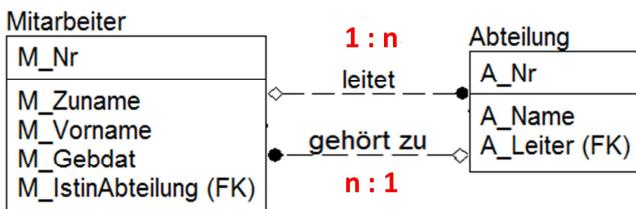
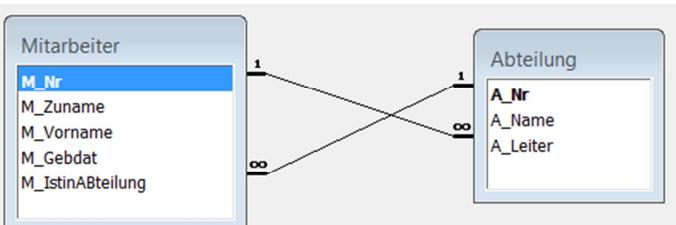


DBIS2 – Datenbanken und Informationssysteme



Gibt es sinnvolle Beispiele für
mehrere Beziehungen
zwischen 2 Tabellen



A01

Die Startfolie enthält wieder eine Modellfrage (bezüglich unserer aktuellen Datenbank)

Ist es möglich, dass zwischen 2 Tabellen mehr als eine Beziehung existiert

ja, z.B., in diesem Beispiel

Jeder Mitarbeiter hat in M_IstinAbteilung eingetragen zu welcher Abteilung er gehört

In der Abteilung steht in A_Leiter jener Mitarbeiter, der die Abteilung leitet

JA, die beiden Beziehungen stellen aber 2 unterschiedliche Sachverhalte dar.

Hinweis

Access erlaubt zwar solche Beziehungen (nach Rückfrage),

es wird aber in der grafischen Darstellung

eine Tabelle 2mal gezeichnet (obwohl physisch nur einmal vorhanden)

Man stellt die dann einfach übereinander

Hinweis 2: Dies hat aber starke Auswirkungen auf die Entwurfsansicht von Abfragen

Hat man beide Tabellen im FROM (genauer zur Entwurfsansicht hinzugefügt)

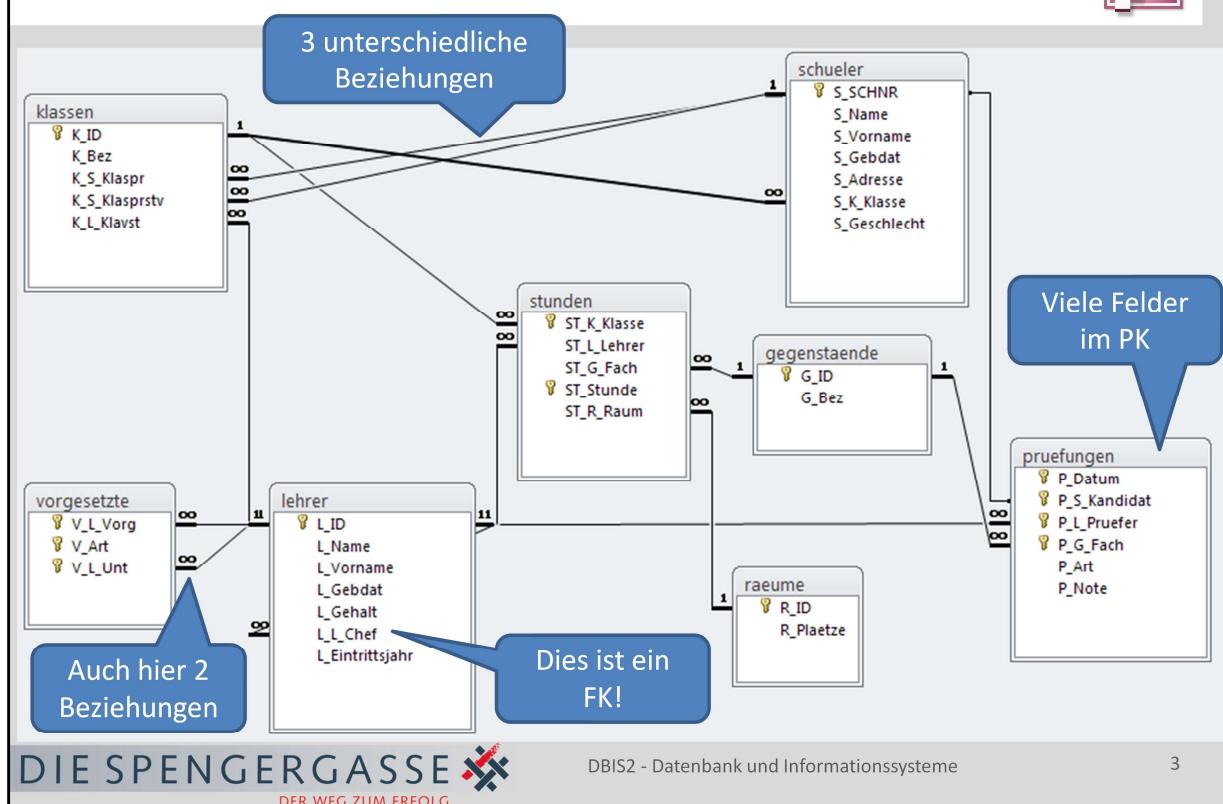
werden auch beide Beziehungen für den ON Zweig vorgeschlagen.

