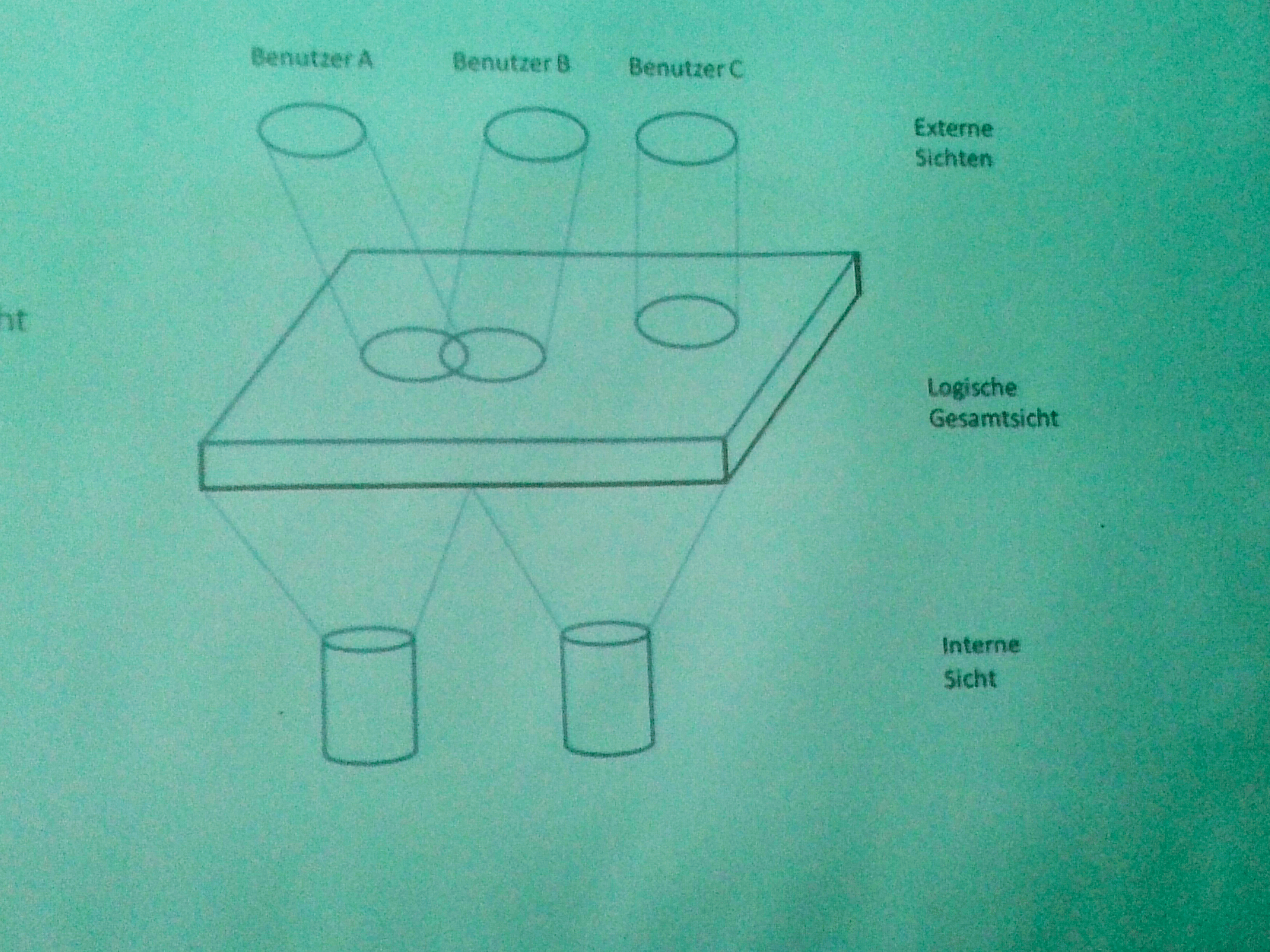
# Themenübersicht

* Was ist eine Datenbank?
* Anforderungen an eine Datenbank
* SQL

#### Allgemein Datenbanken

* … dienen dazu Daten (nicht Informationen) elektronisch zu verwalten
* Unterschied Daten / Informationen
  + Daten: z.B.: Zahl 42
  + Information: Im Lager sind noch 42 Stücke des Produktes XY
* Wesentliche Aufgaben:
  + Große Datenmengen effizient, widerspruchsfrei und dauerhaft zu speichern
  + Teilmengen des Datenbestandes soll bestimmten Benutzern/Anwendungen zugänglich gemacht werden.
* Datenbank besteht im wesentlichen aus 2 Teilen:
  + Datenbasis (Menge der zu verwaltenden Daten)
  + Datenbankmanagementsystem (DBMS)
    - dient der Verwaltung der Daten
    - zur Strukturierung der Daten
    - Regelt Zugriffe (read/write) auf die Daten
    - Bietet eine Datenbanksprache zum Zugriff und Verwaltung der Daten an
* Unter Datenbanken versteht man ein System zur Beschreibung, Speicherung und Wiedergewinnung von Umfangreichen Datenmengen
* Eine Datenbank ist eine integrierte Ansammlung von Daten, die allen Benutzern eines Anwendungsbereiches als gemeinsame Basis aktueller Information dient.
* Typische Anwendungen:
  + Bank (Buchungen, Kontoverwaltung)
  + Bibliothek (Volltextsuche, Entleihe)
  + Redaktionssysteme im Internet (CMS) (Dokumente erstellen, Struktur einer Webseite)
  + eBusiness (Auftrag, Katalog, Controlling)
  + ERP (Personal, Buchhaltung, Controlling)
  + Support (Ticketerwaltung)
  + ….
* Datenbanken/DBMS – Anforderungen:
  + Persistenz
    - Daten sollen dauerhaft gespeichert werden und zu einem späteren Zeitpunkt auch wieder abrufbar sein.
  + Anlegen von Datenschemata
    - Daten haben je nach Kontext unterschiedliche Bedeutung. Ein Schema (z.B.: Tabelle) stellt den Zusammenhang zwischen Daten und Kontext her.
  + Einfügen, Ändern, Löschen von Daten
    - Möglichkeit Daten in das Datenbankschema einzutragen, zu ändern oder auch wieder zu löschen. Eine Zeile im Datenschema wird als Datensatz bezeichnet.
  + Lesen von Daten
    - Es muss möglich sein Daten aus einer Datenbank wieder aufzufinden.
