

**BHT** Berliner Hochschule für Technik

## Wirtschaftsinformatik 2

### LE 06 – Relationales Modell (Teil 3)

#### SQL

Prof. Dr. Thomas Off  
<http://www.ThomasOff.de/lehre/beuth/wi2>

1

**BHT**

### Ziel

- Relationale Algebra und ihre wichtigsten Operationen kennenlernen
- Grundbestandteile der Datenbanksprache SQL kennenlernen
- SQL-Anweisungen der Data Manipulation Language (DML) nutzen
  - Abfrage von Daten mit SQL zur Umsetzung von Operationen der Relationalen Algebra
  - Hinzufügen, Löschen und Ändern von Daten mit SQL
- Praktische Anwendung in MS Access üben
  - Verwendung der verschiedenen SQL-Anweisungen in verschiedenen Abfragearten von MS Access
  - Integration von SQL-Anweisungen in eigene Funktionen, Prozeduren und Formulare

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

3

**BHT**

### Einordnung

LE 12 - Zusammenfassung

LE 11 - SQL (DDL & DCL)

LE 10 - Normalformen

Erweiterte Konzepte

LE 07 Recordsets

LE 08 Transaktionen

LE 09 - Auswertungen

LE 04 - Relationales Modell Teil 1 Anwendung

LE 05 - Relationales Modell Teil 2 Anwendung

LE 06 - Relationales Modell Teil 3 Anwendung

Grundlagen: Relationen

Grundlagen: Beziehungen und Integrität

Grundlagen: SQL (DML)

LE 03 - Datenmodellierung mit ER-Diagrammen

LE 02 - Modellierung, Unternehmens- und Datenmodellierung

LE 02 - Datenbanksystem und -anwendung

LE 01 - Grundlagen

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

4

**BHT**

### Inhalt

#### Ziel und Einordnung

#### Rückblick

- Beziehungen im Relationalen Modell
- Fremdschlüssel und Referenzielle Integrität

#### Relationale Algebra und ihre Operationen

#### SQL kennenlernen

- Bestandteile
- Sprachumfang zum Auswählen, Einfügen, Ändern und Löschen
- Zusammenfassung

#### SQL in MS Access anwenden

- Formulare und SQL-Auswahlabfragen
- Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL
- Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen
- Zusammenfassung

#### Ausblick

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

5

### Rückblick

Beziehungen werden über Fremdschlüssel hergestellt

- ER-Modell
- Relationen

Mitarbeiter	Nr	Name	VName	Verkäufe	MaNr	PrdNr	Produkte	Nr	Name
	123	Huber	Mike		123	23456		23456	KB6314
	234	Mittag	Michael		123	34567		34567	AT1224
	345	Albers	Heidi		345	34567		45678	AT1218
					345	45678		56789	MB4711

Es gibt keinen Fremdschlüssel (ungleich "leer"), dessen Wert im zugehörigen Primärschlüssel nicht existiert (2. Integritätsregel, Referenzielle Integrität)

- was beim Entwurf der Relationen und
- beim Anlegen von Daten bedacht werden muss.

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 6

6

### Rückblick

#### Umsetzung von Beziehungen in MS Access

- Schritt 1: Beziehungen zwischen vorhandenen Relationen werden über zusätzliche Spalten für Fremdschlüssel umgesetzt
- Schritt 2: Anlegen der Beziehung in der Beziehungsansicht, Festlegen der Integritätsbedingungen
- Schritt 3: Erfassen von Daten in der Reihenfolge in der die Beziehung dies erfordert

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 7

7

### Rückblick

#### Selektion (Restriktion)

#### Projektion

#### Join

Nur relevante Tupel werden ausgewählt.

Nur relevante Attribute werden ausgewählt.

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 8

8

### Rückblick

#### Auswählen von Tupeln über grafische Abfragen in Access

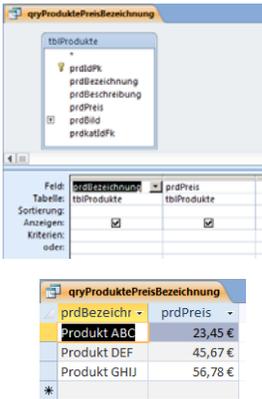
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 9

9

**Rückblick** BHT

**Auswählen von Tupeln über grafische Abfragen in Access**

- ...
- Projektion
- ...



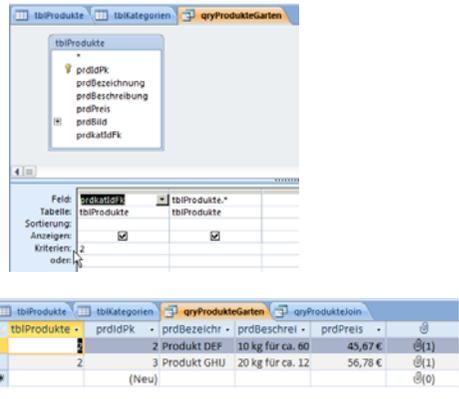
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 10

10

**Rückblick** BHT

**Auswählen von Tupeln über grafische Abfragen in Access**

- ...
- Projektion
- ...



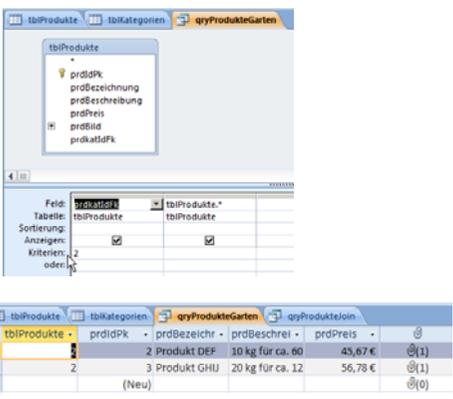
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 11

11

**Rückblick** BHT

**Auswählen von Tupeln über grafische Abfragen in Access**

- Restriktion
- Projektion
- ...



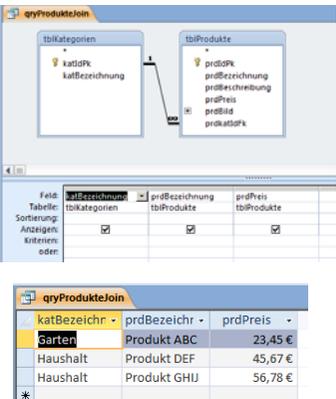
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 12

12

**Rückblick** BHT

**Auswählen von Tupeln über grafische Abfragen in Access**

- Restriktion
- Projektion
- Join



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 13

13

**Rückblick** BHT

**Hinweis:**  
Wenn LE online, dann auch hier schon SQL-Ansicht zeigen!

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 14

14

**Rückblick** BHT

**Komponenten eines Datenbanksystems**

- Datenbasis (syn. Datenbank, DB)
- Data Dictionary (DD)
- Datenbankmanagementsystem (DBMS)
- Datenbanksprache

**Datenbanksystem besteht aus DBMS + DD + mind. einer DB + Datenbanksprache**

Wirtschaftsinformatik 2 - LE 02 - Grundlagen von Datenbanksystemen 15

15

**Rückblick** BHT

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 16

16

**Inhalt** BHT

**Ziel und Einordnung**

**Rückblick**

- Beziehungen im Relationalen Modell
- Fremdschlüssel und Referenzielle Integrität

**Relationale Algebra und ihre Operationen**

**SQL kennenlernen**

- Bestandteile
- Sprachumfang zum Auswählen, Einfügen, Ändern und Löschen
- Zusammenfassung

**SQL in MS Access anwenden**

- Formulare und SQL-Auswahlabfragen
- Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL
- Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen
- Zusammenfassung

**Ausblick**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 17

17

**Relationale Algebra** BHT

**Relationale Algebra**

- ist Bestandteil des Relationalen Modells
- definiert eine Menge von Operationen, mit denen Relationen verknüpft und abgefragt werden können

**Operationen der relationalen Algebra**

- werden immer auf Relationen angewandt (Ausgangsrelationen)
- erzeugen als Ergebnis eine neue Relation (Ergebnisrelation)
  - ist (im Gegensatz zu Ausgangsrelationen) nicht gespeichert, d.h. nur vorübergehend verfügbar und
  - kann durch erneute Ausführung der Operation wieder erzeugt werden
- können nacheinander (geschachtelt) angewandt werden

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 18

18

**Operationen der Relationalen Algebra** BHT

**Mengenoperationen**

- Vereinigung
- Schnitt
- Differenz

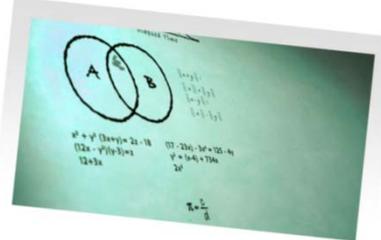
**Projektion**

**Selektion/Restriktion**

**Kartesisches Produkt**

**Verbund (Join)**

...



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 19

19

**Mengenoperationen** BHT

**Vereinigung**

- Ausgangsrelationen müssen die gleichen Attribute mit passenden Datentypen haben (vereinigungskompatibel)
- Ergebnisrelation
  - hat alle Attribute der Ausgangsrelation und
  - umfasst alle Tupel der Ausgangsrelationen, wobei gleiche Tupel nur einmal enthalten sind
- ...