!!!!!! Je nach Fragestellung ist nur eine sinnvoll, die andere muss aus der Entwurfsansicht entfernt werden (Strich durch Klicken markieren und Entf drücken)



- 1) TEST (ODER WIEDERHOLUNG)
NACHBESPRECHUNG**
- 2) „ABFRAGEN“, DIE ZÄHLEN
(GRUPPIEREN) KÖNNEN**

Wiederholung praktischer Übungstest die DB



Dies ist das Modell der Übungsdatenbank, man muss es gut studieren
 Niemand kann auf einer unbekannten DB Abfragen erstellen,
 teilweise muss man auch die Daten ansehen, ST_Stunde enthält z.B. Werte wie MO1, DI2 (Wochentag und Stunde)

Achtung, Hier gibt es zwischen Schüler und Klasse 3 Beziehungen,
 ebenso mehrere Beziehungen bei Lehrer und Vorgesetzte

Die dabei (leider vom Access) mehrfach gezeichneten Tabellen liegen im Bild übereinander

Wie schon im Startbild gefragt gibt es hier mehrere Situationen von unterschiedlichen Beziehungen zwischen 2 Tabellen

Verwenden Sie Klassen und Schüler müssen Sie entscheiden

- Sie wollen die Klasse, welche der Schüler besucht
- Sie wollen die Klasse wo der Schüler Klassensprecher ist
- Sie wollen die Klasse wo der Schüler Klassensprecher-Stellvertreter ist

und die richtige Beziehung im INNER JOIN benutzen.

Wiederholung praktischer Übungstest



Aufgabe 1 die Datenbank erweitern

Die Direktion ist es leid, auf Zetteln zu vermerken, welche Sonderfunktionen (Bibliothekar, Rot-Kreuz-Referent, Umweltreferent ...) Lehrer/innen in der Schule haben. Aus diesem Grunde sollen Sie diese Datenbank entsprechend erweitern!

Die Anforderung:

- für Lehrer/innen soll seine/ihre 'Sonderfunktionen' erfaßt werden können
- ein/e Lehrer/in kann mehr als eine 'Sonderfunktion' haben.
- mehrere Lehrer/innen können die gleiche Sonderfunktion haben

Die möglichen Sonderfunktionen liegen als xls File vor und sollen in die DB importiert werden

- Xls File aufbereiten (1. Zeile Feldnamen, darunter Daten) erst dann importieren (dabei Datentypen prüfen/ setzen und PK wählen bzw. erzeugen lassen)
- Beziehung zu Tabelle Lehrer setzen – geht nicht direkt mit einem Fremdschlüssel weil **n:m**, es muss daher eine Zwischentabelle erstellt werden!

Wenn Sie die Übungs DB öffnen, dann erscheint ein Formular mit den Fragestellungen

Wiederholung praktischer Übungstest



	A	B	C	D
1	So_Nr	So_Name	So_mindestanzahl	
2	1	Bibliothekar	2	
3	2	Rot-Kreuz-Referent	2	
4	3	Umweltreferent	1	
5	4	Geografiekustos	1	
6	5	Branschutzbeauftragter	1	
7	6	Schulballorganisator	1	

Import-Assistant für Kalkulationstabellen

Sie können Informationen zu jedem zu importierenden Feld angeben. Wählen Sie dazu Felder aus der Liste aus und bearbeiten Sie dann die Feldinformationen im Bereich 'Feldoptionen'.

