

LF05 - 22.05.24 / 25.06.24

Normalisieren nach Codd

1. Atomisieren (1NF): keine zusammengesetzten, mengnewertigen oder geschalteten Wertebereiche, Werte/Daten in kleinstmögliche Wertebereiche aufteilen
 - Vertikal: keine Zusammengesetzten Werte mit Teilbereichen unterschiedlicher Funktionalität, z.B. Vorname und Nachname getrennt, Straße und Hausnummer getrennt, PLZ und Ort getrennt, ...
 - Horizontal: nur ein Element mit gleicher Funktionalität, keine Liste von funktional gleichwertigen aber unterschiedlichen Daten in einem Datensatz, z.B. pro Artikel (Hefter, Ordner, Stift) ein eigener Datensatz
2. Vollfunktionale Abhängigkeit (2NF): Nach vollfunktionalen Abhängigkeiten der Nichtschlüsselattribute von Teilschlüsseln diese in eigene Entitäten aufspalten (z.B. Person -> Name, Adresse, ...)
 - Primärschlüssel (Primary Key) - Verwendung in relationalen Datenbanken, ist derjenige Schlüsselkandidat mit möglichst wenig Stufen und dabei die Betriebsumgebung möglichst optimal abbildet
 - Schlüsselkandidat (Candidate Key) - mit minimaler Anzahl Attribute in jedem Satz einzigartige Wertekombination
 - Ersatzschlüssel (Surrogate Key) - wird aus Daten abgeleitet bzw. ersetzt sie, z.B. das Datum
 - Superschlüssel - besteht aus einem oder mehreren Attributen und identifiziert jeden Datensatz eindeutig
3. Transitive Abhängigkeit (3NF): Nach transitiven Abhängigkeiten zwischen Nichtschlüsselattributen suchen und in weitere Entitäten zerlegen (z.B. PLZ -> Ort)
4. Redundanzen: Keine redundanten Informationen
5. Anomalien: Katalogisieren von wiederkehrenden Bezeichnern in eigenen Entitäten, ggf. Einbindung per Referenz (bei zweispaltigen Katalogen)

Anomalien

- Einfügeanomalien: beim Einfügen von neuen Datensätzen z.B. falsche oder unterschiedliche Schreibweisen
- Änderungsanomalie: bei einigen Datensätzen etwas ändern, aber nicht alle betroffenen Datensätze werden entsprechend angepasst
- Löschanomalien: löschen des letzten Satzes mit einer bestimmten Information führt zum dauerhaften Verlust (z.B. eines Ortes beim Löschen eines Kundensatzes), Referenzen bleiben bestehen

Redundanzen

- wenn Informationen zum selben Objekt mehrfach gespeichert werden

Katalog

- dient zur Vermeidung von Anomalien
- einspaltig: Daten sind ihr eigener Schlüssel, damit direkt in den referenzierenden Tabellen ablesbar
- zweispaltig: Bezeichnungen werden einem Schlüssel zugeordnet, Änderungen in der Bezeichnung müssen nur im Katalog erfolgen