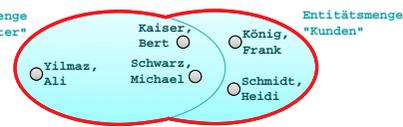
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 20

20

**Mengenoperationen** BHT

**Vereinigung**

- ...
- Beispiel: Sowohl Kunden als auch Mitarbeiter sollen eine Weihnachtskarte bekommen. Die Vereinigungsmenge ist zu bilden.



**Ausgangsrelationen**

Mitarb	Name	VName
	Kaiser	Bert
	Schwarz	Michael
	Yilmaz	Ali

Kunden	Name	VName
	Kaiser	Bert
	Schwarz	Michael
	König	Frank
	Schmidt	Heidi

➔ Vereinigung

**Ergebnisrelation**

Name	VName
Kaiser	Bert
Schwarz	Michael
Yilmaz	Ali
König	Frank
Schmidt	Heidi

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 21

21

## Mengenoperationen

BHT

### Schnitt

- Ausgangsrelationen müssen die gleichen Attribute mit passenden Datentypen haben (vereinigungskompatibel)
- Ergebnisrelation
  - hat alle Attribute der Ausgangsrelationen,
  - umfasst alle Tupel, die in beiden Relationen enthalten sind
- ...

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 22

22

## Mengenoperationen

BHT

### Schnitt

- ...
- Beispiel: Mitarbeiter, die auch Kunden sind, bekommen 10% Rabatt auf ihre Einkäufe am Jahresende ausbezahlt. Die Schnittmenge ist zu bilden.

Ausgangsrelationen

Mitarb	Name	VName
	Kaiser	Bert
	Schwarz	Michael
	Yilmaz	Ali

Kunden	Name	VName
	Kaiser	Bert
	Schwarz	Michael
	König	Frank
	Schmidt	Heidi

Schnitt

Name	VName
Kaiser	Bert
Schwarz	Michael

Ergebnisrelation

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 23

23

## Mengenoperationen

BHT

### Differenz

- Ausgangsrelationen müssen die gleichen Attribute mit passenden Datentypen haben (vereinigungskompatibel)
- Ergebnisrelation
  - hat alle Attribute der Ausgangsrelationen,
  - umfasst alle Tupel, die in der ersten Relationen enthalten sind und in der zweiten Relation nicht enthalten sind
- ...

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 24

24

## Mengenoperationen

BHT

### Differenz

- ...
- Beispiel 1
  - An alle Mitarbeiter, die noch keine Kunden sind, soll eine Information über den Mitarbeiterabbat versandt werden. Die Differenz von Mitarbeitern ohne Kunden ist zu bilden.

Ausgangsrelationen

Mitarb	Name	VName
	Kaiser	Bert
	Schwarz	Michael
	Yilmaz	Ali

Kunden	Name	VName
	Kaiser	Bert
	Schwarz	Michael
	König	Frank
	Schmidt	Heidi

Differenz: Mitarbeiter ohne Kunden

Name	VName
Yilmaz	Ali

Ergebnisrelation

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 25

25

## Mengenoperationen

**Differenz**

- ...
- Beispiel 2
  - Für Kunden gibt es einen Weihnachtsrabatt, der nicht für Mitarbeiter gewährt wird. Die Differenz von Kunden ohne Mitarbeiter ist zu bilden.

**Ausgangsrelationen**

Kunden	Name	VName	Mitarb	Name	VName
	Kaiser	Bert		Kaiser	Bert
	Schwarz	Michael		Schwarz	Michael
	König	Frank		Yilmaz	Ali
	Schmidt	Heidi			

**Ergebnisrelation**

Name	VName
König	Frank
Schmidt	Heidi

Differenz: Kunden ohne Mitarbeiter

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 26

26

## Projektion und Selektion (Restriktion)

**Projektion**

- bestimmt Attribute der Ausgangsrelation, die in der Ergebnisrelation enthalten sein sollen
- Reihenfolge der Attribute in der Ergebnisrelation festgelegt
- entstehen durch Wegfallen von Attributen doppelte Tupel, so werden diese per Definition aus Ergebnisrelation entfernt
- Beispiel: Ort, Name (in dieser Reihenfolge) in Ergebnisrelation

**Ausgangsrelationen**

Kunden	ID	Name	VName	Plz	Ort	StrasseNr
	9	Kaiser	Bert	12345	Berlin	Bergstr. 8
	8	Schwarz	Michael	12345	Berlin	Schloßstr. 7
	7	Kaiser	Frank	12360	Berlin	Turmstr. 23
	6	Schmidt	Heidi	14482	Potsdam	Badstr. 9

**Ergebnisrelation**

Ort	Name
Berlin	Kaiser
Berlin	Schwarz
Potsdam	Schmidt

Projektion (Ort, Name)

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 27

27

## Projektion und Selektion (Restriktion)

**Selektion**

- bestimmt die Tupel der Ausgangsrelation, die in der Ergebnisrelation enthalten sein sollen
- Formulierung einer Bedingung, die Attributwerte in Tupeln der Ausgangsrelation erfüllen müssen
- Ergebnisrelation umfasst nur solche Tupel, deren Attributwerte die Bedingung erfüllen
- Beispiel: Alle Kunden aus Berlin.

**Ausgangsrelationen**

Kunden	ID	Name	VName	Ort
	9	Kaiser	Bert	Berlin
	8	Schwarz	Michael	Berlin
	7	König	Frank	Berlin
	6	Schmidt	Heidi	Potsdam

**Ergebnisrelation**

ID	Name	VName	Ort
9	Kaiser	Bert	Berlin
8	Schwarz	Michael	Berlin
7	König	Frank	Berlin

Selektion (Ort = Berlin)

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 28

28

## Kartesisches Produkt

**Ergebnisrelation hat alle Attribute der Ausgangsrelationen und umfasst alle möglichen Kombinationen von Tupeln der Ausgangsrelationen**

**Ausgangsrelationen**

Namen	Name	VNamen	VName
	Kaiser		Thomas
	Schwarz		Mike
	König		Susi
	Schmidt		

**Ergebnisrelationen**

Name	VName
Kaiser	Thomas
Kaiser	Mike
Kaiser	Susi
Schwarz	Thomas
Schwarz	Mike
Schwarz	Susi
König	Thomas
König	Mike
König	Susi
Schmidt	Thomas
Schmidt	Mike
Schmidt	Susi

Kartesisches Produkt

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 29

29

**Verbund (Join)** BHT

**Allgemeine Form (Theta Join)**

**Gleichverbund (Equi Join)**

**Natürlicher Verbund (Natural Join)**

**Innerer Verbund (Inner Join)**

**Äußerer Verbund (Outer Join)**

- Linker äußerer Verbund (Left Outer Join)
- Rechter äußerer Verbund (Right Outer Join)



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 30

30

**Verbund (Join)** BHT

**Allgemeine Form (Theta Join)**

- Verbindung von Ausgangsrelationen anhand der Werte ausgewählter Attribute
- Formulierung einer Bedingung, die diese Attributwerte erfüllen müssen

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 31

31

**Verbund (Join)** BHT

**Gleichverbund (Equi Join)**

- Verbindung von Ausgangsrelationen anhand gleicher Werte der ausgewählten Attribute
- Nur wenn die Werte der ausgewählten Attribute in den Tupeln gleich sind, ist das durch eine Verbindung entstehende neue Tupel in der Ergebnisrelation enthalten
- Häufig wird in der praktischen Anwendung zusätzlich verlangt, dass die am Vergleich beteiligten Attribute gleiche Bezeichnungen haben müssen
- ...

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 32

32

**Verbund (Join)** BHT

**Gleichverbund (Equi Join)**

- ...
- Beispiel: Produkte.LID = Lieferanten.LID

Ausgangsrelationen

Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

Ergebnisrelation

PID	Name	LID	LID	Firma	Ort
123	Multi AB	987	987	Müller AG	Berlin
234	Flexi 123	987	987	Müller AG	Berlin
345	Mega+	876	876	Meier GmbH	Potsdam

} (Red bracket indicates the result relation is derived from the join of the two input relations)

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 33

33

**Verbund (Join)** BHT

**Natürlicher Verbund (Natural Join)**

- Verbindung von Ausgangsrelationen anhand gleicher Werte ausgewählter Attribute (analog zum Gleichverbund)
- die Attribute, deren Werte verglichen wurden, sind nur einmal in der Ergebnisrelation enthalten
- Sehr häufig wird in der praktischen Anwendung zusätzlich verlangt, dass die am Vergleich beteiligten Attribute gleiche Bezeichnungen haben müssen
- ...

