsanger Str. 77

**25. Aufgabe:** Bei welchen Mitarbeitern steht an der 2. Stelle der dreistelligen Mitarbeiternummer eine 1?



# Datenbanksysteme

MNr PS	Name	Vorname
611	Kammer	Ulrike
710	Schmidt	Beate
711	Bauer	Klaus
810	Messner	Bärbel

**26. Aufgabe:** Welche Mitarbeiterin stammt aus dem Raum Köln und wurde 1998 eingestellt?

MNr PS	Name	Vorname	Strasse	PLZ	Ort	Telefon	Geschlecht	Einstellungsdatum	Abteilung FS (ID Abteilung)
807	Görner	Heidrun	Waldstr.97	51145	Köln-Porz	02203/912778	w	1998-06-01	02

**27. Aufgabe:** Welche nicht stornierten Projekte haben einen Auftragswert unter 50.000€ und werden von dem Mitarbeiter mit der Mitarbeiternummer 200 geleitet?

PNr PS	Bezeichnung	Auftragswert	Anzahlung	Beginn	Ende	Storno	P_Leiter FS (MNr.Mitarbeiter)
79	Aufbau Intranet, Druckerei Wolf	31500.00	7500.00	2006-09-01	0000-00-00	n	200
90	Umstellung Windows- auf Linux-Server, Zu	9975.00	0.00	2006-10-08	0000-00-00	n	200

**28. Aufgabe:** Welche männlichen Mitarbeiter haben Mitarbeiternummern unter 200 oder über 700 oder wohnen in Hamburg?

MNr PS	Name	Vorname	Strasse	PLZ	Ort
209	Schmidt	Jürgen	Osdorfer Weg 7	22607	Hamburg
711	Bauer	Klaus	Spielhagenstr. 113	30107	Hannover
800	Meier	Anton	Tiroler Weg 22	79111	Freiburg
805	Schmidt	Klaus	Vogelsanger Str. 77	50666	Köln
806	Lausen	Werner	Münstereifeler Str. 22	53879	Euskirchen
809	Preuß	Karl	Peter-Paul-Str. 67	52249	Eschweiler

**29. Aufgabe:** Welche Projekte wurden im 3. Quartal eines Jahres begonnen?

Die Funktion Quarter(Datum) gibt das Quartal des Jahres im Bereich 1 bis 4 zurück.

PNr PS	Bezeichnung	Auftragswert	Anzahlung	Beginn
79	Aufbau Intranet, Druckerei Wolf	31500.00	7500.00	2006-09-01
80	Vernetzung Bau AG	105000.00	NULL	2006-07-01
81	Vernetzung, Großmann	25200.00	5000.00	2006-07-05
82	DV-Labor, BBS III	42000.00	0.00	2006-08-01
84	Umzug Stadtanzeiger	189000.00	30000.00	2006-08-01
85	Umrüstung Neumann OHG	12600.00	0.00	2006-07-01
86	Linux Server Stadtverwaltung Brühl	15750.00	0.00	2006-08-10
88	Adabas SQL-Server, MTech	21000.00	20000.00	2006-09-01

**30. Aufgabe:** Erstellen sie eine Liste der nicht stornierten Projekte, die nach der Projektbezeichnung sortiert ist.

**31. Aufgabe:** Welche Mitarbeiter heißen Meier oder so ähnlich?

**32. Aufgabe:** Bei welchen Mitarbeitern beginnt der Name mit 'B' oder 'H' und endet mit 'er'?

**33. Aufgabe:** Erstellen sie eine Liste mit den 3 jüngsten Kindern des Betriebskindergartens.

Hinweis: Mit LIMIT kann die Anzahl an Datensätzen angegeben werden.

Vorname	Geburtsdatum
PS	
Fritz	2013-11-12
Patrick	2003-09-01
Pia	2003-08-05

## 4.3.9 Abfragen mit Berechnungen

**Beispiel:** Ab einer Bestellmenge von 10 Titeln wird ein Rabatt von 10 % gewährt. Erstellen sie eine Tabelle, die den reduzierten Preis in einer separaten Spalte ausgibt.

```
SELECT titel, preis, preis * 0.9 As red_Preis
FROM `alben`
```

Mit AS wird eine neue Spaltenüberschrift erzeugt.

titel	preis	red_Preis
Asterix der Gallier	2.80	2.520
Asterix und Kleopatra	2.80	2.520
Asterix als Legionär	3.00	2.700
Die Trabantenstadt	3.80	3.420
Der große Graben	5.00	4.500
Das Kriminalmuseum	8.80	7.920
Das Meisterwerk	8.80	7.920
Gespenster Geschichten	1.20	1.080
Der geheimnisvolle Stern	NULL	NULL
Tim und der Haifischsee	NULL	NULL

## 4.3.10 Aufgaben:

**34. Aufgabe:** Wie hoch sind die Auftragswerte, die MwSt (19%) und die Auftragswerte inkl. MwSt der einzelnen Projekte? Die Tabelle soll folgendermaßen formatiert sein.

