

## Relationale Datenbanken Normalformen

Ausgangspunkt ist die folgende Tabelle für die Personalverwaltung einer Firma.

Name	AbtNr	Abteilung	Projekte
Müller	5	EDV II	(6, MySQL, 50%), (3, DV2010, 50%)
Schulze	3	Rechenzentrum	(3, DV2010, 100%)

Man erkennt deutlich, dass es schwer ist, Teile der Daten zu löschen, einzufügen oder zu ändern. Daher überführt man solche Tabellen in die sogenannten Normalformen nach Codd.

### 1. Normalform (1NF)

Eine Relation befindet sich in der ersten Normalform, wenn alle Attribute einen *atomaren Wertebereich* haben.

**Jeder Datensatz muss durch den Primärschlüssel eindeutig identifizierbar sein.**

Das Attribut *Projekt* ist nicht atomar, mehrere Werte enthält. Außerdem fehlt ein Primärschlüssel. Wir ergänzen *PersNr* und teilen das Mehrfachattribut in seine atomaren Bestandteile auf. Damit der Primärschlüssel weiterhin eindeutig ist nehmen wir die Projektnummer in den Schlüssel auf (sonst gäbe es die *PersNr* „1“ zweimal).

<u>PersNr</u>	Name	AbtNr	Abteilung	<u>ProjNr</u>	Projekt	Zeit
1	Müller	5	EDV II	6	MySQL	50%
1	Müller	5	EDV II	3	DV2010	50%
2	Schulze	3	Rechenzentrum	3	DV2010	100%

Die Tabelle ist nun in der 1NF. Allerdings wurden bei der Umformung zusätzliche Redundanzen eingefügt.

### 2. Normalform (2NF)

Eine Relation befindet sich in der zweiten Normalform, wenn sie in der 1NF ist und jedes Nichtschlüsselattribut funktional abhängig ist vom Gesamtschlüssel, nicht dagegen von Schlüsselteilen.

*Funktional abhängig* heißt, dass die Werte des Attributs durch ein anderes eindeutig festgelegt sind. Besteht der Primärschlüssel nur aus einem Attribut ist dies immer der Fall. Im Beispiel zeigt sich, dass *Name*, *AbtNr*, und *Abteilung* nur durch *PersNr* festgelegt werden, nicht aber von *ProjNr* abhängig sind. Daher werden die drei Attribute mit dem Primärschlüssel *PersNr* in eine neue Tabelle ausgelagert. Beim Rest sieht man, dass *Projekt* eindeutig durch *ProjNr* festgelegt wird. Auch hier erstellen wir eine neue Tabelle.

<u>PersNr</u>	Name	AbtNr	Abteilung
1	Müller	5	EDV II
1	Müller	5	EDV II
2	Schulze	3	Rechenzentrum

  

<u>PersNr</u>	<u>ProjNr</u>	Zeit
1	6	50%
1	3	50%
2	3	100%

  

<u>ProjNr</u>	Projekt
6	MySQL
3	DV2010

### 3. Normalform (3NF)

Eine Relation befindet sich in der dritten Normalform, wenn sie in der 2NF ist und es kein Nichtschlüsselattribut gibt, das *transitiv abhängig* von einem Schlüsselattribut ist.

Es darf also keine funktionalen Abhängigkeiten zwischen Nichtschlüsselattributen geben.

Das Attribut *Abteilung* ist nur indirekt vom Primärschlüssel *PersNr* abhängig. Vielmehr ist das Attribut *Abteilung* von der *AbtNr* abhängig. Also Aufspaltung in zwei Tabellen:

<u>PersNr</u>	Name	AbtNr
1	Müller	5
2	Schulze	3

<u>AbtNr</u>	Abteilung
5	EDV II
3	Rechenzentrum

### Zusammenfassung

Durch die Transformation in die dritte Normalform wurde also aus der Tabelle

Name	AbtNr	Abteilung	Projekte
Müller	5	EDV II	(6, MySQL, 50%), (3, DV2010, 50%)
Schulze	3	Rechenzentrum	(3, DV2010, 100%)

diese Aufteilung vorgenommen:

<u>PersNr</u>	Name	AbtNr
1	Müller	5
2	Schulze	3

<u>PersNr</u>	<u>ProjNr</u>	Zeit
1	6	50%
1	3	50%
2	3	100%

<u>AbtNr</u>	Abteilung
5	EDV II
3	Rechenzentrum

<u>ProjNr</u>	Projekt
6	MySQL
3	DV2010