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 34

34

**Verbund (Join)** BHT

**Natürlicher Verbund (Natural Join)**

- ...
- Beispiel: Beispiel: Produkte.LID = Lieferanten.LID

Ausgangsrelationen

Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

Lieferanten	LID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus

Ergebnisrelation

PID	Name	LID	Firma	Ort
123	Multi AB	987	Müller AG	Berlin
234	Flexi 123	987	Müller AG	Berlin
345	Mega+	876	Meier GmbH	Potsdam

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 35

35

**Verbund (Join)** BHT

**Innerer Verbund (Inner Join)**

- Verbindung von Ausgangsrelationen anhand der Werte ausgewählter Attribute
- Formulierung einer Bedingung, die diese Attributwerte erfüllen müssen
- Nur wenn die Werte der ausgewählten Attribute in den verglichenen Tupeln die Bedingung erfüllen, ist das durch eine Verbindung entstehende neue Tupel in der Ergebnisrelation enthalten
- Hinweis: Im Folgenden prüft die Bedingung auf Gleichheit. Es muss aber im Inner Join nicht immer Gleichheit sein! Deshalb erscheinen beide in der Bedingung geprüften Spalten in der Ergebnisrelation.

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 36

36

**Verbund (Join)** BHT

**Innerer Verbund (Inner Join)**

- ...
- Beispiel: Produkte.LID = Lieferanten.ID

Ausgangsrelationen

Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus

Ergebnisrelation

PID	Name	LID	ID	Firma	Ort
123	Multi AB	987	987	Müller AG	Berlin
234	Flexi 123	987	987	Müller AG	Berlin
345	Mega+	876	876	Meier GmbH	Potsdam

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 37

37

## Verbund (Join) BHT

### Äußerer Verbund (Outer Join)

- Verbindung von Ausgangsrelationen anhand der Werte ausgewählter Attribute
- Formulierung einer Bedingung, die diese Attributwerte erfüllen müssen
- Im Ergebnis sind mindestens die Tupel einer der beteiligten Relation enthalten und wurden ergänzt, um
  - Attributwerte aus der anderen Relation, wenn die Attributwerte der verglichenen Tupel die Bedingung erfüllen
  - NULL-Werte (leere Attributwerte), wenn die Attributwerte der verglichenen Tupel die Bedingung nicht erfüllen
- Es werden der linke und der rechte äußere Verbund unterschieden (nächste Folie).

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 38

38

## Verbund (Join) BHT

### Linker äußerer Verbund (Left Outer Join)

- Mindestens die Tupel der linken Relation im Ergebnis vorhanden; wenn sie keinen Partner haben, dann mit NULL
- Beispiel: Produkte.LID = Lieferaten.ID

Ausgangsrelationen

Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus
	654	Berg GmbH	Berlin

Ergebnisrelation

PID	Name	LID	ID	Firma	Ort
123	Multi AB	987	987	Müller AG	Berlin
234	Flexi 123	987	987	Müller AG	Berlin
345	Mega+	876	876	Meier GmbH	Potsdam
456	Super XL				

Mindestens alle linken Tupel

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 39

39

## Verbund (Join) BHT

### Rechter äußerer Verbund (Right Outer Join)

- Mindestens die Tupel der rechten Relation im Ergebnis vorhanden; wenn sie keinen Partner haben, dann mit NULL
- Beispiel: Produkte.LID = Lieferaten.ID

Ausgangsrelationen

Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus
	654	Berg GmbH	Berlin

Ergebnisrelation

PID	Name	LID	ID	Firma	Ort
123	Multi AB	987	987	Müller AG	Berlin
234	Flexi 123	987	987	Müller AG	Berlin
345	Mega+	876	876	Meier GmbH	Potsdam
			765	Bach&Sohn	Cottbus
			654	Berg GmbH	Berlin

Mindestens alle rechten Tupel

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 40

40

## Verbund (Join) BHT

### Hinweise

- Die vorherigen Darstellungen zeigten immer
  - zwei Tabellen, der Verbund ist aber analog auch mit mehreren Tabellen möglich (indem er zunächst für die ersten zwei Tabellen durchgeführt wird und das Ergebnis mit der nächsten Tabelle verbunden wird usw.)
  - die Verbindung anhand einer Bedingung für Werte von zwei Attributen, mehrere Bedingungen und mehrere Attribute sind auch möglich
- Es gibt weitere Arten des Verbunds, die hier nicht relevant sind
  - Auto-/Self Join einer Relation mit sich selbst (z.B. Person mit ihrem Ehepartner, der ebenfalls als Person gespeichert ist)
  - Semi Join: Natürlicher Verbund ohne die zusätzlichen Attribute der rechten Relation
  - ...

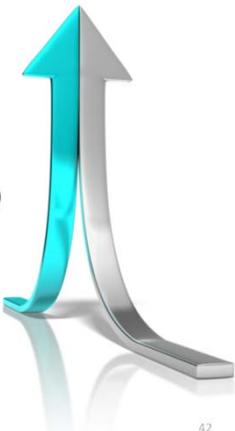
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 41

41

**Verbund (Join)** BHT

**Allgemeine Form (Theta Join)**  
**Gleichverbund (Equi Join)**  
**Natürlicher Verbund (Natural Join)**  
**Innerer Verbund (Inner Join)**  
**Äußerer Verbund (Outer Join)**

- Linker äußerer Verbund (Left Outer Join)
- Rechter äußerer Verbund (Right Outer Join)



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 42

42

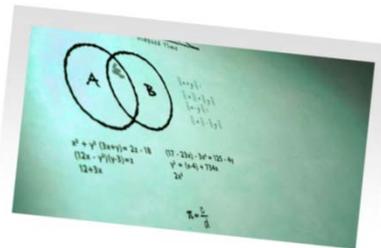
**Operationen der Relationalen Algebra** BHT

**Mengenoperationen**

- Vereinigung
- Schnitt
- Differenz

**Projektion**  
**Selektion/Restriktion**  
**Kartesisches Produkt**  
**Verbund (Join)**

...



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 43

43

**Inhalt** BHT

**Ziel und Einordnung**  
**Rückblick**

- Beziehungen im Relationalen Modell
- Fremdschlüssel und Referenzielle Integrität

**Relationale Algebra und ihre Operationen**  
**SQL kennenlernen**

- Bestandteile
- Sprachumfang zum Auswählen, Einfügen, Ändern und Löschen
- Zusammenfassung

**SQL in MS Access anwenden**

- Formulare und SQL-Auswahlabfragen
- Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL
- Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen
- Zusammenfassung

**Ausblick**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 44

44

**Bestandteile des SQL-Sprachumfangs** BHT

**Data Manipulation Language (DML):** dient zur Abfrage, zum Hinzufügen, zur Veränderung und zum Löschen von Daten

- SELECT
- INSERT
- UPDATE
- DELETE

**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):** dient zum Einrichten, Festlegen und Entziehen von Zugriffsrechten für Benutzer und Gruppen auf den Strukturen und auf Aktionsmöglichkeiten der Datenbanken

*Thema von LE 11*

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 46

46

### Arbeiten mit Relationen und Tupeln per SQL BÄT

**Auswählen**

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

**Einfügen**

- neue Tupel werden zur Relation hinzugefügt

**Ändern**

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

**Löschen**

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 47

47

### Auswählen von Tupeln mit SQL BÄT

**SQL-Syntax ähnlich der natürlichen Sprache (in Englisch), d.h. es kann bspw. formuliert werden**

- "Wähle alles aus Tabelle *Kunden*."
- "Wähle *Name, Vorname* aus Tabelle *Kunden*." (Projektion)
- "Wähle alle aus der Tabelle *Kunden* bei denen der *Ort 'Berlin'* ist." (Selektion)
- "Wähle *Gesamtpreis* der *Bestellung* und *Name* sowie *Vorname* des *Kunden*, der die Bestellung bestellt hat." (Join)

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 48

48

### Auswählen von Tupeln mit SQL BÄT

**Allgemeine Abfrage in SQL (Bsp. 1):**

- Deutsch: "Wähle alle Spalten aus Tabelle *Kunden*."
- Englisch: "Select all Cols from table *Kunden*."
- SQL: **SELECT \* FROM *kunden*;**

Ausgangsrelation

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

SELECT \* FROM Kunden;

Ergebnis

KndNr	Name	Vorname	Ort
123	Albers	Willi	Aachen
234	Boehrs	Vera	Berlin
345	Dinkel	Ulrike	Berlin
456	Dinkel	Thomas	Berlin
567	Esser	Thomas	Dessau

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 49

49

### Auswählen von Tupeln mit SQL BÄT

**Projektion in SQL (Bsp. 2):**

- Deutsch: "Wähle *Name, Vorname* aus Tabelle *Kunden*."
- Englisch: "Select *Name, Vorname* from table *Kunden*."
- SQL: **SELECT name, vorname FROM *kunden*;**

Ausgangsrelation

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

SELECT Name, Vorname FROM Kunden;

Ergebnis

Name	Vorname
Albers	Willi
Boehrs	Vera
Dinkel	Ulrike
Dinkels	Thomas
Esser	Thomas

↑ ↑  
Projektion

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 50

50

### Auswählen von Tupeln mit SQL

**Selektion in SQL (Bsp. 3):**

- Deutsch: "Wähle alle Spalte aus der Tabelle *Kunden* bei denen der Ort 'Berlin' ist."
- Englisch: "Select all columns from table *Kunden* where Ort is 'Berlin'."
- SQL: `SELECT * FROM kunden WHERE ort='Berlin';`

