Vorüberlegungen

Wir versuchen uns, in das folgende Szenario einzudenken:

- Wir sind ein Zwischenhändler, der Zahnärzte mit allem ausstattet, was die brauchen, um ihre Patienten zu foltern.
- Wir müssen Buch führen über unsere Kunden (Doktoren), unsere Lieferanten, angebotene Produkte und unsere Bestellungen.
- Unser sehr unerfahrene Sekretär hat mal was von mysql gehört und die derzeit bekannten Daten in eine Tabelle geschrieben.

Die Ausgangslage

Da unser Sekretär normal nur mit Tabellenkalkulationen arbeitet, hat er alle Infos einfach mal in eine Tabelle gepackt:

id	produkt	preis	nummer	anzahl	doktor	telefon_fax	adresse	hersteller
1	Schlagbohrer	199.95	1000-1	1	Blutgesicht, Hans	123-550,123-551	Kariesweg 1, 12345 Zahnstein	Hilti, Bohrmaschinenstr. 58a, 88887 Hilterhausen
2	Zement	5.95	1000-2	5	Quälstein, Bertha	456-777,456-778	Lochpfad 23, 23458 Schmerzstadt	Hoch & Tief, Am Höhenzug 22, 12345 Tiefenbau
3	Kneifzange	19.95	1000-3	3	Eisenfaust, Viktoria	789-250,789-251	Zahnwurzel 3, 87454 Dolomostadt	Eisen-Karl, Karlstrasse 5, 76542 Eisenhafen
4	Brecheisen	49.95	1000-4	7	Rostzange, Eberhart	234-100,234-101	Peinweg 5, 74512 Reissheim	Eisen-Karl, Karlstrasse 5, 76542 Eisenhafen
5	Hammer	19.95	1000-5	4	Frankenstein, Emilia	567-200,567-201	Am Dom 5, 50670 Köln	Stahl AG, Blechweg 102b,89452 Alustetten
6	Zement	5.95	1000-2	9	Eisenfaust, Viktoria	789-250,789-251	Zahnwurzel 3, 87454 Dolomostadt	Hoch & Tief, Am Höhenzug 22, 12345 Tiefenbau
7	Brecheisen	49.95	1000-4	2	Blutgesicht, Hans	123-550,123-551	Kariesweg 1, 12345 Zahnstein	Eisen-Karl, Karlstrasse 5, 76542 Eisenhafen

Probleme...

Auf den ersten Blick fällt auf, dass in unserer Tabelle etliche Informationen mehrfach gespeichert sind, das kann zu Problemen führen.

Im Moment sieht alles noch sehr übersichtlich aus. Aber was passiert, wenn wir 10.000 oder 100.000 Datensätze verwalten müssen? Was, wenn ein Kunde den Händler wechselt oder sich die Adresse eines Herstellers ändert? Wie kann jemand etwas bestellen, wenn nicht irgendwo ersichtlich ist, welche Produkte es überhaupt gibt?

Redundanzen, Anomalien, Inkonsistenzen

Hinter diesen Begriffen verbirgt sich alles, was den logischen Aufbau unserer Datenbank gefährden könnte.

- Von **Redundanz** spricht man, wenn Informationen mehrfach gespeichert sind. Rdundanzen führen zu einem höheren Speicherplatzbedarf und gefährden die Konsistenz der Informationen, da Änderungen häufig mehrfach vorgenommen werden müssen.
- Von **Anomalie** spricht man, wenn sich beispielsweise Datensätze in ihren Angaben "unregelmäßig" unterscheiden oder die Bedeutung von Attributen unklar ist und z.B. erst aus dem Kontext erschlossen werden kann.

- Von **Inkonsistenz** spricht man, wenn sich Informationen widersprechen.

Ein DBMS soll Fehler dieser Art eigentlich vermeiden, dazu muss die Datenbank jedoch sinnvoll konzipiert sein - was im Falle der allumfassenden Tabelle nicht der Fall ist.

