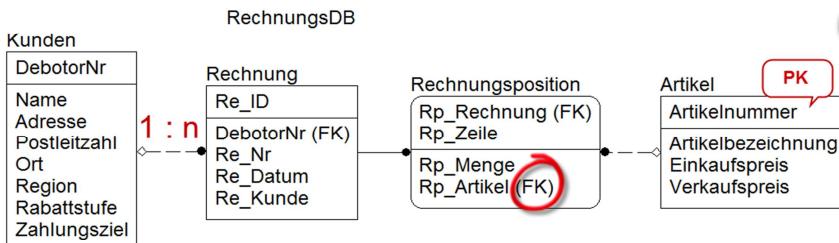


# Lektion 3

## DBIS2 – Datenbanken und Informationssysteme



Können Sie das Bild erklären ??



A01

Die Startfolie enthält ab jetzt wieder(immer) ein Kurzbeispiel zum Aussehen (Modell oder Schema) einer Datenbank.

Hier eine im Erwin erstellte Darstellung der RechnungsDB vom letzten Mal

Die 4 Tabellen sind optisch Kästchen

Es ist immer der Tabellenname angeführt (steht oben darüber)  
und in vielen Fällen auch die Felder (innen drin im Kästchen)

Der Primärschlüssel ist dabei oberhalb des Querstrichs, darunter die normalen Felder

Zur Erinnerung: Primärschlüssel muß immer einen Wert haben und eindeutig sein

- (keine 2 Zeilen mit gleichem PK möglich!!)
- ist nötig um bestimmte Zeilen eindeutig wiederzufinden
- ist der Ausgangspunkt für Beziehungen

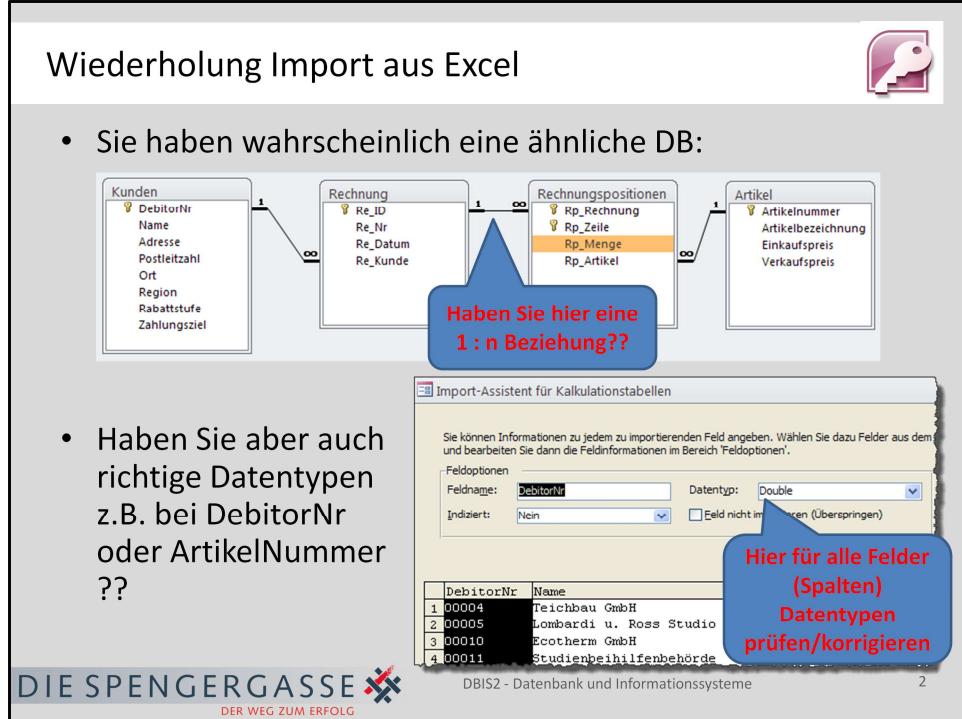
Die Beziehung dazwischen ist als Strich dargestellt,  
wobei (jeweils unterschiedliche Symbole) die Richtung festlegen  
Alle Beziehungen in Datenbanken sind ja 1:n

- der schwarze Knödel ist das n Symbol
- kein schwarzer Knödel somit die 1 er Seite

Wie erkennt man die Fremdschlüssel --- hier durch den zusatz (FK)

Wie liste man eine Beziehung??