Ausgangsrelation

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

`SELECT * FROM Kunden WHERE Ort='Berlin';`

Ergebnis

KndNr	Name	Vorname	Ort
234	Boehrs	Vera	Berlin
345	Dinkel	Ulrike	Berlin
456	Dinkels	Thomas	Berlin

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 51

51

### Auswählen von Tupeln mit SQL

**Hinweis:**

**Wenn LE online, dann hier analog zu Folien 101 bis 105 die SQL-Ansicht einführen und folgende Beispiele zeigen**

- Ermittle alle Bestellungen vom 11.10.2013
- Ermittle alle Bestellungen, des Kunden Yilmaz
- Ermittle alle Bestellungen, mit einem Preis von mind. 200 EUR
- Ermittle alle Bestellungen, die bereits geliefert wurden
- Ermittle alle Bestellungen, die keine Lieferadresse haben

**Anschließend Übungen Ü6.1 Teil 1 inkl. Fortsetzung**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 52

52

### Auswählen von Tupeln mit SQL

**Innerer Verbund (Inner Join) in SQL (Bsp. 5):**

- Name sowie Vorname des Kunden, der eine Bestellung bestellt hat und den Gesamtpreis der Bestellung.
- SQL (Variante 1):

```
SELECT kunden.name, kunden.vorname, bestellungen.gesamt
FROM kunden, bestellungen
WHERE kunden.kndnr = bestellungen.kndnr;
```

Ausgangsrelationen

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

Bestellungen	BstID	KndNr	Gesamt	Datum
	987	456	80€	1.1.2012
	876	345	320€	7.3.2011
	765	234	120€	1.8.2011

`SELECT kunden.name, kunden.vorname, bestellungen.gesamt FROM kunden, bestellungen WHERE kunden.kndnr = bestellungen.kndnr;`

Ergebnis

Name	Vorname	Gesamt
Boehrs	Vera	120€
Dinkel	Ulrike	320€
Dinkels	Thomas	80€

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 53

53

### Auswählen von Tupeln mit SQL

**Innerer Verbund (Inner Join) in SQL (Bsp. 5):**

- Name sowie Vorname des Kunden, der eine Bestellung bestellt hat und den Gesamtpreis der Bestellung.
- SQL (Variante 2):

```
SELECT kunden.name, kunden.vorname, bestellungen.gesamt
FROM kunden INNER JOIN bestellungen
ON kunden.kndnr = bestellungen.kndnr;
```

Ausgangsrelationen

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

Bestellungen	BstID	KndNr	Gesamt	Datum
	987	456	80€	1.1.2012
	876	345	320€	7.3.2011
	765	234	120€	1.8.2011

`SELECT kunden.name, kunden.vorname, bestellungen.gesamt FROM kunden INNER JOIN bestellungen ON kunden.kndnr = bestellungen.kndnr;`

Ergebnis

Name	Vorname	Gesamt
Boehrs	Vera	120€
Dinkel	Ulrike	320€
Dinkels	Thomas	80€

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 54

54

### Auswählen von Tupeln mit SQL BHT

**Linker äußerer Verbund (Left Outer Join) in SQL (Bsp. 5):**

- Alle Lieferadressen mit Straße, Ort und ggf. vorhandene Bestellungen mit ID, Datum
- SQL:

```
SELECT LAdr.StraßeNr, LAdr.PlzOrt, Bestell.BstID, Bestell.Datum
FROM LAdr LEFT JOIN Bestell
ON LAdr.AdrNr = Bestell.AdrNr;
```

Ausgangsrelationen

LAdr	AdrNr	StraßeNr	PlzOrt	Bestell	BstID	AdrNr	Liefern	Gesamt	Datum
	123	Bergstr 7	14476 Potsdam		987	234	J	80€	1.1.2012
	234	Badstr 1a	12345 Berlin		876		N	320€	7.3.2011
	345	Schloßstr 8	12456 Berlin		765	345	J	120€	1.8.2011

Linke Tabelle

Ergebnisrelation

StraßeNr	PlzOrt	BstID	Datum
Bergstr 7	14476 Potsdam		
Badstr 1a	12345 Berlin	987	1.1.2012
Schloßstr 8	12456 Berlin	765	1.8.2011

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 55

55

### Auswählen von Tupeln mit SQL BHT

**Rechter äußerer Verbund (Right Outer Join) in SQL (Bsp. 5):**

- Alle Bestellungen mit ID, Datum und ggf. vorhandene Lieferadressen mit Straße, Ort
- SQL:

```
SELECT LAdr.StraßeNr, LAdr.PlzOrt, Bestell.BstID, Bestell.Datum
FROM LAdr RIGHT JOIN Bestell
ON LAdr.AdrNr = Bestell.AdrNr;
```

Rechte Tabelle

Ausgangsrelationen

LAdr	AdrNr	StraßeNr	PlzOrt	Bestell	BstID	AdrNr	Liefern	Gesamt	Datum
	123	Bergstr 7	14476 Potsdam		987	234	J	80€	1.1.2012
	234	Badstr 1a	12345 Berlin		876		N	320€	7.3.2011
	345	Schloßstr 8	12456 Berlin		765	345	J	120€	1.8.2011

Ergebnisrelation

StraßeNr	PlzOrt	BstID	Datum
Badstr 1a	12345 Berlin	987	1.1.2012
Schloßstr 8	12456 Berlin	765	1.8.2011

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 56

56

### Auswählen von Tupeln mit SQL BHT

**Hinweis**

**Wenn LE online, dann hier Beispiele zeigen**

- Kunden (Name, Vorname) und Bestellungen (Datum, Preis)
- Kunden (Name, Vorname) mit und ohne Bestellung sowie die ggf. vorhandenen Bestellungen (Datum, Preis)
- Bestellungen (Datum, Preis) mit und ohne Kunden, sowie die ggf. vorhandenen Kunden mit (Name, Vorname)
- Warum liefert die letzte Abfrage das gleiche Ergebnis wie die erste?

**Anschließend weiter mit Übung Ü6.1 Teil2**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 57

57

### Auswählen von Tupeln mit SQL BHT

**Mengenoperationen**

- Vereinigung
- Differenz
- Schnitt

**Kartesisches Produkt**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 58

58

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
  - Mengen A und B werden in einer Menge vereinigt
  - Voraussetzung: Gleiche Spalten (und Datentypen)
- Differenz
- Schnitt

**Kartesisches Produkt**

Mitarbeiter	Name	Vorname	Kunden	Name	Vorname
	Wurst	Hans		Albers	Willi
	Wurst	Ida		Boehrs	Vera
	Vetter	Jan		Dinkel	Ulrike

```
SELECT * FROM mitarbeiter
UNION
SELECT * FROM kunden;
```

Name	Vorname
Albers	Willi
Boehrs	Vera
Dinkel	Ulrike
Wurst	Hans
Wurst	Ida
Vetter	Jan

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 59

59

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
  - Eliminieren der Tupel in A, die auch in B sind
  - Voraussetzung: Gleiche Spalten (und Datentypen)
- Schnitt

**Kartesisches Produkt**

Personen	Name	Vorname	Kunden	Name	Vorname
	Albers	Willi		Albers	Willi
	Boehrs	Vera		Boehrs	Vera
	Dinkel	Ulrike		Dinkel	Ulrike
	Wurst	Hans			
	Wurst	Ida			
	Vetter	Jan			

```
SELECT * FROM personen
EXCEPT
SELECT * FROM kunden;
```

Name	Vorname
Wurst	Hans
Wurst	Ida
Vetter	Jan

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 60

60

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)
  - Gemeinsame Tupel in A und B
  - Voraussetzung: Gleiche Spalten (und Datentypen)

**Kartesisches Produkt**

Kunden	Name	Vorname	Mitarbeiter	Name	Vorname
	Wurst	Ida		Albers	Willi
	Vetter	Jan		Boehrs	Vera
	Dinkel	Ulrike		Dinkel	Ulrike
	Wurst	Hans		Wurst	Hans

```
SELECT * FROM kunden
INTERSECT
SELECT * FROM mitarbeiter;
```

Name	Vorname
Dinkel	Ulrike
Wurst	Hans

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 61

61

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

```
SELECT *
FROM
R1, R2;
```

R1	S1	S2
A	B	
B	C	
C	D	
D	E	

R2	S3	S4
1	2	
2	3	
3	4	

S1	S2	S3	S4
A	B	1	2
A	B	2	3
A	B	3	4
B	C	1	2
B	C	2	3
B	C	3	4
C	D	1	2
C	D	2	3
C	D	3	4
D	E	1	2
D	E	2	3
D	E	3	4

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 62

62

**Auswählen von Tupeln mit SQL** BHT

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

**SELECT \***  
**FROM**  
**R1, R2;**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 63

63

**Auswählen von Tupeln mit SQL** BHT

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

**SELECT \***  
**FROM**  
**R1, R2;**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 64

64

**Auswählen von Tupeln mit SQL** BHT

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

**SELECT \***  
**FROM**  
**R1, R2;**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 65

65

**Auswählen von Tupeln mit SQL** BHT

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

**SELECT \***  
**FROM**  
**R1, R2;**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 66

66

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

```
SELECT *
FROM R1, R2;
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 67

67

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