  + Integrität und redundanzfreie Datenhaltung
    - Möglichkeit dass ein Datum, welches an mehreren Stellen benutzt wird, nur an einer Stelle hinterlegt ist. Sicherstellung das Änderungen des Datums an alle benutzten Stellen im System propagiert wird.
  + Koordination der parallelen Nutzung
    - Sicherstellung dass Integrität der Datenbank bei parallelen Zugriffen nicht verloren geht. Jeder Nutzer muss den Eindruck haben, das ihm die Datenbank exklusiv gehört.
  + Rechteverwaltung
    - Unterschiedliche Benutzer der Datenbank sollen unterschiedliche Berechtigungen besitzen
  + Datensicherung
    - DBMS ermöglicht eine Datensicherung des aktuellen Datenbankbestandes herzustellen und diesen auch wieder in das System zurückspielen.
  + Katalog
    - Möglichkeit die Struktur des gesamten Systems bestimmten Benutzern zugänglich zu machen. (Datenschema, Nutzerrechte, …)

Je nach Anwender, ergeben sich unterschiedliche Sichten auf die DB/DBMS. Ein Benutzer sieht z.B.

 die Daten völlig anders als der Systemprogrammierer, der die Organisation der Daten auf den Speichern festlegt.

1. Externe Ebene/Schicht
2. Logische Ebene/Schicht
3. Physikalische/Interne Ebene/Schicht

#### Ebenen eines DBMS

1. Externe Ebene/Schicht
   * Jede Benutzergruppe sieht den Ausschnitt der Datenbank, der für sie von Bedeutung ist. Die Daten werden so dargestellt, wie es für die Benutzer wünschenswert oder natürlich ist.
2. Logische Ebene/Schicht
   * In der Datenbank sind alle wichtigen Daten zusammengefasst. Um die Datenbank erstellen zu können, ist eine Gesamtschau der Daten notwendig. Alle Daten müssen zunächst auf logischer Ebene in Form von Informationseinheiten und deren Beziehung untereinander beschreiben werden, unabhängig von EDV-Gesichtspunkten. Diese Beschreibung der Gesamtheit der Unternehmensdaten nennen wir logische Gesamtsicht.
3. Physikalische / Interne Ebene
   * Die Daten müssen auf den Speicher so organisiert werden, dass die Zugriffsanforderungen der verschiedenen Benutzer möglichst effizient erfüllt werden können.

