

1.7 Datenmodellierung

Einstieg:

- individuelle Lösungen, z. B.:
 - Eine einmal erstellte Datenbank kann für die folgenden Schuljahre wiederverwendet werden.
 - Es können Tabellen mit unterschiedlichen Informationen (z.B. für Lehrkräfte, Eltern oder Schüler) ausgedruckt werden.
 - Änderungen sind leichter durchführbar, da diese nur an geeigneter Stelle getätigt werden müssen und nicht auf verschiedenen „Planungszetteln“.
- individuelle Lösungen, z. B.:
Projekte, Projektwahllisten für Schüler, Betreuungsliste für Lehrer, Übersicht der Projekte eines Schülers für die Eltern bzw. die Schüler, Raumbelungsplan

A1:

a) und b)

Hinweis:

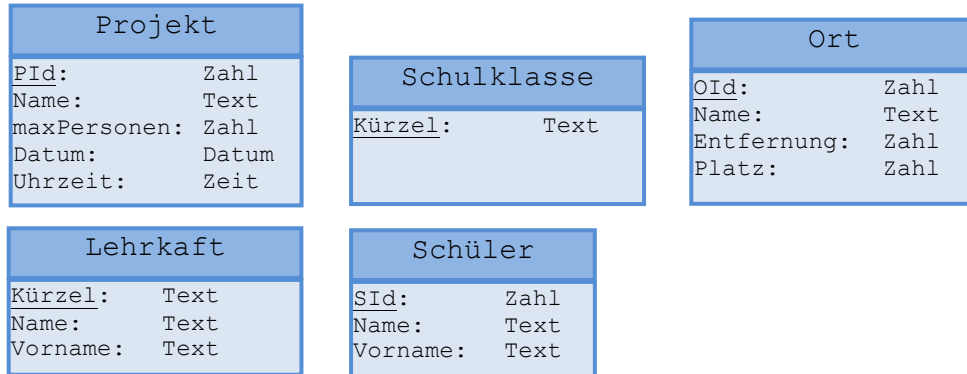
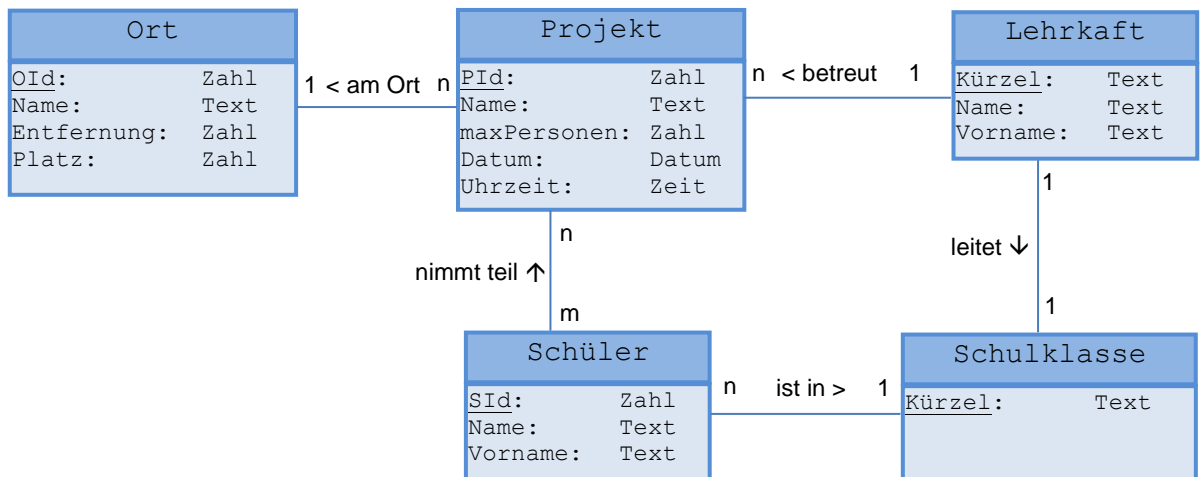
Die Aufgabe lässt sich gut mit einer Pinnwand und Zetteln lösen oder digital mit einer virtuellen Pinnwand wie z.B. einem Padlet.

Folgende Objekte können erkannt werden, sie wurden auch gleich passenden Klassen zugeordnet:

Bohr: Schüler	Handball: Projekt
10a: Schulklasse	Chor: Projekt
Bayer: Lehrkraft	Gitarre: Projekt
Sportplatz: Ort	Handball: Projekt
Schwimmbad: Ort	Weitwurf: Projekt
Turnhalle: Ort	Gymnastik: Projekt

A2:**a) Hinweis:**

Die Aufgabe lässt sich gut mit einer Pinnwand und Zettel lösen oder digital mit einer Modellierungssoftware wie z.B. umlet.