```
SELECT *
FROM R1, R2;
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 68

68

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

```
SELECT *
FROM R1, R2;
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 69

69

### Auswählen von Tupeln mit SQL

**Mengenoperationen**

- Vereinigung (UNION)
- Differenz (EXCEPT)
- Schnitt (INTERSECT)

**Kartesisches Produkt**

- Kombination jedes Elements von Menge A mit jedem Element von B

```
SELECT *
FROM R1, R2;
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 70

70

## Auswählen von Tupeln mit SQL

BHT

### Mengenoperationen

- Vereinigung
- Differenz
- Schnitt

### Kartesisches Produkt

71

## Weiterführendes SQL zum Auswählen von Tupeln

BHT

### SELECT ... WHERE <Bedingung>

- Bedingungen mit NOT und verknüpft mit AND und OR möglich
- Bedingungen können Prüfung umfassen
  - gleich =
  - ungleich <> bzw. !=
  - größer als > bzw. kleiner als <
  - größer gleich >= bzw. kleiner gleich <=
  - nicht größer als !> bzw. nicht kleiner als !<
- Beispiele
  - ```
SELECT name FROM kunden
  WHERE plz >= 12000 AND plz <= 15000
     AND name='Thomas';
```

72

## Weiterführendes SQL zum Auswählen von Tupeln

BHT

### SELECT ... WHERE <Bedingung>, weitere Operatoren für die Bedingung

- IN-Operator
  - ```
SELECT * FROM kunden
  WHERE name IN ('Wurst', 'Vetter', 'Dinkel');
```
- BETWEEN-Operator
  - ```
SELECT * FROM bestellungen
  WHERE gesamtpreis BETWEEN 500 AND 1000;
```
- LIKE-Operator
  - ```
SELECT * FROM kunden
  WHERE name LIKE 'Dink';
```
- NULL-Operator
  - ```
SELECT * FROM kunden
  WHERE email IS NOT NULL;
```

73

## Weiterführendes SQL zum Auswählen von Tupeln

BHT

### Alias für Tabellen verwenden

```
SELECT a.*, u.* FROM artikel AS a, umsatz AS u
```

### Sortieren der Ergebnismenge (absteigend, engl. descending)

```
SELECT b.gesamt, b.datum FROM bestellungen as b
ORDER BY b.gesamt DESC;
```

### Berechnungen in Abfragen und Sortierung (aufsteigend, engl. ascending)

```
SELECT einzelpreis, stueckzahl, einzelpreis *
  stueckzahl AS gesamt
FROM bestellpositionen
ORDER BY gesamt ASC;
```

### Entfernen doppelter Einträge (z.B. gleiche Names-Kombinationen bei Kunden)

```
SELECT DISTINCT name, vorname FROM kunden;
```

74

## Überblick über die SELECT-Anweisung<sup>1)</sup>

BÄT

```
SELECT[DISTINCT]  
    <Spaltenname>  
    <Berechnung>  
    <Konstante>  
    <einer der obigen Ausdrücke> As Alias  
    [, weitere der obigen Ausdrücke]  
FROM <Ausdruck, der Tabelle zurückgibt> As Alias  
[WHERE ...]  
[GROUP BY ...]  
[HAVING ...]  
[UNION [ALL]]  
[Weitere SELECT-Anweisung, welche dieselbe Zahl  
von  
    Spalten und Datentypen liefert]  
[ORDER BY [Order-By-Ausdruck] ASC | DESC]  
[, weitere Sortierungen]
```

<sup>1)</sup>nach [1]

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

75

75

## Arbeiten mit Relationen und Tupeln per SQL

BÄT

### Auswählen

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

### Einfügen

- neue Tupel werden zur Relation hinzugefügt

### Ändern

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

### Löschen

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

76

76

## Arbeiten mit Relationen und Tupeln per SQL

BÄT

### Auswählen

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

### Einfügen

- neue Tupel werden zur Relation hinzugefügt

### Ändern

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

### Löschen

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

77

77

## Einfügen von Tupeln mit SQL

BÄT

### SQL-Syntax ähnlich der natürlichen Sprache (in Englisch), d.h. es kann bspw. formuliert werden

- "Füge in Tabelle Kunden die Werte 345, 'Simon', 'Jakob' ein."
- "Füge in Tabelle Kunden für die Spalte Vorname, Name, Kundennummer die Werte 'Simon', 'Jakob', 345 ein."

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

78

78

### Einfügen von Tupeln mit SQL BÄT

**Einfügen in vorgegebener Spaltenreihenfolge (Bsp. 1):**

- Deutsch: "Füge in Tabelle Kunden die Werte 345, 'Simon', 'Jakob' ein."
- Englisch: "Insert into table *Kunden* values 345, 'Simon', 'Jakob'."
- SQL: `INSERT INTO kunden  
VALUES (345, 'Simon', 'Jakob');`

Ausgangsrelation

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Boehrs | Vera    |

`INSERT INTO kunden  
VALUES (345, 'Simon', 'Jakob');`

Ergebnisrelation

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Boehrs | Vera    |
|        | 345   | Simon  | Jakob   |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 79

79

### Einfügen von Tupeln mit SQL BÄT

**Einfügen in abweichender Spaltenreihenfolge (Bsp. 1):**

- Deutsch: "Füge in Tabelle Kunden für die Spalte Vorname, Name, Kundennummer die Werte 'Simon', 'Jakob', 345."
- Englisch: "Insert into table *Kunden* in columns Vorname, Name, Kundennummer values 'Simon', 'Jakob', 345."
- SQL: `INSERT INTO kunden(Vorname, Name, KndNr)  
VALUES ('Simon', 'Jakob', 345);`

Ausgangsrelation

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Boehrs | Vera    |

`INSERT INTO kunden(Vorname, Name, KndNr)  
VALUES ('Simon', 'Jakob', 345);`

Ergebnisrelation

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Boehrs | Vera    |
|        | 345   | Jakob  | Simon   |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 80

80

### Weiterführendes SQL zum Einfügen von Tupeln BÄT

**Wenn Primärschlüssel auf Relation (als AutoWert) definiert, muss beim INSERT kein Wert für den Schlüssel angegeben werden**

```
INSERT INTO personen (vorname, name)
VALUES ('Barbara', 'Beispiel');
```

**Kombination von INSERT INTO mit dem Ergebnis einer Auswahl mittels SELECT**

```
INSERT INTO personen (vorname, name)
SELECT vorname, name FROM kunden
WHERE ort='Berlin';
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 81

81

### Einfügen von Tupeln mit SQL BÄT

**Hinweis**

**Wenn LE online, dann hier Beispiele zeigen**

- Einfügen einer neuen Bestellung für einen vorhandenen Kunden
- Einfügen einer neuen Bestellung für einen neuen Kunden

**Anschließend weiter mit Übung Ü6.1 (Teil 3 inkl. Fortsetzung)**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 82

82

## Überblick über die INSERT-Anweisung<sup>1)</sup>

BHT

### Entweder

```
INSERT INTO  
  <Tabellenname>(<Spaltenname> [,weitere Spalte])  
VALUES  
  (<Wert für die erste Spalte> [, weitere Werte])
```

### Oder

```
INSERT INTO  
  <Tabellenname>(<Spaltenname> [, weitere Spalte])  
SELECT <erste Spalte> [, weitere Ausgaben]  
  [FROM ...]  
  [WHERE ...]  
  [GROUP BY ...]  
  [HAVING ...]
```

<sup>1)</sup> siehe [1]

83

## Arbeiten mit Relationen und Tupeln per SQL

BHT

### Auswählen

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

### Einfügen

- neue Tupel werden zur Relation hinzugefügt

### Ändern

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

### Löschen

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

84

## Arbeiten mit Relationen und Tupeln per SQL

BHT

### Auswählen

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

### Einfügen

- neue Tupel werden zur Relation hinzugefügt

### Ändern

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

### Löschen

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

85

## Ändern von Tupeln mit SQL

BHT

### SQL-Syntax ähnlich der natürlichen Sprache (in Englisch), d.h. es kann bspw. formuliert werden

- "Ändere in der Tabelle Kunden den Wert der Spalte Name in 'Albers', wo die Kundennummer '234' ist."

86

## Ändern von Tupeln mit SQL

BHT

### Änderungen (Bsp. 1):

- Deutsch: "Ändere in der Tabelle *Kunden* den Wert der Spalte *Name* in 'Albers', wo die *Kundennummer* '234' ist."
- Englisch: "Update in table *Kunden* value of column *Name* to 'Albers' where *Kundennummer* is '234'."
- SQL: **UPDATE kunden SET name='Albers' WHERE kndnr=234;**

Ausgangsrelation

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Boehrs | Vera    |
|        | 345   | Muster | Michael |

UPDATE kunden SET name='Albers'  
WHERE kndnr=234;

Ergebnisrelation

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Albers | Vera    |
|        | 345   | Muster | Michael |

87

## Überblick über die UPDATE-Anweisung<sup>1)</sup>

BHT

