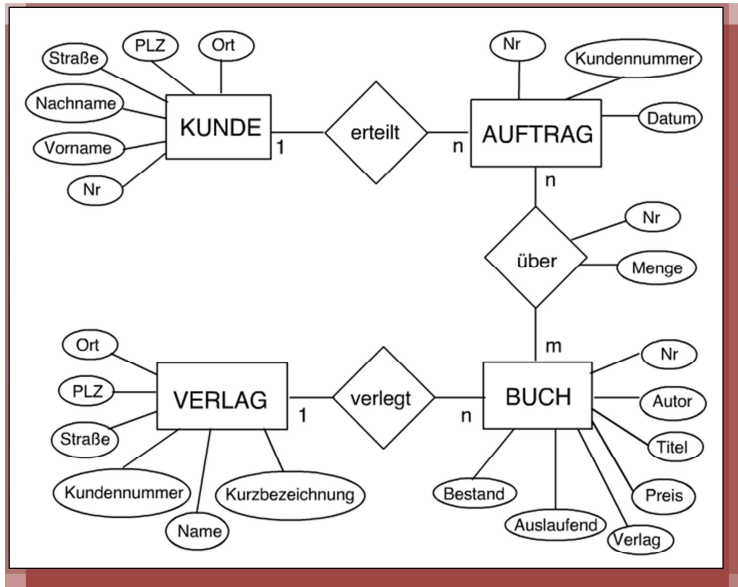


## DBIS2 – Datenbanken und Informationssysteme

## Lektion 6



Erläutern Sie das Modell



A01

Die Startfolie enthält wieder eine Modellfrage  
(es handelt sich um das gleiche Modell wie schon in Lektion 4)

Kunde, Auftrag sind die Namen der Tabellen (stehen hier in den Kästchen),  
Im Kreisen (Ellipsen) daneben angeordnet stehen die Namen der Felder,  
der Primary Key (PK) kann man hier nicht klar erkennen, sie sollten eigentlich unterstrichen sein  
Foreign Keys (FK) sind zwar zu sehen, aber nicht entsprechend gekennzeichnet.

Die Beziehungsstriche haben 1:n oder n:m direkt angegeben,  
Ein Text für die Beziehung erscheint in einer Raute

Also:

- 1 Kunde erteilt n Aufträge      aber 1 Auftrag gehört zu genau einem Kunden
- 1 Verlag verlegt n Bücher      aber 1 Buch wird nur von genau einem Verlag verlegt
- 1 Auftrag hat n Auftragspositionen
- 1 Buch wird in n Auftragspositionen verkauft

Zwischen Buch und Auftrag wird hier auch klar eine n:m Beziehung dargestellt.  
Um diese n : m Beziehung in wirklichen Datenbanken auch speichern zu können  
muss noch die neue Tabelle Auftragsposition zur Auflösung der n:m Beziehung geschaffen werden.



**1) ABFRAGEN ÜBERBLICK –  
WIE GEHE ICH VOR**

**2) PROBE – „TEST“**

# Überblick – wie geht man vor

1. FROM, Tabellen und Beziehungen auswählen
2. Inner Join oder Outer Join (LEFT join)
3. WHERE schränkt auszugebende Zeilen ein

Je nachdem

	Ausgabe	Daten ändern	$\Sigma$ Summen	Gruppieren
4.	SELECT bestimmt Ausgabespalten	Wähle (nach Select) die Abfragegeart (Aktualisieren, Anfügen)		Drücke (nach Select) den Summen-Knopf
5.	ORDER sortiert	Fülle die nötigen „Zeilen“ in Entwurfsansicht		Fülle die Zeile Funktion

3

Entwirft man einen SQL Befehl, so geht man praktisch immer in dieser Reihenfolge vor, auch die interne Abarbeitung des Befehls ist in dieser Reihenfolge

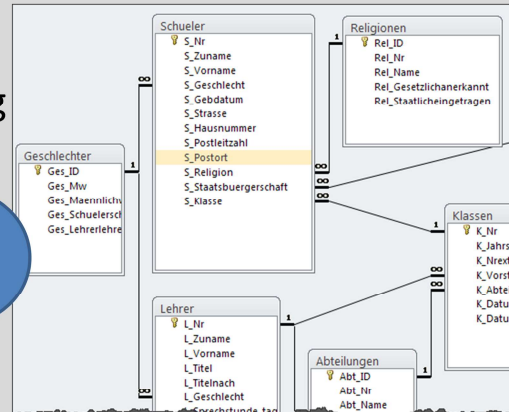
Schritt 4 und 5 sind unterschiedlich, je nachdem für welche Art der Abfrage man sich entscheidet,

Die dritte Spalte „Gruppieren“ kommt erst nächste Woche im Unterricht vor



- **SQL** - Abfragesprache in allen Datenbanken  
in Access mittels Abfragen Entwurf- oder SQLansicht
- **Wie geht man grundsätzlich vor?**
  - Man kennt die Datenbank!
  - Aufgrund der Fragestellung wählt man die benötigten Tabellen
  - und die dazwischen nötigen Beziehungen

1.