Feldoptionen

Feldname: So_Nr Datentyp: Long Integer
Indiziert: Nein Feld nicht importieren (überspringen)

So_Nr	So_Name	So_mindestanzahl
1	Bibliothekar	2
2	Rot-Kreuz-Referent	2
3	Umweltreferent	1
4	Geografiekustos	1

(radio buttons) Primärschlüssel soll von Access hinzugefügt werden
Eigenen Primärschlüssel auswählen So_Nr
Kein Primärschlüssel

- N : M Beziehung, daher eine Zwischentabelle erstellen



Wenn Sie beim Import bereits die richtigen Einstellungen treffen ist die importierte Tabelle "Sonderfunktionen" sofort benutzbar, allerdings können Sie die Tabelle in der Tabellen Entwurfsansicht auch später ändern.

Entscheidend ist Schritt 1 aufbereiten der Daten im Excel

Klickt man sich (ohne Benutzung des Gehirns) durch die Importschritte Hat man am Schluss den sinnlosen Datentyp double sowie eine generierte ID Spalte Obwohl die Importdaten bereits einen tauglich PK aufweisen

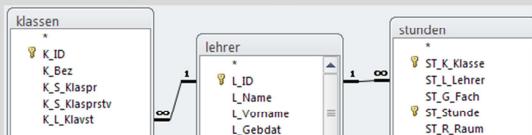
Bei jeder n:m Beziehung muss immer eine Zwischentabelle gebildet werden, Welche die PKs der ursprünglichen Tabellen als Fremdschlüssel enthält .

Was könnte der PK der Zwischentabelle sein ?? Auf jeden Fall können die beiden Fremdschlüssel zusammen den PK bilden. Alternativ kann auch eine eigene ID als Pk benutzt werden.

Wiederholung praktischer Übungstest



- Für die Abfragen muss man natürlich zuerst die benötigten Tabellen und die geeigneten Joins auswählen.
 - 1 – nur die Tabelle Schueler
 - 2 – Schueler, Pruefungen, (Gegenstaende für G_Bez)
 - 3 – Lehrer, Stunden (Klassen nicht nötig, Feld St_Klasse reicht)
 - 4 – Stunden, Lehrer, Klassen
(Achtung: Klasse-Stunden Beziehung nicht nutzen!)
- 5 - Outer Join, Fragestellung betrifft PK Seite daher FROM lehrer LEFT JOIN klassen ON lehrer.L_ID = klassen.K_L_Klavst;
- 6 – Fragepattern: Zeige Datensätze wo PK nicht im FK vorkommt
FROM raeume LEFT JOIN stunden ON raeume.R_ID = stunden.ST_R_Raum
WHERE ST_Stunde Is Null;



- 1: Welche Schülerinnen an der Schule sind vor 1985 geboren?
- 2: Welche Schüler haben an welchem Prüfungsdatum die Note 1 erhalten (Name, P-Datum, Fach)
- 3: Welche Lehrer (Namen) unterrichten in welchen Klassen am Nachmittag in Stunden 7 und 8
- 4: Welche Stunden werden von Klassenvorständen (mindestens auszugeben: Vorname, Nachname, Stunde, Raum,
Fach, Klasse) unterrichtet ? Die Ausgabe soll sortiert nach Nachname, Klasse und Stunde erfolgen.
- 5: Liefere Daten aller Lehrer mit Angabe der Klasse(n) wo sie Klassenvorstand sind
- 6: Gibt es Räume, in denen keine einzige Unterrichtsstunde abgehalten wird ??
- 7: Gibt es eine/n oder mehrere Lehrer/innen, der/die exakt gleich alt ist/sind wie ein/e Schüler/in ?
- 8: Gib den Namen aller Lehrer mit dem Namen des chefs aus (!! Tabelle Lehrer 2 mal verwenden ist nötig!!)



- An mehreren Stellen ist es wichtig die bereits in den Unterrichtseinheiten 2 und 3 vorgestellten Funktionen und Operatoren zu kennen
 - 1 - WHERE Year(S_Gebdat)<1985
 - 3 - WHERE St_Stunde Like "*7" Or St_Stunde "*8"
 - 6 - WHERE ST_Stunde Is Null
- ORDER BY sollte kein grosses Problem sein
- Abfrageentwurf oder SQL - bleibt Ihnen überlassen zumindest der Beginn (FROM) ist im Entwurf oft übersichtlicher, danach ist auch SQL Ansicht recht tauglich

In vielen Fällen sind Funktionen hilfreich

WHERE S_Gebdat < #1/1/1985# Achtung: im SQL Text #mm/dd/yyyy#
WHERE Year(S_Gebdat)<1985 mit Funktion einfacher Month(...), Day(...)