```
UPDATE <Tabelle>  
  SET <Name einer Spalte> = <Ausdruck aus Spalten,  
    Konstanten, Funktionen>  
    [, weitere Spaltennamen = Ausdruck]  
  WHERE <Bedingung>
```

<sup>1)</sup> siehe [1]

88

## Ändern von Tupeln mit SQL

BHT

### Hinweis

#### Wenn LE online, dann hier Beispiele zeigen:

- Die Bestellung mit dem Primärschlüsselwert 6 ist geliefert worden
- Erhöhung des Preises alle Bestellungen, die geliefert werden um 5 EUR (für Lieferkosten)

Anschließend weiter mit Übung Ü6.1 (Teil 4 inkl. Fortsetzung)

89

## Arbeiten mit Relationen und Tupeln per SQL

BHT

### Auswählen

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

### Einfügen

- neue Tupel werden zur Relation hinzugefügt

### Ändern

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

### Löschen

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

90

## Arbeiten mit Relationen und Tupeln per SQL BHT

**Auswählen**

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

**Einfügen**

- neue Tupel werden zur Relation hinzugefügt

**Ändern**

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

**Löschen**

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 91

91

## Löschen von Tupeln mit SQL BHT

**SQL-Syntax ähnlich der natürlichen Sprache (in Englisch), d.h. es kann bspw. formuliert werden**

- "Lösche alles aus der Tabelle Kunden."
- "Lösche aus der Tabelle Kunden wo der Name 'Albers' ist."

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 92

92

## Ändern von Tupeln mit SQL BHT

**Alle Zeilen löschen (Bsp. 1):**

- Deutsch: "Lösche alles aus der Tabelle *Kunden*."
- Englisch: "Delete all from table *Kunden*."
- SQL: **DELETE FROM kunden;**

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Boehrs | Vera    |
|        | 345   | Muster | Michael |

DELETE FROM kunden;



| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Boehrs | Vera    |
|        | 345   | Muster | Michael |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 93

93

## Ändern von Tupeln mit SQL BHT

**Alle Zeilen, die einer Bedingung genügen löschen (Bsp. 2):**

- Deutsch: "Lösche aus der Tabelle *Kunden* wo der *Name* 'Albers' ist."
- Englisch: "Delete from table *Kunden* where *Name* is 'Albers'."
- SQL: **DELETE FROM kunden WHERE name='Albers';**

| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 123   | Albers | Willi   |
|        | 234   | Albers | Vera    |
|        | 345   | Muster | Michael |

DELETE FROM kunden  
WHERE name='Albers';



| Kunden | KndNr | Name   | Vorname |
|--------|-------|--------|---------|
|        | 345   | Muster | Michael |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 94

94

## Überblick über die DELETE-Anweisung<sup>1)</sup>

```
DELETE FROM <Tabelle>  
[WHERE <Bedingung>];
```

<sup>1)</sup> siehe [1]

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 95

95

## Ändern von Tupeln mit SQL

### Hinweis

**Wenn LE online, dann hier Beispiele**

- Löschen einer Bestellung
- Löschen eines Kunden
- Löschen aller Bestellungen eines Kunden

**Anschließend weiter mit Übung Ü6.1 Teil 5 inkl. Fortsetzung**  
**Dann weiter bei Folie Nr. 100 und dann weiter bei Nr. 121**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 96

96

## Arbeiten mit Relationen und Tupeln per SQL

### Auswählen

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

### Einfügen

- neue Tupel werden zur Relation hinzugefügt

### Ändern

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

### Löschen

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 97

97

## Arbeiten mit Relationen und Tupeln per SQL

### Auswählen

- vorhandene Tupel werden gelesen
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel bereitgestellt werden

### Einfügen

- neue Tupel werden zur Relation hinzugefügt

### Ändern

- vorhandene Tupel werden geändert
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

### Löschen

- vorhandene Tupel werden gelöscht
- ggf. nach bestimmten Kriterien festgelegt, welche Tupel zu ändern sind

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 98

98

**Inhalt** BHT

**Ziel und Einordnung**

**Rückblick**

- Beziehungen im Relationalen Modell
- Fremdschlüssel
- Integritätsregeln
- Operationen auf Relationen und Tupeln

**Grundlagen des Relationalen Datenmodell (Teil 3) - SQL**

- Bestandteile
- Sprachumfang zum Auswählen, Einfügen, Ändern und Löschen
- Zusammenfassung

**Arbeiten mit dem Relationalen Modell (Teil 3) - SQL in MS Access**

- Formulare und SQL-Auswahlabfragen
- Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL
- Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen
- Zusammenfassung

**Ausblick**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 99

99

**Zusammenfassung** BHT

**SQL (Structured Query Language) als Datenbanksprache**

**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): dient zum Erzeugen, Verändern und Löschen der Strukturen, die für die Speicherung der Daten benutzt werden (z.B. Tabellen, Spalten)
- Data Control Language (DCL): dient zum Einrichten, Festlegen und Entziehen von Zugriffsrechten für Benutzer und Gruppen auf den Strukturen und auf Aktionsmöglichkeiten der Datenbank

*Thema von LE 11*

*Thema von LE 11*

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 100

100

**Inhalt** BHT

**Ziel und Einordnung**

**Rückblick**

- Beziehungen im Relationalen Modell
- Fremdschlüssel
- Integritätsregeln
- Operationen auf Relationen und Tupeln

**Grundlagen des Relationalen Datenmodell (Teil 3) - SQL**

- Bestandteile
- Sprachumfang zum Auswählen, Einfügen, Ändern und Löschen
- Zusammenfassung

**Arbeiten mit dem Relationalen Modell (Teil 3) - SQL in MS Access**

- Formulare und SQL-Auswahlabfragen
- Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL
- Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen
- Zusammenfassung

**Ausblick**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 101

101

**SQL in MS Access** BHT

**SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen**

**SQL SELECT-Abfragen als Datenquelle für Formulare**

**Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 102

102

### SQL in MS Access BHT

SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen

- SELECT
- INSERT
- UPDATE
- DELETE

SQL SELECT-Abfragen als Datenquelle für Formulare

Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 103

103

### Grafische Abfragen und SQL-Abfragen BHT

MS Access stellt dem Sprachumfang von SQL vergleichbare grafische Abfragen bereit

- Auswahlabfrage für Restriktion, Projektion und Join
  - entspricht: SELECT
  - vgl. letzte LE
- Anfügeabfrage: entspricht INSERT
- Aktualisierungsabfrage: entspricht UPDATE
- Löschartabfrage: entspricht DELETE



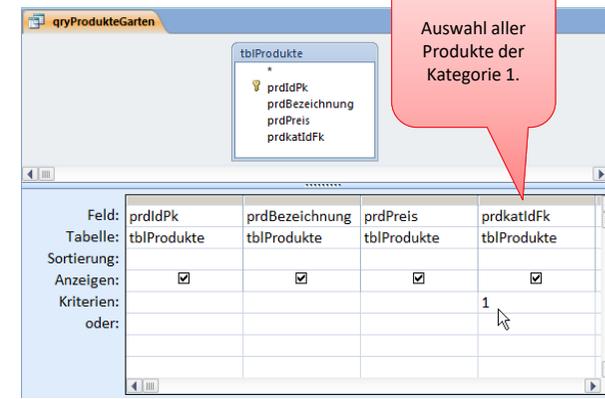
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 105

105

### SQL-Ansicht grafischer Abfragen BHT

Jede Abfrage kann in mehreren Ansicht dargestellt werden

- Entwurfsansicht



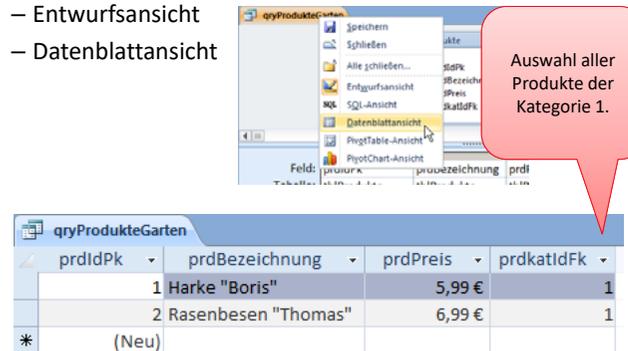
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 106

106

### SQL-Ansicht grafischer Abfragen BHT

Jede Abfrage kann in mehreren Ansicht dargestellt werden

- Entwurfsansicht
- Datenblattansicht



| prdIdPk | prdBezeichnung      | prdPreis | prdkatIdFk |
|---------|---------------------|----------|------------|
| 1       | Harke "Boris"       | 5,99 €   | 1          |
| 2       | Rasenbesen "Thomas" | 6,99 €   | 1          |
| *       | (Neu)               |          |            |

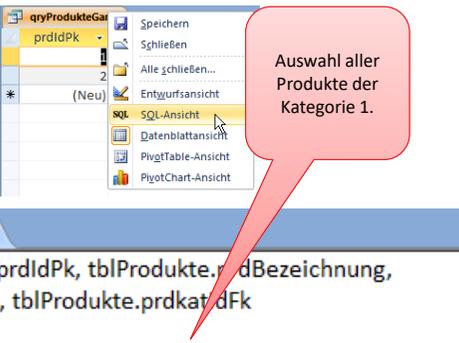
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 107

107

### SQL-Ansicht grafischer Abfragen

Jede Abfrage kann in mehreren Ansicht dargestellt werden

- Entwurfsansicht
- Datenblattansicht
- SQL-Ansicht