Die Benutzer arbeiten mit der Datenbank ausschließlich über ihre externen Sichten. Sie sehen weder die logische Gesamtschicht noch die interne Organisation der Daten. Die notwendigen Umsetzungen von einer externen Sicht in die logische Gesamtschicht und von dort in die interne Sicht erledigt das DBMS. Es benötigt dazu Beschreibungen der jeweiligen Sichten und Regeln, die die Umsetzung von einer Sicht in die andere ermöglichen.

Jede Ebene der Daten modelliert die Daten auf einem anderen Abstraktionsniveau. Diese Modelle der Datenwelt werden mit Hilfe sogenannter Datenbeschreibungssprachen in einer für das DBMS verständlichen Form beschrieben; diese Beschreibung heißt dann Schema. Es gibt also verschiedene externe Schemata, ein konzeptuelles Schema, das die logische Gesamtsicht beschreibt, und ein internes Schema.

Die einzelnen Schemata werden mittels Transformation in das jeweils andere übergeführt. Die Transformation definiert die Abbildung zwischen den einzelnen Schichten.

Für die Modellierung einer Datenbank ist die logische Ebene ausschlaggebend. In ihr werden Modelle der „realen Welt“ abgebildet.

Das ER-Modell ist das wichtigste für die Abbildung der realen Welt. Es beschreibt Objekte (**Entity**) der Realwelt und ihre Beziehung (**Relationship**) zueinander.

Es wurde 1976 von Peter Chen vorgestellt. Von ihm stammt auch die **Chen-Notation**, welche für die grafische Darstellung von Datenmodellen verwendet wird.

#### Begriffe des ER-Modells:

* **Entity** (Entität):
  + Individuell identifizierbares Objekt der Wirklichkeit
* **Entitätstyp:**
  + Typisierung gleichartiger Entitäten z.B. Angestellter, Projekt, Buch, Autor, Verlag
* **Beziehungen:**
  + Verknüpfung / Zusammenhang zwischen zwei oder mehrerer Entitäten
* **Eigenschaften** (Attribute):
  + Was über eine Entität (im Kontext) von Interesse ist; z.B. das Eintrittsdatum des Angestellten Müller. Beschreibt eine gewisse Eigenschaft
* **Beziehungstyp:**
  + Gibt die Art der Beziehung zwischen Entitäten an

Artikel

beinhaltet

Kunde

bestellt

Bestellung

### ER-Modell – Entity Relationship Modell

#### Studierende

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MatrNr | Vname | Nname | GebDat | Studiengang |
| … |  |  |  |  |
| 1717 | Anna | Technik | 17.12.81 | TKS |
| 1212 | Hans | Kunst | 12.01.75 | MMA |
| … |  |  |  |  |

#### Studierende

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MatrNr | Vname | Nname | GebDat | Studiengang |
| … |  |  |  |  |
| 1717 | Anna | Technik | 17.12.81 | TKS |
| 1212 | Hans | Kunst | 12.01.75 | MMA |
| … |  |  |  |  |

**Entität**: Eine Entität ist ein individuelles Objekt der zu modellierenden Welt, das eindeutig von anderen Objekten unterscheidbar ist.

**Entitätstyp**: Ein Entitätstyp repräsentiert eine Klasse von Entitäten mit gleichen Attributen, aber unterschiedlichen Attributswerten.

**Attribut**: Ein Attribut beschreibt eine bestimmte Eigenschaft eines Entitätstyps. Es besitzt einen eindeutigen Namen.

**Attributswert**: Ein Attributswert ist eine konkrete Ausprägung eines Attributs (einer Eigenschaft einer Entität).

#### Identifikationsschlüssel:

Ist ein Attribut oder eine Kombination aus Attributen. Er identifiziert jede Entität einer Entitätsmenge eindeutig und ändert sich während der Existenz einer Entität nicht. Gibt es mehrere voneinander unabhängige Attribute, die identifizierend wirken können, spricht man von Schlüssel-Kandidaten.

