



Universität Münster  
Institut für Wirtschaftsinformatik

Lehrstuhl für Wirtschaftsinformatik  
und Interorganisationssysteme  
Prof. Dr. Stefan Klein

[www.wi-ios.de](http://www.wi-ios.de)  
[mail@wi-ios.de](mailto:mail@wi-ios.de)

## EWI Übungsveranstaltung 6: Datenbanken III

24. November 2004



## Kontrollfragen

---

- Was versteht man unter referentieller Integrität?
- Wie kann referentielle Integrität zerstört werden?
- Nennt 3 Beispiele für „Aktualisierungsweitergabe an Detailfeld“!
- Warum sind Löschweitergabe und Aktualisierungsweitergabe nicht in allen Fällen erwünscht? Beispiele?
- Welche Vorteile bietet die Definition von Beziehungen in Access?
- Wie lassen sich n:m-Beziehungen in Datenbanken realisieren?

# Agenda

---

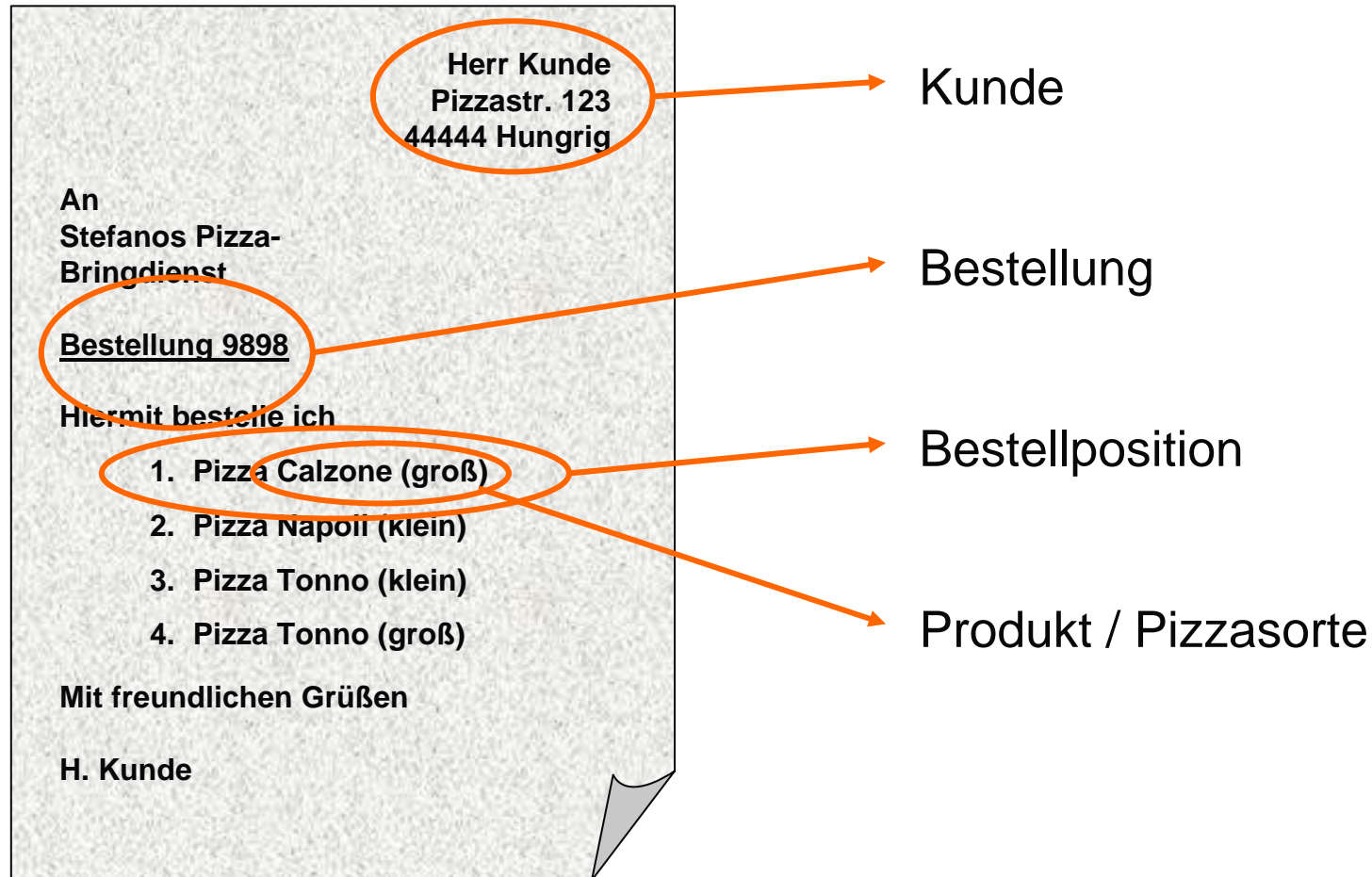
1. Aufgabenblatt 5

2. Normalisierungsprozess

3. Aggregatfunktionen

# Hinweis zu Aufgabenblatt 5: Aufbau einer Bestellung

1. Blatt 5
2. Normalisierung
3. Aggregatfkt.



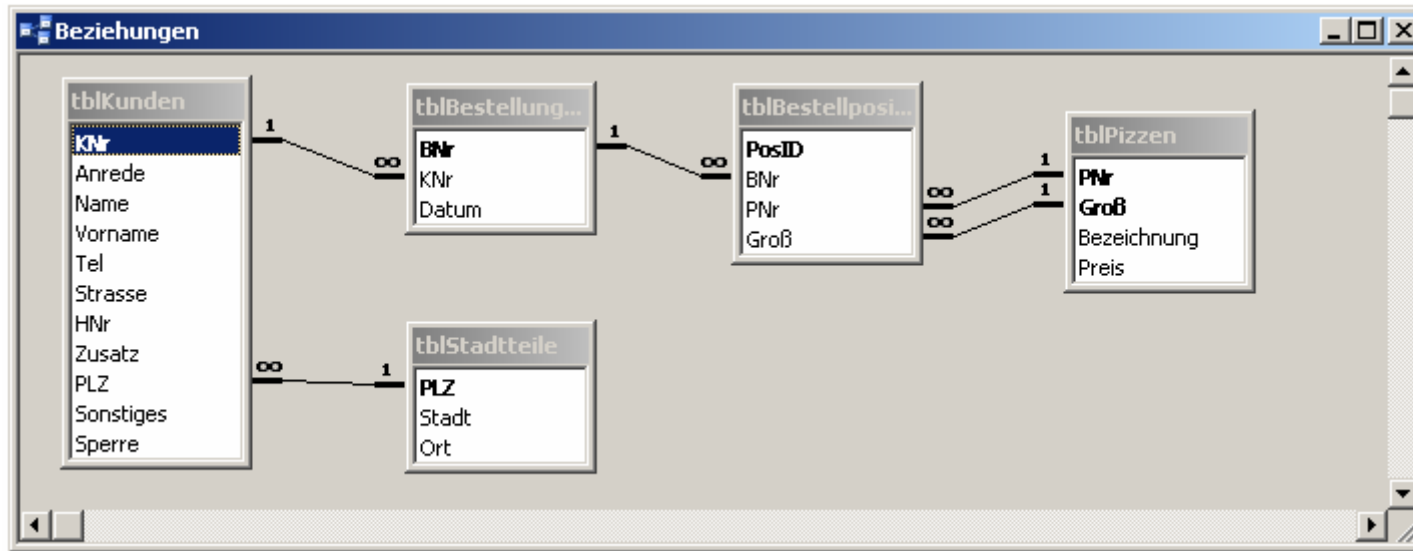
→ Überlegen:  
Welche Beziehungen bestehen zwischen Kunde,  
Bestellung, Bestellposition und Pizzasorte?