WHERE (Date() – S_Gebdat) / 365.25 > 18 wer ist heute schon über 18 Jahre

WHERE St_Stunde Like "*7" Or St_Stunde "*8"

WHERE ST_Stunde Is Null;

WHERE P_Note in (1,2,3)

ORDER BY L_Name, ST_K_Klasse, ST_Stunde;
ORDER BY P_Note DESC

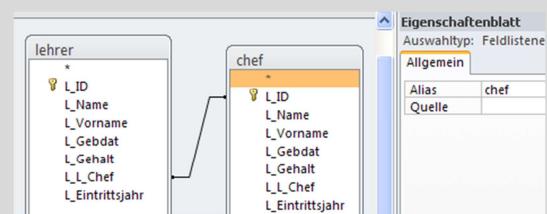
Wiederholung praktischer Übungstest



- Frage 7 Schüler gleich alt wie Lehrer
dies ist eine der seltenen Ausnahmen, wo der JOIN nicht das übliche PK = FK benutzt

```
FROM lehrer INNER JOIN schueler ON
      lehrer.L_Gebdat = schueler.S_Gebdat;
```
- Frage 8 Lehrer mit Namen des Chefs
Tabelle Lehrer 2 mal im FROM (1 mal umbenennen)
- FROM lehrer INNER JOIN
lehrer AS chef

```
ON lehrer.L_L_Chef =
      chef.L_ID
```



7: Gibt es eine/n oder mehrere Lehrer/innen, der/die exakt gleich alt ist/sind wie ein/e Schüler/in ?

Zwischen Lehrer und Schüler gibt es keine direkte Beziehungen, 3 indirekte Beziehungen sind denkbar

- Lehrer, welche Schüler geprüft haben (- Pruefungen -)
- Lehrer welche Schüler unterrichten (- Stunden – Klasse -)
- Lehrer, die KV von Schülern sind (- Klasse -)

Paßt bei dieser Frage alles nicht

```
FROM lehrer INNER JOIN schueler ON lehrer.L_Gebdat = schueler.S_Gebdat;
```

8: Gib den Namen aller Lehrer mit dem Namen des chefs aus (!! Tabelle Lehrer 2 mal verwenden ist nötig!!)

Diese DB enthält den Sonderfall, dass bei einer Beziehung der Fremdschlüssel (L_L_Chef) in der gleichen Tabelle wie der PK (L_ID) steht !!!

Benötigt man so eine Beziehung im Rahmen einer Abfrage,
muss man die Tabelle zweimal im FROM haben und einmal umbenennen

```
SELECT lehrer.L_Name, "der Chef ist" AS Ausdr1, chef.L_Name
FROM lehrer INNER JOIN lehrer AS chef
      ON lehrer.L_L_Chef = chef.L_ID;
```

Gruppierung

The diagram illustrates the process of grouping data. On the left, a table shows individual student records with columns 'S_Geschlecht' and 'S_Nr'. An arrow points to the right, leading to a summary table where the data is grouped by 'S_Geschlecht'. The summary table has columns 'S_Geschlecht' and 'Anzahl_Schueler', showing the count of students for each gender: 130 for gender 1 and 23 for gender 2.

S_Geschlecht	S_Nr
1	1
1	2
1	3
1	4
1	5
2	6
2	7
2	8
1	9
1	10
1	11
1	12
1	13

S_Geschlecht	Anzahl_Schueler
1	130
2	23

Gruppenfunktionen
COUNT(), SUM(),
AVG(), MIN(), MAX()

9

Mit Gruppierung (oder Summierung)

kann man nach verschiedenen Kriterien

- Zählen COUNT()
- Summieren SUM()
- Durchschnitt AVG()
- kleinster MIN()
- größter MAX()

Das sind die sogenannten Gruppen- oder Aggregatsfunktionen,

die sich deutlich anders verhalten als normale Funktionen

→ sie ermitteln aus mehreren Werten (aus verschiedenen Zeilen einer Spalten) einen Gruppenergebniswert

Abfragen Gruppenfunktionen



- Am einfachsten ist es im SELECT eine Gruppenfunktion (Count, sum, Avg, Min, Max) einzusetzen.
- Folgendes gibt nicht die Schüler im Detail sondern nur (in einer Zeile) die Anzahl der Schüler aus
- Das geht auch mit anderen Gruppenfunktionen

```
SELECT Count(*) as Anzahl FROM Schueler
```