Man ermittelt also zuerst den FROM Zweig mitsamt der nötigen ON Klausel

```
FROM Lehrer INNER JOIN Klassen  
ON Lehrer.L_Nr = Klassen.K_Vorstand
```

```
FROM tabelle1 INNER JOIN tabelle2  
ON Klausel definiert, welche Datensätze aus den beiden Tabellen kombiniert werden,  
praktisch wird hier immer eine existierende Beziehung verwendet,  
also tabelle1.PrimaryKey = tabelle2.ForeignKey
```

## Abfragen Überblick Beispiele



- Geben Sie alle Lehrervornamen sortiert aus  
→ wohl nur FROM lehrer
- Geben Sie alle Schüler der Klasse des KV „LO“ aus  
→ Schülerdaten aus Tabelle schueler,  
Klassenvorstand(Kürzel) steht in Tabelle Klasse  
→ daher FROM Klassen INNER JOIN Schueler  
ON K\_Nr = S\_Klasse  
Hinweis1: erst im WHERE wird auf LO eingeschränkt.  
Hinweis2: Frage bezieht sich auf schueler, also die  
Tabelle auf der Fremdschlüsselseite, daher INNER JOIN  
(statt outer join)

Man ermittelt also zuerst den FROM Zweig mitsamt der nötigen ON Klausel

### Beispiele

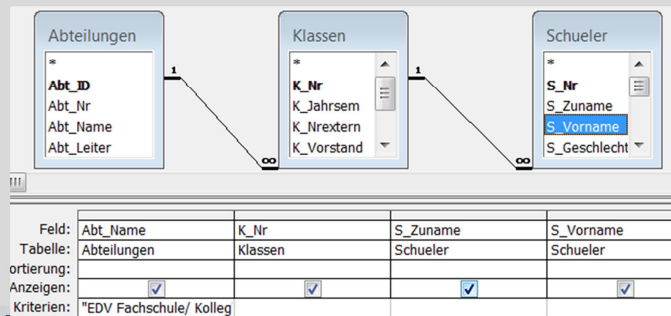
in Access muss man im ON immer tabellenname.feldname schreiben

```
FROM Klassen INNER JOIN Schueler  
ON Klassen.K_Nr = Schueler.S_Klasse
```

## Abfragen Überblick



- Geben Sie alle Schüler der Abteilung „EDV Fachschule/ Kolleg “ aus  
→ Schülerdaten aus Tabelle Schueler,  
Abteilungsname aus Tabelle Abteilungen  
dazwischen gibt es keine direkte Beziehung, also  
muss auch noch die Tabelle Klassen dazu.



DIE SPENGERGASSE  
DER WEG ZUM ERFOLG

DBIS2 - Datenbank und Informationssysteme

6

Auch in diesem Beispiel wird nach Schüler gefragt (steht rechts an der Fremdschlüsselseite), daher INNER JOINS

Bei 3 Tabellen ist in Access leider eine Klammerung wie folgt nötig:

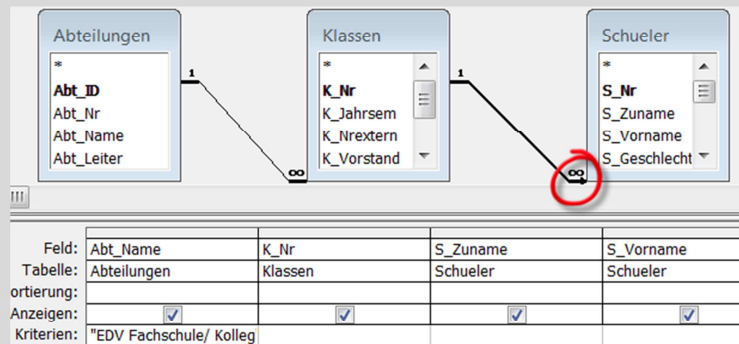
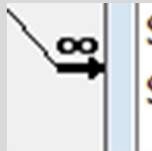
```
SELECT Abteilungen.Abt_Name, Klassen.K_Nr, Schueler.S_Zuname
FROM (Abteilungen INNER JOIN Klassen ON Abteilungen.Abt_ID = Klassen.K_Abteilung)
     INNER JOIN Schueler ON Klassen.K_Nr = Schueler.S_Klasse
WHERE Abt_Name = "EDV Fachschule/ Kolleg,,
```