# Aufgabe 17: DB mit Pizzen, Bestellungen, Bestellpositionen, Kunden, Stadtteilen

1. Blatt 5

2. Normalisierung

3. Aggregatfkt.



## Beachten

- Geeignete Primärschlüssel wählen
- Tabelle Pizzen hat einen zusammengesetzten Schlüssel
- Bestell-Datum und -zeit können in einem Feld gespeichert werden

## Aufgabe 18: Dateneingabe

---

### Aufgabe 18:

- Eingabe von: mind. 3 Kunden  
5 Bestellungen  
7 Bestellpositionen
- Dabei: eine Pizza von mehreren Kunden  
eine weitere mehrmals in einer Bestellung
- Abfrage: Umsätze für Kunde im Jahr 2004
- Darauf aufbauend:  
Weihnachtsgeschenk (für Kunden mit Umsatz über 30 €) inklusive  
Adressausgabe

## Aufgabe 18: Unterdatenblatt

- Darstellung zusammenhängender oder verknüpfter Daten
- Bei 1:1-Beziehung oder 1:n-Beziehung: automatische Erstellung von Unterdatenblättern in Tabellen (Voraussetzung: *Unterdatenblattname: [automatisch]*)
- Einblenden über Klick auf **Erweiterungssymbol**

tblBestellungen : Tabelle			
	Bestellungs ID	Kunde	Bestellt am
▶	18	9	05.01.2004
	PosID	Pizza ID	Groß
▶	1	2	<input checked="" type="checkbox"/>
	2	11	<input checked="" type="checkbox"/>
	4	12	<input checked="" type="checkbox"/>
	15	11	<input checked="" type="checkbox"/>
	16	8	<input type="checkbox"/>
*	(AutoWert)		<input type="checkbox"/>
+	20	2	03.03.2004
+	21	12	04.05.2004

- Weiteres Beispiel für Einsatz von Unterdatenblättern: in Unterformularen

# Aufgabe 18: Bestellformular

- 1. Blatt 5
- 2. Normalisierung
- 3. Aggregatfkt.

**Stefano's Bestellungen**

Bestellungs ID:

Kunde:

Bestellt am:

Firma Lehrstuhl WI (9)  
0251-838122  
Leonardo-Campus 3  
48149 Münster

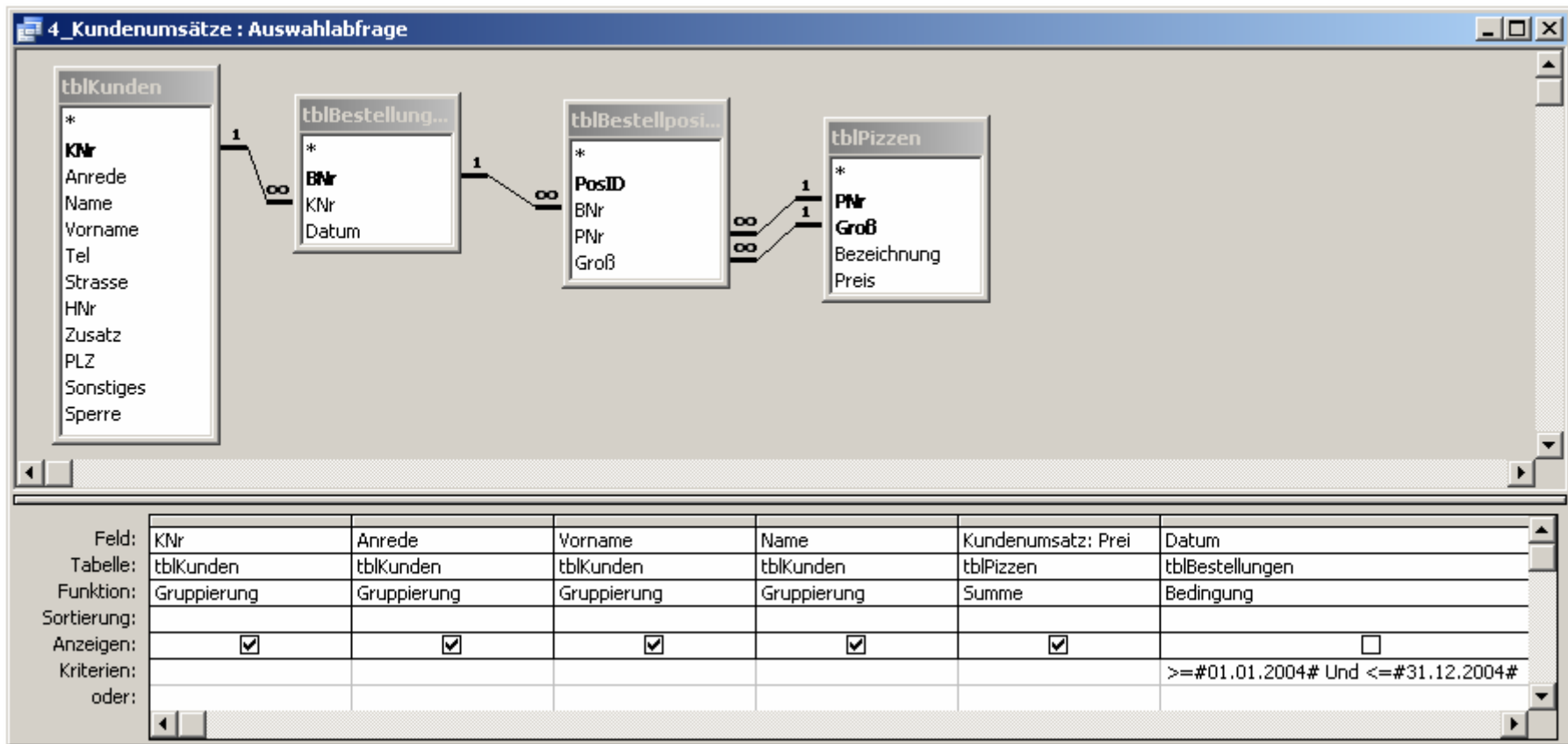
Bestellpositionen:

Pizza	Groß?	Preis	
<input type="checkbox"/> Pizza Salami	<input checked="" type="checkbox"/>	<input type="text" value="6,95 €"/>	<input type="button" value="🗑"/>
<input type="checkbox"/> Pizza Calzone	<input checked="" type="checkbox"/>	<input type="text" value="9,95 €"/>	<input type="button" value="🗑"/>
<input type="checkbox"/> Pizza Prosciutto	<input checked="" type="checkbox"/>	<input type="text" value="6,95 €"/>	<input type="button" value="🗑"/>
<input type="checkbox"/> Pizza Calzone	<input checked="" type="checkbox"/>	<input type="text" value="9,95 €"/>	<input type="button" value="🗑"/>
<input type="checkbox"/> Pizza Vegetaria	<input type="checkbox"/>	<input type="text" value="5,95 €"/>	<input type="button" value="🗑"/>
* <input type="text"/>	<input type="checkbox"/>	<input type="text"/>	<input type="button" value="🗑"/>

