

Wirtschaftsinformatik 2

LE 11 – SQL DDL und DCL

Prof. Dr. Thomas Off

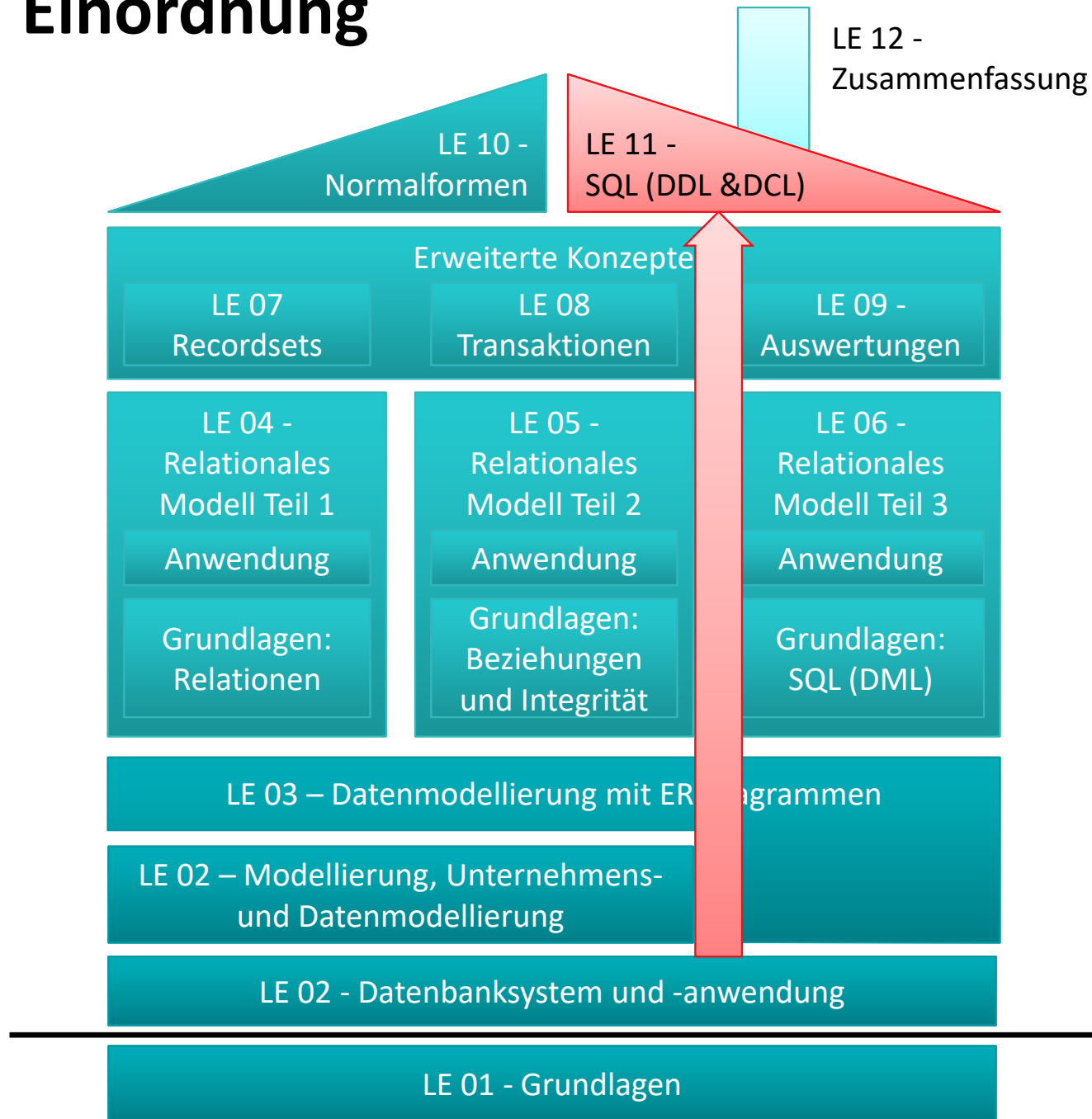
<http://www.ThomasOff.de/lehre/beuth/wi2>

Ziel

Ziel dieser Lehreinheit

- Kennenlernen und Anwenden weiterer SQL Befehle
- Erzeugen von Datenbankstrukturen (Data Definition Language) mit SQL
- Einrichten und Entziehen von Zugriffsrechten (Data Control Language) mit SQL

Einordnung



Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Rückblick: Normalisierung und Normalformen

Normalisierung als Prozess der anhand definierter Regeln, der Normalformen der verlustfreien Optimierung dient

Normalformen 1 bis 3¹⁾

– In Anlehnung an:

"Die Wahrheit, die ganze Wahrheit und nichts als die Wahrheit.
So wahr mir Gott helfe!"

– Bedeutet Normalisierung:

"Der Schlüssel, der ganze Schlüssel und nichts als der Schlüssel.
So wahr mir Codd helfe!"

- alle atomaren Attribute beziehen sich auf den Schlüssel (1. NF)
- sie beziehen sich auf den gesamten Schlüssel (2. NF)
- die Attribute hängen nur vom Schlüssel ab und nicht von weiteren Attributen (3. NF)

1) nach Wikipedia: Normalisierung (Datenbanken). http://de.wikipedia.org/wiki/Normalisierung_%28Datenbank%29#Merkspruch

Antworten auf Frage: "In welcher NF ist Relation R?"

- Ausgehend von 1. NF prüfen
 - 1. NF bedeutet, dass ...
 - Die Tabelle befindet sich in 1. NF, weil .../ nicht in 1. NF, weil ...
- Ist sie in 1. NF, dann weiter prüfen ob 2. NF
 - ...
- Ist sie in 2. NF, dann weiter prüfen, ob 3. NF
 - ...
- ...
- Abschließende Argumentation: Sie ist in n. NF, weil Sie ist aber nicht in n+1. NF, weil

LE 00 - ~~Belegungen~~



Rückblick: SQL DML

Bestandteile des SQL-Sprachumfangs

- Data Manipulation Language (DML)
- Data Definition Language (DDL)
- Data Control Language (DCL)

Rückblick (LE05)

Bestandteile des SQL-Sprachumfangs

- Data Manipulation Language (DML): dient zur Abfrage, zum Hinzufügen, zur Veränderung und zum Löschen von Daten
 - SELECT
 - UPDATE
 - INSERT
 - DELETE
- Data Definition Language (DDL)
- Data Control Language (DCL)

Rückblick (LE05)

Bestandteile des SQL-Sprachumfangs

- Data Manipulation Language (DML)
- Data Definition Language (DDL): dient zum Erzeugen, Verändern und Löschen der Strukturen, die für die Speicherung der Daten benutzt werden (z.B. Tabellen, Spalten)
 - ?
 - ?
 - ?
- Data Control Language (DCL)

Rückblick (LE05)

Bestandteile des SQL-Sprachumfangs

- Data Manipulation Language (DML)
- Data Definition Language (DDL)
- Data Control Language (DCL): dient zum Einrichten, Festlegen und Entziehen von Zugriffsrechten für Benutzer und Gruppen auf den Strukturen und Aktionsmöglichkeiten der Datenbank
 - ?
 - ?

Rückblick (LE05)

Bestandteile des SQL-Sprachumfangs

- Data Manipulation Language (DML)
- Data Definition Language (DDL)
- Data Control Language (DCL)

LE 08 - SQL Programm



Rückblick (LE02)

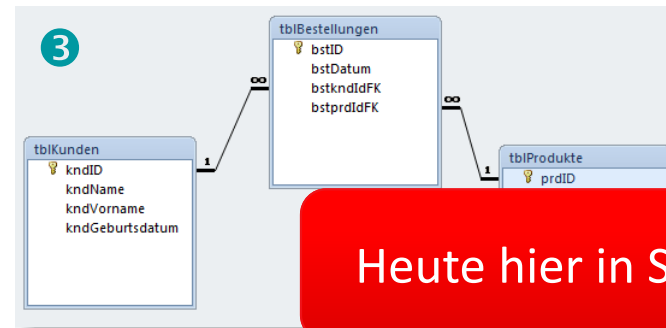
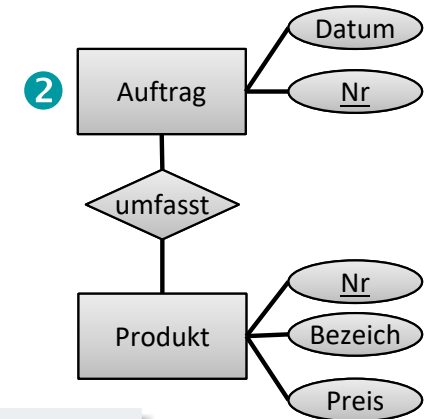
Datenmodellierung als Prozess, in dem

- die relevanten Informationsobjekte mit ihren Eigenschaften
- auf Modelle abgebildet werden

Prozessphasen

- Diskursbereich beschreiben
- Informationsmodell erstellen
- Datenmodell ableiten
- Schema implementieren

Schema wird anschließend in einem DBS umgesetzt und eine Anwendung zur Nutzung der Datenbank implementiert



Heute hier in Schritt 4!

```

4 CREATE TABLE tblBestellungen
  (bstID AUTOINCREMENT,
   bstDatum DATETIME,
   bstkndIdFK INTEGER,
   bstprdIdFK INTEGER
  )

-- ...

```

Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Ausgangssituation

Entwurf des Datenmodells in MS Access

- grafische Benutzeroberfläche ermöglicht
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - Löschen von Tabellen
- Tabellen werden in verschiedenen Sichten dargestellt
- Tabellen können mit Daten gefüllt werden
 - durch manuelle Erfassung
 - durch Import strukturierter Daten/Dateien

Werkzeugfunktionen und Handhabung ist herstellerspezifisch und kann sich bei unterschiedlichen Herstellern unterscheiden

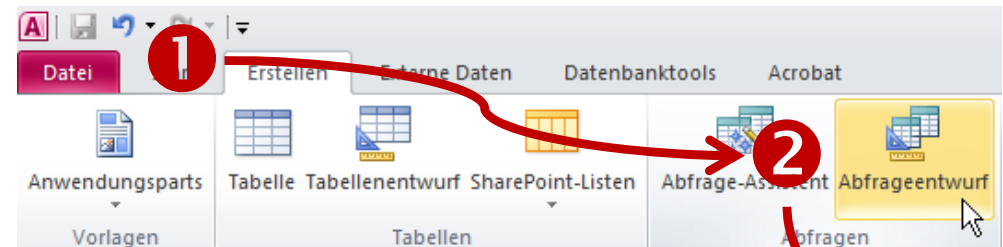
SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - Löschen von Tabellen

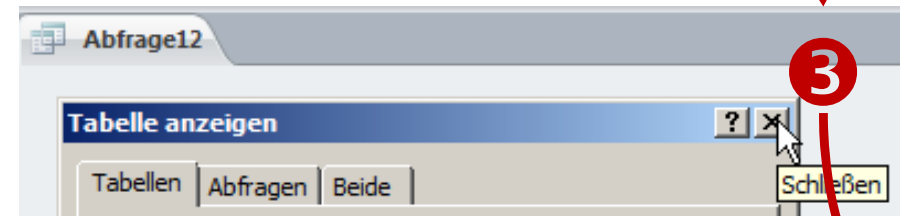
SQL DDL in MS Access

Befehle der SQL Data Definition Language werden als Abfrage vom Typ "Datendefinition" ausgeführt

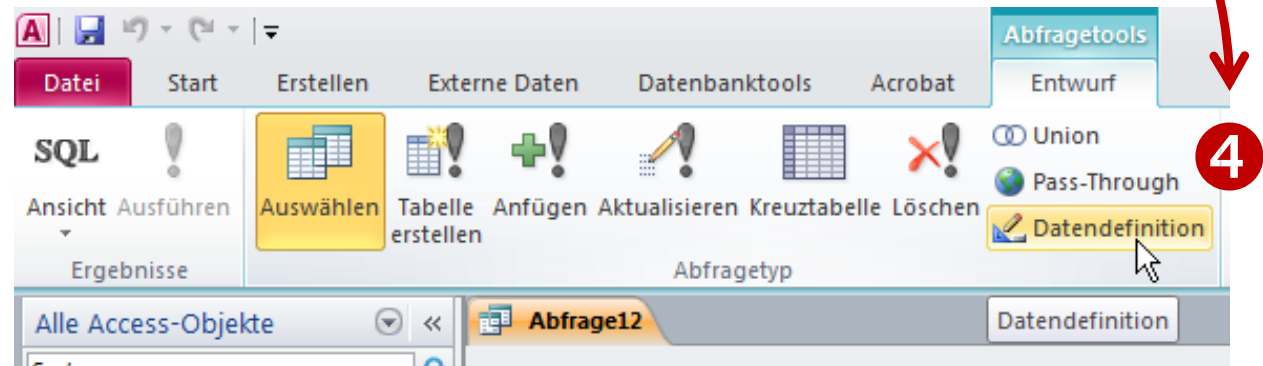
– Erstellen "Abfrageentwurf"



– Dialog "Tabellen anzeigen" schließen



– Abfragetyp umschalten



SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - Löschen von Tabellen

SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - **Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen**
 - Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - Löschen von Tabellen

Überblick über die Syntax zur Erzeugung neuer Relationen

```
CREATE TABLE <Tabellename>
(
  <Spaltenname1> <Datentyp> <Optionale Bedingung>,
  <Spaltenname2> <Datentyp> <Optionale Bedingung>,
  <...>
  <SpaltennameN> <Datentyp> <Optionale Bedingung>

  <Optionale Tabellenbedingung>
)
```

Relationen erzeugen

Ausgewählte SQL-Datentypen für Tabellenspalten

Datentyp (ggf. mit Kurz- und Langform)	Erläuterung	Hinweis zu MS Access
INT oder INTEGER, SMALLINT	Ganzzahlige Werte	Als Long Integer bzw. Integer umgesetzt
CHAR(x) oder CHARACTER(x)	Zeichenketten fester Länge	-
VARCHAR(x)	Zeichenketten variabler Länge	-
DATE, TIME	Datum (Jahr, Monat, Tag), Uhrzeit (Stunde, Minute, Sekunde)	Als Datum/Uhrzeit umgesetzt
NUMERIC(x,y), FLOAT(x,y)	Gebrochene Zahl mit Angabe der Stellen und Nachkommastellen	Nur ohne Angabe der Stellen und als Double umgesetzt.
BIT(x)	Bitfolge fester Länge	Nur ohne Abgabe der Länge als Bit-Wert und Umsetzung als Ja/Nein

Relationen erzeugen

Weitere MS Access spezifische Datentypen

Datentypen	Erläuterung
TEXT, MEMO	Textfeld und langes, mehrzeiliges Textfeld
BYTE, LONG	Ganzzahlige Werte vom Typ Byte und Long, als Ergänzung zu SHORTINT (für Integer) und INTEGER für Long
SINGLE, DOUBLE	Gebrochene Werte für einfache und doppelte Genauigkeit
CURRENCY	Währungsbeträge
AUTOINCREMENT	AutoWert
DATETIME	Datum/Uhrzeit
YESNO	Ja/Nein
OLEOBJECT	Bild oder anderes Objekt, das als Binärdaten vorliegt

