- special cases
  + $min = 0$: an entity does not have to be in relationship (optional)
  + $max = *$: an entity may be in relationship arbitrarily many times
- example: conceptual university schema with $(min, max)$-notations
- **multivalued attributes**
  - optional attribute: minimal cardinality is equal to 0
  - simple attribute: cardinality is equal to 1
  - prescribed attribute: minimal cardinality is equal to 1
  - **multivalued attribute**: maximal cardinality is equal to \( n \)
  - example:

```
<table>
<thead>
<tr>
<th></th>
<th>first-name</th>
<th>last-name</th>
<th>phone-no</th>
</tr>
</thead>
<tbody>
<tr>
<td>( (1,n) )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( (0,n) )</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

- **composite** attributes
  - grouping of attributes of the same entity set or relationship set which are closely related
  - antonym: simple attribute
  - example:

```
person
  - birth-date
  - address
    - street
    - number
    - zipcode
    - city
  - name
```
derived attributes
- attribute that can be derived from one or more attributes
- antonym: base/stored attribute
- graphical representation:
- example:

```
  person
  /      
 name    birth-date
  |      /    
  age
```
Generalization

- goals
  - abstraction at the set level: better (i.e., more understandable and more concise) structuring of entity sets
  - abstraction at the instance level: similar entities are to be modeled by a common entity set

- „factoring“ (extracting) properties (attributes, relationships) of similar entity sets (subclass, subtypes, categories) to a common superclass (supertype)

- properties that cannot be extracted remain with the respective subclass, i.e., the subclass is a specialization of the superclass

- inheritance as the key concept of generalization: a subclass inherits all properties of a superclass
entities of a subclass are implicitly considered as entities of the superclass, therefore is-a in the graphical representation

→ set of entities of the subclass is a subset of the set of entities of the superclass

two special cases

- disjoint/overlapping specialization: all subclasses of a superclass are pairwise disjoint/overlapping
- total specialization: the superclass does not contain explicit elements, but is only given by the union of its subclasses (antonym: partial specialization)
Aggregation

- goal: distinct entity sets which together form a structured superclass are associated with each other
- an aggregation is a special relationship set which associates each superior entity set with several subordinate entity sets
- part-of-relationship
- example: construction of a bicycle

```
  bicycles
    ├── part-of: frame
    │     ├── part-of: tube
    │     └── part-of: handle-bar
    └── part-of: wheels
        ├── part-of: rims
        └── part-of: spokes
```
4. Relational Data Model

4.1 Introduction


- commercial DBMSs like Oracle, Informix, SQL Server, Sybase, DB/2 are based on the relational model

- reasons for the success of the relational data model
  - flat tables (relations) as the simple underlying data structure
  - no nested complicated structures
  - set oriented processing of data in contrast to record oriented processing prevailing until then (hierarchical model, network model)
  - simple comprehensibility also for the unskilled user
  - good performance for standard database applications
  - existence of a mature, formal theory (in contrast to other data models), in particular with respect to the design of relational databases and with respect to an efficient processing of user queries