Datensatz:      von 8

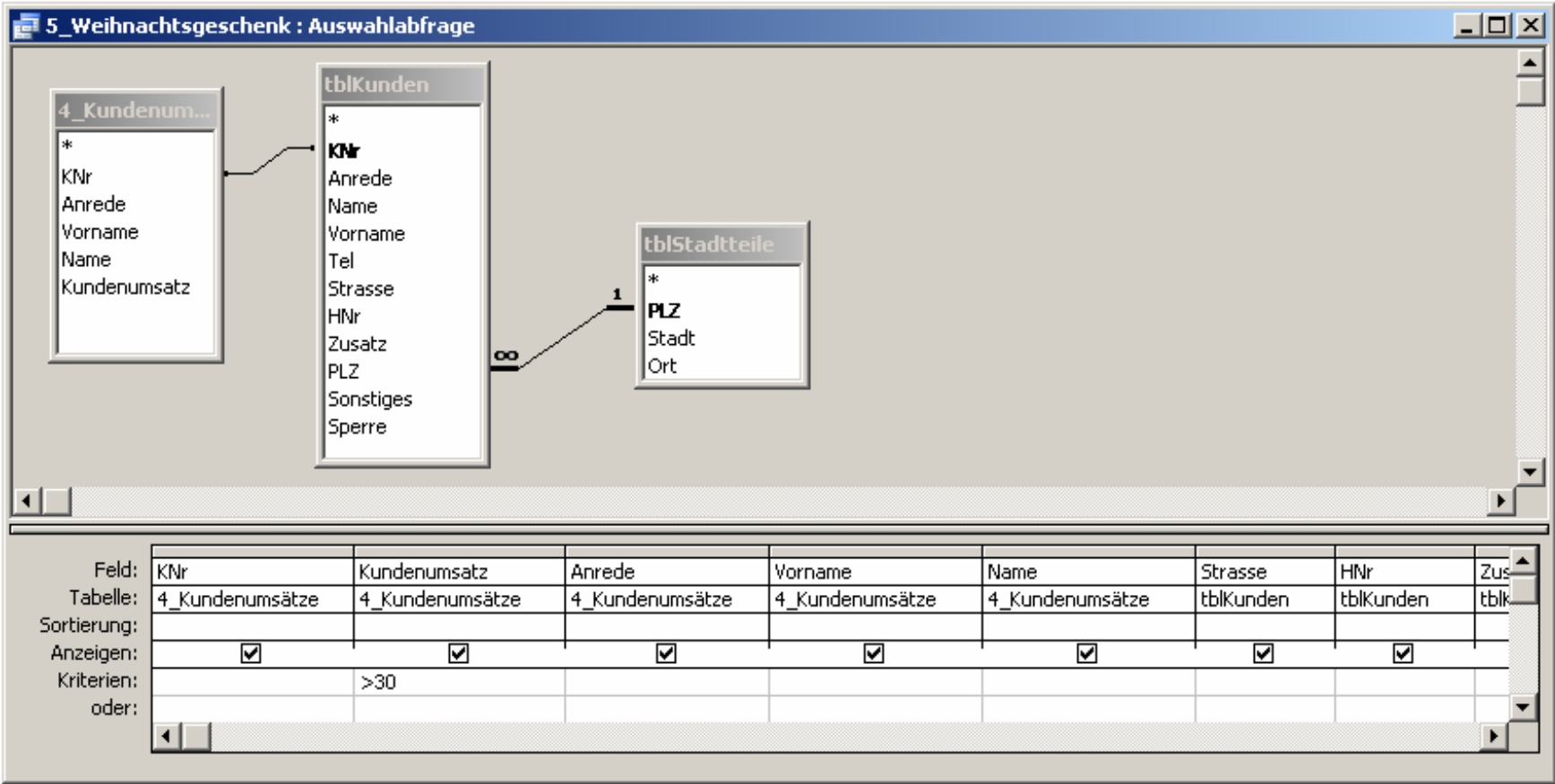
- 1. Blatt 5
- 2. Normalisierung
- 3. Aggregatfkt.

# Aufgabe 18: Kundenumsätze 2004



- 1. Blatt 5
- 2. Normalisierung
- 3. Aggregatfkt.

# Aufgabe 18: Weihnachtsgeschenk



# Agenda

---

1. Aufgabenblatt 5

2. Normalisierungsprozess

3. Aggregatfunktionen

1. Blatt 5
2. Normalisierung
3. Aggregatfkt.

## Normalisierungsprozess

---

- Regeln, um Daten zu strukturieren, bevor sie in Datenbank überführt werden → „Normalisierung“
  - **Ziele**
    - Redundanzen in Nicht-Schlüsselattributen vermeiden
    - Konsistenz der Datenbank bei Änderungen erhalten
- ➔ Weitere Details in der Vorlesung Datenmanagement (3. Semester)

## Relationales Datenbankmodell

- Ausgangsdaten mit zwei Datensätzen 12345 und 54321:

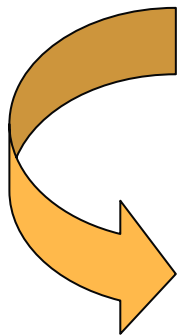
SNr	SName	VNr	VName	Note	DNr	DName
12345	Müller	1	EWI	2.3	1	Klein
		2	CSCW	1.7	2	Teubner
		3	KomSys	2.0	1	Klein
54321	Meier	1	EWI	2.7	1	Klein
		2	CSCW	2.0	2	Teubner
		4	DM	1.7	3	Becker

- Problem Redundanzen
  - Erhöhter Speicherplatzbedarf
  - Verlängerung der Datensuche
  - Inkonsistenzen bei der Änderung von Daten

# Erste Normalform

- Sich wiederholende Felder bzw. Feldgruppen in eigene Relation überführen (funktionale Abhängigkeit)
- Vom ursprünglichen Datensatz bleiben nur der Schlüssel und die nicht zur Wiederholungsgruppe gehörenden Felder über

SNr	SName	VNr	VName	Note	DNr	DName
12345	Müller	1	EWI	2.3	1	Klein
		2	CSCW	1.7	2	Teubner
		3	KomSys	2.0	1	Klein
54321	Meier	1	EWI	2.7	1	Klein
		2	CSCW	2.0	2	Teubner
		4	DM	1.7	3	Becker



**R. Student**

<u>SNr</u>	SName
12345	Müller
54321	Meier

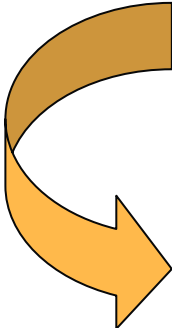
**R. Student-Vorlesung-Dozent-Zuord.**

<u>SNr</u>	<u>VNr</u>	VName	Note	DNr	DName
12345	1	EWI	2.3	1	Klein
12345	2	CSCW	1.7	2	Teubner
12345	3	KomSys	2.0	1	Klein
54321	1	EWI	2.7	1	Klein
54321	2	CSCW	2.0	2	Teubner
54321	4	DM	1.7	3	Becker

# Zweite Normalform

- Eigene Relationen für die Attribute bilden, die durch einen Teil des Schlüssels identifiziert werden (volle funktionale Abhängigkeit)
- Relation in 1. NF mit einem Schlüsselattribut ist automatisch in 2. NF

<u>SNr</u>	<u>VNr</u>	VName	Note	DNr	DName
12345	1	EWI	2.3	1	Klein
12345	2	CSCW	1.7	2	Teubner
12345	3	KomSys	2.0	1	Klein
54321	1	EWI	2.7	1	Klein
54321	2	CSCW	2.0	2	Teubner
54321	4	DM	1.7	3	Becker

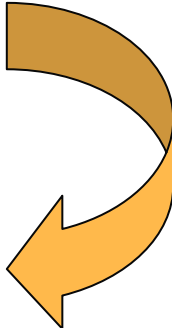


**R. SV-Zuord.**

<u>SNr</u>	<u>VNr</u>	Note
12345	1	2.3
12345	2	1.7
12345	3	2.0
54321	1	2.7
54321	2	2.0
54321	4	1.7

**R. Vorlesung-Dozent-Zuord.**

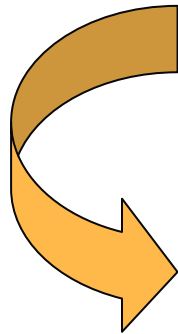
<u>VNr</u>	VName	DNr	DName
1	EWI	1	Klein
2	CSCW	2	Teubner
3	KomSys	1	Klein
4	DM	3	Becker



## Dritte Normalform

- Eigene Relationen für die Attribute bilden, die durch ein Nichtschlüsselattribut identifiziert werden (keine transitive Abhängigkeiten)
- Relation in 2. NF mit einem Nichtschlüsselattribut ist automatisch in 3. NF

<u>VNr</u>	VName	DNr	DName
1	EWI	1	Klein
2	CSCW	2	Teubner
3	KomSys	1	Klein
4	DM	3	Becker