```
SELECT tblProdukte.prldPk, tblProdukte.prdBezeichnung,
tblProdukte.prdPreis, tblProdukte.prdkatIdFk
FROM tblProdukte
WHERE (((tblProdukte.prdkatIdFk)=1));
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 108

108

### SQL-Ansicht grafischer Abfragen

SQL-Ansicht steht auch für Anfüge-, Aktualisierungs- und Löschartfragen zur Verfügung

Beispiel

- Aktualisierung der Kategorienbezeichnung "Haushalt" in "Haushalt/Balkon"



```
UPDATE tblKategorien SET katBezeichnung = "Haushalt/Balkon"
WHERE ((tblKategorien.katBezeichnung)="Haushalt");
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 109

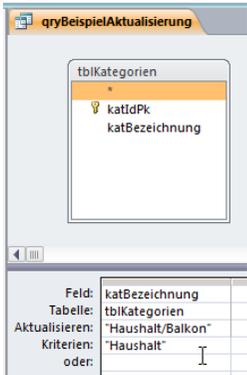
109

### SQL-Ansicht grafischer Abfragen

SQL-Ansicht steht auch für Anfüge-, Aktualisierungs- und Löschartfragen zur Verfügung

Beispiel

- Aktualisierung der Kategorienbezeichnung "Haushalt" in "Haushalt/Balkon"



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 110

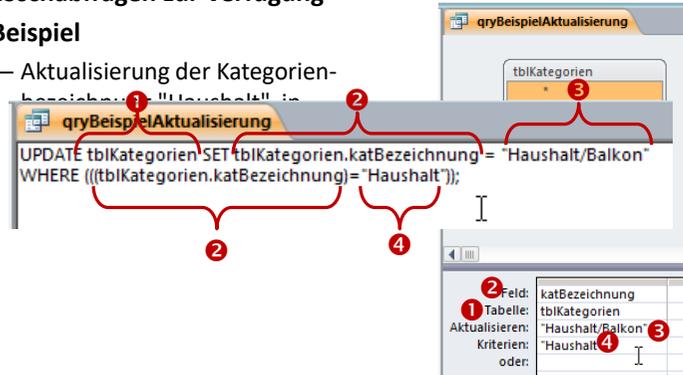
110

### SQL-Ansicht grafischer Abfragen

SQL-Ansicht steht auch für Anfüge-, Aktualisierungs- und Löschartfragen zur Verfügung

Beispiel

- Aktualisierung der Kategorienbezeichnung "Haushalt" in "Haushalt/Balkon"



```
UPDATE tblKategorien SET katBezeichnung = "Haushalt/Balkon"
WHERE ((tblKategorien.katBezeichnung)="Haushalt");
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 111

111

### SQL-Ansicht grafischer Abfragen

SQL-Ansicht steht auch für Anfüge-, Aktualisierungs- und Löschartfragen zur Verfügung

**Beispiel**

- Aktualisierung der Kategorienbezeichnung "Haushalt" in "Haushalt/Balkon"

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 112

112

### SQL-Ansicht grafischer Abfragen

SQL-Ansicht steht auch für Anfüge-, Aktualisierungs- und Löschartfragen zur Verfügung

**Beispiel**

- Aktualisierung der Kategorienbezeichnung "Haushalt" in "Haushalt/Balkon"

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 113

113

### SQL-Ansicht grafischer Abfragen

SQL-Ansicht steht auch für Anfüge-, Aktualisierungs- und Löschartfragen zur Verfügung

**Beispiel**

- Aktualisierung der Kategorienbezeichnung "Haushalt" in "Haushalt/Balkon"

| katIdPk | katBezeichnung  | Zum |
|---------|-----------------|-----|
| 1       | Garten          |     |
| 2       | Haushalt/Balkon |     |
| *       | (Neu)           |     |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 114

114

### SQL-Ansicht grafischer Abfragen

Verbund (Join) in der Entwurfs- und SQL-Ansicht von Auswahlabfragen

| Feld:       | bstIdPk                             | bstDatum                            | adrStrasseNr                        | adrPlzOrt                           |
|-------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Tabelle:    | tblBestellungen                     | tblBestellungen                     | tblLieferAdresser                   | tblLieferAdresser                   |
| Sortierung: |                                     |                                     |                                     |                                     |
| Anzeigen:   | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Kriterien:  |                                     |                                     |                                     |                                     |
| oder:       |                                     |                                     |                                     |                                     |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 115

115

### SQL-Ansicht grafischer Abfragen

Verbund (Join) in der Entwurfs- und SQL-Ansicht von Auswahlabfragen

The dialog box 'Verknüpfungseigenschaften' shows the configuration for a join between 'tblLieferAdressen' (left) and 'tblBestellungen' (right). The 'Linker Spaltenname' is 'adrIdPK' and the 'Rechter Spaltenname' is 'bstIdPK'. The radio button for '1: Behält nur die Datensätze, bei denen die Inhalte der verknüpften Felder beider Tabellen gleich sind.' is selected. A red callout box labeled 'Left Outer Join' points to the '1' radio button.

| Feld:       | bstIdPK                             | bstDatum                            | adrStrasseNr                        | adrPlzOrt                           |
|-------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Tabelle:    | tblBestellungen                     | tblBestellungen                     | tblLieferAdresser                   | tblLieferAdresser                   |
| Sortierung: |                                     |                                     |                                     |                                     |
| Anzeigen:   | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Kriterien:  |                                     |                                     |                                     |                                     |
| oder:       |                                     |                                     |                                     |                                     |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 116

116

### SQL-Ansicht grafischer Abfragen

Verbund (Join) in der Entwurfs- und SQL-Ansicht von Auswahlabfragen

The dialog box 'Verknüpfungseigenschaften' shows the configuration for a join between 'tblLieferAdressen' (left) and 'tblBestellungen' (right). The 'Linker Spaltenname' is 'adrIdPK' and the 'Rechter Spaltenname' is 'bstIdPK'. The radio button for '3: Behält ALLE Datensätze aus 'tblBestellungen' und nur die Datensätze aus 'tblLieferAdressen', bei denen die Inhalte der verknüpften Felder beider Tabellen gleich sind.' is selected. A red callout box labeled 'Right Outer Join' points to the '3' radio button.

| Feld:       | bstIdPK                             | bstDatum                            | adrStrasseNr                        | adrPlzOrt                           |
|-------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Tabelle:    | tblBestellungen                     | tblBestellungen                     | tblLieferAdresser                   | tblLieferAdresser                   |
| Sortierung: |                                     |                                     |                                     |                                     |
| Anzeigen:   | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Kriterien:  |                                     |                                     |                                     |                                     |
| oder:       |                                     |                                     |                                     |                                     |

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 117

117

### SQL-Ansicht grafischer Abfragen

Verzicht auf grafische Abfragen, jetzt kommt SQL!

#### Generelle Vorgehensweise

1. Erstellen eines neuen Abfrageentwurfs
2. Schließen des Dialogs zum Hinzufügen von Tabellen
3. Sofortiges Wechseln in SQL-Ansicht
4. Erstellen der SQL-Anweisung bzw. SQL-Abfrage
5. Speichern als Abfrage (Präfix: qry)
6. Ausführen der Abfrage
  - Klick auf Symbol für Ausführen (bei SQL-Anweisungen)
  - Wechsel in Datenblattansicht (bei SQL-Abfragen)

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 118

118

### SQL-Ansicht grafischer Abfragen: Demo D06.03

#### D06.03

- Erstellen von drei SQL-Anweisungen
  1. zum Hinzufügen eines neuen Benutzers
  2. zum Ändern des Passworts des hinzugefügten Benutzers
  3. zum Löschen des geänderten Benutzers
- Vorgehensweise
  - Erstellen eines neuen Abfrageentwurfs
  - Schließen des Dialogs zum Hinzufügen von Tabellen
  - Sofortiges Wechseln in SQL-Ansicht
  - Speichern als Abfrage
  - Ausführen der Abfrage
- Warum kann die Anweisung Nr. 1 nicht mehrfach hintereinander ausgeführt werden?