Beispiel 1 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Produkte
(
  Name VARCHAR(15),
  LieferTermin DATE,
  Hersteller VARCHAR(25),
  Preis CURRENCY
);
```

Als Relation

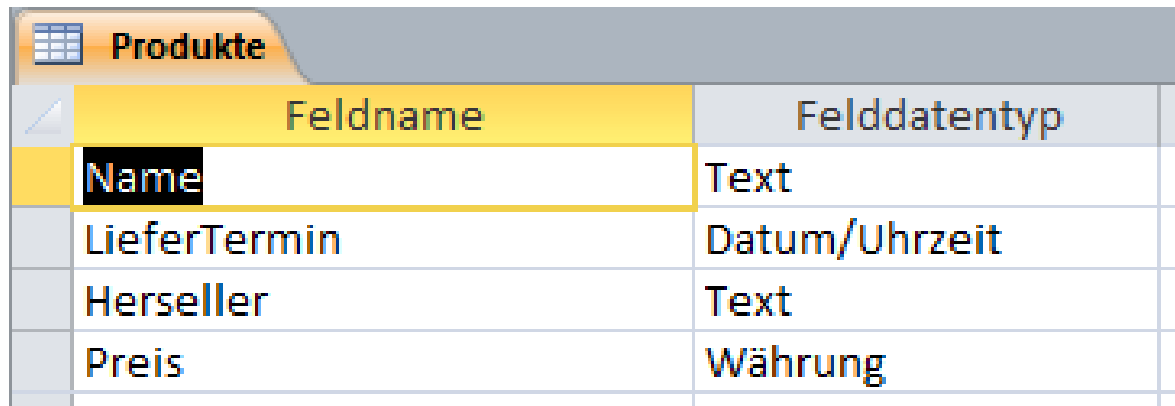
Produkte	Name	LieferTermin	Hersteller	Preis

Beispiel 1 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Produkte
(
  Name VARCHAR(15),
  LieferTermin DATE,
  Hersteller VARCHAR(25),
  Preis CURRENCY
);
```

In MS Access Entwurfsansicht



Feldname	Felddatentyp
Name	Text
LieferTermin	Datum/Uhrzeit
Hersteller	Text
Preis	Währung

Relationen erzeugen

Bedingungen, Primär- und Fremdschlüssel für einzelne Spalten anlegen

- Einordnung in den CREATE TABLE-Befehl

```
CREATE TABLE <Tabellename> (  
    <Spaltenname> <Datentyp> <Optionale Bedingung> ,  
    <...>  
);
```

- Varianten der optionalen Bedingung

```
CONSTRAINT <Bedingungsname> NOT NULL  
CONSTRAINT <Bedingungsname> PRIMARY KEY  
CONSTRAINT <Bedingungsname>  
    REFERENCES <Tabellename> (<opt.Spalte1>, ...)  
    ON DELETE CASCADE oder SET NULL  
    ON UPDATE CASCADE oder SET NULL  
CONSTRAINT <Bedingungsname> CHECK <Bedingung>
```

Beispiel 2 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Produkte
(
  IdPk INTEGER CONSTRAINT PrimSchluessel PRIMARY KEY,
  Name VARCHAR(15) CONSTRAINT Produktname NOT NULL,
  LieferTermin DATE,
  Hersteller VARCHAR(25),
  Preis CURRENCY
);
```

Als Relation


Produkte	<u>IdPk</u>	Name	LieferTermin	Hersteller	Preis

Beispiel 2 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Produkte
(
  IdPk INTEGER CONSTRAINT PirmSchluessel PRIMARY KEY,
  Name VARCHAR(15) CONSTRAINT Produktname NOT NULL,
  LieferTermin DATE,
  Herseller VARCHAR(25),
  Preis CURRENCY
);
```

In MS Access Entwurfsansicht

Feldname	Felddatentyp
 IdPk	Zahl
Name	Kurzer Text
LieferTermin	Datum/Uhrzeit
Herseller	
Preis	

Eingabe erforderlich	Ja
	Ja

Beispiel 3 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Bestellung
(
  IdPk INTEGER CONSTRAINT PirmSchluessel PRIMARY KEY,
  BestellDatum DATE,
  Anzahl SMALLINT,
  prdIdFk INTEGER CONSTRAINT FremdschluesselPrd
    REFERENCES Produkte(IdPk)
);
```

Als Relation

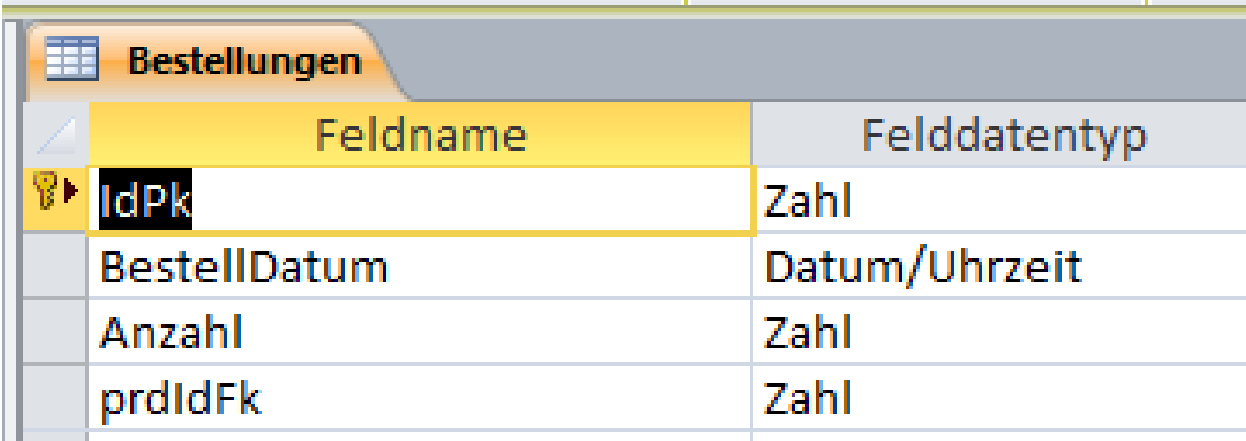
Bestellung	<u>IdPk</u>	BestellDatum	Anzahl	<u>prdIdFk</u>


Beispiel 3 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Bestellung
(
  IdPk INTEGER CONSTRAINT PirmSchluessel PRIMARY KEY,
  BestellDatum DATE,
  Anzahl SMALLINT,
  prdIdFk INTEGER CONSTRAINT FremdschluesselPrd
      REFERENCES Produkte(IdPk)
);
```

In MS Access Entwurfsansicht



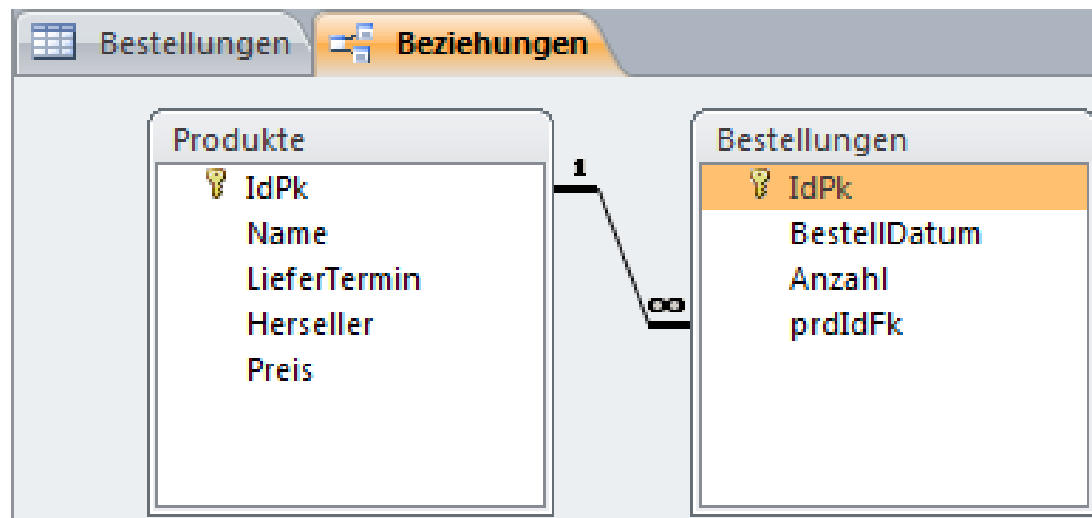
Feldname	Felldatentyp
 IdPk	Zahl
BestellDatum	Datum/Uhrzeit
Anzahl	Zahl
prdIdFk	Zahl

Beispiel 3 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Bestellung
(
  IdPk INTEGER CONSTRAINT PirmSchluessel PRIMARY KEY,
  BestellDatum DATE,
  Anzahl SMALLINT,
  prdIdFk INTEGER CONSTRAINT FremdschluesselPrd
    REFERENCES Produkte(IdPk)
);
```

In MS Access Beziehungsansicht



Relationen erzeugen

Bedingungen, Primär- und Fremdschlüssel für die gesamte Tabelle anlegen

- Einordnung in den CREATE TABLE-Befehl

```
CREATE TABLE <Tabellename> (  
    <Spaltenname> <Datentyp> <Optionale Bedingung>,  
    <...>  
    <Optionale Bedingung>  
);
```

- Varianten der optionalen Bedingung

```
CONSTRAINT <Bedingungsname> PRIMARY KEY (<Spalte1>, ...)  
CONSTRAINT <Bedingungsname>  
    FOREIGN KEY (<Spaltenname>)  
        REFERENCES <Tabellename> (<opt.Spalte1>, ...)  
        ON DELETE CASCADE oder SET NULL  
        ON UPDATE CASCADE oder SET NULL  
CONSTRAINT <Bedingungsname> CHECK <Bedingung>
```


Beispiel 4 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Verkaeufer (  
  VerkIdPk INTEGER,  
  AbtIdPk INTEGER,  
  Name VARCHAR(25),  
  Gehalt CURRENCY,  
  AbtLeister BIT,  
  
  CONSTRAINT PrimSchluesselVerk  
    PRIMARY KEY (VerkIdPk, AbtIdPk));
```

Als Relation

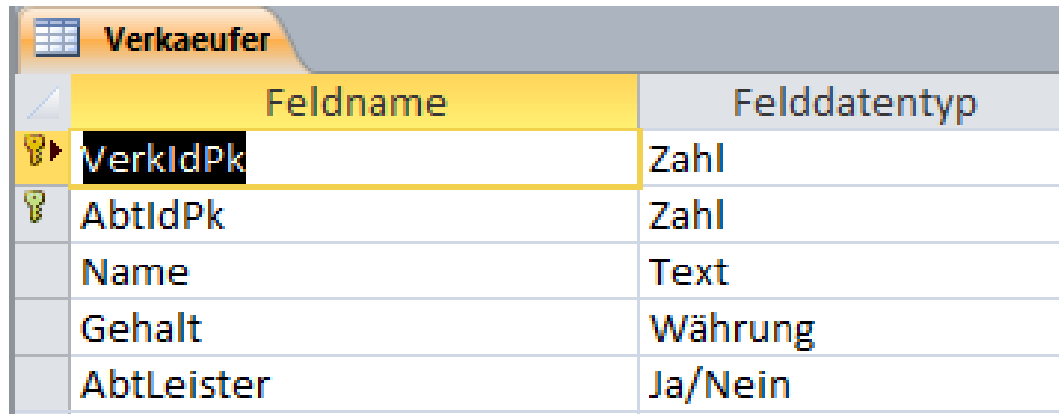
Verkaeufer	<u>VerkIdPk</u>	<u>AbtIdPk</u>	Name	Gehalt	AbtLeiter

Beispiel 4 (für MS Access)

SQL (für MS Access)

```
CREATE TABLE Verkaeufer (  
    VerkIdPk INTEGER,  
    AbtIdPk INTEGER,  
    Name VARCHAR(25),  
    Gehalt CURRENCY,  
    AbtLeister BIT,  
  
    CONSTRAINT PrimSchluesselVerk  
        PRIMARY KEY (VerkIdPk, AbtIdPk));
```

In MS Access Entwurfsansicht



Feldname	Felddatentyp
VerkIdPk	Zahl
AbtIdPk	Zahl
Name	Text
Gehalt	Währung
AbtLeister	Ja/Nein

D11.1: Aufgabe

- Erstellen Sie in einer leeren MS Access Datenbank jeweils eine Datendefinitionsabfrage zu Erzeugen einer Tabelle für
 - Personen (Name, Geburtsdatum) mit dem Primärschlüssel perIdPk
 - Telefonnummern (Rufnummer inkl. Vorwahl), dem Primärschlüssel telIdPk und dem Fremdschlüssel zur Person telperIdFk
- Führen Sie die beiden Abfragen aus und erfassen Sie dann einige Testdaten für beide Tabellen.

Demo 11.1: SQL DDL – Erzeugen von Tabellen

D11.1: Lösungsvorschlag

– Tabelle Personen

```
CREATE TABLE tblPersonen (  
    perIdPk INTEGER CONSTRAINT PersonenPerIdPk PRIMARY KEY,  
    perName VARCHAR(25),  
    perGebDatum DATE  
);
```

– Tabelle Telefonnummern

```
CREATE TABLE tblTelefonnummern  
(  
    telIdPk INTEGER CONSTRAINT PirmSchluessel PRIMARY KEY,  
    telNummer TEXT,  
    telperIdFk INTEGER CONSTRAINT TelefonnummernTelPerIdFk  
        REFERENCES tblPersonen(perIdPk)  
);
```

SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - **Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen**
 - Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - Löschen von Tabellen

SQL-Standard

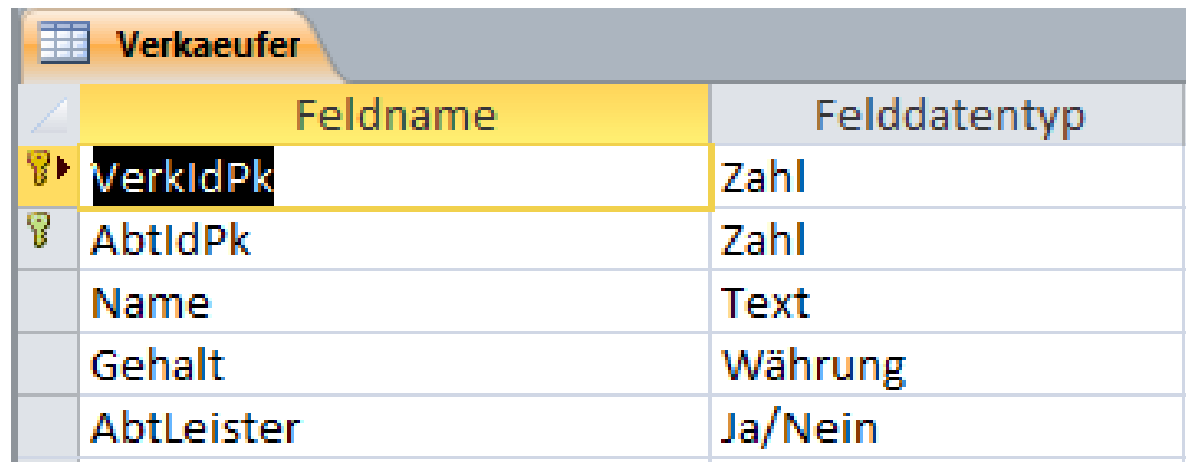
- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - **Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen**
 - Löschen von Tabellen

Überblick über die Syntax zur Änderung von Relationen

```
ALTER TABLE <Tabellenname>  
  ADD COLUMN <Spaltenname> <Datentyp> <Optionale Bedingung>  
  oder  
  DROP COLUMN <Spaltenname>  
  oder  
  ADD CONSTRAINT <Bedingung>  
  oder  
  DROP CONSTRAINT <Bedingungsname>  
;
```

Beispiel 5 (für MS Access)

Vorher
(Entwurfsansicht)