**R. Vorlesung**

<u>VNr</u>	VName	DNr
1	EWI	1
2	CSCW	2
3	KomSys	1
4	DM	3

**R. Dozent**

<u>DNr</u>	DName
1	Klein
2	Teubner
3	Becker



1. Blatt 5
2. Normalisierung
3. Aggregatfkt.

# Ergebnis des Normalisierungsprozesses

**R. Student**

<u>SNr</u>	SName
12345	Müller
54321	Meier

**R. SV-Zuord.**

<u>SNr</u>	<u>VNr</u>	Note
12345	1	2.3
12345	2	1.7
12345	3	2.0
54321	1	2.7
54321	2	2.0
54321	4	1.7

**R. Vorlesung**

<u>VNr</u>	VName	DNr
1	EWI	1
2	CSCW	2
3	KomSys	1
4	DM	3

**R. Dozent**

<u>DNr</u>	DName
1	Klein
2	Teubner
3	Becker

- Wie erhält man wieder die originäre Relation?

# Agenda

---

1. Aufgabenblatt 5

2. Normalisierungsprozess

3. Aggregatfunktionen

1. Blatt 5
2. Normalisierung
3. Aggregatfkt.

# Aggregatfunktionen

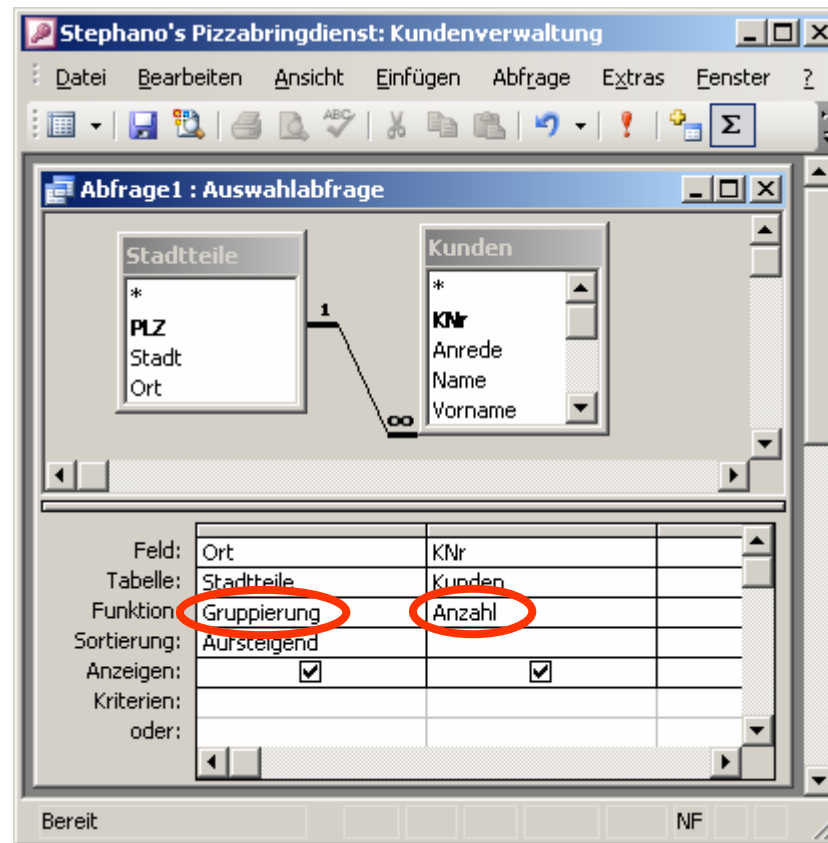
---

## Idee

- Zusammenfassung mehrerer zusammengehörige Datensätze in einer Abfrage (z.B. alle Pizzen, die ein Kunde bestellt hat)  
→ In der Entwurfsansicht einer Abfrage  $\Sigma$ -Zeichen aktivieren
- Die Abfrage-Definition wird daraufhin um die Möglichkeit von Aggregatfunktionen erweitert, z.B.:
  - Gruppierung: Zusammenfassung gleicher Werte
  - Anzahl: Anzahl der Einträge (ohne Null-Werte)
  - Summe: Summe der Werte eines numerischen Feldes
  - Mittelwert: Mittelwert der Werte eines numerischen Feldes
  - Min bzw. Max: Minimum bzw. Maximum