Zeigt 52 Schüler-Datensätze aus 2 Klassen aus der Abteilung EDV Fachschule/ Kolleg

## Abfragen Überblick



- Entscheidung INNER JOIN oder LEFT JOIN 2.
  - hängt von der Fragestellung ab,  
fragt man nach Tabelle auf FK Seite -> INNER JOIN  
nach Tabelle auf der PK Seite eher LEFT JOIN  
z.B. Zeige alle Klassen (aus Abt. ...) mit Schüler



**DIE SPENGERGASSE**  
DER WEG ZUM ERFOLG

DBIS2 - Datenbank und Informationssysteme

7

Auch in diesem Beispiel wird nach Schüler gefragt (steht rechts an der Fremdschlüsselseite), daher INNER JOINS

Bei 3 Tabellen ist in Access leider eine Klammerung wie folgt nötig:

```
SELECT Abteilungen.Abt_Name, Klassen.K_Nr, Schueler.S_Zuname, Schueler.S_Vorname
FROM (Abteilungen INNER JOIN Klassen ON Abteilungen.Abt_ID = Klassen.K_Abteilung)
LEFT JOIN Schueler ON Klassen.K_Nr = Schueler.S_Klasse
WHERE Abt_Name = "EDV Fachschule/ Kolleg"
```

Aufgrund des LEFT JOIN kommen jetzt alle Klassen, auch wenn sie keine Schüler haben  
Womit sich 57 Ausgabedatensätze ergeben (5 Klassen haben keine Schüler)

PS.: diese 5 Klassen kann man leicht herausfinden, wenn man  
zum WHERE Zweig S\_Klasse IS NULL dazufügt

## Abfragen Überblick



- Mit WHERE Zweig die Ausgabe(-zeilen) einschränken falls das nötig ist

Abt_Name	S_Nr
Abteilungen	Schueler
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
"EDV Fachschule/ Kolleg"	Ist Null

3.

- Im SELECT Zweig gibt man die gewünschten Ausgabespalten an

Feld:	Abt_Name	K_Nr	S_Zuname
Tabelle:	Abteilungen	Klassen	Schueler

4.

- Im ORDER BY kann man noch sortieren

Abt_Name	S_Zuname	S_Vorname
Abteilungen	Schueler	Schueler
Aufsteigend	Aufsteigend	Aufsteigend
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

5.

Im WHERE Zweig kann man (wie in Programmiersprachen im if) beliebige Bedingungen bilden.

Alle Felder aus den im From stehenden Tabellen können benutzt werden

WHERE Abt\_Name = "EDV Fachschule/ Kolleg,, AND S\_Klasse IS NULL

Im Select definiert man die gewünschten Ausgabespalten, man kann alle Felder und auch Berechnungen zwischen diesen angeben

```
Select  S_Klasse,
        S_Zuname + „ „ + S_Vorname as Name,
        S_Gebdat – Date() as AlterinTagen
```

```
ORDER BY S_gebdat desc
```



## Abfragen Überblick



- Will man aber Daten verändern, so muss man nach dem Select die gewünschte Abfrageart wählen und die jeweiligen zusätzlichen Angaben erledigen (z.B. bei Aktualisieren Zeile mit den gewünschten neuen Werten befüllen oder bei Anfügen an die Feldnamen (Einfügen) angeben

5a.

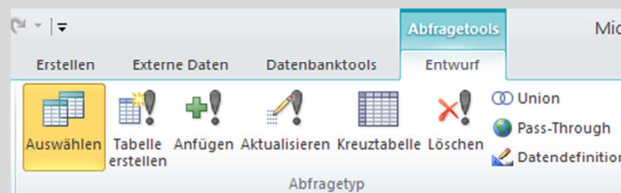


Tabelle erstellen fügt nur ein INTO neuetabelle vor dem FROM ein

Anfügen schreibt vor den SELECT ein INSERT INTO anderetabelle (feld1,feld2,...)

Aktualisieren macht aus dem SELECT(und FROM) ein UPDATE tabelle SET feld1 = neuerwert, feld2 = ....  
!!! ein WHERE muss unbedingt erhalten bleiben

Löschen macht aus dem SELECT(und FROM) ein DELETE FROM tabelle  
!!! ein WHERE muss unbedingt erhalten bleiben