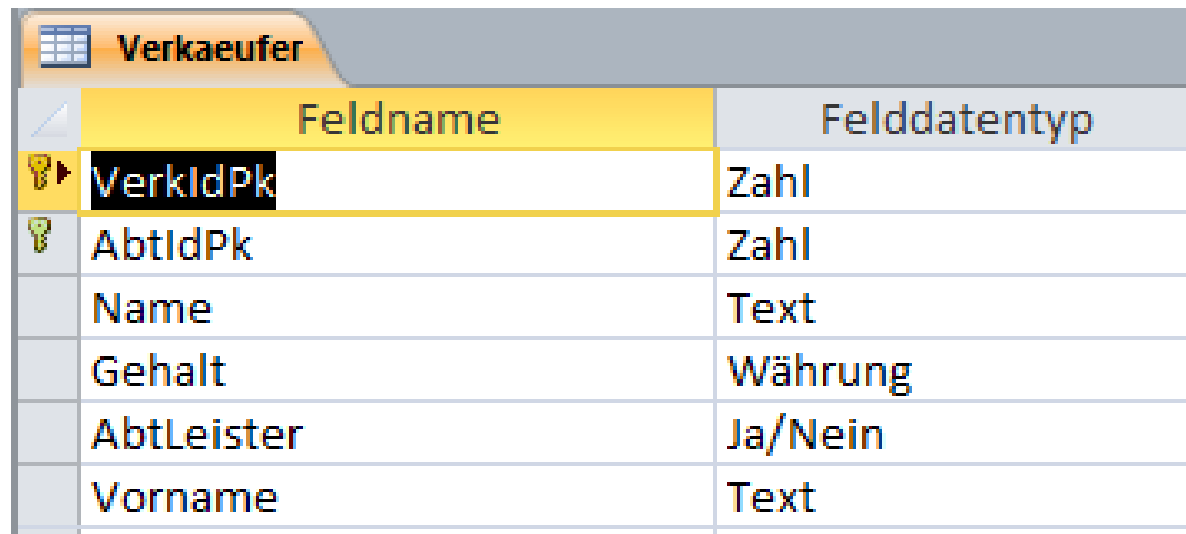


	Feldname	Felddatentyp
🔑	VerkIdPk	Zahl
🔑	AbtIdPk	Zahl
	Name	Text
	Gehalt	Währung
	AbtLeister	Ja/Nein

SQL (für MS Access)

```
ALTER TABLE Verkaeufer  
ADD COLUMN Vorname VARCHAR(25);
```

Nachher
(Entwurfsansicht)



	Feldname	Felddatentyp
🔑	VerkIdPk	Zahl
🔑	AbtIdPk	Zahl
	Name	Text
	Gehalt	Währung
	AbtLeister	Ja/Nein
	Vorname	Text

Beispiel 5 (für MS Access)

Vorher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe
1	1	Mustermann, Mike	60.000,00 €	0
2	1	Beispiel, Frank	80.000,00 €	0
3	1	Musterfrau, Ulrike	70.000,00 €	0

SQL (für MS Access)

```
ALTER TABLE Verkaeufer  
ADD COLUMN Vorname VARCHAR(25);
```

Nachher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann, Mike	60.000,00 €	0	
2	1	Beispiel, Frank	80.000,00 €	0	
3	1	Musterfrau, Ulrike	70.000,00 €	0	

Exkurs in SQL DDL (MS Access)

Vorher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann, Mike	60.000,00 €	0	
2	1	Beispiel, Frank	80.000,00 €	0	
3	1	Musterfrau, Ulrike	70.000,00 €	0	

SQL (mit speziellen MS Access-Funktionen)

```
UPDATE Verkaeufner SET
Verkaeufner.Vorname =
  Right(Verkaeufner.Name, Len(Verkaeufner.Name) -
  InStr(Verkaeufner.Name, ", ") - 1),
Verkaeufner.Name =
  Left(Verkaeufner.Name,
  InStrRev(Verkaeufner.[Name], ", ") - 1);
```

Nachher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann	60.000,00 €	0	Mike
2	1	Beispiel	80.000,00 €	0	Frank
3	1	Musterfrau	70.000,00 €	0	Ulrike

Beispiel 5 (für MS Access)

Vorher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe
1	1	Mustermann, Mike	60.000,00 €	0
2	1	Beispiel, Frank	80.000,00 €	0
3	1	Musterfrau, Ulrike	70.000,00 €	0

SQL (für MS Access)

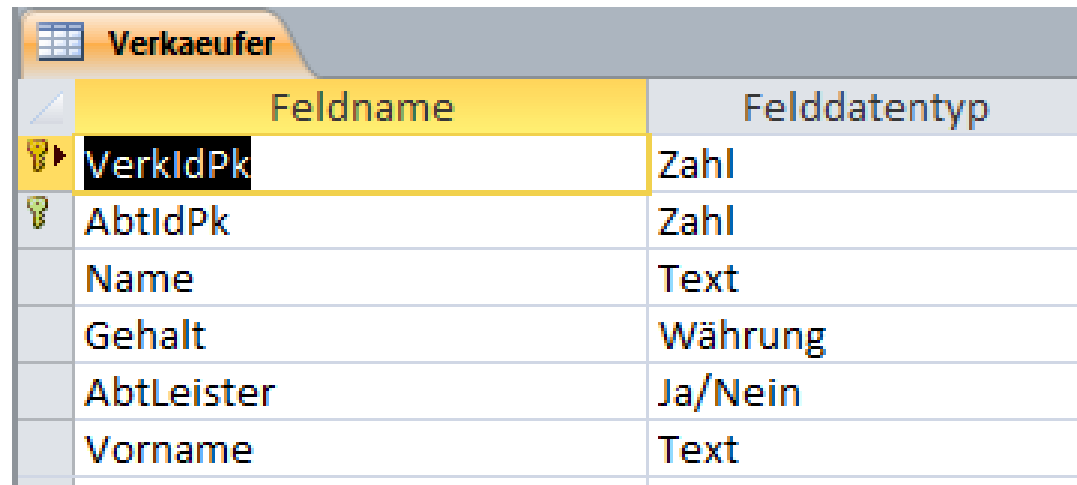
```
ALTER TABLE Verkaeufer  
ADD COLUMN Vorname VARCHAR(25);
```

Nachher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann, Mike	60.000,00 €	0	
2	1	Beispiel, Frank	80.000,00 €	0	
3	1	Musterfrau, Ulrike	70.000,00 €	0	

Beispiel 6 (für MS Access)

Vorher
(Entwurfsansicht)



	Feldname	Felddatentyp
🔑	VerkIdPk	Zahl
🔑	AbtIdPk	Zahl
	Name	Text
	Gehalt	Währung
	AbtLeister	Ja/Nein
	Vorname	Text

SQL (für MS Access)

```
ALTER TABLE Verkaeufer  
ADD COLUMN StraÙeNr VARCHAR(25),  
Plz VARCHAR(5), Ort VARCHAR(25);
```

Nachher
(Entwurfsansicht)



	Feldname	Felddatentyp
🔑	VerkIdPk	Zahl
🔑	AbtIdPk	Zahl
	Name	Text
	Gehalt	Währung
	AbtLeiter	Ja/Nein
	Vorname	Text
	StraÙeNr	Text
	Plz	Text
	Ort	Text

Beispiel 6 (für MS Access)

Vorher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann, Mike	60.000,00 €	0	
2	1	Beispiel, Frank	80.000,00 €	0	
3	1	Musterfrau, Ulrike	70.000,00 €	0	

SQL (für MS Access)

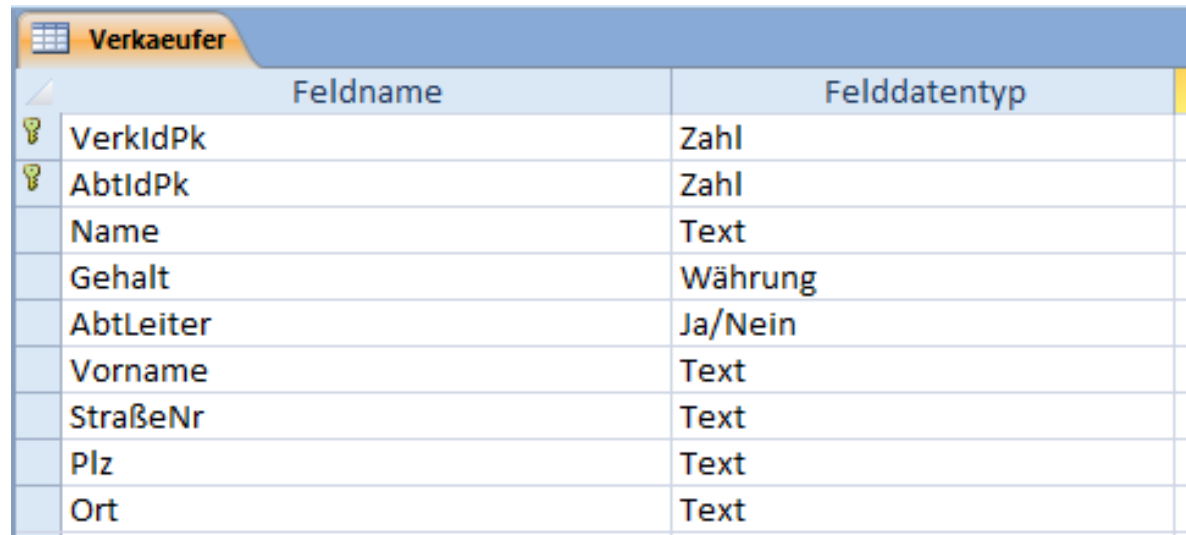
```
ALTER TABLE Verkaeufer  
ADD COLUMN StraÙeNr VARCHAR(25),  
Plz VARCHAR(5), Ort VARCHAR(25);
```

Nachher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname	StraÙeNr	Plz	Ort
1	1	Mustermann	60.000,00 €	0	Mike			
2	1	Beispiel	80.000,00 €	0	Frank			
3	1	Musterfrau	70.000,00 €	0	Ulrike			

Beispiel 7 (für MS Access)

Vorher
(Entwurfsansicht)

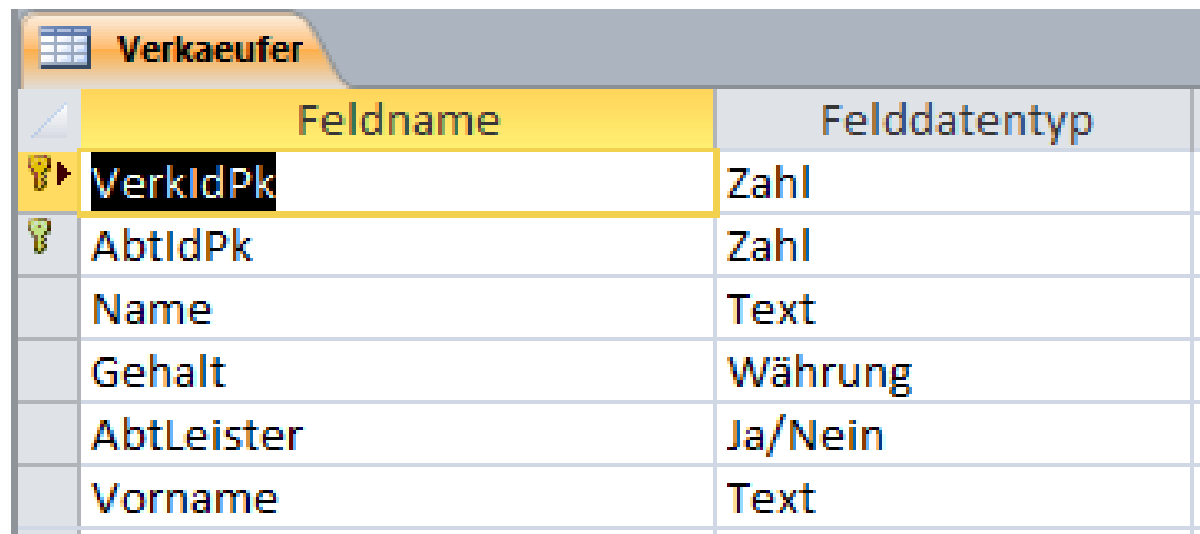


	Feldname	Felddatentyp
🔑	VerkIdPk	Zahl
🔑	AbtIdPk	Zahl
	Name	Text
	Gehalt	Währung
	AbtLeiter	Ja/Nein
	Vorname	Text
	StraßeNr	Text
	Plz	Text
	Ort	Text

SQL (für MS Access)

```
ALTER TABLE Verkaeuer  
DROP COLUMN Ort, Plz, StraßeNr;
```

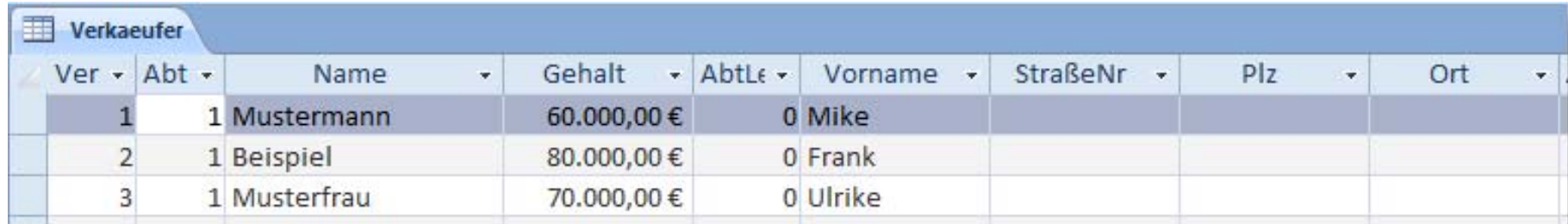
Nachher
(Entwurfsansicht)



	Feldname	Felddatentyp
🔑	VerkIdPk	Zahl
🔑	AbtIdPk	Zahl
	Name	Text
	Gehalt	Währung
	AbtLeister	Ja/Nein
	Vorname	Text

Beispiel 7 (für MS Access)

Vorher (Datenblattansicht)



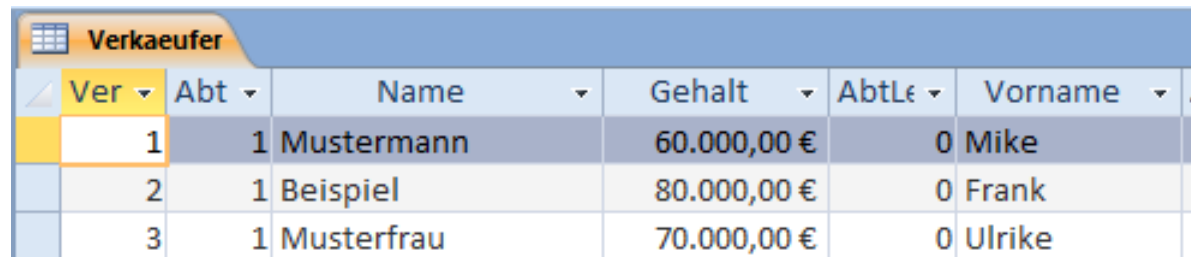
The screenshot shows the 'Verkaeufer' table in MS Access data sheet view. The table has 9 columns: Ver, Abt, Name, Gehalt, AbtLe, Vorname, StraÙeNr, Plz, and Ort. The data is as follows:

Ver	Abt	Name	Gehalt	AbtLe	Vorname	StraÙeNr	Plz	Ort
1	1	Mustermann	60.000,00 €	0	Mike			
2	1	Beispiel	80.000,00 €	0	Frank			
3	1	Musterfrau	70.000,00 €	0	Ulrike			

SQL (für MS Access)

```
ALTER TABLE Verkaeufer  
DROP COLUMN Ort, Plz, StraÙeNr;
```

Nachher (Datenblattansicht)

