

Normalisierungsformen (NF1 bis NF3)

Description

In diesem Artikel beschreibe ich die Normalisierungsformen 1 bis 3 in Relationalen Datenbanken. Die Normalisierungsformen helfen uns eine Datenbank zu konzipieren, die eine perfekte Struktur aufweist um Probleme und Inkonsistenzen in der Zukunft zu vermeiden.

Die Regeln müssen nach und nach erfüllt werden, das bedeutet es muss erst die **NF1**, dann die **NF2** und zum Schluss die **NF3** erfüllt sein. Also eine Tabelle, die sich in der **Normalisierungsform 2** befindet, hat automatisch die Anforderungen der **Normalisierungsform 1** erfüllt.

Normalform 0 (0NF)

Eine Tabelle liegt in der **Nullten Normalform** vor, wenn die Daten sich einfach in der Tabelle befinden. Das bedeutet, die Daten sind nicht atomar. Das bedeutet wir haben **Redundanzen** in der Datenbank ohne **signifikanten Informationsgewinn**. Dadurch ist unsere Tabelle durch sogenannte Anomalien gefährdet.

In dem ersten Beispiel haben wir eine Tabelle mit Kundendaten mit den Bestellungen, die diese getätigt haben. Es gibt in der Tabelle Redundanzen, z.B. **DHL => Deutsche Post DHL Group** als auch im Merkmal / Attribut **Adresse** bei der Postleitzahl mit der Ortsangabe. Im Weiteren haben wir Probleme, wenn wir z.B. Kunden nach Nachnamen sortieren wollen. Dies ist laut unserer aktuellen Tabelle nicht möglich.

Name	Adresse	Geburtstag	Bestellte Artikel
Peter Petersen	Mühlenstraße 30, 24944 Flensburg	01. Jan 1990	5x Stifte, 3x Bücher
Hans Hansen	Hauptstraße 60, 24943 Flensburg	25.06.2000	30x Eddings, 6x Kopierpapier
Hans Petersen	Apenrader Straße 90, 25001 Berlin	30.05.1999	25x Kinderriegel, 2x Radiergummis
Petra Hansen	Bundesstraße 90, 24943 Flensburg	05. Jun 1975	50x Klebezettel, 1x Collegeblock

Normalform 1 (1NF)

Die Tabelle in der **ersten Normalform** existiert, wenn folgende Bedingungen erfüllt sind:

1. Jedes Merkmal atomar ist
2. Nicht weiter unterteilbar ist
3. Listen aufgelöst sind

Es befinden sich in der Tabelle dennoch weiterhin Redundanzen, da diese noch nicht entfernt werden. In der 1. Normalisierungsform werden die Daten nur aufgeteilt, dass wir eine reine Tabelle erhalten, in dem wir die entsprechenden Daten in einzelnen Merkmalen aufgeteilt haben.

Ab wann ein Wert **atomar** ist, hängt vom Nutzungskontext ab. Wenn man gewisse Teilungen nicht benötigt, ist dies nicht unbedingt nötig. In der Praxis empfiehlt sich trotzdem alles möglichst klein aufzuteilen.

Vorname	Nachname	Straße	Hausnummer	PLZ	Ort	Geburtstag	Anzahl Artikel	Bestell
Peter	Petersen	Mühlenstraße	30	24944	Flensburg	01.01.1990	5	Stifte
Peter	Petersen	Mühlenstraße	30	24944	Flensburg	01.01.1990	3	Bücher
Hans	Hansen	Hauptstraße	60	24943	Flensburg	25.06.2000	30	Eddings
Hans	Hansen	Hauptstraße	60	24943	Flensburg	25.06.2000	6	Kopierp
Hans	Petersen	Apenrader Straße	90	10179	Berlin	30.05.1999	25	Kinderr
Hans	Petersen	Apenrader Straße	90	10179	Berlin	30.05.1999	2	Radierg
Petra	Hansen	Bundesstraße	90	24943	Flensburg	05.06.1975	50	Klebeze
Petra	Hansen	Bundesstraße	90	24943	Flensburg	05.06.1975	1	College

Normalform 2 (2NF)

In der zweiten Normalform muss die Tabelle in der ersten Normalform vorliegen, und alle Nichtschlüsselmerkmale voll funktional vom Primärschlüssel abhängen. Unter Strich bedeutet dies, dass jedes Nichtschlüsselmerkmal aus dem Primärschlüssel ableitbar ist. Merkmale, die von einem Teilschlüssel abhängig sind, müssen in eine eigene Tabelle geschrieben werden. Hier teilen wir also die Tabellen in einzelne Tabellen auf.