(A1)

Stelle dir vor, unser Sekretär muss einen neuen Zahn-"Doktor" (Emilia Bohrgut, An der Bohrinsel 87, 23918 Diamantspitz, Telefon 763-221, Fax, 762-223) in die Datenbank einfügen.

- Wo kannst du in unserer Datenbanktabelle Redundanzen finden?
- Inwiefern stellt der neue Datensatz für Doktor Bohrgut in unserer Tabelle eine Anomalie dar?
- Ist auf den ersten Blick ersichtlich, was für eine Bedeutung die Tabellenspalte "nummer" hat? Wie kannst du das herausfinden? Warum stellt das eine Anomalie dar?
- Die Firma Eisen-Karl hat eine neue Anschrift, unser Sekretär muss diese in unserer Tabelle korrigieren.
    - Wo erschweren ihm Redundanzen diese Arbeit?
    - Warum können die Redundanzen dazu führen, dass die Daten inkonsistent werden?

Normalisierung

Unter der "**Normalisierung einer Datenbank**" kann man sich eine Sammlung von Regeln vorstellen, die eingehalten werden sollten, um die erläuterten Probleme zu vermeiden. Wir betrachten im Folgenden die ersten drei "Normalformen".

Grundsatz: Eine Entität - eine Tabelle

Ein **Grundsatz** ohne Name, den unser Sekretär mit seinem an die Tabellenkalkulation angelehnten Vorgehen bereits verletzt hat, den wir aber eigentlich kennen, lautet: **Jede Entität bekommt ihre eigene Relation, die Attribute sind die Tabellenspalten**((Erinnerung: Relation ist nur ein anderer Name für Tabelle…). Wenn man sich daran hält, muss man sich nur noch Gedanken und sinnvolle Attribute und die Möglichkeiten der Verknüpfung machen.

(A2)

- Importiere die
  Universaltabelle
  in deine Übungsdatenbank.
- Überführe die Inhalte der "Universaltabelle" in drei Tabellen: `doktoren`, `hersteller`, `produkte`. Verteile die vorhandenen Attribute auf die Tabellen der Entitäten, **verändere die Attribute nicht**, bennene Sie auch nicht um. Gehe wie folgt vor:
  1. Lege die drei Tabellen mit den entsprechenden Attributen an
  2. Überführe die Inhalte aus der Universaltabelle in die jeweilige Tabelle, indem du das Ergebnis einer Abfrage direkt an eine INSERT INTO Statement weitergibst: `INSERT INTO doktoren SELECT doktor, telefon_fax, adresse FROM `zahnarztbedarf` .` Du musst auf die Reihenfolge der Attribute bei der Abfrage achten!
- Zwei Tabellenspalten der Universaltabelle können den Entitätstabellen nicht sinnvoll zugeordet werden - bei einer geht keine wesentliche Information verloren, wenn man sie weglässt, bei der anderen schon. Welche Attribute sind das? Lasse beide Attribute vorerst einfach aus - wir kommen später darauf zurück.
- Fügen nun den Datensatz für Frau Bohrgut ein - du siehst, nun ist ganz klar, welche Informationen in welche Tabelle eingefügt werden müssen.

Lösungshinweise

Tabellenstrukturen

Überführungsabfragen
INSERT INTO produkte SELECT produkt, preis, nummer FROM `zahnarztbedarf`
INSERT INTO hersteller SELECT hersteller FROM `zahnarztbedarf`
INSERT INTO doktoren SELECT doktor, telefon_fax, adresse FROM `zahnarztbedarf`

Nicht übertragbare Attribute Das Attribut `id` ist weitgehend sinnfrei, da überhaupt nicht klar ist, was diese ID eigentlich identifizieren sollte, das kann weggelassen werden. Das Attribut `anzahl` bezieht sich auf die Produkt-Bestellungen der Zahnärzte, das passt zu keiner unserer Tabellen, weil es gewissermaßen zu `doktoren` und zu `produkte` gehört.

From:
https://www.info-bw.de/ -

Permanent link:
https://www.info-bw.de/faecher:informatik:oberstufe:datenbanken:normalisierung:vorueberlegungen:start?rev=1606315209

Last update: **25.11.2020 14:40**