Anzahl
153

```
SELECT Count(*) AS Anzahl, Max(S_Gebdatum) AS Juengster,  
Min(S_Gebdatum) AS Aeltester, Avg(S_Geschlecht) AS Durchschnitt  
FROM Schueler;
```

Anzahl	Juengster	Aeltester	Durchschnitt
153	07.01.1997	27.03.1991	1,1503267974

Durchschnitt geht nur über Zahlen

Gruppierung liegt vor, wenn man
GROUP BY und/oder eine der Gruppenfunktionen einsetzt

Wichtiger Hinweis:

Im Fall einer Gruppierung dürfen Sie im Select nur Gruppenfunktionen (und Group by Felder) einsetzen

Es sollte auch auffallen, dass die zugrundeliegenden Detailsätze nicht mehr ausgegeben werden (können)
Stattdessen gibt es „Gruppensummenzeilen“

Nur eine einzige Zeile wenn man kein Group By hat

Mit Einsatz von Group By entscheiden die verschiedenen Werte in den Group by Feldern
wie viele Gruppen gebildet werden und somit wie viele Ausgabezeilen entstehen (eine pro Gruppe)

Abfragen Gruppierung in Entwurfsansicht



- Ich möchte die Anzahl Jungs und Mädels (der Schüler)

Schueler

S_Nr
S_Zuname
S_Vorname
S_Geschlecht
S_Gebdatum
C_Straße

Feld: S_Geschlecht
Tabelle: Schueler
Sortierung:
Anzeigen:
Kriterien:
oder:

Σ
Summen

Schueler

S_Nr
S_Zuname
S_Vorname
S_Geschlecht
S_Gebdatum
C_Straße

Feld: S_Geschlecht Anz: s_nr
Tabelle: Schueler Schueler
Funktion: Gruppierung Anzahl
Sortierung:
Anzeigen:
Kriterien:
oder:

Überschrift

Gruppierung nur für die Gruppenfelder, ansonsten Gruppenfunktionen

```
SELECT S_Geschlecht, Count(s_nr) AS Anz
FROM Schueler
GROUP BY S_Geschlecht;
```

DIE SPENGERGASSE
DBIS2 - Datenbank u
11

Tabelle Schueler reicht aus, wenn man sich mit 1(=m) und 2(=w) zufriedengibt.
 andernfalls braucht man noch die Tabelle Geschlechter und einen inner join

Es gibt keinen Where Zweig (man will ja nichts einschränken)

Weil man eine Anzahl ermitteln will, die so nicht in der DB gespeichert ist muss man Gruppierung anwenden

Also zuerst die relevanten Felder in den Select Zweig
 S_Geschlecht und S_nr (PK ist immer gut zum zählen)

Jetzt die Summentaste drücken
 und in der neu entstandenen Zeile Funktion befüllen

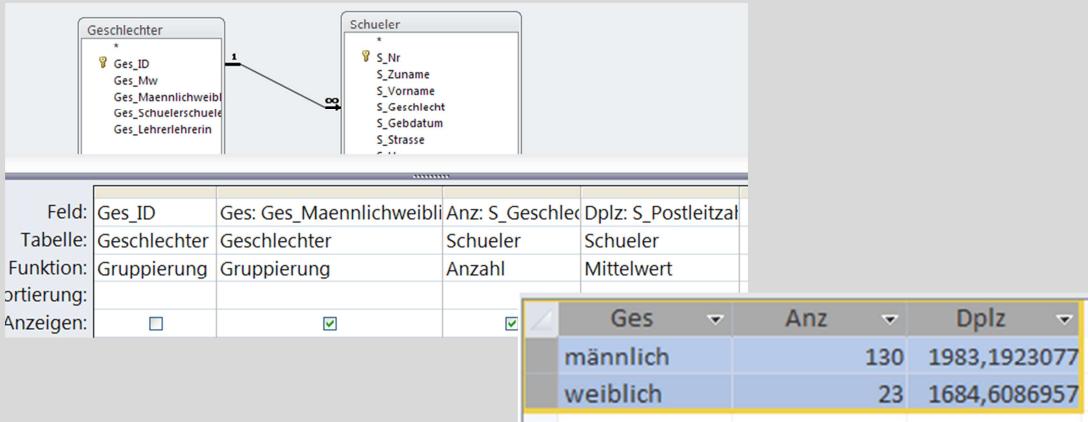
Man könnte auch sagen Pro vorhandenem Geschlecht will ich die Anzahl der Schüler
 Daher wird S_Geschlecht das Gruppenwechselfeld

```
SELECT S_Geschlecht, Count(s_nr) AS Anz
FROM Schueler
GROUP BY S_Geschlecht;
```