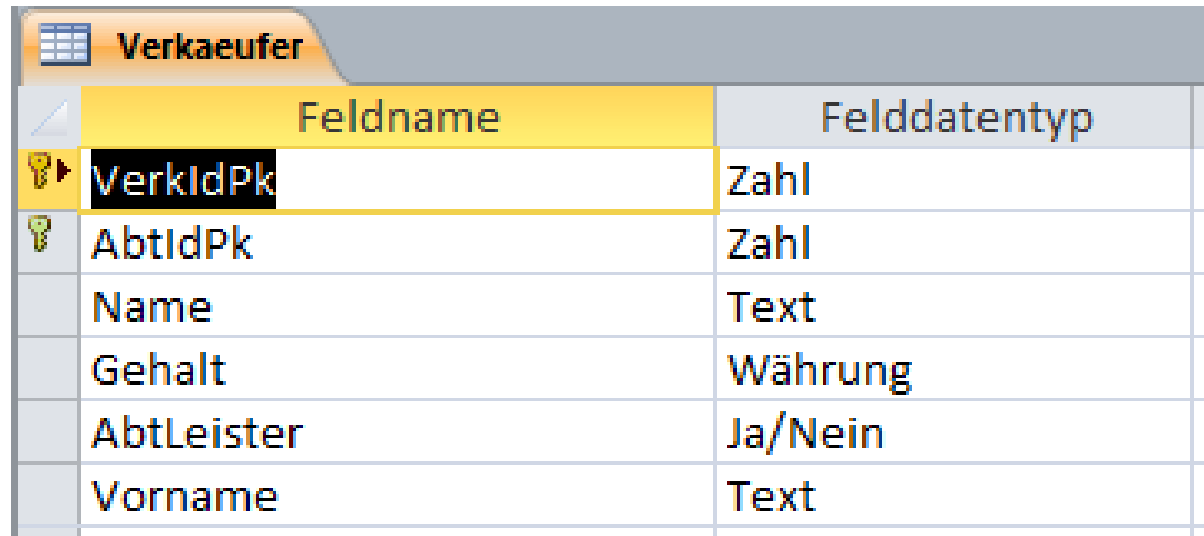


The screenshot shows the 'Verkaeufer' table in MS Access data sheet view after the SQL command. The columns 'Ort', 'Plz', and 'StraÙeNr' have been removed. The data is as follows:

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann	60.000,00 €	0	Mike
2	1	Beispiel	80.000,00 €	0	Frank
3	1	Musterfrau	70.000,00 €	0	Ulrike

Beispiel 8 (für MS Access)

Vorher
(Entwurfsansicht)

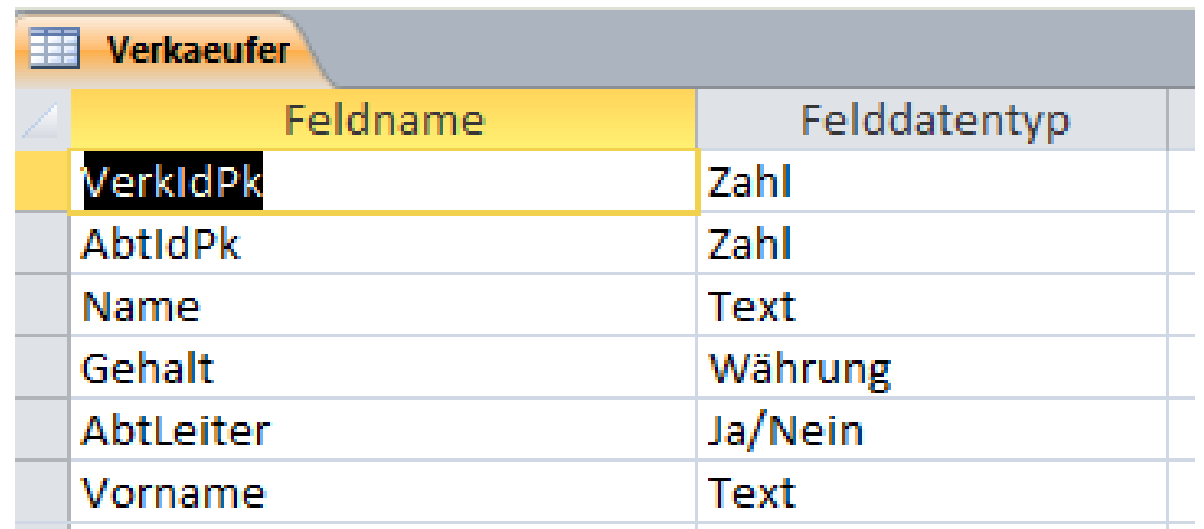


Feldname	Felddatentyp
VerkIdPk	Zahl
AbtIdPk	Zahl
Name	Text
Gehalt	Währung
AbtLeister	Ja/Nein
Vorname	Text

SQL (für MS Access)

```
ALTER TABLE Verkaeuf  
DROP CONSTRAINT PrimSchluesselVerk;
```

Nachher
(Entwurfsansicht)



Feldname	Felddatentyp
VerkIdPk	Zahl
AbtIdPk	Zahl
Name	Text
Gehalt	Währung
AbtLeiter	Ja/Nein
Vorname	Text

Beispiel 8 (für MS Access)

Vorher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann, Mike	60.000,00 €	0	
2	1	Beispiel, Frank	80.000,00 €	0	
3	1	Musterfrau, Ulrike	70.000,00 €	0	

SQL (für MS Access)

```
ALTER TABLE Verkaeufer  
DROP CONSTRAINT PrimSchluesselVerk;
```

Nachher
(Datenblattansicht)

Ver	Abt	Name	Gehalt	AbtLe	Vorname
1	1	Mustermann, Mike	60.000,00 €	0	
2	1	Beispiel, Frank	80.000,00 €	0	
3	1	Musterfrau, Ulrike	70.000,00 €	0	

Vergleich

- Wenn die Spalten einer Tabelle bereits existiert und nur als Primär- oder Fremdschlüssel festgelegt werden soll:

```
ALTER TABLE tblKunden
  ADD CONSTRAINT CkndIdPk PRIMARY KEY(kndIdPk)

ALTER TABLE tblAuftraege ADD CONSTRAINT CaufkndIdFk
  FOREIGN KEY(aufkndIdFk) REFERENCES tblKunden(kndIdPk)
```

- Wenn die Spalte noch nicht existiert und gleich als Primär- oder Fremdschlüssel angelegt werden soll:

```
ALTER TABLE tblKunden ADD COLUMN kndIdPk INTEGER
  CONSTRAINT CkndIdPk PRIMARY KEY(kndIdPk)

ALTER TABLE tblAuftraege ADD COLUMN aufkndIdFk INTEGER
  CONSTRAINT CaufkndIdFk FOREIGN KEY(aufkndIdFk) REFERENCES
  tblKunden(kndIdPk)
```

Relationen ändern

Bei Änderungen auf Relationen müssen die Daten der Relation beachtet werden, z.B.

- Hinzufügen von Spalten, die nicht leer sein dürfen
 - Zunächst Hinzufügen der Spalte (ohne die Bedingung)
 - Anpassen der Daten in der Tabelle, so dass die Spalte für alle Datensätze gefüllt ist
 - Hinzufügen der Bedingung, dass die Spalte nicht leer sein darf
- Entfernen eines Teilschlüssels eines zusammengesetzten Primärschlüssels
 - Daten müssen auch mit dem neuen Primärschlüssel eindeutig sein
- beliebige weitere Konstellationen denkbar...

D11.2

- Nehmen Sie in der MS Access Datenbank per Datendefinitionsabfrage eine Änderung der Tabelle Personen vor, indem Sie die Spalte Vorname einfügen
- Welche Wirkung hat dies auf die bereits erfassten Daten?
- Was wäre zu tun, damit jede Personen in der Tabelle einen Vornamen bekommt? Nennen Sie ein Beispiel für eine beliebige Personen in der Tabelle!

SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - **Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen**
 - Löschen von Tabellen

SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - **Löschen von Tabellen**

Relationen löschen

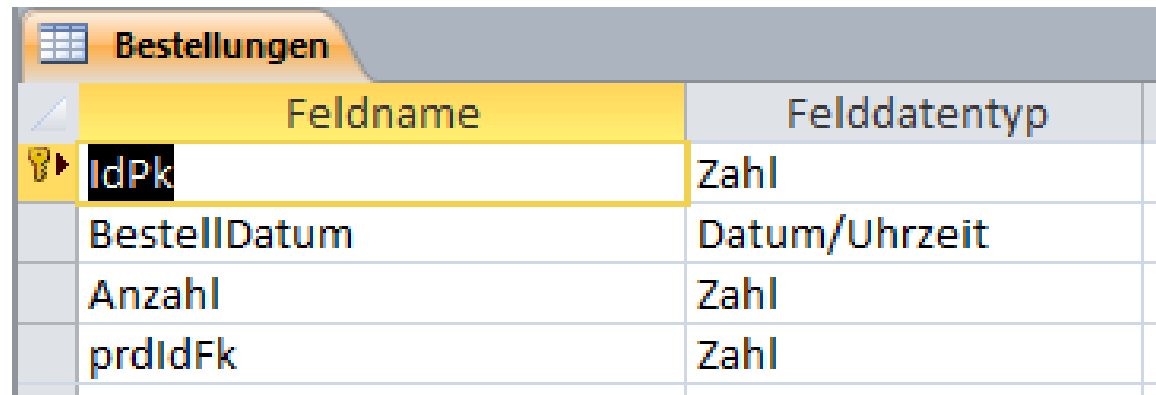
Überblick über die Syntax zum Löschen Relationen

```
DROP TABLE <Tabellename>;
```

Hinweis: Löschen von Relationen ist erst dann möglich, wenn Beziehungen zuvor aufgelöst wurden (2. Integritätsbedingung/Referenzielle Integrität)

Beispiel 7 (für MS Access)

Vorher



	Feldname	Felddatentyp
🔑	IdPk	Zahl
	BestellDatum	Datum/Uhrzeit
	Anzahl	Zahl
	prdIdFk	Zahl

SQL (für MS Access)

```
DROP TABLE Bestellungen;
```

Nachher

Demo 11.3: SQL DDL – Ändern von Tabellen

D11.3

- Löschen Sie in der MS Access Datenbank per Datendefinitionsabfrage die Tabelle Personen

SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - **Löschen von Tabellen**

SQL-Standard

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- ermöglicht unabhängig vom verwendeten Werkzeug
 - Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
 - Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
 - Löschen von Tabellen

Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Zusammenfassung

SQL Data Definition Language

- umfasst Befehle zum Anlegen und Ändern des Datenbankschemas
- CREATE TABLE: Anlegen von Relationen mit Spalten, deren Datentypen, Schlüsseln und ggf. Integritätsbedingungen
- ALTER TABLE: Ändern, Hinzufügen oder Löschen von Spalten oder Integritätsbedingungen
- DROP TABLE: Löschen von Tabellen

Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Hintergrund

Gefährdung von Datenbankanwendungen und Datenbanken

- durch vorsätzlich Handelnde
 - außerhalb der Organisation
 - innerhalb der Organisation
- durch Sicherheitsmängel
 - im Entwurf/in der Programmierung der Anwendung
 - im Datenbanksystem
 - im Betriebssystem
 - in Serverkomponenten
 - in Clientkomponenten (z.B. Browser)
 - ...



Hintergrund

Neben Schutzmechanismen auf verschiedenen Ebenen ...

- Betriebssystem
- Server-/Clientkomponenten
- Netzwerkkommunikation
- Anwendungen
- usw.

... ist speziell der Schutz der Datenbanken von zentraler Bedeutung.



Hintergrund

Zugriff auf Daten

- muss stets restriktiv gehandhabt werden
- jeder Nutzer einer Datenbankanwendung
 - darf stets nur die Daten verfügbar haben, die er zur Erledigung seiner Aufgaben benötigt
 - dies muss auch im Fehlerfall einzelner Komponenten gewährleistet sein



Negativ-Beispiele – Ist das nötig?

- ein Administratorzugang von außerhalb der Organisation bietet vollen Zugriff auf die Datenbankanwendung
- ein angemeldeter Kunden hat auch schreibenden Zugriff auf die Preisinformation von Produkten
- in einer im Betrieb befindlichen Datenbankanwendung können Benutzer SQL DDL-Befehle ausführen (z.B. ALTER TABLE, DROP TABLE)
- ...



Lösungsansatz

Für die Datenbank

- Benutzer und/oder
- Gruppen

einrichten, explizit

- ermöglichen oder
- entziehen

der Aktionsmöglichkeiten auf

- Inhalten und
- Strukturen

der Datenbank.



Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

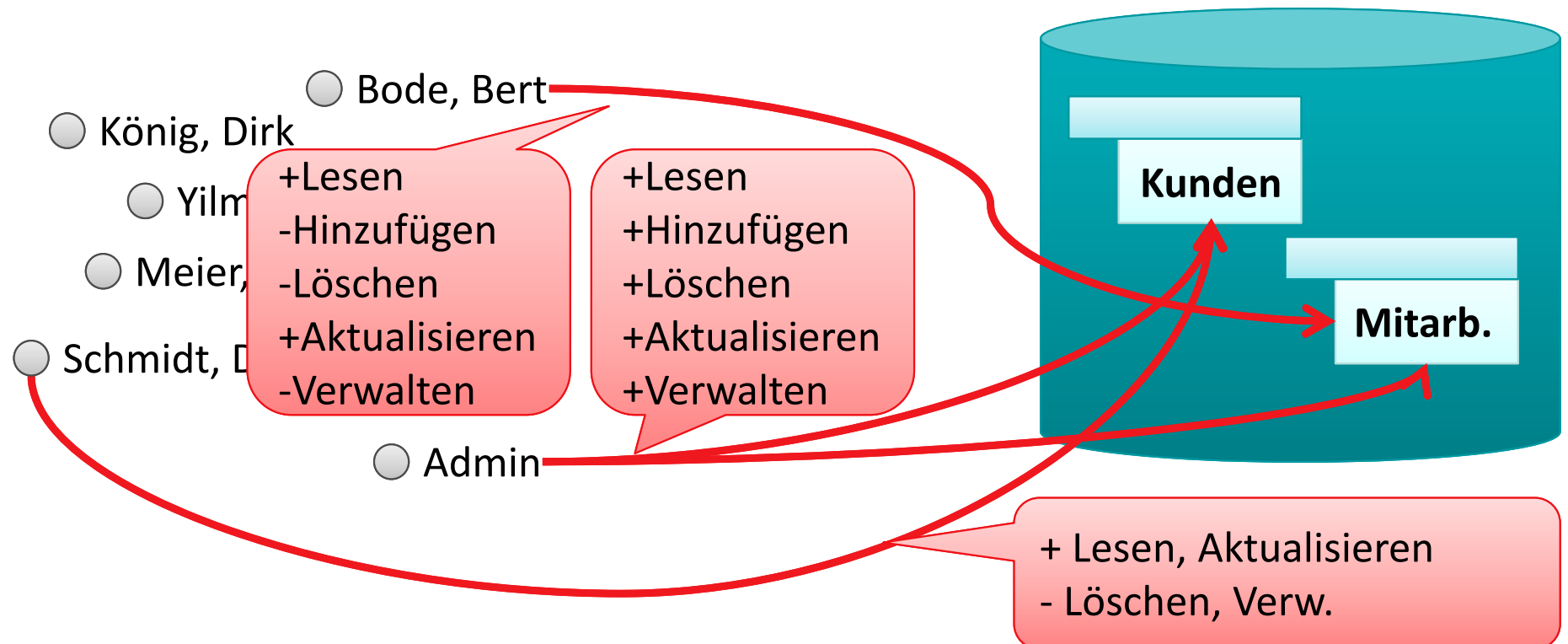
- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Benutzer und Gruppen