Ein Schlüssel dient in einer relationalen Datenbank dazu, Datensätze (Entitäten) einer Relation (Tabelle) eindeutig zu identifizieren.

##### Primärschlüssel (Primary Key):

Attribut welche eine Entität eindeutig kennzeichnet. (Attributwert ist für alle Entitäten unterschiedlich). Wird in referenzierten Tabellen als Fremdschlüssel verwendet.

##### Sekundärschlüssel (Secundary Key):

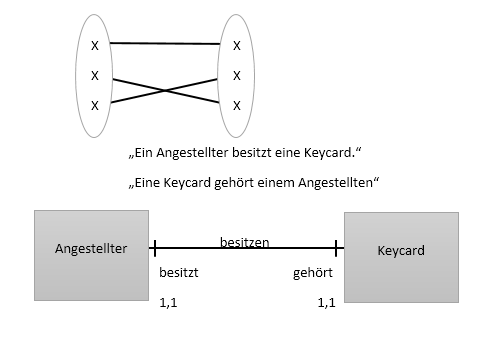
Attribut welches einige Entitäten kennzeichnet. Z.B.: PLZ. Dient meist als Indizes zum schnellen suchen in einer Tabelle.

##### Fremdschlüssel (Foreign Key):

Dient als Verweis zwischen 2 Relationen. Fremdschlüssel der einen Relation ist meist Primärschlüssel der anderen Relation.

###### Einfachste Beziehung

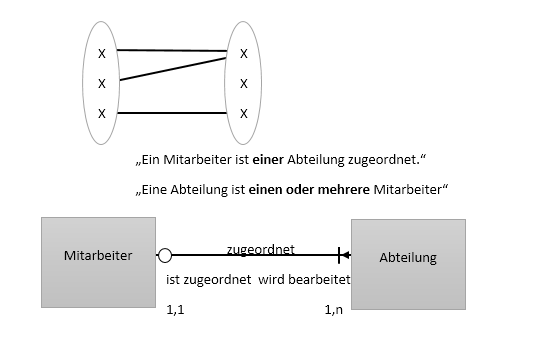
#### 1:1 Beziehung:



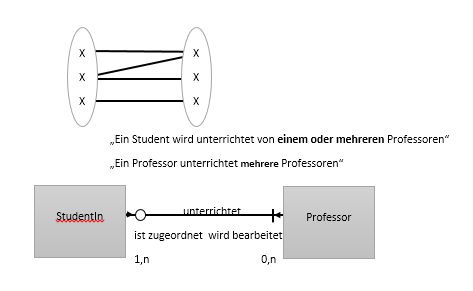
#### 1:n Beziehung:



#### n:1 Beziehung:



#### m:n Beziehung:



### ER-Modell Modellierung

Wurde der Text analysiert bietet sich folgende Vorgehensweise an:

1. Abstecken des Problemrahmens (welche Teile der Anforderung betreffen das Datenbankmodell)
2. Auswahl der Entitätstypen
3. Festlegung der Beziehungen
4. Definition der Attribute
5. Definition der Schlüssel
6. Überführung in ein Relationales Modell

#### Beispiel:

Ein Kunde soll eine Bestellung im Webshop tätigen können. Er muss sich zuvor im System registrieren. Dabei sollen sein Name, Alter, Wohnort erfasst werden. Eine Bestellung soll eindeutig zu einem Kunden zugewiesen werden können indem sie eine eindeutige Nummer bekommt. Zusätzlich soll das Datum der Bestellung, der Status und der Gesamtpreis erfasst werden. Die einzelnen Status einer Bestellung sollen einheitlich erfasst werden.

### Structured Query Language – SQL

SQL ist eine Datenbanksprache zur Definition von Datenstrukturen in relationalen Datenbanken sowie zum Bearbeiten (Einfügen, Verändern, Löschen) und Abfragen von darauf basierenden Datenbeständen.

#### SQL-Befehle lassen sich in 3 Gruppen einteilen:

* **DDL – Data Definition Language**
  + Befehle zur Definition von Tabellen und anderen Datenstrukturen
* **DCL – Data Control Language**
  + Befehle zur Kontrolle von Zugriffsberechtigungen
* **DML – Data Manipulation Language**
  + Befehle zur Datenmanipulation und Datenabfrage