# Beispiel: Anzahl Kunden pro Ort

1. Blatt 5
2. Normalisierung
3. Aggregatfkt.



# Funktionsweise von Aggregatfunktionen

In der Datenbank

- Eine Bestellung von Kunde Nr. 1 (Meyer)
- Bestellpositionen: 2 große & 1 kleine Calzone, 1 große Salami

Query configuration:

Feld:	KNr	Name	BNr
Tabelle:	Kunden	Kunden	Bestellungen
Funktion:	Gruppierung	Gruppierung	Anzahl
Sortierung:			Absteigend
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Result table:

Kunden ID	Name	Anzahl von BNr
1	Meyer	1

Query configuration:

Feld:	KNr	Name	BNr
Tabelle:	Kunden	Kunden	Bestellungen
Funktion:	Gruppierung	Gruppierung	Anzahl
Sortierung:			Absteigend
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Result table:

Kunden ID	Name	Anzahl von BNr
1	Meyer	4

➔ Unterschiedliche Ergebnisse, obwohl gleiche Felder ausgewählt!

# Funktionsweise von Aggregatfunktionen

1. Blatt 5
2. Normalisierung
3. Aggregatfkt.

## 1. Zusammenführen der Tabellen:

<u>KNr</u>	Name	...	...
1	Meyer	...	...
...	...	...	...

<u>BNr</u>	KNr	Datum
15	1	1.1.2002
...	...	...



K.KNr	K.Name	K. ...	B.BNr	B.Knr	B.Datum
1	Meyer	...	15	1	1.1.2002
...	...	...	...	...	...

2. Gruppierung
3. Berechnung der Anzahl je Gruppe: *Anzahl (B.Bnr)*
4. Darstellung der ausgewählten Spalten

# Funktionsweise von Aggregatfunktionen

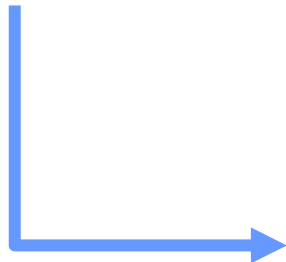
1. Blatt 5
2. Normalisierung
3. Aggregatfkt.

## 1. Zusammenführen der Tabellen:

<u>KNr</u>	Name	...	...
1	Meyer	...	...
...	...	...	...

<u>BNr</u>	KNr	Datum
15	1	1.1.02
...	...	...

<u>BNr.</u>	PNr	Groß
15	11	✓
15	11	✓
15	11	
15	2	✓
...	...	...



K.KNr	K.Name	K. ...	B.BNr	B.Knr	B.Datum	P.Bnr	P.Pnr	P.Groß
1	Meyer	...	15	1	1.1.02	15	11	✓
1	Meyer	...	15	1	1.1.02	15	11	✓
1	Meyer	...	15	1	1.1.02	15	11	
1	Meyer	...	15	1	1.1.02	15	2	✓
...	...	...	...	...	...	...	...	...

## 2. Gruppierung

3. Berechnung der Anzahl je Gruppe: *Anzahl (B.Bnr)*

4. Darstellung der ausgewählten Spalten

## Kontakt

---

Marcel Gogolin  
wimago@wi.uni-muenster.de  
+49 (0)251 83-38 122  
Raum 201

Universität Münster  
Institut für Wirtschaftsinformatik

Lehrstuhl für Wirtschaftsinformatik und  
Interorganisationssysteme (IOS)  
Prof. Dr. Stefan Klein  
Leonardo-Campus 3  
D-48149 Münster

Tel.: +49 (0)251 83-38110  
Fax: +49 (0)251 83-38119

<http://www.wi-ios.de>