Benutzer

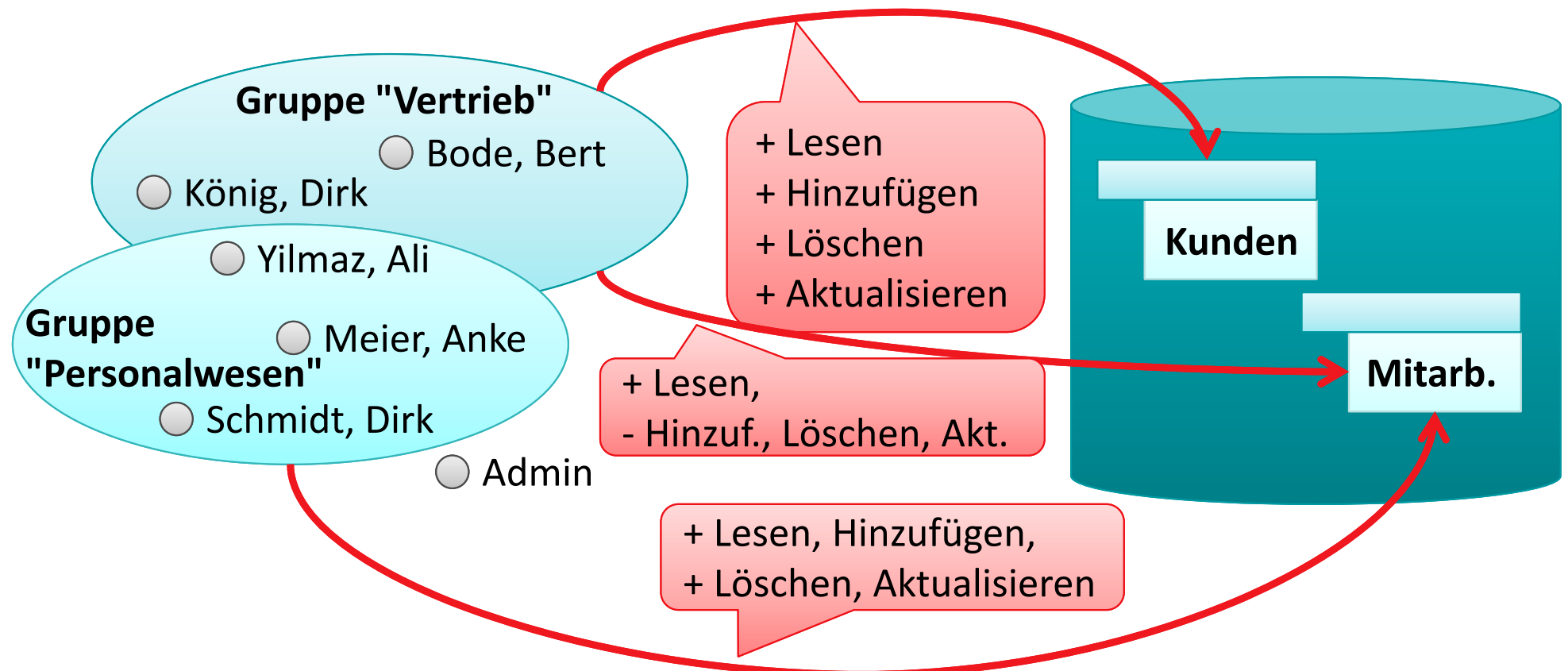
- Identifizierbar und authentifizierbar
- für Zugang zur Datenbank
- bestimmte Aktionsmöglichkeiten eingerichtet oder entzogen
- kann einer oder mehreren Gruppen zugeordnet werden



Benutzer und Gruppen

Gruppen

- umfasst einen oder mehrere Benutzer
- für Zugang zur Datenbank
- bestimmte Aktionsmöglichkeiten eingerichtet oder entzogen



Benutzer und Gruppen in SQL

Anlegen von Benutzern/Gruppen

```
CREATE USER <Benutzername> <Passwort>  
CREATE GROUP <Gruppenname>
```

Zuordnen von Benutzern zu Gruppen

```
ADD USER <Benutzername> TO <Gruppenname>  
ADD USER <Benutzername1>, <Benutzername2>, ... TO <Gruppe>
```

Entfernen von Benutzer aus Gruppen

```
DROP USER <Benutzername> FROM <Gruppenname>
```

Entfernen von Benutzern/Gruppen

```
DROP USER <Benutzername>  
DROP GROUP <Gruppenname>
```


Anlegen von Dirk König und Andrea Meier als Benutzer

```
CREATE USER dkoenig geheim123  
CREATE USER ameier geheim234
```

Anlegen der Gruppen Vertrieb und Personalwesen

```
CREATE GROUP vertrieb  
CREATE GROUP pesonalwesen
```

Zuordnen von Dirk König und Andrea Meier zu Gruppen

```
ADD USER dkoenig TO vertrieb  
ADD USER ameier TO pesonalwesen
```

Benutzer und Gruppen in MS Access

MS Access unterstützt SQL DCL-Befehle, aber ...

- nicht mittels üblicher Abfragen
- nicht mit der "normalen" Datenbankverbindung
- ohne sichtbare Wirkung in jüngeren Versionen als 2003

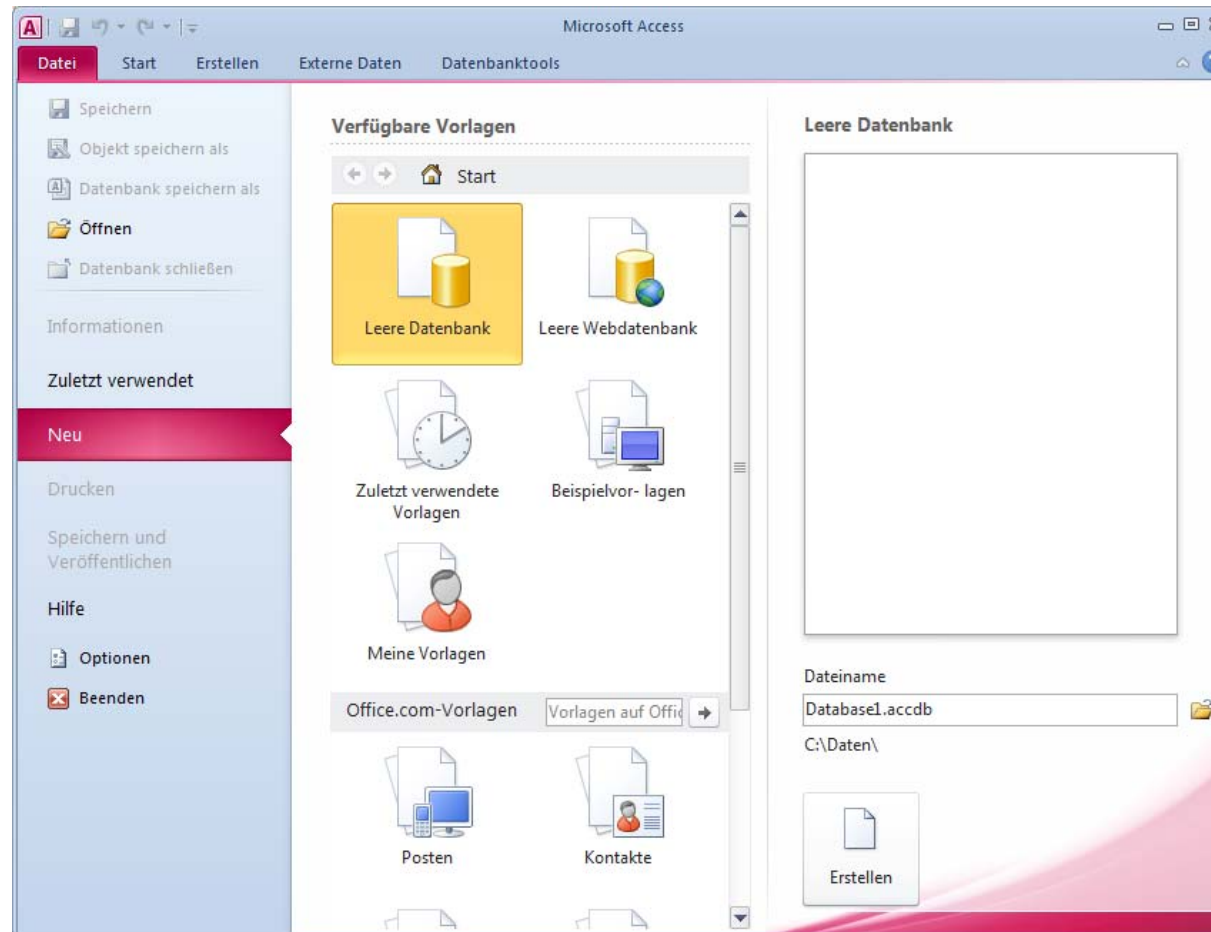
nur per Workaround durch Verwendung

- einer "alten" Datenbank im Format 2003 (*.mdb)
- und ActiveX Data Objects (ADO) über Jet SQL for MS Access
- als User-Level Security (Sicherheit auf Benutzerebene)