Bezeichnung	Auftragswert	MWSt	Auftragswert2
Umzug Stein AG	50000.00	9,500.00	59,500.00
Einrichtung Apache-Web-Server, Moberg Gm	8400.00	1,596.00	9,996.00

**35. Aufgabe:** Wie hoch sind die Außenstände der einzelnen Projekte (Differenz von Auftragswert und bereits bezahltem Betrag).

Bezeichnung	Auftragswert	Außenstand
Umzug Stein AG	50000.00	38000.00
Einrichtung Apache-Web-Server, Moberg Gm	8400.00	NULL
Aufbau Intranet, Druckerei Wolf	31500.00	24000.00
Vernetzung Bau AG	105000.00	NULL
Vernetzung, Großmann	25200.00	20200.00
DV-Labor, BBS III	42000.00	42000.00

## 4.3.11 Abfragen mit Gruppierungen

Datensätze können zu Gruppen zusammengefasst und tabellarisch angeordnet werden. Dabei lassen sich für jede Gruppe unter Einsatz so genannter Aggregatfunktionen zusammenfassende Berechnungen (z.B. Summe, Mittelwert, Minimalwert, Maximalwert, Anzahl, usw.) ausgeben.

**Beispiel:** Wie hoch ist die nach Projektleitern zusammengefasste Summe der Auftragswerte?

Sum bildet die Summe des ausgewählten Attributs.

```
SELECT P_Leiter, SUM(Auftragswert) As SummeAuftragswert
FROM `projekt`
GROUP BY P_Leiter
```

Group By legt fest, welche Einträge summiert werden sollen

P_Leiter FS (MNr.Mitarbeiter)	SummeAuftragswert
100	75200.00
101	42000.00
200	41475.00
700	12600.00
800	105000.00
803	57750.00
804	21000.00
805	189000.00
809	15750.00
810	13650.00

**Beispiel:** Wie viele Mitarbeiter stammen aus den einzelnen Postleitzahl-Gebieten? Die Gebiete sollen anhand der ersten Stelle der Postleitzahl gruppiert werden.

Mit der Funktion LEFT wird aus einer Zeichenkette eine beliebige Anzahl Zeichen von links beginnend herausgelöst.

```
SELECT LEFT (PLZ,1) AS PLZ_Gebiet, COUNT(PLZ) As Anzahl1
FROM Mitarbeiter
GROUP BY LEFT (PLZ,1)
```

Hier wird angegeben nach welchem Kriterium die Einträge gezählt werden sollen. In diesem Fall wird die erste Stelle der PLZ ausgewertet.

PLZ_Gebiet	Anzahl
0	2
2	1
3	1
4	1
5	10
6	1
7	2
8	2
9	1

COUNT zählt die vorkommenden Einträge des ausgewählten Attributes.

## 4.3.12 Aufgaben

**36. Aufgabe:** Wie hoch ist die nach Projektleitern zusammengefasste Summe der Auftragswerte der nicht stornierten Projekte?

**37. Aufgabe:** Wie viele Mitarbeiter stammen aus dem Raum Köln?

Anzahl
4

**38. Aufgabe:** Wie hoch ist die Anzahl der weiblichen und die Anzahl der männlichen Mitarbeiter?

Geschlecht	Anzahl
m	9
w	12

**39. Aufgabe:** Wie viele Mitarbeiter sind in den einzelnen Projekten tätig?

PNr PS_FS (PNr.Projekt)	Anzahl
77	3
78	2
79	2
80	6
81	4
82	3
84	3
85	1
86	2

**40. Aufgabe:** Wie viele Mitarbeiter wurden in den einzelnen Jahren eingestellt? Die Funktion YEAR ermittelt aus einem Datum die Jahreszahl.

Einstellungsjahr	Anzahl
1995	4
1996	5
1997	3
1998	9

**41. Aufgabe:** Ermitteln sie in einer Abfrage die Höhe des maximalen Auftragswertes, die Höhe des minimalen Auftragswertes und die durchschnittliche Höhe der Auftragswerte. Der Mittelwert wird über die Funktion AVG bestimmt.

MAX_Auftragswert	MIN_Auftragswert	Durchschnittlicher_Auftragswert
189000.00	5250.00	40958.928571

## 4.4 Verbundabfragen

Eine Abfrage, an der mehr als eine Tabelle beteiligt ist nennen wir Verbund oder Verknüpfung. Bei Verbundabfragen kann das Problem auftreten, dass Attributnamen nicht mehr eindeutig sind. In diesem Fall muss dem Attributnamen der Name der Tabelle vorangestellt werden.