Um jetzt eine Tabelle in die zweite Normalform zu übernehmen, müssen wir folgende Schritte durchführen:

1. Alle Nichtschlüsselmerkmale, die von einem Teilschlüssel funktional abhängig sind, bestimmen.
2. Aus den Teilschlüsseln mit allen funktional abhängigen Nichtschlüsselmerkmalen eigene Tabellen bilden.
3. Im letzten Schritt entfernen wir alle nicht voll funktional abhängigen Nichtschlüsselmerkmale.

<i>tblKunden</i>				
<u>ID_Kunde</u>	Vorname	Nachname	Straße	Hausnummer
1	Peter	Petersen	Mühlenstraße	30
2	Hans	Hansen	Hauptstraße	60
3	Hans	Petersen	Apenrader Straße	90
4	Petra	Hansen	Bundesstraße	90
<i>tblSpediteur</i>				
<u>ID_Spediteur</u>	Spediteur	Spediteur Name		
1	DHL	Deutsche Post DHL Group		
2	Hermes	Hermes Germany GmbH		
<i>tblBestellungen</i>				
<u>ID_Bestellungen</u>	ID_Kunde	ID_Spediteur		
1	1	1		
2	2	2		
3	3	1		
4	4	1		
<i>tblBestellteArtikel</i>				
<u>ID_BestellteArtikel</u>	ID_Bestellung	Menge	Artikel	
1	1	5	Stifte	
2	1	3	Bücher	
3	2	30	Eddings	
4	2	6	Kopierpapier	
5	3	25	Kinderriegel	
6	3	2	Radiergummi	
7	4	50	Klebezettel	
8	4	1	Collegeblock	

Normalform 3 (3NF)

Im letzten Schritt entfernen wir die letzten Redundanzen. Laut der Definition ist eine Tabelle in der Normalform 3, wenn jedes Schlüsselmerkmal nicht transitiv vom Primärschlüssel abhängig ist.

Im Weiteren bedeutet **Transitiv**, dass ein Merkmal einen Umweg nutzen kann, um funktional abhängig von einem anderen Merkmal abhängig ist. Unterm Strich sollen alle Merkmale nur von einem Primärschlüssel abhängig sein.

Um eine Tabelle in die dritte Normalform zu übernehmen, verwenden wir folgende Schritte:

1. Alle Nichtschlüsselmerkmale, die transitiv vom Schlüssel abhängen, bestimmen.

2. Im nächsten Schritt sollen aus diesen transitiv abhängigen Nichtschlüsselmerkmalen und den Nichtschlüsselmerkmalen, von denen diese funktional abhängig sind, eigene Tabellen bilden.
3. Im letzten Schritt entfernen wir alle transitiv abhängigen Nichtschlüsselmerkmale aus der Ursprungstabelle.

<i>tblKunden</i>					
ID_Kunde	Vorname	Nachname	Straße	Hausnummer	PLZ
1	Peter	Petersen	Mühlenstraße	30	24944
2	Hans	Hansen	Hauptstraße	60	24943
3	Hans	Petersen	Apenrader Straße	90	10179
4	Petra	Hansen	Bundesstraße	90	24943
<i>tblSpediteur</i>					
ID_Spediteur	Spediteur	Spediteur Name			
1	DHL	Deutsche Post DHL Group			
2	Hermes	Hermes Germany GmbH			
<i>tblBestellungen</i>					
ID_Bestellungen	ID_Kunde	ID_Spediteur			
1	1	1			
2	2	2			
3	3	1			
4	4	1			
<i>tblBestellteArtikel</i>					
ID_BestellteArtikel	ID_Bestellung	Menge	Artikel		
1	1	5	Stifte		
2	1	3	Bücher		
3	2	30	Eddings		
4	2	6	Kopierpapier		
5	3	25	Kinderriegel		
6	3	2	Radiergummi		
7	4	50	Klebezettel		
8	4	1	Collegeblock		
<i>tblPlzOrt</i>					
PLZ	Ort				
10179	Berlin				
24943	Flensburg				
24944	Flensburg				

Category

1. Allgemein SQL
2. Datenbanken

Date Created

10.03.2025

Author
administrator