Ein Kunde hat n Rechnungen, ABER eine Rechnung gehört NUR zu einem Kunden



Bestehende Daten beim Erstellen einer DB zu benutzen ist praktisch.

Trotzdem müssen Tabellen mit Überlegung erstellt werden

Die Datentypen sollen für die entsprechenden Felder geeignet sein  
 Währung für Einkaufspreis, Verkaufspreis  
 long integer (integer ist in Access nur 2 Byte) für Postleitzahl, Zahlungsziel, ev auch Rabattstufe (falls keine Kommastellen geplant sind)

Die Primary Keys DebitorNr und ArtikelNr sehen in Excel zwar wie Texte aus,  
 das liegt aber nur an den benutzerdefinierten Formatierungen  
 !!!!! In Wahrheit sind sie normale Ganzzahlen --- also long integer  
 Autowert wäre auch praktisch, leider geht das aber nur wenn man noch keine Daten hat

Natürlich gilt auch, dass PK (primary keys) und Beziehungen (1 : n) notwendig sind,  
 welche vorher das Erstellen eines FK (foreign key) mit passendem Datentyp erfordern.

Frage hier: was sind die Datentypen von Re\_ID und Rp\_Rechnung?

## Wiederholung Import aus Excel 2



- Prüfen Sie bitte die Datentypen in Ihrer DB und korrigieren Sie sie bitte falls notwendig!

- Vergleichen Sie mit der ausgegebenen rechnungdb.mdb

Artikel		Feldname	Felddatotyp
		Artikelnummer	Zahl
		Artikelbezeichnung	Text
		Einkaufspreis	Währung
		Verkaufspreis	Währung

Allgemein		Nachschlagen
Feldgröße	Long Integer	
Format	'AN-'00000	
Dezimalstellenanzeige	Automatisch	
Eingabeformat		
Beschriftung		
Standardwert		
Gültigkeitsregel		
Gültigkeitsmeldung		
Eingabe erforderlich	Nein	
Indiziert	Ja (Ohne Duplikate)	
Smarttags		
Textausrichtung	Standard	

Kunden		Feldname	Felddatotyp
		DebitorNr	Zahl
		Name	Text
		Adresse	Text
		Postleitzahl	Zahl
		Ort	Text
		Region	Text
		Rabattstufe	Zahl
		Zahlungsziel	Zahl

Allgemein		Nachschlagen
Feldgröße	Long Integer	
Format	00000	
Dezimalstellenanzeige	Automatisch	
Eingabeformat		
Beschriftung		
Standardwert		
Gültigkeitsregel		
Gültigkeitsmeldung		
Eingabe erforderlich	Nein	
Indiziert	Ja (Ohne Duplikate)	
Smarttags		
Textausrichtung	Standard	

## Wiederholung Import aus Excel 3



- Access ist (in Vergleich zu anderen DBs) sehr großzügig, wenn Sie Datentypen in Tabellen verändern wo schon Datensätze enthalten sind. Wenn möglich werden die Datenfelder konvertiert, andernfalls aber gelöscht – beachten Sie Warnungen.
- PK und FK Typen kann man nur ändern wenn die Beziehung vorübergehend entfernt wird.
- In schwierigen Fällen kann man eine neue leere Tabelle (mit richtigen Typen) anlegen, die Beziehungen umstellen, die Daten transferieren und die alte Tabelle löschen.

Daten transferieren kann man mittels Abfragen (know how vorausgesetzt)

Würde man z.B. ArtikelNr von Text nach Zahl umwandeln wollen und die ArtikelNr enthielte Einträge wie „AN-00022“, dann geht das mal nicht!

Felder mit Buchstaben werden bei der Typumwandlung gelöscht.