**Beispiel:** Welche Mitarbeiter (Vorname, Nachname) leiten die einzelnen Projekte.

Vorgehensweise:

- Welche Tabellen benötigen wir?  
*Mitarbeiter, Projekt*
- Welches ist das gemeinsame Attribut über das die beiden Tabellen verknüpft werden können (Primär- Fremdschlüssel Verknüpfung) ?  
*MNr in der Tabelle Mitarbeiter wird zu P\_Leiter in der Tabelle Projekt*

```
SELECT Name, Vorname
FROM mitarbeiter, projekt
WHERE Mitarbeiter.MNr=Projekt.P_Leiter
```

Name	Vorname
Meyer	Inge
Meyer	Inge
Schneider	Ute
Huber	Sepp
Huber	Sepp
Klein	Kurt
Meier	Anton
Schmidt	Barbara
Schmidt	Barbara
Walczak	Hildegard
Schmidt	Klaus
Preuß	Karl
Messner	Bärbel
Messner	Bärbel

Um die doppelten Einträge zu vermeiden wird der Befehl **DISTINCT** verwendet:

```
SELECT DISTINCT Name, Vorname
FROM mitarbeiter, projekt
WHERE Mitarbeiter.MNr=Projekt.P_Leiter
```

Die Gleichheitsbedingung hinter der WHERE Klausel kann alternativ als INNER-JOIN Verknüpfung hinter der ON Klausel ausgeführt werden.

```
SELECT DISTINCT Name, Vorname
FROM mitarbeiter INNER JOIN projekt
ON Mitarbeiter.MNr=Projekt.P_Leiter
```

**Beispiel:** Mit welchen Stundenanteilen arbeiten welche Mitarbeiter in den einzelnen Projekten. Die Tabelle (Name, Vorname, Projektbezeichnung und Zeitanteil) soll aufsteigend nach Name und Vorname der Mitarbeiter geordnet sein.

```
Select Mitarbeiter.Name, Mitarbeiter.Vorname, Projekt.Bezeichnung, Projektmitarbeiter.Zeitanteil
From Mitarbeiter, Projekt, Projektmitarbeiter
Where Projektmitarbeiter.MNr=Mitarbeiter.MNr and Projektmitarbeiter.PNr = Projekt.PNr
Order By Mitarbeiter.Name, Mitarbeiter.Vorname
```



Name	Vorname	Bezeichnung	Zeitanteil
Bauer	Klaus	Vernetzung Bau AG	3
Bauer	Klaus	Adabas SQL-Server, MTech	5
Berg	Karin	Vernetzung Bau AG	20
Görner	Heidrun	Vernetzung, Großmann	10
Görner	Heidrun	Office-Schulung, BATIX	10
Huber	Anke	Umzug Stein AG	5
Huber	Anke	Aufbau Intranet, Druckerei Wolf	30
Huber	Sepp	Umstellung Windows- auf Linux-Server, Zu	5
Huber	Sepp	Aufbau Intranet, Druckerei Wolf	10
Klein	Kurt	Umrüstung Neumann OHG	35
Klein	Kurt	Umzug Stadtanzeiger	3
Lausen	Werner	Vernetzung, Großmann	10
Meier	Anton	Schulung, Naumann & Co.	12
Meier	Anton	Vernetzung Bau AG	5
Meier	Doris	Umstellung Windows- auf Linux-Server, Zu	25
Messner	Bärbel	Einrichtung Apache-Web-Server, Moberg Gm	8
Messner	Bärbel	Schulung, Naumann & Co.	5

Um sich Schreibarbeit zu sparen können sinnvolle Aliasnamen verwendet werden.

```

Select M.Name, M.Vorname, P.Bezeichnung, PM.Zeitanteil
From Mitarbeiter as M, Projekt as P, Projektmitarbeiter as PM
Where M.MNr=PM.MNr and P.PNr = PM.PNr
Order BY M.Name, M.Vorname

```

Als Alternative mit INNER JOIN:

```

SELECT M.Name, M.Vorname, P.Bezeichnung, PM.Zeitanteil
FROM `mitarbeiter` AS M
INNER JOIN Projektmitarbeiter AS PM
ON M.MNr = PM.MNr
INNER JOIN Projekt as P
ON PM.PNr = P.PNr
ORDER BY M.Name, M.Vorname

```

## 4.4.1 Aufgaben

Die **Ausgaben** sollen genau den abgebildeten Tabellen entsprechen!!!