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 119

119

**SQL in MS Access** BHT

SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen

- SELECT
- INSERT
- UPDATE
- DELETE

SQL SELECT-Abfragen als Datenquelle für Formulare

Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 120

120

**SQL in MS Access** BHT

SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen

SQL SELECT-Abfragen als Datenquelle für Formulare

Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 121

121

**SQL in MS Access** BHT

SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen

SQL-Abfragen als Datenquelle für Formulare

- nur SELECT

Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 122

122

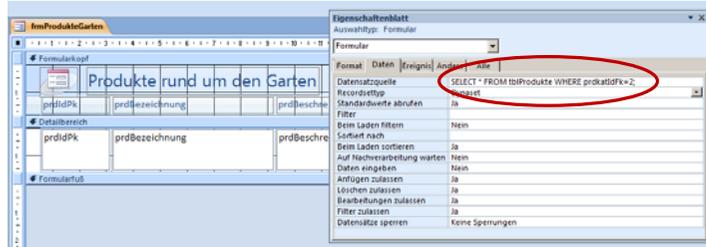
**SQL-Abfragen als Datenquelle für Formulare** BHT

**Schritt 1**

- SQL-Abfrage überlegen

**Schritt 2**

- Formular erstellen
- SQL-Abfrage als Datensatzquelle eintragen



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 123

123

## SQL-Abfragen als Datenquelle für Formulare

BHT

### Schritt 1

- SQL-Abfrage überlegen

### Schritt 2

- Formular erstellen
- SQL-Abfrage als Datensatzquelle eintragen  
oder  
SQL-Abfrage per Programmierung als Datensatzquelle festlegen

```
' Datensatzquelle als Eigenschaft des aktuellen Formulars  
' kann so verändert werden  
Me.RecordSource = "SELECT * FROM tblProdukte " & _  
                  "WHERE prdkatIdFk=" & bytAktKategorie & ";"  
  
' Aktualisierung des Formulars (Neuladen) erforderlich  
Me.Requery
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

124

124

## SQL in MS Access: Demo 06.01

BHT

### D06.01 (Teil 1)

- Vorhanden ist das Formular eines Produktkatalogs, dass
  - bisher alle Produkte zeigt und
  - die Datensatzquelle tblProdukte verwendet
- Es soll zukünftig nur die Produkte der Kategorie 4 (Lebensmittel) zeigen
- Wie muss die SQL-Abfrage lauten?
- Ändern der Datensatzquelle, um die SQL-Abfrage zu verwenden
- Prüfen des Formularinhalts (nur noch Lebensmittel)
- ...

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

125

125

## SQL in MS Access: Demo 06.01

BHT

### D06.01 (Teil 2)

- ...
- Formular im Kopfbereich um zwei Schaltflächen "Alle Produkte" und "Getränke" erweitern
- zwei SQL-Abfragen überlegen
  - alle Produkte auswählen
  - nur Getränke (Produktkategorie 4) auswählen
- Ereignisprozedur für Klicken anlegen
  - Variable für SQL-Abfrage deklarieren (String)
  - Variable mit SQL-Abfrage initialisieren
  - Datensatzquelle des Formulars die SQL-Anweisung zuweisen
- Formular erproben: Umschalten zwischen allen Produkten und nur Getränken möglich

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

126

126

## SQL in MS Access

BHT

### SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen

- SELECT
- INSERT
- UPDATE
- DELETE

### SQL SELECT-Abfragen als Datenquelle für Formulare Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML

127

127

### SQL in MS Access BHT

SQL-Abfragen als Datenquelle für Formulare  
Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL  
Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 128

128

### SQL in MS Access BHT

SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen  
SQL-Abfragen als Datenquelle für Formulare  
Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL

- INSERT
- UPDATE
- DELETE
- kein SELECT!

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 129

129

### Vordefinierte VBA-Funktionen für SQL BHT

Einfügen, Ändern und Löschen mit SQL durch Nutzung von

```
' Standardfunktionen aufrufen, als Parameter  
' String mit SQL-Abfrage übergeben  
Call CurrentDb.Execute ("<SQL-Anweisung>")  
Call DoCmd.RunSQL ("<SQL-Anweisung>")
```

Um Auswirkungen in einem Formular sichtbar zu machen

```
' Aktualisierung des Formulars (Neuladen)  
' erforderlich  
Call Me.Requery
```

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 130

130

### Vordefinierte VBA-Funktionen für SQL BHT

Beispiel

- Löschen des aktuellen Warenkorb-Eintrags über neue Schaltfläche
- ID des aktuellen Eintrags kann mit Me.<FeldMitPrimärschlüssel> abgefragt werden
- Wird in SQL-Anweisung "eingebaut" und Anweisung wird ausgeführt
- Neuladen des Formulars notwendig

```
Option Compare Database  
Option Explicit  
  
Private Sub btnLoeschen_Click()  
CurrentDb.Execute ("DELETE FROM tblWarenkoerbe WHERE wkbidPk = " & Me.txtWkbIdPk)  
Me.Requery  
End Sub
```

Hinweis: Nur zur einfachen Demonstration wird hier SQL direkt aus der Oberfläche ausgeführt. Eine saubere Trennung in Schichten ist empfehlenswert.  
Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 131

131

**Vordefinierte VBA-Funktionen für SQL: Demo 06.02** BHT

**D06.02**

- Schaltfläche zum Löschen von Produkten im Formular des Warenkorbs einfügen
- Bei Klicken soll der in der aktuellen Zeile des Formulars dargestellte Kunde gelöscht werden
  - SQL-Abfrage überlegen und
  - in Ereignisprozedur ausführen

132

**SQL in MS Access** BHT

**SQL in der SQL-Ansicht von grafischen "Abfragen" zum Auswählen, Einfügen, Ändern und Löschen**

**SQL-Abfragen als Datenquelle für Formulare**

**Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL**

- INSERT
- UPDATE
- DELETE
- kein SELECT!

133

**SQL in MS Access** BHT

**SQL-Abfragen als Datenquelle für Formulare**

**Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL**

**Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen**

134

**Inhalt** BHT

**Ziel und Einordnung**

**Rückblick**

- Beziehungen im Relationalen Modell
- Fremdschlüssel
- Integritätsregeln
- Operationen auf Relationen und Tupeln

**Grundlagen des Relationalen Datenmodell (Teil 3) - SQL**

- Bestandteile
- Sprachumfang zum Auswählen, Einfügen, Ändern und Löschen
- Zusammenfassung

**Arbeiten mit dem Relationalen Modell (Teil 3) - SQL in MS Access**

- Formulare und SQL-Auswahlabfragen
- Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL
- Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen
- Zusammenfassung

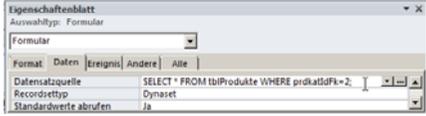
**Ausblick**

135

**Zusammenfassung** BHT

**SQL-Abfragen können als Datenquelle für Formulare verwendet werden**

**Einfügen, Ändern und Löschen mit SQL mit vordefinierter VBA-Funktionen**



**Generelle Syntax**

Call `CurrentDB.Execute (<strSQL-Anweisung>)`

**Grafische Abfragen und SQL**

- werden von MS Access in SQL übersetzt
- SQL-Ansicht zeigt das generierte SQL
- Ausführung SQL ist direkt über den entsprechenden Abfragetyp in der SQL-Ansicht möglich

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 136

136

**Inhalt** BHT

**Ziel und Einordnung**

**Rückblick**

- Beziehungen im Relationalen Modell
- Fremdschlüssel
- Integritätsregeln
- Operationen auf Relationen und Tupeln

**Grundlagen des Relationalen Datenmodell (Teil 3) - SQL**

- Bestandteile
- Sprachumfang zum Auswählen, Einfügen, Ändern und Löschen
- Zusammenfassung

**Arbeiten mit dem Relationalen Modell (Teil 3) - SQL in MS Access**

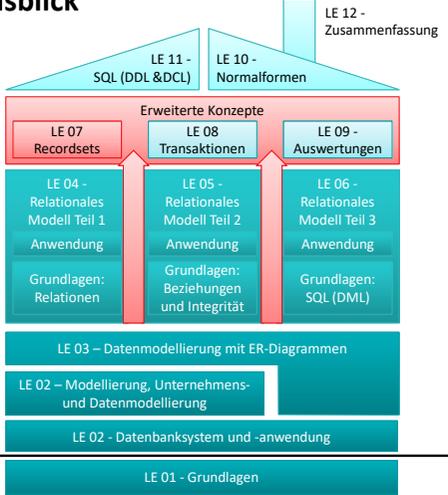
- Formulare und SQL-Auswahlabfragen
- Vordefinierte VBA-Funktionen zum Einfügen, Ändern und Löschen mit SQL
- Zusammenhang und Abgrenzung grafischer Abfragen von SQL-Abfragen
- Zusammenfassung

**Ausblick**

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 137

137

**Ausblick** BHT



Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 138

138

**Quellen** BHT

**Literatur**

- [1] Jürgen Auer: Sql-Tutorial. <http://www.sql-und-xml.de/sql-tutorial/>

**Abbildungen**

- [1] Computer History Museum: Hall of Fellows, Donald Chamberlin;  
<http://www.computerhistory.org/fellowawards/hall/bios/DonaId,Chamberlin/>
- [2] Homepage of Don Chamberlin;  
<http://www.almaden.ibm.com/cs/people/chamberlin/>

Wirtschaftsinformatik 2 - LE 05 - Relationales Modell (Teil 3): SQL DML 139

139



**Wirtschaftsinformatik 2**  
**LE 06 – Relationales Modell (Teil 3)**  
**SQL**

Prof. Dr. Thomas Off

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