Daher zuerst die ersten 3 Stellen entfernen  
geht händisch oder mit einer Änderungsabfrage

(update Artikel set ArtikelNr = mid(ArtikelNr;4;100))

Danach kann der verbliebene Ziffernteil in eine Zahl umgewandelt werden und wir brauchen gar keine neue Tabelle

## Abfragen in Access (und anderen Datenbanken)



- **SQL** (Structured query language) ist die Abfragesprache für alle (relationalen) Datenbanken
- In verschiedenen DBs gibt es zwar syntaktische Unterschiede, die Befehle und die Logik von SQL sind aber überall gleich.
- SELECT liest Daten, UPDATE, INSERT, DELETE verändert diese
- Access unterstützt uns auch hier mit Assistenten und Abfrageentwurfsfenstern

MS-SQL oder MySQL tragen ja ihre Abfragesprache schon im Namen.

SQL zeichnet sich gleichzeitig durch Einfachheit und Mächtigkeit aus.

SQL ist keine prozedurale sondern eine deklarative Sprache,  
daher gibt es keine Schleifen, kein if

In anderen DBs erstellt man auch die Tabellen mit Befehlen wie CREATE, ALTER, DROP

wir haben dazu die Access Tools verwendet  
(Tabellenentwurf und Beziehungsfenster)

SELECT welche spalten  
FROM welche Tabelle  
WHERE unter welcher Bedingung

Mittels SQL greifen auch Programmiersprachen (wie Java, php)  
auf Datenbanken zu

Abfragen in Access 2

The screenshot shows two windows. The top window is titled 'Artikel' and contains a table with columns: Artikelnummer, Artikelbezeichnung, Einkaufspreis, and Verkaufspreis. A red arrow points from the 'Artikelnummer' column header to the SQL code. Another red arrow points from the 'Einkaufspreis' column header to the condition in the WHERE clause. A blue callout box labeled 'Welche Spalten in die Ausgabe' points to the 'Ausgabe' section. A blue callout box labeled 'Welche Zeilen in die Ausgabe (Bedingung)' points to the WHERE clause. The bottom window is titled 'Abfrage1' and shows a table with the same four columns, containing three rows of data. A blue callout box labeled 'Ausgabe' points to this window.

Artikelnummer	Artikelbezeichnung	Einkaufspreis	Verkaufspreis
AN-00002	Hitachi Deskstar 2000GB, SATA II (H3K20003272SE/0502574)	92,00	110,00
AN-00003	Hitachi Deskstar 7K1000.B 1000GB, SATA II (HDT721010SLA360)	95,00	101,20
AN-00004	Samsung EcoGreen F4 2000GB, SATA II (HD204UI)	45,00	60,00
AN-00005	Western Digital Caviar Black 2000GB, SATA II (WD2001FASS)	89,00	120,00
AN-00006	Samsung EcoGreen F3 2000GB, SATA II (HD203WI)	59,00	70,00
AN-00007	Intel Core 2 Quad Q9650, 4x 3.00GHz, boxed (BX80569Q9650)	195,00	220,00
AN-00008	Intel Celeron Dual-Core E3400, 2x 2,60GHz, boxed (BX80571E3400)	22,00	30,00
AN-00009	Intel Core 2 Quad Q8400, 4x 2,67GHz, tray (AT80580P0674ML)	89,00	99,80
AN-00010	ASUS P5Q3, P45 (dual PC3-8500U DDR3) (90-MIIB6B0-GOEAY00Z)	70,00	92,00
AN-00011	Gigabyte GA-P43T-ES3G, P43 (dual PC3-8500U DDR3)	38,00	50,00
AN-00012	Elitegroup G41T-R3, G41 (dual PC3-10667U DDR3)	17,00	25,00

Datensatz: 14 von 12 | Kein Filter | Suchen

Artikelnummer	Einkaufspreis
AN-00008	22,00
AN-00011	38,00
AN-00012	17,00

Abfrage1

DIE SPENGERGASSE

DBIS2 - Datenbank und Informationssysteme 6

```

SELECT Artikelnummer, Einkaufspreis
FROM Artikel
WHERE Einkaufspreis < 45
  