**b)****A3:****a)** Für alle n:m-Beziehungen müssen zusätzliche Klassen erstellt werden um diese in einer Datenbank umsetzen zu können. Somit wird eine Klasse `nimmtTeil` benötigt.**b)** Primärschlüssel wurden in den Diagrammen unter A2 bereits eingetragen.

Hinweis:

Normalerweise sind Primärschlüssel nicht Gegenstand der Modellierung, werden also z. B. in Klassenkarten nicht berücksichtigt. Da hier klar ist, dass sie im relationalen Modell benötigt werden, wurden sie in A2 bereits mit angegeben.

c) Fremdschlüssel werden bei 1:n- bzw. 1:1-Beziehungen wie folgt hinzugefügt:

Klasse `Projekt`

- `amOrt` bezieht sich auf `Ort.OId`
- `betreutVon` bezieht sich auf `Lehrkraft.Kürzel`

Klasse Schüler

- istIn **bezieht sich auf** Schulklasse.Kürzel

Klasse Schulklasse

- Klassleiter **bezieht sich auf** Lehrkraft.Kürzel

neue Klasse nimmtTeil

- Schüler **bezieht sich auf** Schüler.Kürzel
- Projekt **bezieht sich auf** Projekt.Kürzel

d) Schemata

```

Projekt(PIId: Zahl; Name: Text; maxPersonen: Zahl;
       Datum: Datum; Uhrzeit: Zeit; amOrt: Zahl;
       betreutVon: Text)
Ort(OId: Zahl; Name: Text; Entfernung: Zahl;
    Platz: Zahl)
Lehrkraft(Kürzel: Text; Name: Text; Vorname: Text)
Schüler(SId: Text; Name: Text; Vorname: Text;
        istIn: Text)
Schulklasse(Kürzel: Text; Klassleiter: Text)
nimmtTeil(Schüler: Zahl; Projekt: Zahl)

```

A4:**a) individuelle Lösungen, z. B.:**

Suche nach den Projekten (mit Ort), an denen der Schüler Bohr teilnimmt:

```

SELECT Schüler.Name, Projekt.Name, Ort.Name
FROM   Schüler, Projekt, nimmtTeil, Ort
WHERE  Schüler.Name = 'Bohr'
      AND Schüler.SId      = nimmtTeil.Schüler
      AND nimmtTeil.Projekt = Projekt.PId
      AND Projekt.amOrt     = Ort.OId

```

Alle Schüler, die am Projekt Handball teilnehmen:

```

SELECT Schüler.Name, Schüler.Vorname, Schüler.istIn
FROM   Schüler, Projekt, nimmtTeil
WHERE  Projekt.Name      = 'Handball'
      AND Schüler.SId     = nimmtTeil.Schüler
      AND nimmtTeil.Projekt = Projekt.PId

```

Alle Projekte (mit Zeitangabe), die in der Turnhalle stattfinden:

```

SELECT Projekt.Name, Projekt.Datum, Projekt.Uhrzeit
FROM   Projekt, Ort
WHERE  Ort.Name      = 'Turnhalle'
      AND Ort.OId    = Projekt.amOrt

```

Alle Orte mit Entfernung:

```

SELECT Ort.Name, Ort.Entfernung
FROM   Ort

```

b) Alle Lehrer angeben, die bereits ein Projekt betreuen

```
SELECT DISTINCT Lehrkraft.Name
FROM    Projekt, Lehrkraft
WHERE   Projekt.betreutVon = Lehrkraft.Kürzel
```

Diejenigen Projekte finden, die an Orten stattfinden, deren Kapazität evtl. nicht groß genug ist:

```
SELECT DISTINCT Projekt.Name, Ort.Name
FROM    Projekt, Ort
WHERE   Ort.OId          = Projekt.amOrt
        AND Ort.Platz < Projekt.maxPersonen
```

Aufgabe 1:

- a) Da draußen bzw. drinnen vom Ort abhängt, fügt man zur Klasse `Ort` das Attribut `drinnen` z. B. als Wahrheitswert hinzu.
- b) Die Beziehung `betreut` ist dann eine n:m-Beziehung und muss durch eine zusätzliche Klasse z.B. `betreut` implementiert werden. Der Fremdschlüssel `betreutVon` wird aus der Klasse `Projekt` entfernt.

Schema von `betreut`:

```
betreut(Lehrkraft: Text; Projekt: Zahl)
```

- c) Die Klasse `Projekt` erhält ein zusätzliches Attribut `minPersonen`.
Ergibt sich bei der Suche nach den Schülern eines Projekts dann eine Ergebnisanzahl kleiner als `minPersonen`, kann das Projekt abgesagt werden.

Hinweis:

Wurden die Aggregationsfunktionen besprochen, so kann dies mithilfe von `COUNT(*)` auch automatisiert werden.

Aufgabe 2:

Beispiel für ein passendes Klassendiagramm. Die Fremdschlüssel wurden auch hier bereits ergänzt:

