



BEUTH HOCHSCHULE FÜR TECHNIK BERLIN  
University of Applied Sciences

# Übungen zu Wirtschaftsinformatik 2

## LE 06 – Relationales Modell (Teil 3)

### SQL

Prof. Dr. Thomas Off

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



# Lernziel und Lehrinhalte

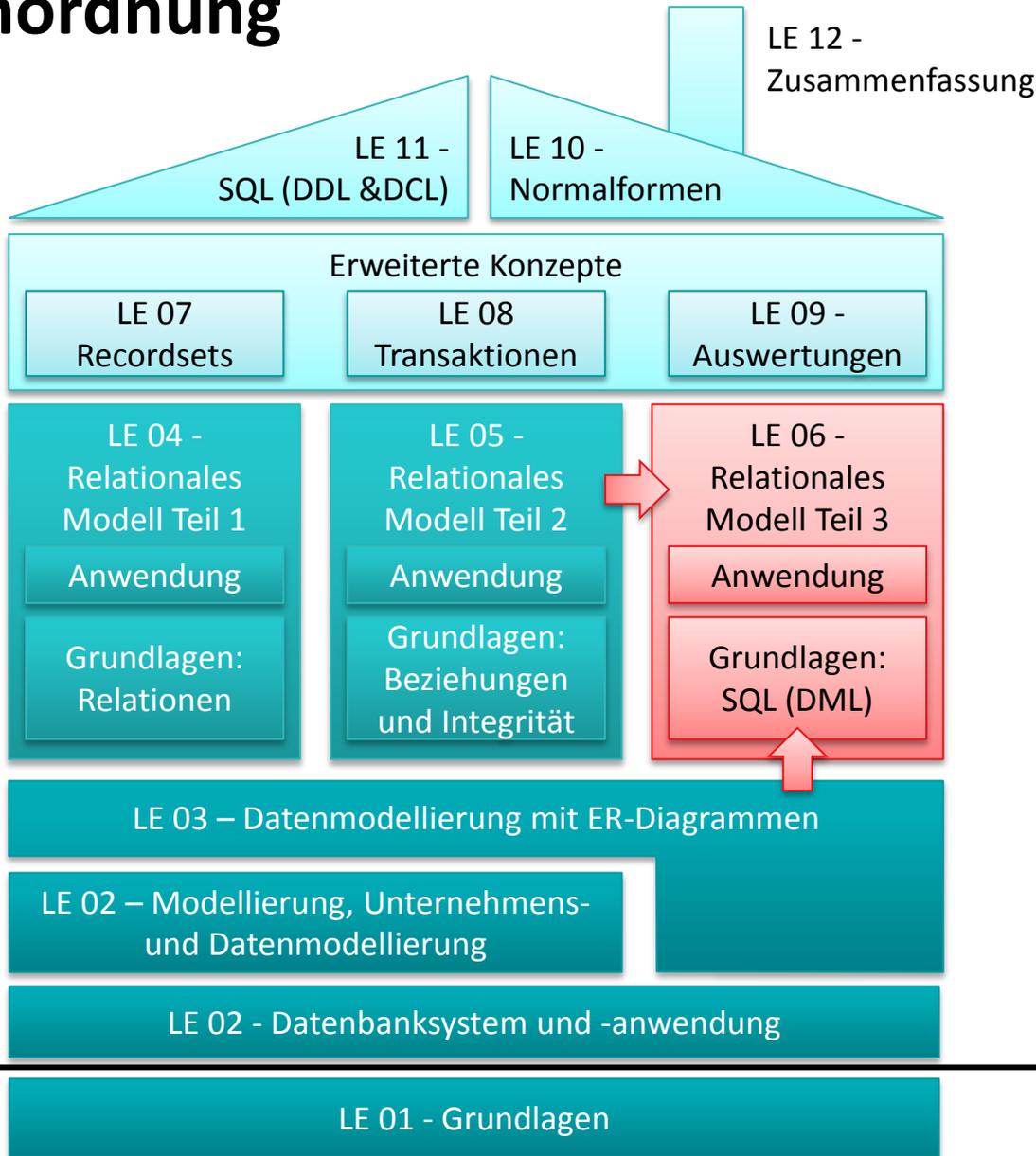
## Lernziel

- Grundlegende Erfahrung in der Anwendung von SQL-Anweisungen sammeln
- Eigene SQL-Anweisungen entwerfen und in eigenen Formularen und Prozeduren verwenden

## Lehrinhalte

- Anwendung der relationalen Algebra mit SQL
- Ausführung von SQL-Anweisungen zum Auswählen, Einfügen, Ändern und Löschen von Datensätzen (Data Manipulation Language, DML)
- Integration der Anweisungen in eigene Formulare und Prozeduren

# Einordnung





# Inhalt

## Lernziel, Lehrinhalte und Einordnung

## SQL-Anweisungen zur Auswahl und Manipulation

- Einfache Anweisungen
- Auswählen von Produkten einer Produktkategorie
- Einfügen von Produkten zum Warenkorb
- Ändern der Anzahl eines Produktes im Warenkorb
- Löschen eines Produktes aus dem Warenkorb
- Verbinden (Join) von Produkten und Produktkategorien

## Ausblick



# Inhalt

## Lernziel, Lehrinhalte und Einordnung

### **SQL-Anweisungen zur Auswahl und Manipulation**

- Einfache Anweisungen
- Auswählen von Produkten einer Produktkategorie
- Einfügen von Produkten zum Warenkorb
- Ändern der Anzahl eines Produktes im Warenkorb
- Löschen eines Produktes aus dem Warenkorb
- Verbinden (Join) von Produkten und Produktkategorien

### **Ausblick**

# Auswahl und Manipulation mit SQL



# Auswahl und Manipulation mit SQL: Übung 6.1



## Ü6.1 (Teil 1)

- Sofern noch nicht vorhanden, erfassen Sie in ihrer Kundentabelle die folgenden Testdaten für Kunden. Sollten dazu weitere Datensätze in der Tabelle `tblBenutzer` benötigt werden, legen Sie diese zuvor an.

	kndIdPk	kndName	kndVorname	kndGebDatum	kndbenIdFk
+	1	Albers	Willi	29.11.1998	user1
+	2	Boehrs	Thomas	21.03.2000	user2
+	3	Dinkel	Ulrike	28.04.1993	user3
+	9	Dinkels	Thomas	21.07.1994	user4
*	(Neu)				

- Ermitteln Sie alle Kunden, deren Vorname "Thomas" ist. Gehen Sie wie folgt vor
  - Erstellen Sie einen neuen Abfrageentwurf,
  - schließen Sie den Dialog zum Hinzufügen von Tabellen und
  - wechseln Sie gleich in die SQL-Ansicht.
  - Erfassen Sie eine SQL-Abfrage die alle Kunden ermittelt, deren Vorname "Thomas" ist.
  - Führen Sie die Abfrage aus

	kndName	kndVorname
	Boehrs	Thomas
	Dinkels	Thomas
*		

*Beispiel*

# Auswahl und Manipulation mit SQL: Übung 6.1



## Ü6.1 (Teil 2)

- ...
- Sie wollen mit einer SQL-Anweisung Boehrs, Vera (geb. 03.12.1997) in die Tabelle der Kunden einfügen

tblKunden					
	kndIdPk	kndName	kndVorname	kndGebDatum	kndbenBenutzernameFk
+	1	Albers	Willi	29.11.1998	user1
+	2	Boehrs	Thomas	21.03.2000	user2
+	4	Dinkels	Thomas	01.05.2001	user4
→+	9	Boehrs	Vera	03.12.1997	user10
*		(Neu)			

*Beispiel*

- Warum müssen Sie zuerst einen neuen Benutzer für Vera anlegen?
- Erstellen Sie in der SQL-Ansicht eines neuen Abfrageentwurfs/oder einer "Einfügeabfrage" eine SQL-Anweisung die einen weiteren Benutzer "user10" mit Passwort "test10" in die Tabelle Benutzer einfügt
- Erstellen Sie in der SQL-Ansicht eines neuen Abfrageentwurfs/oder einer "Einfügeabfrage" eine SQL-Anweisung die Boehrs, Vera mit einer Beziehung zu user10 in die Tabelle Kunden einfügt

– ...

# Auswahl und Manipulation mit SQL: Übung 6.1



## Ü6.1 (Teil 3)

- ...
- Erstellen Sie in der SQL-Ansicht eines neuen Abfrageentwurfs/oder in der SQL-Ansicht einer "Aktualisierungsabfrage" eine SQL-Anweisung die bei allen Kunden mit dem Namen "Boehrs" den Namen in "Böhrs" ändert.

– ...

**Beispiel: Vorher**

kndIdPk	kndName	kndVorname	kndGebDatum	kndbenBenutzernameFk
1	Albers	Willi	29.11.1998	user1
2	Boehrs	Thomas	21.03.2000	user2
4	Dinkels	Thomas		
9	Boehrs	Vera		
*	(Neu)			

**Beispiel: Nachher**

kndIdPk	kndName	kndVorname	kndGebDatum	kndbenBenutzernameFk
1	Albers	Willi	29.11.1998	user1
2	Böhrs	Thomas	21.03.2000	user2
4	Dinkels	Thomas	01.05.2001	user4
9	Böhrs	Vera	03.12.1997	user10
*	(Neu)			

# Auswahl und Manipulation mit SQL: Übung 6.1



## Ü6.1 (Teil 5)

– ...

– Sie wollen mit einer SQL-Anweisung den Benutzer "user10" löschen.

- Warum müssen Sie zuerst Vera Böhrs aus der Tabelle Kunden löschen?
- Erstellen Sie in der SQL-Ansicht eines neuen Abfrageentwurfs/oder einer "Löschabfrage" eine SQL-Anweisung, die Vera Böhrs aus der Tabelle Kunden löscht.

tblBenutzer						
	benBenutzernamePk	benPasswor	benIstAdmi	Zum Hinz		
+	admin1	test1	<input checked="" type="checkbox"/>			
+	admin2	test2	<input checked="" type="checkbox"/>			
+	user1	test1	<input type="checkbox"/>			
+	user10	test10	<input type="checkbox"/>			
	+	kndIdPk	kndName	kndVorname	kndGebDatum	Zu
		9	Boehrs	Vera	03.12.1997	
	*	(Neu)				
+	user2	test2	<input type="checkbox"/>			
+	user3	test3	<input type="checkbox"/>			
+	user4	test4	<input type="checkbox"/>			

- Erstellen Sie in der SQL-Ansicht eines neuen Abfrageentwurfs/oder einer "Löschabfrage" eine SQL-Anweisung, die user10 aus der Tabelle löscht.

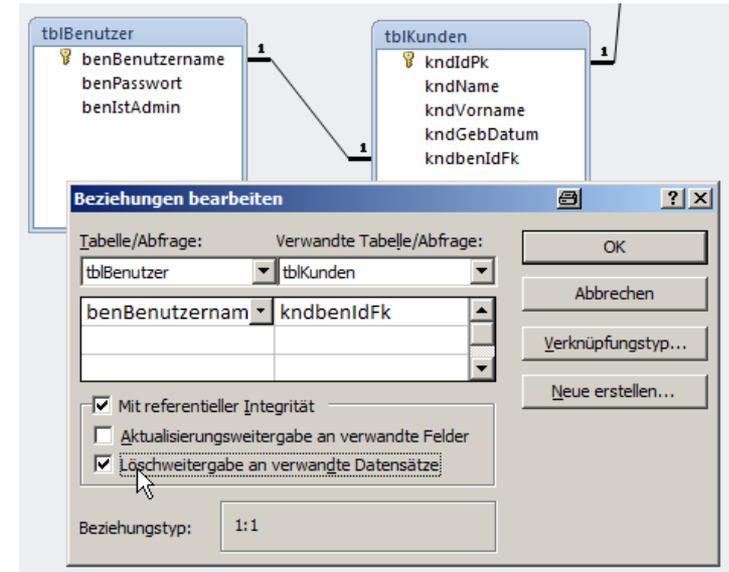
– ...

# Auswahl und Manipulation mit SQL: Übung 6.1



## Ü6.1 (Teil 4)

- ...
- Aktivieren Sie an der Beziehung von Kunde zu Benutzer und von Kunde zum Warenkorb die Löschweitergabe (Beziehungsansicht > Dialog "Beziehung bearbeiten" > Löschweitergabe an andere Felder aktivieren)
- Erstellen Sie in der SQL-Ansicht eines neuen Abfrageentwurfs/ oder einer "Löschabfrage" eine SQL-Anweisung, die einen Kunden aus der Tabelle Kunden löscht, zu dem es einen Warenkorbbeitrag und einen Benutzernamen gibt. Was stellen Sie fest?



**Wichtig!** Löschen Sie nicht den Kunden, den Sie in Übung Ü5.7 bei der Abfrage des Warenkorbinhalt verwenden.

Feld:	kndIdPk
Tabelle:	tblKunden
Sortierung:	
Anzeigen:	<input checked="" type="checkbox"/>
Kriterien:	2
oder:	

# Auswahl und Manipulation mit SQL



# Auswahl von Tupeln mit SQL



# Auswahl von Tupeln mit SQL: Übung 6.2



## Ü6.2 (Teil 1)

– Erweitern Sie das Formular ihres Produktkatalogs um eine Aufklappliste im Formularkopf, die alle Produktkategorien zeigt



- Folgen Sie dem Assistenten, der sich bei Aufklapplisten öffnet
- Wählen Sie als Datenquelle die Tabelle tblProduktKategorien
- Als Ergebniswert soll die Aufklappliste den Primärschlüssel aus der Tabelle liefern
- Es soll nur die Bezeichnung der Produktkategorie sichtbar sein

– ...

# Auswahl von Tupeln mit SQL: Übung 6.2



## Ü6.2 (Teil 2)

- Überlegen Sie, wie Sie den Wert des in der Aufklappliste ausgewählten Eintrags erhalten. (Hinweis: Dieser Wert entspricht dem Primärschlüssel der ausgewählten Produktkategorie.)
- Überlegen Sie sich eine SQL-Abfrage, die nur solche Produkte auswählt, die einer bestimmten Kategorie zugeordnet sind.
- Wie kann in VBA der Wert aus der Aufklappliste in der SQL-Abfrage berücksichtigt werden? Notieren Sie Ihre Abfrage.
- ...

# Auswahl von Tupeln mit SQL: Übung 6.2



## Ü6.2 – Lösungsvorschlag

- Ansatz: SQL-Anweisung als String speichern und mit Werten aus Variablen zusammenbauen.
- Beispiel:

```
' ...  
  
Dim strSql As String ' Variable für SQL-Abfrage  
Dim intKat As Integer ' Variable für ausgewählte Kategorie  
  
' Ermitteln des aktuellen Wertes der Aufklappliste,  
' liefert den Primärschlüssel der ausgewählten Kategorie  
Let intKat = Me.cmbKategorie.Value  
  
' SQL-Abfrage zur Auswahl der Produkte einer Kategorie mit dem Wert  
' aus der Aufklappliste verbinden  
Let strSql = "SELECT * FROM tblProdukte WHERE prdkatIdFk=" & intKat & ";"  
  
' ...
```

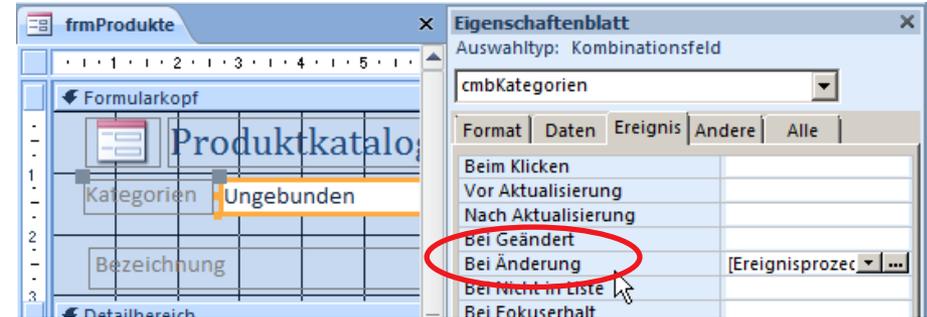
# Auswahl von Tupeln mit SQL: Übung 6.2



## Ü6.2 (Teil 3)

– ...

– Implementieren Sie eine Ereignisprozedur, die bei der Änderung der Auswahl in dieser Liste aufgerufen wird und



- aus der Aufklappliste den dort aktuell ausgewählten Wert ermittelt und in einer Variablen speichert,
- diese Variable dann in Ihre SQL-Anweisung (von oben) eingebaut und die gesamte Anweisung als String gespeichert
- die SQL-Anweisung im Direktbereich ausgibt
- dem Formular die SQL-Anweisung dann als Datenquelle zuweist (Nutzen Sie dafür die Eigenschaft Me.RecordSource)
- die Daten des Formulars neu lädt (mit Me.Requery).

– Hinweis: Wenn etwas nicht funktioniert vergleichen Sie, ob die Ausgabe im Direktbereich mit Ihrer beabsichtigten SQL-Abfrage übereinstimmt.

# Auswahl von Tupeln mit SQL



# Einfügen von Tupeln mit SQL



# Einfügen von Tupeln mit SQL: Übung 6.3



## Ü6.3 (Teil 1)

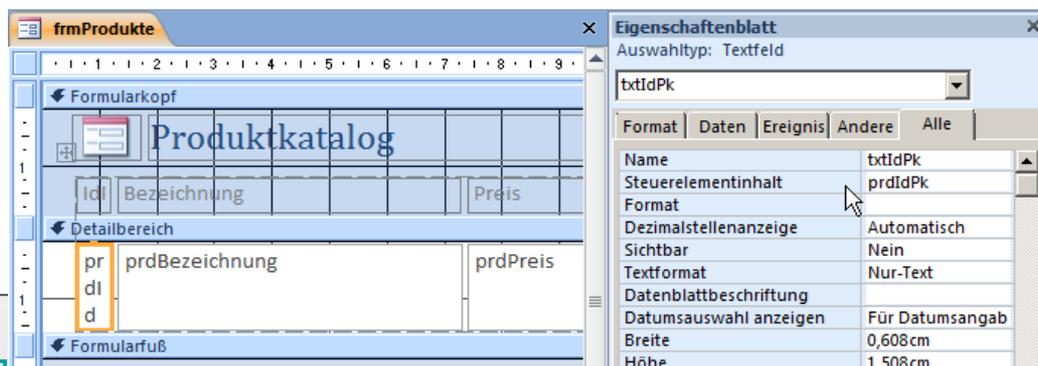
- Überlegen Sie sich eine SQL-Anweisung, mit der der aktuelle Artikel in den Warenkorb gelegt werden kann, d.h. in der Tabelle **tblWarenkoerbe** eine neue Zeile hinzugefügt wird
  - mit der Anzahl 1
  - mit dem aktuellen Zeitpunkt (Rückgabewert der VBA-Funktion **Now ( )**)
  - mit einem Fremdschlüssel für die Beziehung zum Produkt (aktueller Artikel, dessen Primärschlüssel in einem versteckten Feld auf der Oberfläche enthalten ist)
  - und einem Fremdschlüssel für die Beziehung zum Kunden (momentan fester Wert, wie in Übung 5.7, später angemeldeter Kunde)
- ...

# Einfügen von Tupeln mit SQL: Übung 6.3



## Ü6.3 – Lösungsvorschlag

- Hinweis: Im Formular frmProudkte existiert ein verstecktes Feld, das den Primärschlüssel wert des Produktes enthält.
- Beispiel:



```
' ...  
Dim strSql As String ' Variable für SQL-Anfrage  
Dim lngPrd As Long ' Variable für Primärschlüsselwert des akt. Produktes  
  
' Ermitteln des Primärschlüssels des aktuellen Produktes  
Let lngPrd = Me.txtIdPk.Value  
  
' SQL-Anweisung zum Einfügen in Warenkorb  
Let strSql = "INSERT INTO tblWarenkoerbe " & _  
            "(wkbAnzahl, wkbZeitpunkt, wkbprdIdFk, wkbkndIdFk) VALUES " & _  
            "(1, '" & Now() & "', " & lngPrd & ", 2 );"  
  
' ...
```

# Einfügen von Tupeln mit SQL: Übung 6.3



## Ü6.3 (Teil 2)

- ...
- Legen Sie das Modul **mdlWarenkoerbe** und darin eine öffentliche Prozedur **hinzufuegen** an, die
  - als Parameter den Primärschlüsselwert eines Produktes und eines Kunden bekommt (ProudktID, KundenID)
  - diese Parameterwerte in die SQL-Anweisung (von oben) einbaut
  - die SQL-Anweisung ausführt

– ...

The screenshot shows the Microsoft Access interface. On the left, the 'Projekt - projekt' window displays a tree view of the 'projekt (youtube\_db)' database. Under the 'Module' folder, the 'mdlWarenkoerbe' module is selected. The main window shows the 'hinzufuegen' procedure in the 'Allgemein' tab. The code is as follows:

```
Option Compare Database
Option Explicit

'Ü6.3
Public Sub hinzufuegen(plngProduktID As Long, plngKundenID As Long)
  ' Deklaration der Variable für die SQL-Anweisung

  ' Initialisierung mit SQL-Anweisung

  ' Ausgabe bei Bedarf (Fehlerkorrektur)

  ' Ausführen der SQL-Anweisung

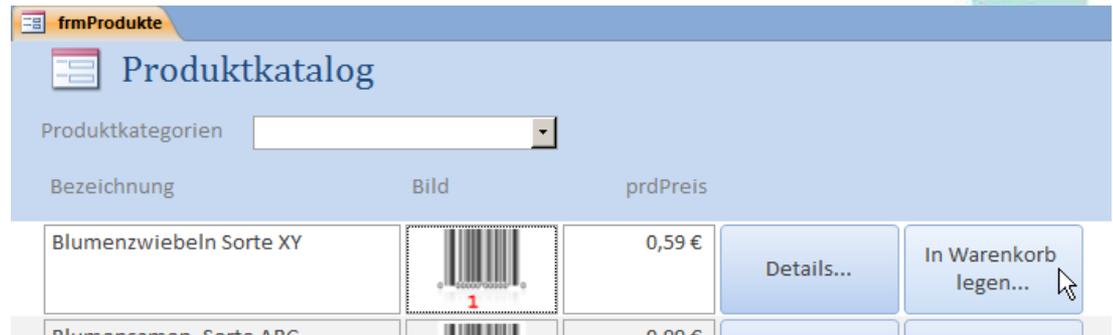
End Sub
```

# Einfügen von Tupeln mit SQL: Übung 6.3



## Ü6.3 (Teil 3)

- ...
- Erweitern Sie ihren Produktkatalog im Detailbereich um eine Schaltfläche "In Warenkorb legen"
- Legen Sie eine Ereignisprozedur für das Klicken auf diese Schaltfläche an, so dass
  - aus dem versteckten Feld mit dem Primärschlüssel des Produktes der Wert gelesen und in einer Variable gespeichert wird
  - in einer zweiten Variable ein fest vorgegebener Wert gespeichert wird, der als Primärschlüssel für einen Kunden in Ihrer Tabelle Kunden existiert.  
**WICHTIG: Nehmen Sie hier den gleichen Wert, wie in der Abfrage in Übung Ü5.7!**
  - und dann die Prozedur `mdlWarenkoerbe.hinzufuegen` mit diesen Variablen als Parameter aufgerufen wird
  - und anschließend das Formular des Warenkorbs geöffnet wird



# Einfügen von Tupeln mit SQL



# Ändern von Tupeln mit SQL



# Ändern von Tupeln mit SQL: Übung 6.4



## Ü6.4 (Teil 1)

- Überlegen Sie sich eine SQL-Anweisung, mit der die Menge des aktuellen Artikels um eins erhöht werden kann.
- Implementieren Sie im Modul **mdlWarenkoerbe** eine Prozedur **erhoehenAnzahl**,
  - die als Parameter einen Primärschlüsselwert des Warenkorbeintrags übergeben bekommt
  - und diesen Parameter in ihre SQL-Anweisung einbaut und ausführt

The screenshot shows the Microsoft Access interface. On the left, the 'Projekt - projekt' window displays a tree view of the 'projekt (youtube\_db)' database, with the 'mdlWarenkoerbe' module selected. The main window shows the 'erhoehenAnzahl' procedure in the 'Allgemein' tab. The code is as follows:

```
'Ü6.4
' Erhöhen der Anzahl eines Warenkorbeintrags
Public Sub erhoehenAnzahl(plngWarenkorbID As Long)
' Deklaration der Variable für die SQL-Anweisung

' Initialisierung mit SQL-Anweisung

' Ausgabe bei Bedarf (Fehlerkorrektur)

' Ausführen der SQL-Anweisung

End Sub
```

# Ändern von Tupeln mit SQL: Übung 6.4



## Ü6.4 (Teil 2)

- ...
- Erweitern Sie das Formular des Warenkorbs im Detailbereich um eine Schaltfläche zum Erhöhen der Menge eines Produktes im Warenkorb
- Implementieren Sie die Ereignisprozedur so, dass beim Klick auf die Schaltfläche
  - aus dem versteckten Feld auf der Oberfläche der Primärschlüssel des Warenkorbeintrags ermittelt und
  - anschließend die Prozedur **mdlWarenkoerbe.erhoeheAnzahl** mit der ID als Parameter aufgerufen wird
  - und das Formular aktualisiert wird.



# Ändern von Tupeln mit SQL



# Löschen von Tupeln mit SQL



# Löschen von Tupeln mit SQL: Übung 6.5



## Ü6.5 (Teil 1)

- Überlegen Sie sich eine SQL-Anweisung, mit der ein Eintrag aus dem Warenkorb gelöscht werden kann.
- Implementieren Sie im Modul **mdlWarenkoerbe** eine Prozedur **loeschen**,
  - die als Parameter den Primärschlüssel des zu löschenden Warenkorbbeitrags übergeben bekommt
  - und diesen Parameter in ihre SQL-Anweisung einbaut und ausführt

```
(Allgemein) loeschen

'Ü6.5
' Löschen eines Warenkorbbeitrags
Public Sub loeschen(plngWarenkorbID As Long)
' Deklaration der Variable für die SQL-Anweisung

' Initialisierung mit SQL-Anweisung

' Ausgabe bei Bedarf (Fehlerkorrektur)

' Ausführen der SQL-Anweisung

End Sub
```

# Löschen von Tupeln mit SQL: Übung 6.5



## Ü6.5 (Teil 2)

- Erweitern Sie das Formular des Warenkorbs im Detailbereich um eine Schaltfläche zum Löschen eines Eintrags im Warenkorb
- Implementieren Sie die Ereignisprozedur so, dass beim Klick auf die Schaltfläche
  - aus dem versteckten Feld auf der Oberfläche den Primärschlüsselwert des Warenkorbeintrags ermittelt und
  - anschließend die Prozedur **mdlWarenkoerbe.loesche** mit dem Primärschlüsselwert als Parameter aufgerufen wird

Bezeichnung	Anzahl	Einzelpreis	Gesamtpreis	
Blumensamen, Sorte ABC	2	0,99 €	1,98 €	Löschen
Blumenzwiebeln Sorte XY	1	0,59 €	0,59 €	

# Löschen von Tupeln mit SQL



# Verbinden (Join) von Relationen mit SQL



# Verbinden (Join) von Relationen in SQL: Übung 6.6



**Ü6.6: Überlegen Sie sich eine SQL-Abfrage die Datensätze aus den Tabellen Benutzer und Kunden kombiniert, so dass**

- alle Spalten aus Tabelle Benutzer und die zugehörige KundenID (**kndIdPk** aus Tabelle Kunden) ausgegeben werden und
- alle Datensätze aus Benutzer enthalten sind, selbst wenn sie keinen Join-Partner in der Tabelle Kunden haben

**Probieren Sie die SQL-Abfrage in der SQL-Ansicht einer Auswahlabfrage aus.**

**Speichern Sie diese Abfrage als:**

Wichtig, weil in weiteren Übungen damit gearbeitet wird.

**qryBenutzerUndKundenID**

# Verbinden (Join) von Relationen mit SQL



# Inhalt



## Lernziel, Lehrinhalte und Einordnung

### SQL-Anweisungen zur Auswahl und Manipulation

- Einfache Anweisungen
- Auswählen von Produkten einer Produktkategorie
- Einfügen von Produkten zum Warenkorb
- Ändern der Anzahl eines Produktes im Warenkorb
- Löschen eines Produktes aus dem Warenkorb
- Verbinden (Join) von Produkten und Produktkategorien



## Ausblick



# Inhalt

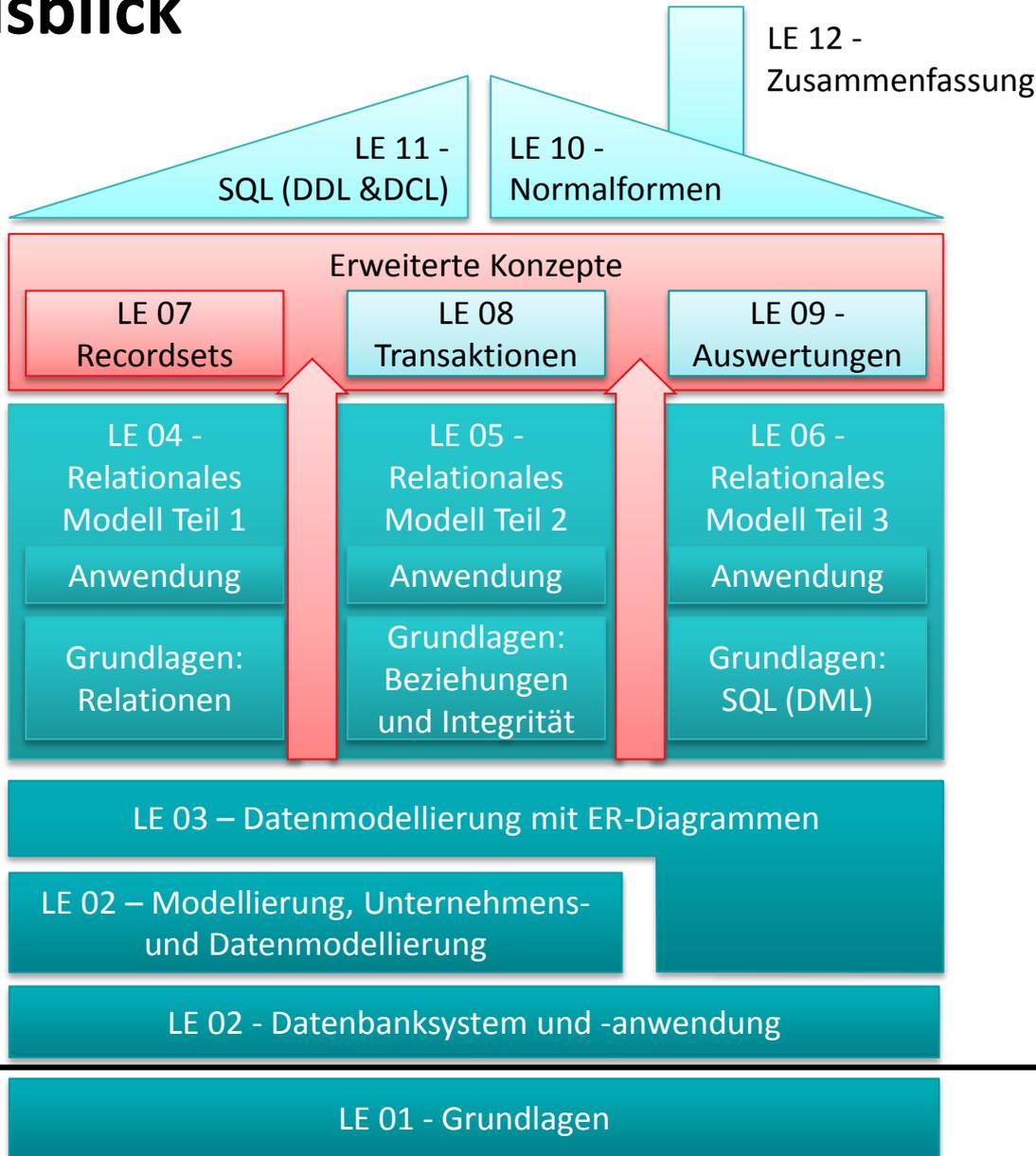
## Lernziel, Lehrinhalte und Einordnung

## SQL-Anweisungen zur Auswahl und Manipulation

- Einfache Anweisungen
- Auswählen von Produkten einer Produktkategorie
- Einfügen von Produkten zum Warenkorb
- Ändern der Anzahl eines Produktes im Warenkorb
- Löschen eines Produktes aus dem Warenkorb
- Verbinden (Join) von Produkten und Produktkategorien

## Ausblick

# Ausblick





BEUTH HOCHSCHULE FÜR TECHNIK BERLIN  
University of Applied Sciences

# Übungen zu Wirtschaftsinformatik 2

## LE 06 – Relationales Modell (Teil 3)

### SQL

Prof. Dr. Thomas Off

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