```

Aus dieser Tabelle

Welche Zeilen in die Ausgabe (Bedingung)

Ausgabe

SELECT welche spalten  
 FROM welche Tabelle  
 WHERE unter welcher Bedingung

Der Output eines SQL Select Statements sieht aus wie eine Tabelle  
 und damit ähnlich wie der Input (die Tabelle beim from)

Hier ist es auch eine echte Teilmenge von Spalten und Zeilen-

Abfragen in Access 3

- Erstellen Abfrageentwurf  
Felder eintragen  
Kriterium eintragen
- Ausführen oder Ansicht Datenblatt

DIE SPENGERGASSE 

DBIS2 - Datenbank und Informationssysteme 7

Zuerst Erstellen Abfrageentwurf

In Tabelle anzeigen die gewünschte Tabelle hinzufügen

In der Abfrage Entwurfsansicht die gewünschten Felder eintragen (oder via drag and drop runterziehen)

In Kriterien die Bedingung <45 eintragen

Mittels Ausführen (oder Ansicht Datenblatt) erhält man das Ergebnis

Mittels Ansicht SQL sieht man den erstellten SQL Select Befehl

```
SELECT Artikel.Artikelnummer, Artikel.Einkaufspreis
FROM Artikel
WHERE (((Artikel.Einkaufspreis)<45));
```

Bei allen Feldnamen wurde tabellenname.feldname geschrieben, was möglich aber nicht unbedingt nötig ist

gelegentlich werden feldnamen in [] eingeschlossen [Verkaufspreis]  
ist nötig wenn Sonderzeichen im feldnamen wären

Eine gewisse Sucht nach Klammern kann in der Bedingung festgestellt werden  
Der ; am Ende ist nicht verboten aber unnötig

Im allgemeinen können Sie Abfragen erstellen wie Sie möchten,  
sollten aber einen geschriebenen Select Befehl lesen können!

## Abfragen in Access 4



- Select ... feldname, tabname.feldname, ausdruck  
SELECT Artikelnummer, Verkaufspreis \* 1,2 as Brutto
- Where bedingung\_wie\_in\_Programmiersprachen  
WHERE (feld < 2 OR feld2 = 5) AND feld3 <> „Huber“

Operatoren im WHERE	
< > <= >= = <>	Übliche Vergleichsoperatoren
IS NULL IS NOT NULL	Feld ist leer / ist nicht leer
BETWEEN 12 AND 16	Im Bereich von 12 – 16 (inklusiv)
IN (12;13;14;16)	Einer der aufgezählten Werte (or)
LIKE "likestring"	* und ? haben eine Sonderbedeutung, * steht für 0 bis n beliebige Zeichen ? Steht für genau ein beliebiges Zeichen
NOT, AND, OR	dasselbe wie !, &&,

Im SELECT Zweig sind feldnamen und Rechenausdrücke (basierend auf feldnamen) erlaubt

```
SELECT      Artikelnummer,  
            Artikel.Einkaufspreis,  
            Verkaufspreis * 1.2 AS Brutto,          -- Brutto ist die  
„Überschrift“  
            Verkaufspreis - einkaufspreis AS Handelsspanne  
WHERE einkaufspreis < 100 and  
      verkaufspreis - einkaufspreis > 0
```

Operatoren in Bedingungen:

>< >= <= = <> // <> ist in manchen DBs auch !=, = ist aber niemals ==  
IS NULL IS NOT NULL // Feld ist leer  
BETWEEN 12 AND 16 // im Bereich also feld >= 12 AND feld <= 16  
IN (12;13;14;16) NOT IN (1;2;3;5;7;11;13;19)  
LIKE "likestring"

Like verwendet eine konstanten String mit Wildcardzeichen

\* steht für 0 bis n beliebige, ? Für genau eine beliebige Stelle (Hinweis: in allen anderen DBs ist dies % und \_)