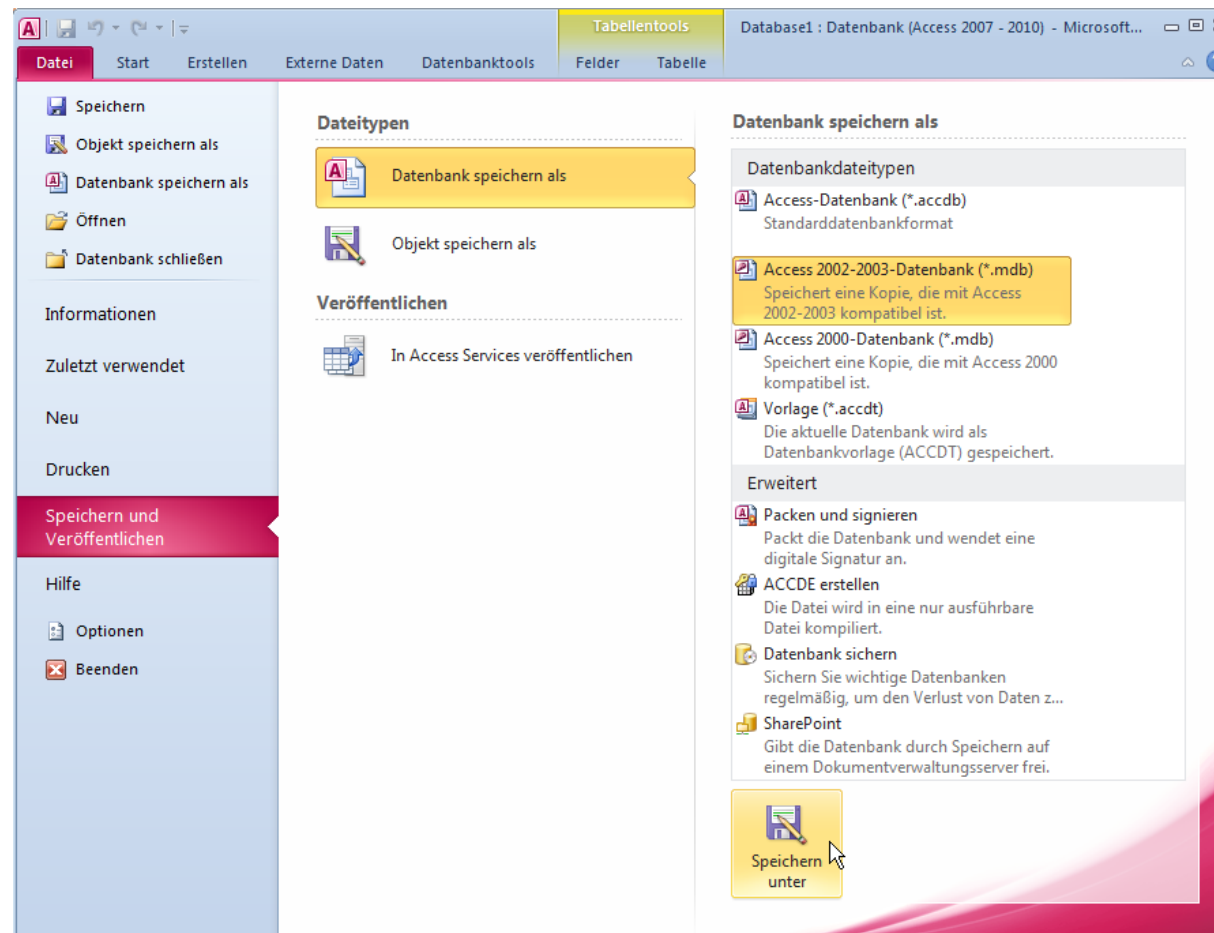
Benutzer und Gruppen in MS Access 2003

Schritt 1: Neue Datenbank erstellen



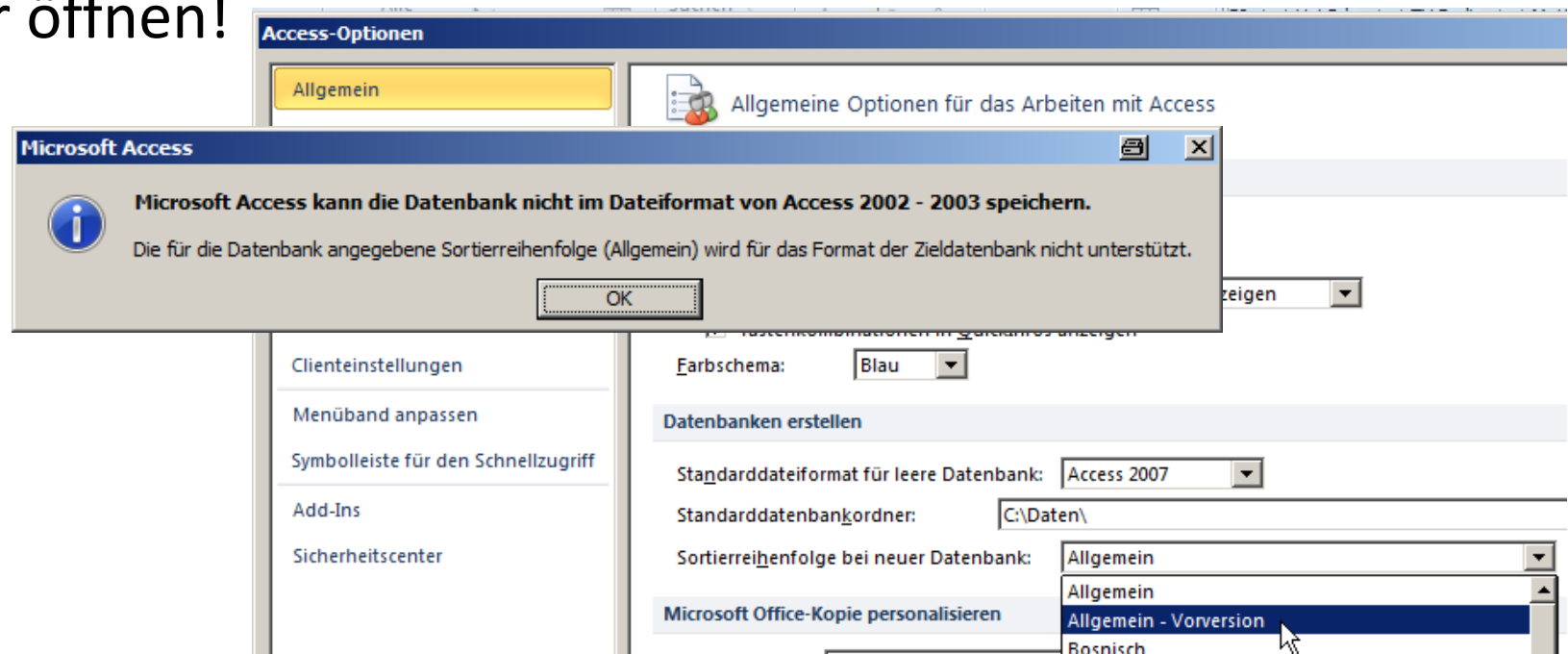
Benutzer und Gruppen in MS Access 2003

Schritt 2: Datenbank im Format "Access 2002-2003" (*.mdb) speichern



Hinweis Bei Fehlermeldung wg. Sortierreihenfolge

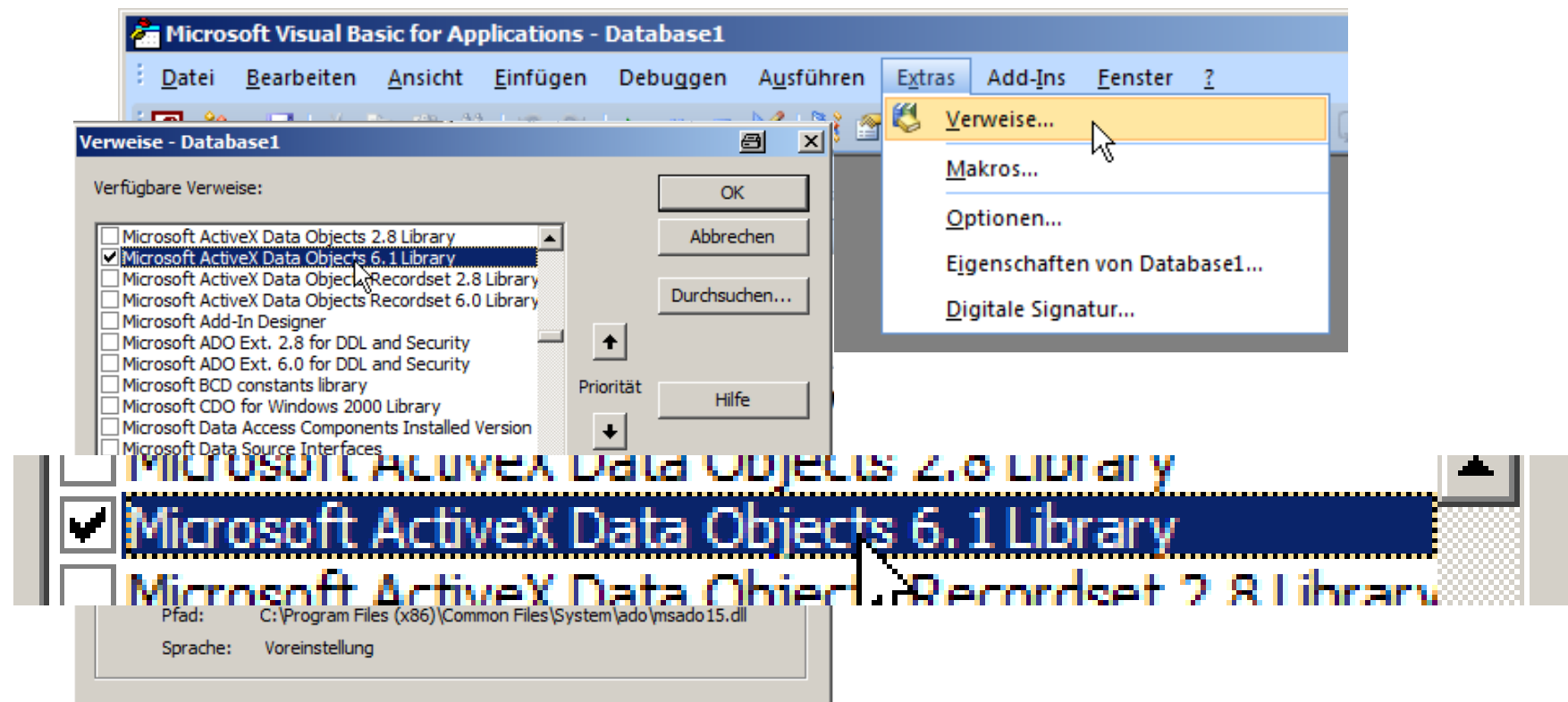
- Menüband "Datei" ► Eintrag "Optionen" ► "Allgemein" ► Gruppe "Datenbanken erstellen" ► Feld "Sortierreihenfolge bei neuer Datenbank" anstelle von "Allgemein" die Auswahl "Allgemein - Vorgängerversion"
- Access neu starten und neu angelegte Datenbank (*.mdb) wieder öffnen!



Benutzer und Gruppen in MS Access 2003

Schritt 3: In Visual Basic die ADO-Bibliothek aktivieren

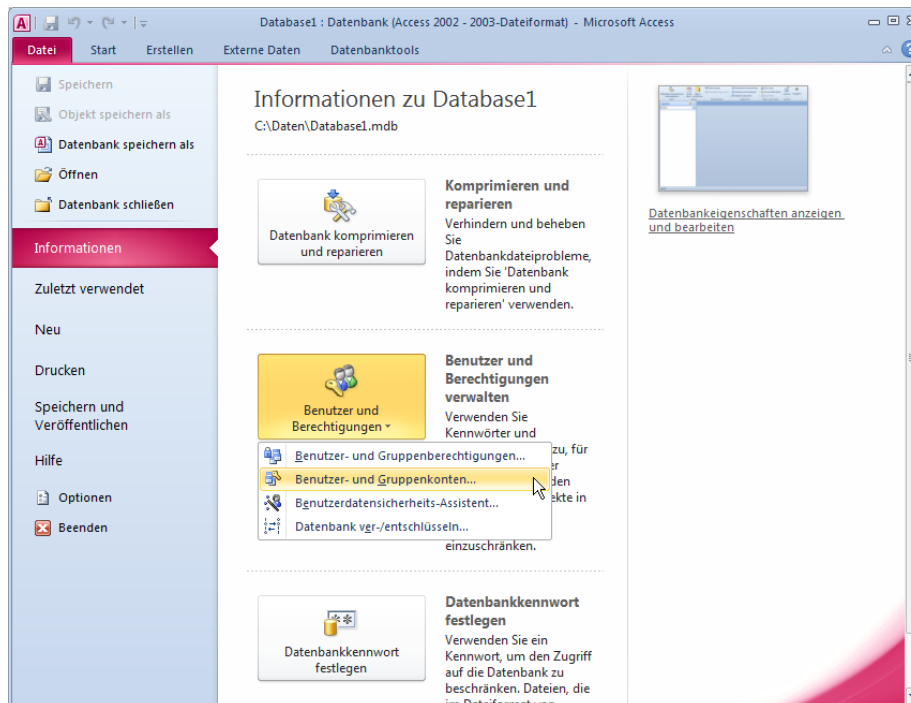
- VBA-Editor öffnen, im Menü "Extras" ► "Verweise..."
- im Dialog "Verweise" ► Liste "Verfügbare Verweise" den Eintrag "Microsoft ActiveX Data Objects 6.1 Library" selektieren (Haken setzen)



Benutzer und Gruppen in MS Access 2003

Schritt 4: Prüfen, ob "Benutzer und Berechtigungen" verfügbar

- in Access 2010/2013 auf in der Backstage "Datei"
- in Access 2007 im Menüband "Datenbanktools" ► Gruppe "Verwalten"



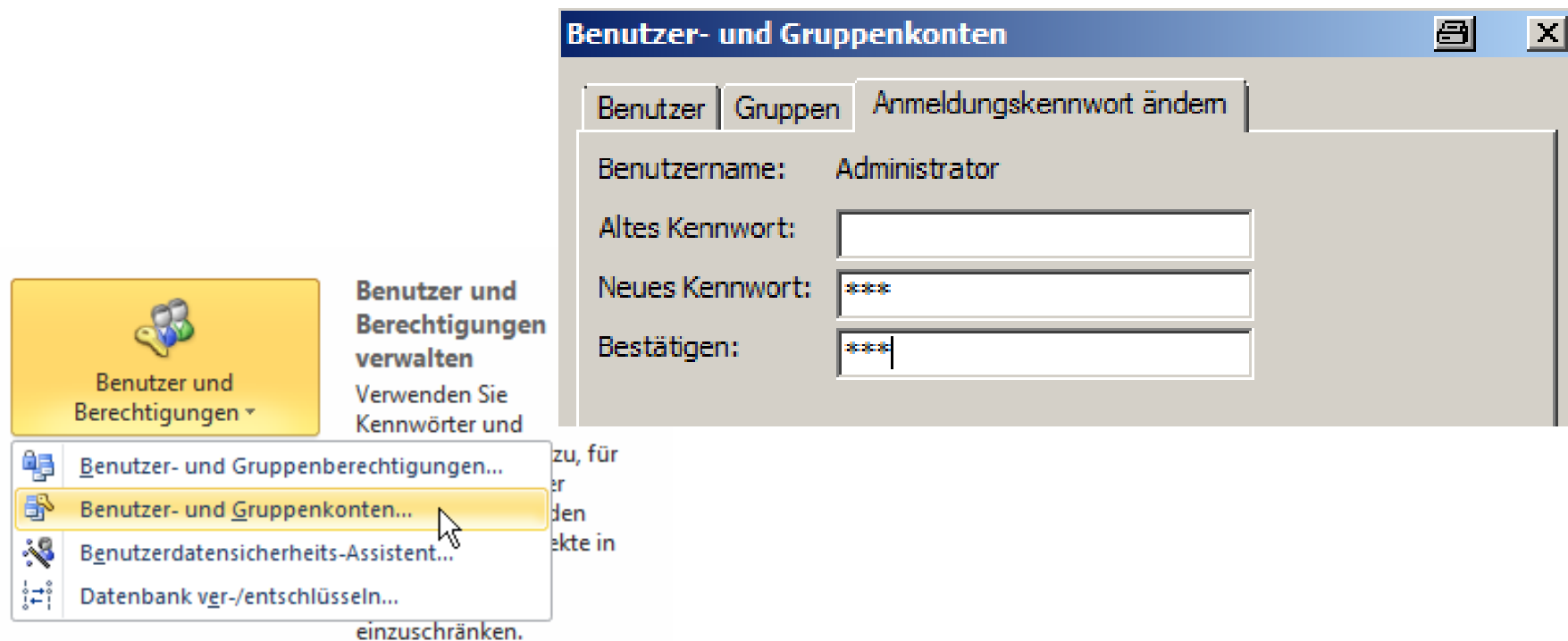
' Wenn nicht angezeigt, dann:
DoCmd.RunCommand acCmdUserAndGroupAccounts
DoCmd.RunCommand acCmdUserAndGroupPermissions

Quelle: <http://www.access-im-unternehmen.de/620.0.html>

Benutzer und Gruppen in MS Access 2003

Schritt 5: Festlegen eines Passwortes für den Administrator

- Altes Kennwort leer
- Neues Kennwort



Hinweise

- Passwort wird beim Öffnen **jeder** Datenbank abgefragt
- Zu Übungszwecken einfaches Passwort (z.B. "123") verwenden
- Löschen Sie das Passwort nach Abschluss der Übungen wieder, um den Ursprungszustand wieder herzustellen (Dialog "Benutzer und Gruppenkonten..." ► Registerkarte "Benutzer" ► Schaltfläche "Kennwort löschen").

Benutzer und Gruppen in MS Access 2003

Schritt 6: Neue Gruppen und Benutzer über SQL DCL per ADO-Datenbankverbindung anlegen

The image shows two overlapping VBA code windows from MS Access 2003. The top window is titled '(Allgemein)' and contains a VBA subroutine named 'anlegenGruppen()'. The code uses an ADO connection to execute SQL DCL statements to create two groups: 'Vertrieb' and 'Personalwesen'. The bottom window is also titled '(Allgemein)' and contains a VBA subroutine named 'anlegenBenutzer()'. This code uses an ADO connection to execute SQL DCL statements to create two users: 'dkoenig' with password 'geheim123' and 'ameier' with password 'geheim234'. Both code windows have a toolbar on the right with a yellow play button icon highlighted by a mouse cursor, and a tooltip that reads 'Sub/UserForm ausführen (F5)'.

```
(Allgemein)

Sub anlegenGruppen()

' ADODB-Verbindung für den Zugriff nutzen
Dim adoCon As ADODB.Connection
Set adoCon = CurrentProject.Connection

' Gruppen anlegen mit SQL DCL
adoCon.Execute ("CREATE GROUP Vertrieb")
adoCon.Execute ("CREATE GROUP Personalwesen")

End Sub
```

```
(Allgemein)

Sub anlegenBenutzer()

' ADODB-Verbindung für den Zugriff nutzen
Dim adoCon As ADODB.Connection
Set adoCon = CurrentProject.Connection

' Benutzer anlegen mit SQL DCL
adoCon.Execute ("CREATE USER dkoenig geheim123")
adoCon.Execute ("CREATE USER ameier geheim234")

End Sub
```

Schritt 7: Prüfen, ob neue Benutzer und Gruppen verfügbar

- Dialog "Benutzer und Gruppenkonten" öffnen
- Einträge auf Registerkarte "Benutzer" und ggf. "Gruppen" kontrollieren
- Benutzer "ameier" und "dkoenig" sind noch keine Mitglieder von Gruppen



Benutzer und Gruppen in MS Access 2003

Schritt 8: Gruppenzugehörigkeit ändern und prüfen

The image shows a screenshot of the MS Access 2003 interface. On the left, a VBA code window is open, displaying the following code:

```
(Allgemein)  
Sub hinzufuegenBenutzerZuGruppen()  
    ' ADODB-Verbindung für den Zugriff nutzen  
    Dim adoCon As ADODB.Connection  
    Set adoCon = CurrentProject.Connection  
  
    ' Benutzer den Gruppen zuordnen mit SQL DCL  
    adoCon.Execute ("ADD USER dkoenig TO Vertrieb")  
    adoCon.Execute ("ADD USER ameier TO Personalwesen")  
  
End Sub
```

On the right, a toolbar is visible with a yellow play button icon highlighted. A tooltip below it reads "Sub/UserForm ausführen (F5)".

In the foreground, the "Benutzer- und Gruppenkonten" dialog box is open. It has three tabs: "Benutzer", "Gruppen", and "Anmeldeskennwort ändern". The "Benutzer" tab is selected. The "Name:" field contains "dkoenig". Below it, a list shows "Administrator", "ameier", and "dkoenig", with "dkoenig" selected. The "Gruppenzugehörigkeit" section has two lists: "Verfügbare Gruppen:" containing "Administratoren", "Personalwesen", "Benutzer", and "Vertrieb"; and "Mitglied von:" containing "Vertrieb". Between these lists are buttons for "Hinzufügen >>" and "<< Entfernen". At the bottom of the dialog is a button labeled "Benutzer und Gruppen drucken". The dialog has "OK", "Abbrechen", and "Übernehmen" buttons at the very bottom.

Benutzer und Gruppen in SQL

Anlegen von Benutzern/Gruppen

```
CREATE USER <Benutzername> <Passwort>  
CREATE GROUP <Gruppenname>
```

Zuordnen von Benutzern zu Gruppen

```
ADD USER <Benutzername> TO <Gruppenname>  
ADD USER <Benutzername1>, <Benutzername2>, ... TO <Gruppe>
```

Entfernen von Benutzer aus Gruppen

```
DROP USER <Benutzername> FROM <Gruppenname>
```

Entfernen von Benutzern/Gruppen

```
DROP USER <Benutzername>  
DROP GROUP <Gruppenname>
```

Aktionsmöglichkeiten

**Für Gruppen und/oder Benutzer können
Aktionsmöglichkeiten eingerichtet oder entzogen werden**

– beispielsweise (MS Access)

Aktion	Beschreibung
SELECT	Lesender Zugriff auf die Datensätze einer Tabelle
INSERT	Hinzufügen neuer Datensätze zu einer Tabelle
DELETE	Löschen vorhandener Datensätze aus einer Tabelle
UPDATE	Ändern vorhandener Datensätze in einer Tabelle
DROP	Löschen von Tabellen (und deren Daten)
...	...