**42. Aufgabe:** Welche Mitarbeiter leiten die einzelnen Projekte?

Name	Vorname	PNr	Bezeichnung
Huber	Sepp	90	Umstellung Windows- auf Linux-Server, Zu
Huber	Sepp	79	Aufbau Intranet, Druckerei Wolf
Klein	Kurt	85	Umrüstung Neumann OHG
Meier	Anton	80	Vernetzung Bau AG
Messner	Bärbel	78	Einrichtung Apache-Web-Server, Moberg Gm
Messner	Bärbel	91	Schulung, Naumann & Co.
Meyer	Inge	77	Umzug Stein AG
Meyer	Inge	81	Vernetzung, Großmann
Preuß	Karl	86	Linux Server Stadtverwaltung Brühl
Schmidt	Barbara	87	Win-Server, Lacke Hansen
Schmidt	Klaus	84	Umzug Stadtanzeiger
Schmidt	Barbara	89	Office-Schulung, BATIX
Schneider	Ute	82	DV-Labor, BBS III
Walczak	Hildegard	88	Adabas SQL-Server, MTech

**43. Aufgabe:** Erstellen sie eine Abfrage, die Auskunft über Projektnummer, Projektbezeichnung, Auftragswert, Leiter und Telefonnummer bietet.

# Datenbanksysteme

PNr	Bezeichnung	Auftragswert	Leiter	Vorname	Telefon
77	Umzug Stein AG	50000.00	Meyer	Inge	0228/124578
78	Einrichtung Apache-Web-Server, Moberg Gm	8400.00	Messner	Bärbel	02238/980021
79	Aufbau Intranet, Druckerei Wolf	31500.00	Huber	Sepp	089/84744397
80	Vernetzung Bau AG	105000.00	Meier	Anton	0761/443444
81	Vernetzung, Großmann	25200.00	Meyer	Inge	0228/124578
82	DV-Labor, BBS III	42000.00	Schneider	Ute	0221/215211
84	Umzug Stadtanzeiger	189000.00	Schmidt	Klaus	0221/561112
85	Umrüstung Neumann OHG	12600.00	Klein	Kurt	0721/9812124
86	Linux Server Stadtverwaltung Brühl	15750.00	Preuß	Karl	02403/151515
87	Win-Server, Lacke Hansen	31500.00	Schmidt	Barbara	0371/525212
88	Adabas SQL-Server, MTech	21000.00	Walczak	Hildegard	0351/788285
89	Office-Schulung, BATIX	26250.00	Schmidt	Barbara	0371/525212
90	Umstellung Windows- auf Linux-Server, Zu	9975.00	Huber	Sepp	089/84744397
91	Schulung, Naumann & Co.	5250.00	Messner	Bärbel	02238/980021

**44. Aufgabe:** Welcher Mitarbeiter (Name, Vorname, Telefon) kann von der Erzieherin angesprochen werden, wenn es Probleme mit einem Kind gibt?

Vorname	Name	Vorname	Telefon
Pia	Meyer	Inge	0228/124578
Benedikt	Schneider	Ute	0221/215211
Heidi	Schneider	Ute	0221/215211
Martin	Schneider	Ute	0221/215211
Gudrun	Schmitz	Hans	0221/814447
Patrick	Huber	Anke	0821/98666
Daniela	Bauer	Klaus	0511/833883
Sarah	Bauer	Klaus	0511/833883
Toni	Bauer	Klaus	0511/833883
Fritz	Meier	Anton	0761/443444
Daniela	Schmidt	Klaus	0221/561112
Tobias	Schmidt	Klaus	0221/561112
Kathrin	Preuß	Karl	02403/151515

**45. Aufgabe:** Wie viel Kinder haben die einzelnen Mitarbeiter?

Name	Vorname	Anzahl
Bauer	Klaus	3
Huber	Anke	1
Meier	Anton	1
Meyer	Inge	1
Preuß	Karl	1
Schmidt	Klaus	2
Schmitz	Hans	1
Schneider	Ute	3

**46. Aufgabe:** Wie viel Mitarbeiter arbeiten in den einzelnen Projekten?

# Datenbanksysteme

---

Bezeichnung	Anzahl
Adabas SQL-Server, MTech	3
Aufbau Intranet, Druckerei Wolf	2
DV-Labor, BBS III	3
Einrichtung Apache-Web-Server, Moberg Gm	2
Linux Server Stadtverwaltung Brühl	2
Office-Schulung, BATIX	2
Schulung, Naumann & Co.	3
Umrüstung Neumann OHG	1
Umstellung Windows- auf Linux-Server, Zu	2
Umzug Stadtanzeiger	3
Umzug Stein AG	3
Vernetzung Bau AG	6
Vernetzung, Großmann	4
Win-Server, Lacke Hansen	1