(z.B. like "2?HIF" steht für alle 2. Klassen, like "\*BM\*" für alle Klassen aus der BM Abteilung,

feld like "Tech\*" für alle Inhalte die mit Tech beginnen

NOT, AND, OR sind dasselbe wie !, &&, ||

## Abfragen in Access 5



- WHERE name = "Mayer" -- " oder ' bei Texten
- WHERE geburtsdatum < #15.02.2011#
- SELECT zuname & " " & vorname as name
- Funktionen

Funktionen	
Left(feld,2) Mid(feld,2,1) Right(feld,3), Instr(feld,'S')	Substring Funktionen um Textteile aus einem Textfeld zu extrahieren / Instr sucht die Stelle, wo 'S' vorkommt
Day(datum), datumsfeld-12, Datumsfeld-datumsfeld2	Day, Month, Year liefern Datumsteile, man kann cien Anzahl Tage addieren/subtrahieren, Datum – Datum liefert die Tagedifferenz
Wert(textfeld)	Konvertiert Text nach Zahl
Count, Min, Max, Sum, Avg	„Gruppenfunktionen“ – werden später behandelt

Textkonstante werden mit Anführungszeichen oder auch mit Hochkomma geschrieben " oder '

Datumskonstante sind unterschiedlich,  
in der Entwurfsansicht < #15.02.2011# , in der SQL Ansicht aber < #2/15/2011#

SELECT zuname & " " & vorname as name  
ist ein Beispiel für Textverkettung, + wie in Java funktioniert auch, allerdings soll dann kein Feld leer sein

SELECT [dummeNamen müssen in eckige Klammern], zuname from tabelle

Zahlreiche Funktionen (aus Visual Basic) können eingesetzt werden  
wichtig ist, dass in der Entwurfsansicht die Funktionsnamen deutsch sind und zwischen den parametern ein Strichpunkt steht  
in SQL Ansicht sind die Funktionsnamen englisch und zwischen Parametern steht ein Beistrich.

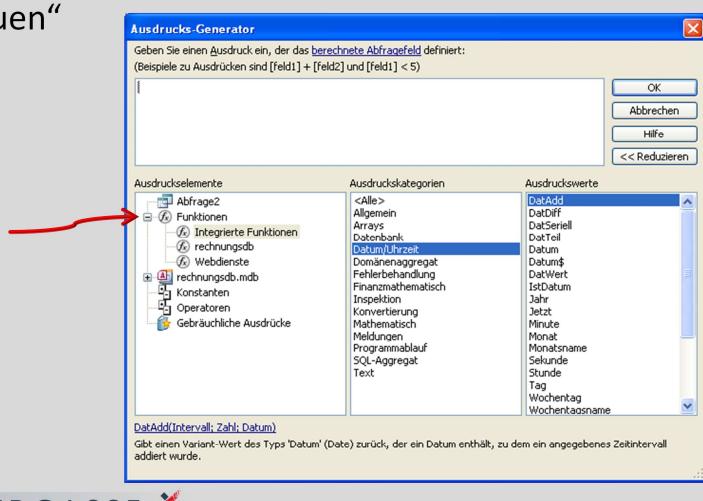
Left(feld,2) Mid(feld,2,1) Right(feld,3) liefern Textteile, Instr(feld,'S') liefert die Stelle, wo S vorkommt

SELECT Left([Name],3) AS Ausdr1, Mid([Region],8) AS Ausdr2, Right([Ort],1) AS Ausdr3,  
Instr(Region,"SÜD") as AnwelcherStelle  
FROM Kunden;

## Abfragen in Access 6



- Hilfe bei Funktionen rechte Maustaste „Aufbauen“



Bei Funktionen gibt es leider gravierende Unterschiede zwischen verschiedenen Datenbanksystemen

## Abfragen in Access 7



- ORDER BY zum beliebigen Sortieren  
ORDER BY feld1, feld2, feld3 DESC
- SELECT name, postleitzahl, Ort, Region FROM Kunden  
order by Region, Postleitzahl, Name

Feld:	Spalte:	Ausdruck:	eigen:
name	Kunden	Kunden	<input checked="" type="checkbox"/> Aufsteigend
postleitzahl	Kunden	Kunden	<input checked="" type="checkbox"/> Aufsteigend
Ort	Kunden	Kunden	<input checked="" type="checkbox"/> Absteigend
Region	Kunden	Kunden	<input type="checkbox"/>
Postleitzahl	Kunden	Kunden	<input type="checkbox"/>
Name	Kunden	Kunden	<input type="checkbox"/>

DBIS2 - Datenbank und Informationssysteme

11

SQL versteht sich als mengenorientierte Sprache.