– andere Datenbanksysteme unterstützen andere/weitere
Aktionsmöglichkeiten

Festlegen von Aktionsmöglichkeiten auf Tabellen

```
GRANT <Aktionsmöglichkeit>  
ON TABLE <Tabellename>  
TO <BenutzerOderGruppe>
```

Entziehen von Aktionsmöglichkeiten auf Tabellen

```
REVOKE <Aktionsmöglichkeit>  
ON TABLE <Tabellename>  
FROM <BenutzerOderGruppe>
```

Festlegen von Aktionsmöglichkeiten auf Tabellen

- Vertrieb darf Kunden lesen und bearbeiten
- D. König darf (zusätzlich) Mitarbeiter lesen
- Personalwesen darf Mitarbeiter hinzufügen, bearbeiten, aber nicht löschen

```
GRANT  
SELECT, INSERT, UPDATE, DELETE  
ON TABLE tblKunden  
TO Vertrieb
```

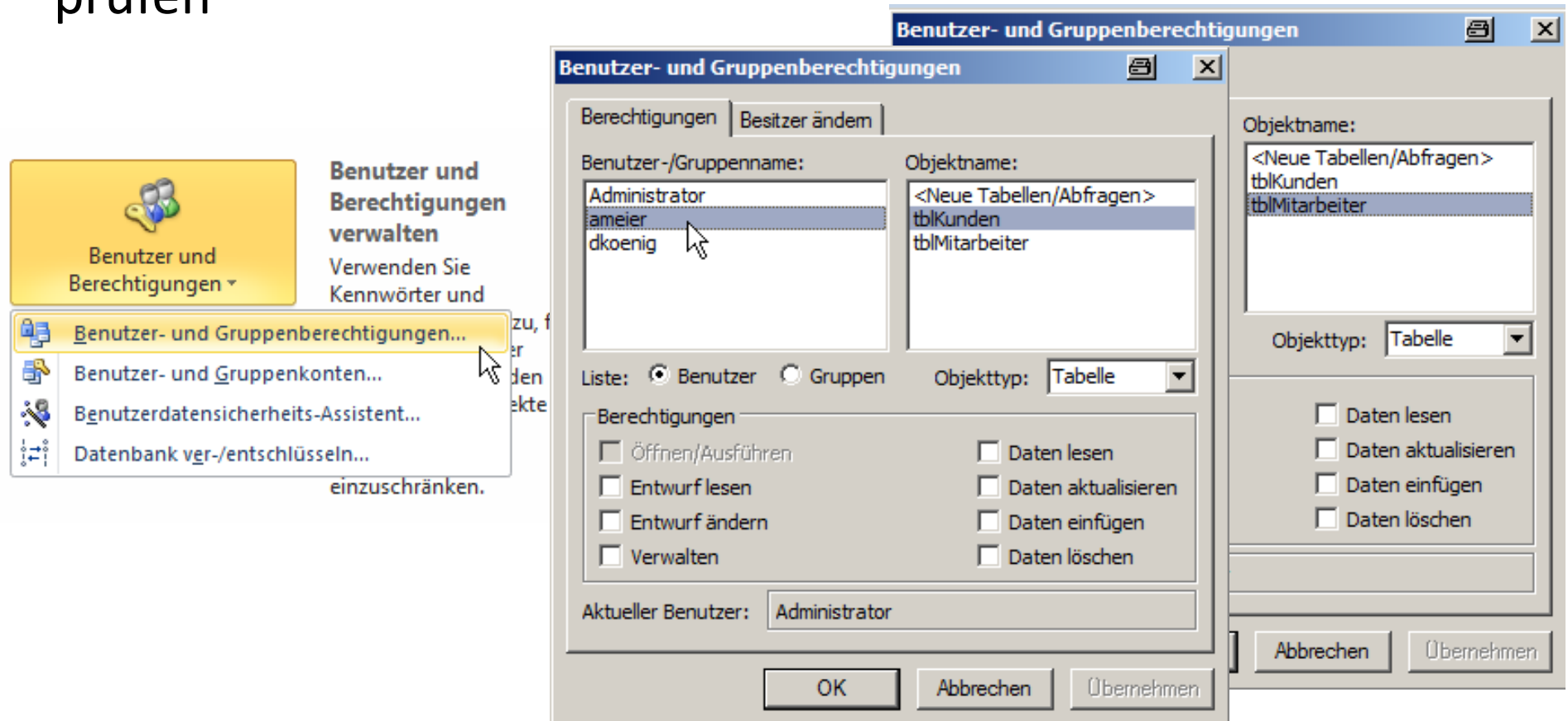
```
GRANT SELECT  
ON TABLE tblMitarbeiter  
TO dkoenig
```

```
GRANT SELECT, INSERT, UPDATE  
ON TABLE tblMitarbeiter  
TO Personalwesen
```


Beispiel: Aktionsmöglichkeiten in Access 2003

Schritt 1: Vorhandene Aktionsmöglichkeiten einsehen

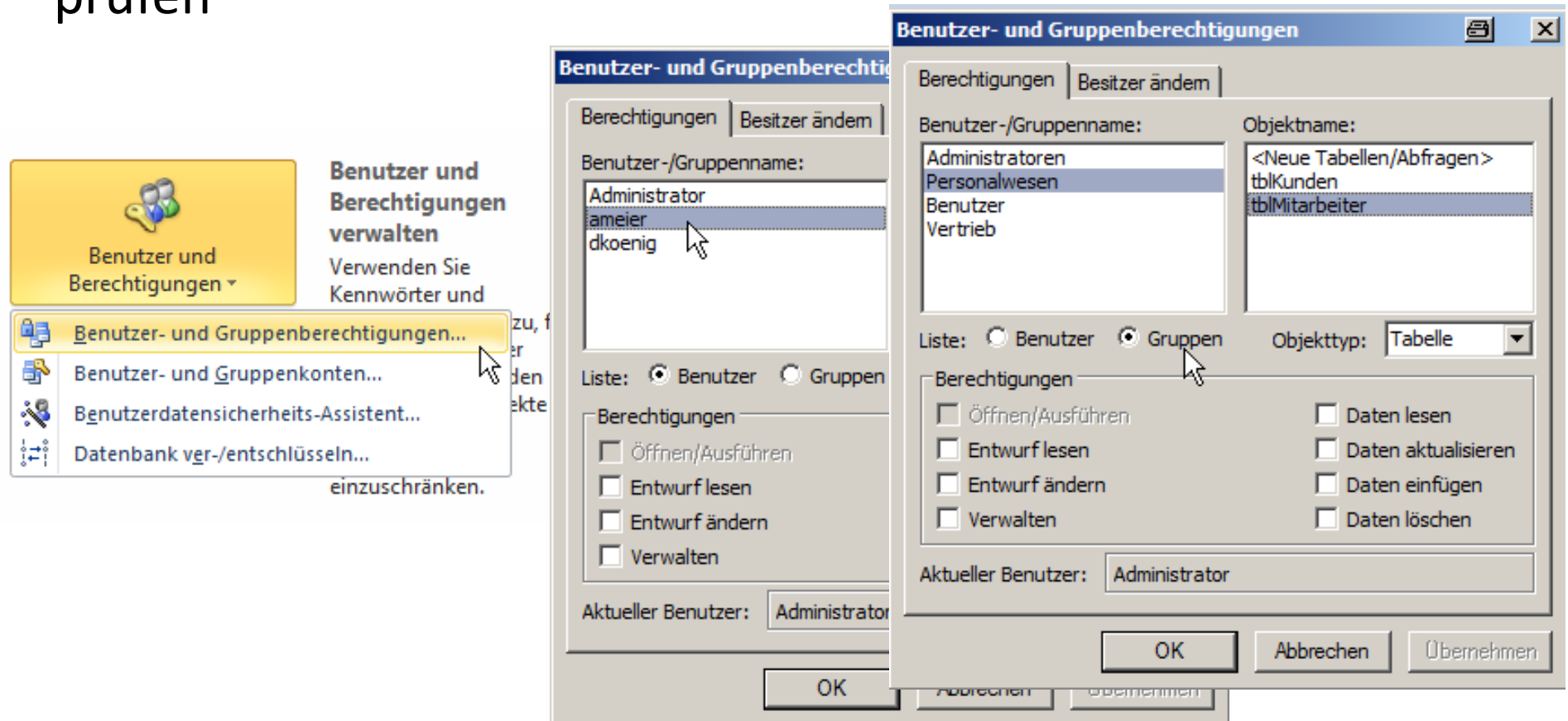
- Dialog "Benutzer- und Gruppenberechtigungen" öffnen
- Berechtigungen für neu angelegte Benutzer und Gruppen prüfen



Beispiel: Aktionsmöglichkeiten in Access 2003

Schritt 1: Vorhandene Aktionsmöglichkeiten einsehen

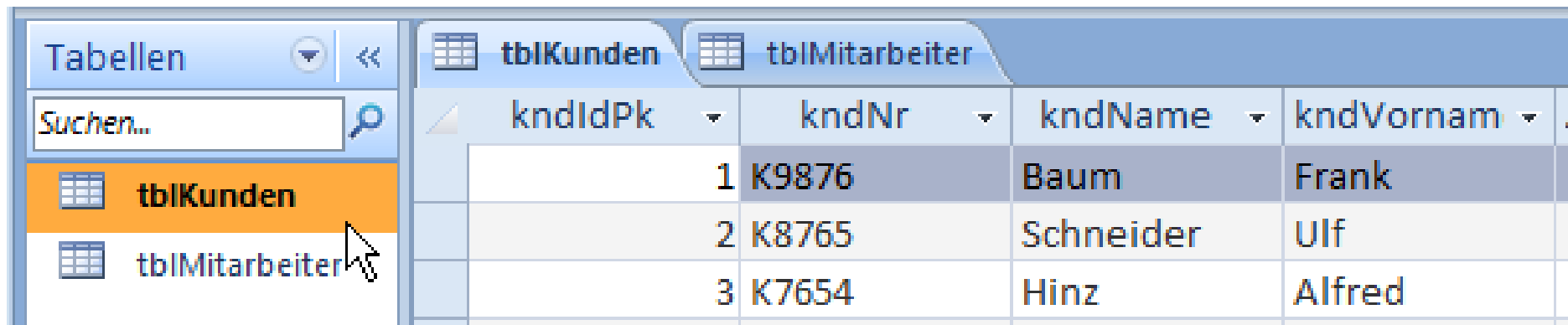
- Dialog "Benutzer- und Gruppenberechtigungen" öffnen
- Berechtigungen für neu angelegte Benutzer und Gruppen prüfen



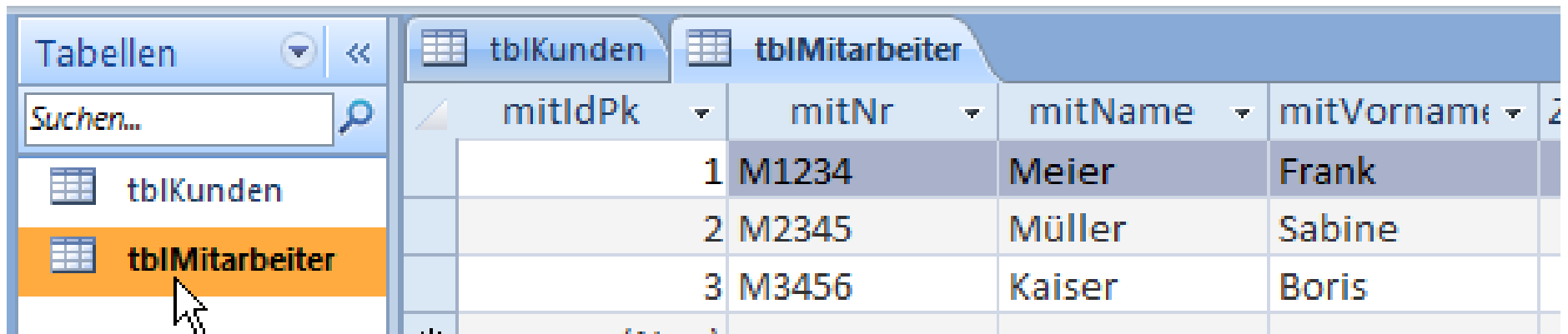
Beispiel: Aktionsmöglichkeiten in Access 2003

Schritt 2: Tabellen anlegen und füllen

- Tabellen "tblKunden" und "tblMitarbeiter" anlegen
- beliebige Beispieldaten erfassen



knIdPk	knNr	knName	knVorname
1	K9876	Baum	Frank
2	K8765	Schneider	Ulf
3	K7654	Hinz	Alfred



mitIdPk	mitNr	mitName	mitVorname
1	M1234	Meier	Frank
2	M2345	Müller	Sabine
3	M3456	Kaiser	Boris

Beispiel: Aktionsmöglichkeiten in Access 2003

Schritt 3: Aktionsmöglichkeiten einrichten, z.B.