Abfragen Gruppierung in Entwurfsansicht



- Geht auch mit mehreren Tabellen



```
SELECT Ges_Maennlichweibl AS Ges, Count(S_Geschlecht) AS Anz,
       Avg(S_Postleitzahl) AS Dplz
FROM Geschlechter LEFT JOIN Schueler ON Geschlechter.Ges_ID = Schueler.S_Geschlecht
GROUP BY Ges_ID, Ges_Maennlichweibl;
```

```
SELECT Ges_Maennlichweibl AS Ges, Count(S_Geschlecht) AS Anz,
       Avg(S_Postleitzahl) AS Dplz
FROM Geschlechter LEFT JOIN Schueler ON Geschlechter.Ges_ID = Schueler.S_Geschlecht
GROUP BY Ges_ID, Ges_Maennlichweibl;
```

Es gelten hier einige Regeln:

LEFT JOIN wenn man auch Einträge zählen will, die 0 sein können

Im Group BY IMMER den Primary Key der Tabelle unterbringen, nach der man zählen will

z.B. Schülerzahlen pro Klasse === GROUP BY K_Nr

Zählen geht am Besten mit Count(Fremdschlüssel) ... ist erst wichtig wenn 0 rauskommen könnte

Die Anzahl Schüler pro Klasse (für jede Klasse) lauten daher :

```
SELECT K_Nr, Count(Schueler.S_Klasse) AS Anz
FROM Klassen LEFT JOIN Schueler ON Klassen.K_Nr = Schueler.S_Klasse
GROUP BY K_Nr;
```

Abfragen Gruppierung in Entwurfsansicht



- Man kann auch ganz was anderes zählen:

```
SELECT S_Vorname, Count(S_Vorname) AS Anz
FROM Schueler
GROUP BY S_Vorname
ORDER BY Count(S_Vorname) DESC
```

Schueler

S_Nr
S_Zuname
S_Vorname
S_Geschlecht
S_Gebdatum

Feld: **S_Vorname** Anz: **S_Vorname** S_Geschlecht
Tabelle: Schueler Schueler Schueler
Funktion: Gruppierung Anzahl Bedingung
Sortierung: Absteigend
Anzeigen:
Kriterien: 2

S_Vorname	Anz
Michael	7
Alexander	6
Florian	6
Lukas	5
Stefan	5
Daniel	4
Dominik	3
Jan	3
Thomas	3
Robert	2
Oliver	2

S_Vorname	Anz
Anna	1
Christina	1
Dergil	1
Eda	1
Jennifer	1
Jessica	1

-- Häufigkeit der Vornamen absteigend

```
SELECT S_Vorname, Count(S_Vorname) AS Anz
FROM Schueler
GROUP BY S_Vorname
ORDER BY Count(S_Vorname) DESC
```

Oder auch

```
SELECT S_Vorname, Count(S_Vorname) AS Anz
FROM Schueler
WHERE S_Geschlecht = 2
GROUP BY S_Vorname
ORDER BY Count(S_Vorname) DESC, S_Vorname
```

- Leider gibt es in diesen Daten keinen weiblichen Vornamen doppelt,
- in den echten Daten der Schule führen Julia, Katharina, Barbara die Rangliste an
- bei den Jungs Alexander, Daniel, Stefan, Michael, Markus, Thomas, Patrick, Lukas



- Ermittle die Anzahl der Klassen pro Schuljahr
- Ermittle die Anzahl der Schüler pro Klasse
(Klassen mit 0 Schüler sollen auch aufscheinen!)
- Ermittle pro Postleitzahl die Anzahl männlicher Schüler
- Ermittle pro Klasse die Namen des ersten und letzten Schülers auf der Schülerliste sowie das Geburtsdatum des jüngsten und ältesten Schülers
- Zeige jene Tage, an denen mehrere Personen geboren wurden
- Zähle pro Klasse Burschen und Mädchen

-- die 2. Frage wird wohl einen outer join brauchen

Lösungspattern:

```
SELECT pk, Count(fk) as Anzahl
      FROM pktable LEFT JOIN fktable ON pktable.pk = fktable.fk
 GROUP BY pk
```

-- die letzte Abfrage soll 2 verschiedene Dinge zur gleichen Zeit zählen

das lässt sich mit count nicht direkt durchführen