# Probe „test“

10

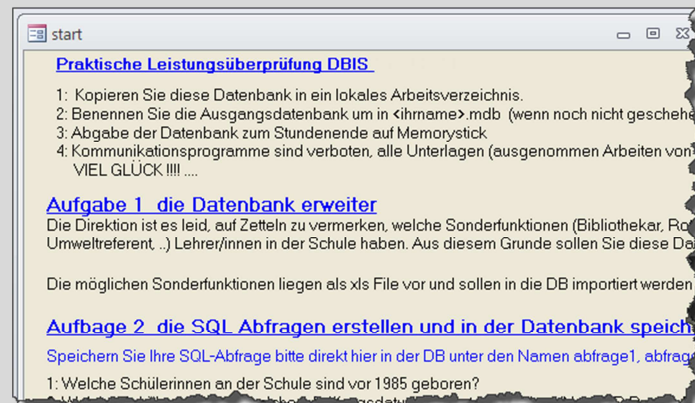
Weil in den letzten Lektionen viel Neues durchgemacht wurde  
gibt es heute keinen neuen Stoff

Damit sie eine Vorstellung haben wie Ihre kommende  
praktische Leistungsfeststellung aussehen kann  
üben wir das heute

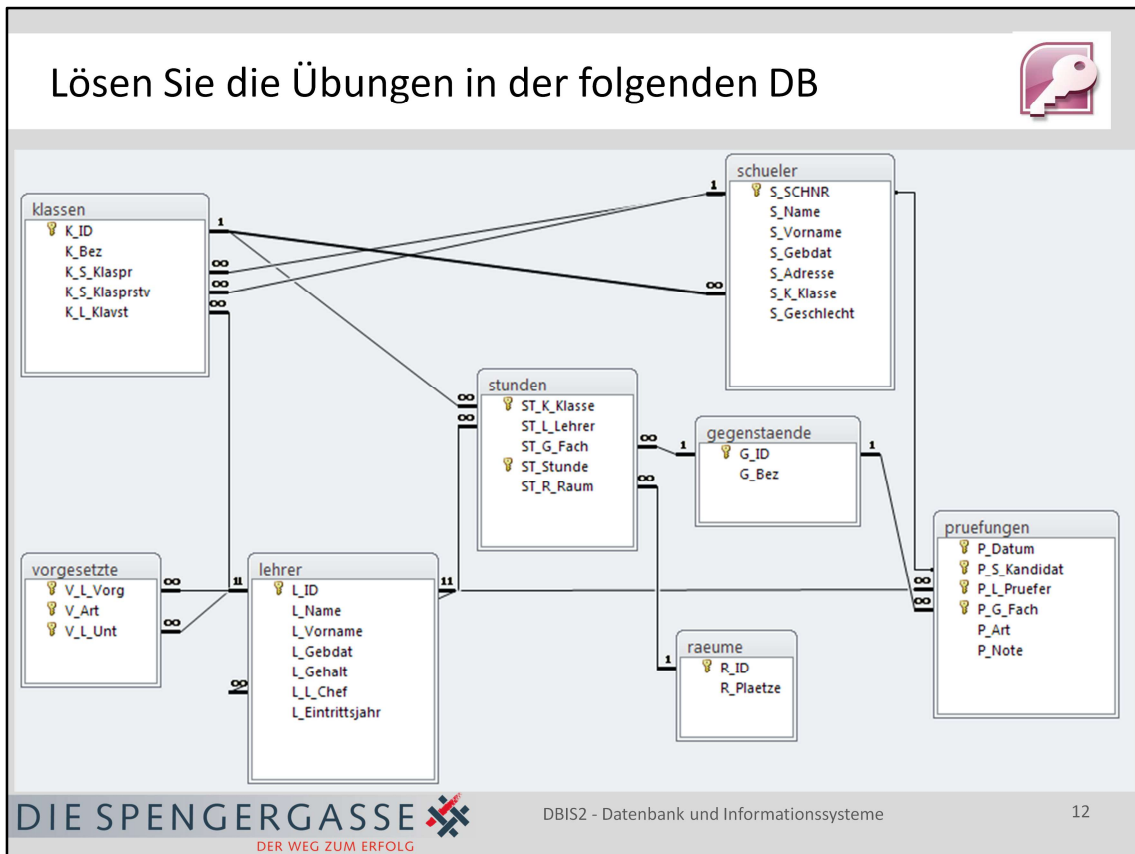
Lösen Sie die Übungen in der folgenden DB



- Verwenden Sie die bereitgestellte Übungs Datenbank und folgen Sie den Angaben am Start Formular!  
Die fertige Db am Stundenende abgeben



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



Dies ist das Modell der Übungsdatenbank

Achtung, Hier gibt es zwischen Schüler und Klasse 3 Beziehungen,

das erfordert auch mitdenken beim Abfrageentwurf

ebenso mehrere Beziehungen bei Lehrer und Vorgesetzte

Leider zeichnet Access in solchen Fällen Tabellen mehrfach ins Beziehungsbild ein

obwohl sie nur einmal real existieren

Die dabei von Access mehrfach gezeichneten Tabellen liegen im Bild übereinander

und sind nur zu sehen, wenn man Schüler oder Lehrertabellen wegzieht