- Mitglieder der Gruppe "Vertrieb" dürfen Lesen, Hinzufügen, Ändern und Löschen auf Tabelle "tblKunden"
- Dirk König darf auch die Tabelle "tblMitarbeiter" lesen
- Mitglieder der Gruppe "Vertrieb" dürfen Lesen, Hinzufügen, Ändern auf Tabelle "tblMitarbeiter" (aber nicht Löschen!)
- Zugriff auf Navigationsbereich einrichten

```
(Allgemein) einrichtenAktionsmoeglichkeiten
Sub einrichtenAktionsmoeglichkeiten()
' ADODB-Verbindung für den Zugriff nutzen
Dim adoCon As ADODB.Connection
Set adoCon = CurrentProject.Connection

' Zugriffsrechte einrichten auf Tabellen mit SQL DCL
adoCon.Execute ("GRANT SELECT, INSERT, UPDATE, DELETE ON TABLE tblKunden TO Vertrieb")
adoCon.Execute ("GRANT SELECT ON TABLE tblMitarbeiter TO dkoenig")
adoCon.Execute ("GRANT SELECT, INSERT, UPDATE ON TABLE tblMitarbeiter TO Personalwesen")

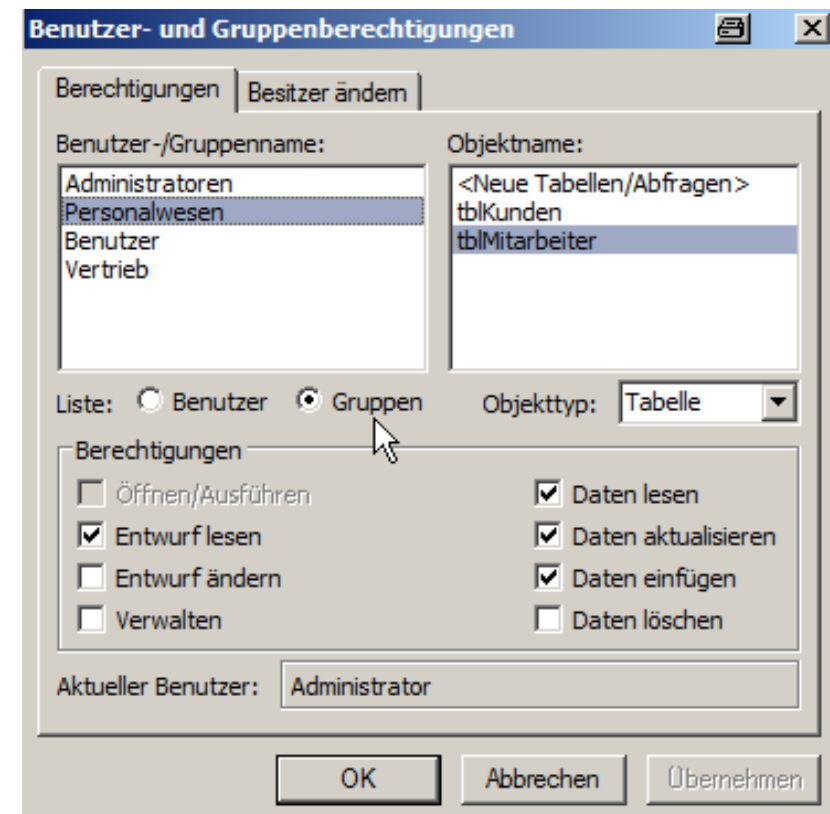
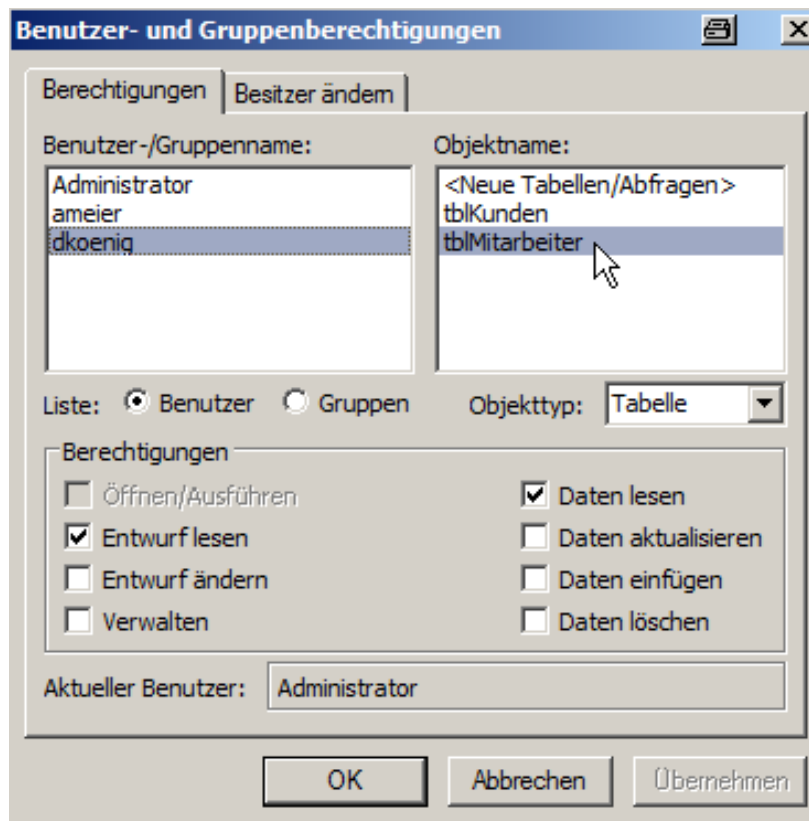
'| Zugriffsrechte auf Navigationsbereich einrichten
adoCon.Execute ("GRANT ALL PRIVILEGES ON TABLE MSysAccessStorage TO Vertrieb, Personalwesen")
adoCon.Execute ("GRANT ALL PRIVILEGES ON TABLE MSysObjects TO Vertrieb, Personalwesen")

End Sub
```

Beispiel: Aktionsmöglichkeiten in Access 2003

Schritt 4: Aktionsmöglichkeiten prüfen

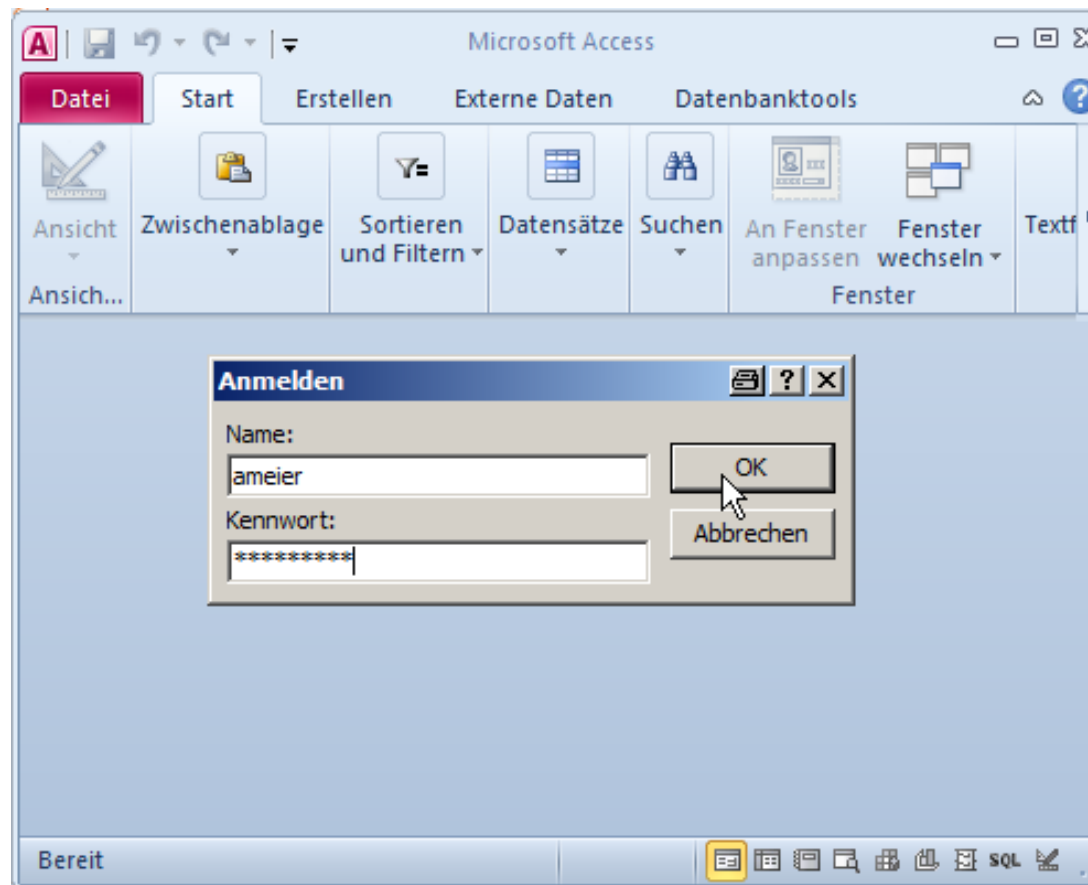
- im Dialog "Benutzer- und Gruppenberechtigungen" die neu gesetzten Aktionsmöglichkeiten prüfen



Beispiel: Aktionsmöglichkeiten in Access 2003

Schritt 5: Access neu starten und Anmelden

- mit Benutzer "ameier"
- mit Passwort: "geheim234"



Beispiel: Aktionsmöglichkeiten in Access 2003

Schritt 6: Aktionsmöglichkeiten teste

– Personalwesen

- Keine Sichtbarkeit von Kundendaten
- Anzeigen, Hinzufügen und Bearbeiten von Mitarbeitern
- Keine Löschmöglichkeit von Mitarbeitern

The screenshot shows the Microsoft Access 2003 interface. In the top-left pane, the 'Tabellen' (Tables) list contains 'tblKunden' and 'tblMitarbeiter'. A context menu is open over 'tblKunden' with options 'Öffnen' (Open) and 'Entwurfsansicht' (Design View). A dialog box is displayed over this menu with the following text:

Microsoft Access

Definitionen konnten nicht gelesen werden. Es ist keine Leseberechtigung für die Tabelle oder Abfrage 'tblKunden' vorhanden.

OK Hilfe

[War diese Information hilfreich?](#)

In the bottom-left pane, the 'tblMitarbeiter' table is shown in a data view. The table has columns: mitIdPk, mitNr, mitName, mitVorname. The data rows are:

mitIdPk	mitNr	mitName	mitVorname
1	M1234	Meier	Frank
2	M2345	Müller	Sabine
		Kaiser	Boris
		Baum	Doris

A context menu is open over the table with options: 'Neuer Datensatz' (New Record), 'Datensatz löschen' (Delete Record), and 'Ausdrucken' (Print). A dialog box is displayed over this menu with the following text:

Microsoft Access

Sie haben nicht die nötigen Berechtigungen, um das Objekt 'tblMitarbeiter' zu bearbeiten. Bitte Sie den Systemadministrator oder die Person, die das Objekt erstellt hat, Ihnen die entsprechenden Berechtigungen zuzuweisen.

OK Hilfe

[War diese Information hilfreich?](#)

Aktionsmöglichkeiten

Festlegen von Aktionsmöglichkeiten auf Tabellen

```
GRANT <Aktionsmöglichkeit>  
ON TABLE <Tabellename>  
TO <BenutzerOderGruppe>
```

Entziehen von Aktionsmöglichkeiten auf Tabellen

```
REVOKE <Aktionsmöglichkeit>  
ON TABLE <Tabellename>  
FROM <BenutzerOderGruppe>
```


Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Zusammenfassung

Schutz durch Nutzung von SQL Data Control Language

- Einrichten von Gruppen und Benutzern

```
CREATE USER/GROUP <BenutzerGruppe>
```

- Hinzufügen von Benutzern zu Gruppen

```
ADD USER <Benutzer> TO <Gruppe>
```

- Einrichten von Zugriffsrechten

```
GRANT <Aktionmgl>  
ON TABLE <Name>  
TO <BenutzerOderGruppe>
```

- Entfernen der Gruppenmitgliedschaft
und eines Benutzers: **DROP**

- Entziehen von Zugriffsrechten mit
REVOKE (ähnlich **GRANT**)



Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

Inhalt

Ziel und Einordnung

Rückblicke

SQL DDL zur Manipulation des Datenmodells

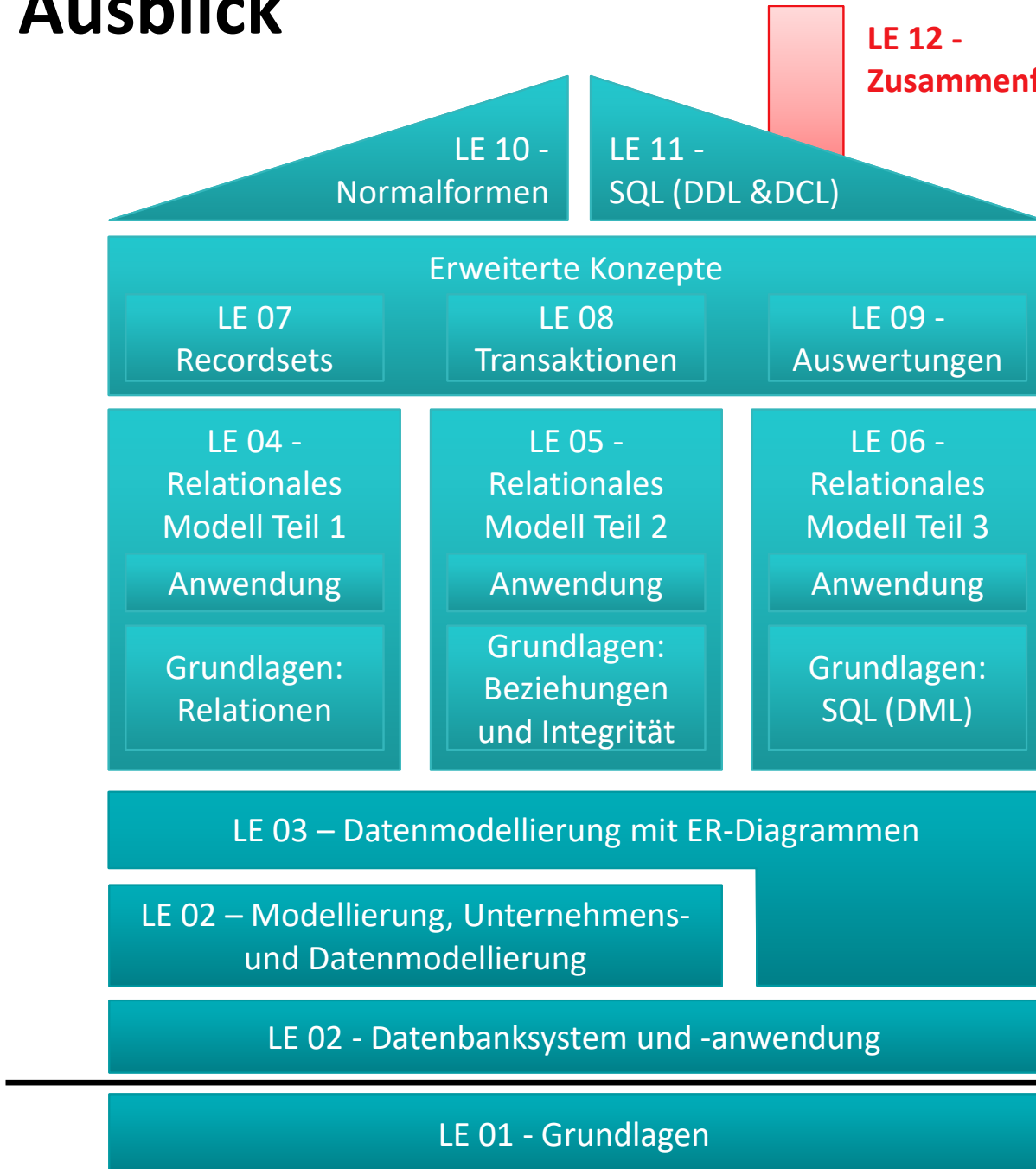
- Manipulation des Datenmodells
 - Erweiterungen und Weiterentwicklung
 - Installation einer Datenbank in einer anderen Umgebung
- Manipulationsmöglichkeiten mit SQL DDL
- SQL DDL in MS Access
- Zusammenfassung

SQL DCL zur Einrichtung von Zugriffsrechten

- Hintergrund
- Gruppen und Benutzer
- Festlegen von Zugriffsrechten
- SQL DCL in MS Access
- Zusammenfassung

Ausblick

LE 12 -
Zusammenfassung



Quellen

[1] "Massiver Datendiebstahl bei Schweizer Geheimdienst":

<http://www.silicon.de/41576696/massiver-datendiebstahl-beim-schweizer-geheimdienst/>

[2] "Patientendaten aus Kliniken verschwunden":

http://www.welt.de/print/die_welt/wirtschaft/article109781722/Wirtschaft-Kompakt-II.html

[3] "Datenklau im Gesundheitsministerium"

<http://www.heise.de/security/meldung/Datenklau-im-Gesundheitsministerium-1766905.html>

[4] "Hacker stehlen Zehntausende Kreditkartendaten"

<http://www.welt.de/wirtschaft/webwelt/article106141798/Hacker-stehlen-Zehntausende-Kreditkartendaten.html?config=print>

[5] "Banken ziehen mehr als 100.000 Kreditkarten aus dem Verkehr"

<http://www.spiegel.de/wirtschaft/service/datendiebstahl-banken-ziehen-mehr-als-100-000-kreditkarten-aus-dem-verkehr-a-661909-druck.html>

[6] "Datenklau im Gesundheitsministerium „Neue Qualität der Spionage“"

<http://www.taz.de/Datenklau-im-Gesundheitsministerium/!107295/>

[7] "Benutzer und Berechtigungen in Access 2007 verwalten" <http://www.access-im-unternehmen.de/620.0.html>

[8] "Tabellen von Access 2007-ACCDBs schützen" <http://www.access-im-unternehmen.de/610.0.html>

[9] "Einführung in die Sicherheit von Access 2010" <http://office.microsoft.com/de-de/access-help/einfuehrung-in-die-sicherheit-von-access-2010-HA010341741.aspx>

Wirtschaftsinformatik 2

LE 11 – SQL DDL und DCL

Prof. Dr. Thomas Off

<http://www.ThomasOff.de/lehre/beuth/wi2>