

# NORMALISIERUNG

I. – 3. Normalform

# NORMALISIERUNG???

- = Aufteilung von Attributen in mehrere Relationen (Tabellen)
- Redundanzfreie Datenspeicherung
- Entfernung von Anomalien
- 5 Normalformen
- Dritte Normalform = Balance aus Redundanz, Performance & Flexibilität

# ZIELE DER NORMALISIERUNG

- Beseitigung von Redundanzen
- Vermeidung von Anomalien (funktionelle und transitive Abhängigkeiten)
- Erstellung eines klar strukturierten Datenbankmodells

# ANOMALIEN

- Insert Anomalie
  - Es kann passieren, dass Daten gar nicht in die Datenbank übernommen werden, wenn zum Beispiel der **Primärschlüssel keinen Wert erhalten hat**, oder eine **unvollständigen Eingabe von Daten** zu Inkonsistenzen führt.
- Update Anomalie
  - Es kann passieren, dass gleiche Attribute eines Datensatzes in einer Transaktion **nicht automatisch geändert werden**. So entsteht eine Inkonsistenz der Daten.
- Delete Anomalie
  - Es kann passieren, dass ein Benutzer einer Datenbank **aktiv Informationen löschen will** und damit indirekt, aufgrund des fehlerhaften Datenbankdesigns, andere **zusammenhängende Informationen parallel mitlöscht**.

# ABHÄNGIGKEITEN

- Funktionale Abhängigkeit
  - Zu jedem X genau ein Y gibt
- Voll funktionale Abhängigkeit
  - Nichtschlüsselattribut sind von allen Teilen eines Relationstyps funktional abhängig
- Transitive Abhängigkeit
  - Y von X funktional abhängig und Z von Y = Z von X funktional Abhängig

# I. NORMALFORM (INF)

- Unsortierte/unstrukturierte Informationen werden einheitlich und klar strukturiert

R.-Nr.	Datum	Name	Straße	Ort	Artikel	Anzahl	Preis
187	01.01.2012	Max Mustermann	Musterstr. 1	12345 Musterort	Bleistift	5	1,00 €



R.-Nr.	Datum	Name	Vorname	Straße	Hnr.	PLZ	Ort	Artikel	Anzahl	Preis	Währung
187	01.01.2012	Mustermann	Max	Musterstr.	1	12345	Musterort	Bleistift	5	1,00	Euro

## 2. NORMALFORM (2NF)

- Prüft ob funktionale- oder voll funktionale Abhängigkeit
- Meistens durch ER-Diagramm indirekt erreicht
- 1NF + Nichtschlüsselattribut voll funktional Abhängig von Schlüsselkandidaten

R.-Nr.	Datum	Name	Vorname	Straße	Hnr.	PLZ	Ort	Artikel	Anzahl	Preis	Währung
187	01.01.2012	Mustermann	Max	Musterstr.	1	12345	Musterort	Bleistift	5	1,00	Euro

Rechnung		
R.-Nr.	Datum	Knr.
187	01.01.2012	007

Kunde						
Knr.	Name	Vorname	Straße	Hnr.	PLZ	Ort
007	Mustermann	Max	Musterstr.	1	12345	Musterort

Rechnungsposition			
R.-P.-Nr.	R.-Nr.	Art.-Nr.	Anzahl
1	187	69	5

Artikel		
Art.-Nr.	Artikel	Preis
69	Bleistift	1,00



Rechnung		
R.-Nr.	Datum	Knr.
187	01.01.2012	007

  

Kunde						
Knr.	Name	Vorname	Straße	Hnr.	PLZ	Ort
007	Mustermann	Max	Musterstr.	1	12345	Musterort

  

Rechnungsposition			
R.-P.-Nr.	R.-Nr.	Art.-Nr.	Anzahl
1	187	69	5

  

Artikel		
Art.-Nr.	Artikel	Preis
69	Bleistift	1,00

- Untertabelle für Spalten die von Schlüsselkand. nicht vollständig funktional abhängig sind.
- Primärschlüssel = Spalte mit funktionaler Abhängigkeit
- Kunde: Nachname nicht eindeutig -> Knr. (Primärschl.)
- Artikel: Artikel nicht eindeutig -> Art.-Nr. (Primärschl.)
- RPT: Eine Rechnung -> Vielzahl an Positionen mit versch. Artikeln



### 3. NORMALFORM (3NF)

- Verhindert Anomalien und Redundanzen
- Bietet genug Performance für SQL-Abfragen
- 2NF + kein Nichtschlüsselattribut transitiv von nicht Kandidatenschlüssel abhängt

Kunde						
Knr.	Name	Vorname	Straße	Hnr.	PLZ	Ort
007	Mustermann	Max	Musterstr.	1	12345	Musterort

Kunde					
Knr.	Name	Vorname	Straße	Hnr.	PLZ
007	Mustermann	Max	Musterstr.	1	12345

Postleitzahl	
PLZ	Ort
12345	Musterort



Kunde						
Knr.	Name	Vorname	Straße	Hnr.	PLZ	Ort
007	Mustermann	Max	Musterstr.	1	12345	Musterort

Kunde					
Knr.	Name	Vorname	Straße	Hnr.	PLZ
007	Mustermann	Max	Musterstr.	1	12345

Postleitzahl	
PLZ	Ort
12345	Musterort

- Kunde:
  - Vorname, Straße, PLZ = abhängig vom Attribut „Name“ -> nicht von Primärschlüssel
  - Ort abhängig von PLZ
- Transitiv abhängige Spalten -> Untertabelle, da nur indirekt von Schlüsselkandidat abhängig