Daher hat die Ausgabe eine zufällige Sortierung, diese kann sich bei mehrmaligen Ausführen des gleichen Befehls ändern!

Natürlich nicht das Ergebnis sondern nur die Reihenfolge der Ausgabezeilen

Mit ORDER BY legt man eine bestimmte Sortierung fest

beliebige viele Sortfelder sind möglich, jedes Sortfeld kann den Zusatz DESC (für absteigend haben)

auch Rechenausdrücke sind möglich, alle Felder müssen aber aus der Tabelle beim from stammen

```
SELECT name,postleitzahl,Ort,Region
FROM Kunden
order by Region, Postleitzahl, Name
```

Abfragen in Access Übungen für Sie schuldb1\_V3.mdb 

- Lehrer mit Sprechstunde am Montag, sortiert nach Lehrerkürzel
- Welche Lehrer haben keine Sprechstunde eingetragen
- Welche Klassen gehören zur Abteilung 3
- Welche Klassen enden vor dem Schuljahresende (haben Inhalt in K\_Datumbis)

L_Nr	L_Zuname	L_Sprechstu
BI	Binder	Montag
CAM	Camrda	Montag
FM	Furtlechner	Montag
GT	Griesmayer	Montag

L_Nr	L_Zuname	L_Sprechstu
DW	Diewald	
HAR	Harand	
HI	Hickel	
*		

K_Nr	K_Abteilung
1AFID	3
1AHBGS	3
1BFID	3
2AFID	3
3AFID	3
3BFID	3
4AFID	3

K_Nr	K_Abteilung	K_Datumbis
4AFID	3	20.02.2011
SAHDVK	4	15.05.2011
SAHDVM	4	15.05.2011
SAHDVN	4	15.05.2011

**DIE SPENGERGASSE**   
DER WEG ZUM ERFOLG

DBIS2 - Datenbank und Informationssysteme 12

Befehle bitte in einem Textfile sammeln und dem Lehrer vor dem Stundenende zeigen

Hinweis: Falls Sie eine Rückfrage mit Parameter eingeben von Access erhalten,  
dann haben Sie einen Feldnamen falsch geschrieben  
(genau den Text, für den Sie angeblich einen Parameter eingeben sollen)

Natürlich gibt es auch normale Syntaxfehler zu überwinden

## Abfragen in Access Übungen für Sie schuldb1\_V3.mdb

- Welche Klassen sind Fachschulklassen (enthalten FID)
- Welche Klassen gehören zum 1. Jahrgang (1,2 oder 3)
- Welche Klassen hat Skolud (SKO) als Klassenvorstand
- Welche Schuljahre beginnen im Jahr 2009
- Wie lange dauern die verschiedenen Schuljahre
- Welche Schüler (Ausgabe sortieren) besuchen Biomedizin- und Gesundheitstechnik (HBG)
- Welche Schüler der Abteilung HIF haben den Vornamen Michael

Alle Abfragen haben eine Ausgabe, falls nicht dürfte ein logischer Fehler vorliegen

## Abfragen in Access Übungen für Sie Bonuscharakter



- Welche Schüler wohnen an einem „weg“ (z.B. am Rosenweg) in den Bezirken 2, 12 oder 22
- Welche Schüler wohnen nicht in Wien und sind vor 1993 geboren
- Welche Schüler wohnen in Bezirken, die an die Ringstraße bzw. die 2er-Linie grenzen
- Was bedeutet die folgenden Bedingungen?

where s\_geschlecht = month(s\_gebdatum)

where Instr(S\_Hausnummer,'/') = 0

Was ist die 2er Linie?

verläuft etwas ausserhalb parallel zur Ringstraße

<http://de.wikipedia.org/wiki/Zweierlinie>