NORMALIZOWANIE BAZY DANYCH

Normalizacja bazy

Normalizacja bazy danych *jest to proces mający na celu eliminację powtarzających się danych w relacyjnej bazie danych. Główna idea polega na trzymaniu danych w jednym miejscu, a w razie potrzeby linkowania do danych. Taki sposób tworzenia bazy danych zwiększa bezpieczeństwo danych i zmniejsza ryzyko powstania niespójności (w szczególności problemów anomalii).*

Postacie normalne

Istnieją sposoby ustalenia czy dany schemat bazy danych jest „znormalizowany”, a jeżeli jest to jak bardzo. Jednym ze sposobów jest przyrównanie danej bazy do schematów zwanych **postaciami normalnymi** (ang. **normal forms** lub **NF**). Normalizacja bazy danych do konkretnej postaci może wymagać rozbicia dużych tabel na mniejsze i przy każdym wykonywaniu zapytania do bazy danych ponownego ich łączenia. Zmniejsza to wydajność, więc w niektórych przypadkach świadoma denormalizacja (stan bez normalizacji) jest lepsza – zwłaszcza w systemach niekorzystających z **modelu relacyjnego** (np. OLAP).

Celem normalizacji baz danych jest unikanie anomalii.

Anomalie na przykładzie

Załóżmy, że mamy następującą strukturę bazy książek w bibliotece:

Tytuł książki	Autor	Wypożyczający	Adres wypożyczającego	Data wypożyczenia

W tej bazie wystąpią następujące anomalie:

Anomalia	Opis
przy aktualizacji	Jeżeli wypożyczający zmienił adres, trzeba przeszukać całą bazę i we wszystkich komórkach, w których występuje, zmienić ten adres
przy usuwaniu	Jeżeli wypożyczający zwróci ostatnią książkę, zostanie utracona informacja na jego temat (adres i inne dane osobowe)
przy wstawianiu	Nowa osoba nie może zapisać się do biblioteki, jeżeli nie wypożyczy książki (a nie musi od razu wypożyczać)
redundacja	Redundacja, czyli powtarzanie tej samej informacji w kilku miejscach w bazie, powoduje niepotrzebne zajmowanie pamięci (wypożyczenie dwóch książek powoduje, że niepotrzebnie adres jest powtarzany dwa razy)

Anomalie baz danych

Redundancja — ta sama informacja jest niepotrzebnie przechowywana w kilku krotkach.

Anomalia modyfikacji — informacja zostanie zmodyfikowana w pewnych krotkach, a w innych nie. Która informacja jest wówczas prawdziwa?

Anomalia usuwania — usuwanie części informacji powoduje utratę innej informacji, której nie chcielibyśmy stracić.

Anomalia dołączania — wprowadzenie pewnej informacji jest możliwe tylko wtedy, gdy jednocześnie wprowadzamy jakąś inną informację, która może być obecnie niedostępna.

Postacie normalne

Edgar Frank Codd (twórca normalizacji) początkowo wymyślił 3 postacie normalne: **1NF, 2NF i 3NF**. Obecnie istnieją jeszcze inne postacie, ale 3NF jest powszechnie uznawana za wystarczającą do większości projektów. Większość tabel spełniając postać 3NF, spełnia także BCNF (ang. Boyce-Codd normal form). **4NF i 5NF** są następnymi rozszerzeniami, a **6NF** jest używana do baz uwzględniających w modelu relacyjnym wymiar czasowy.

Pierwsza postać normalna 1NF

Mówimy, że tabela (encja) jest w **pierwszej postaci normalnej, kiedy wiersz przechowuje informacje o pojedynczym obiekcie, nie zawiera kolekcji, posiada klucz główny (kolumnę lub grupę kolumn jednoznacznie identyfikujących go w zbiorze) a dane są atomowe**.

Pierwsza postać normalna

- Wyeliminuj powtarzające się grupy w poszczególnych tabelach.
- Utwórz osobną tabelę dla każdego zestawu powiązanych danych.
- Zidentyfikuj każdy zestaw powiązanych danych za pomocą klucza podstawowego.

Druga postać normalna 2NF

2 NF - tabela powinna przechowywać dane dotyczące tylko konkretnej klasy obiektów.

Druga postać normalna

- Utwórz osobne tabele dla zestawów wartości dotyczących wielu rekordów.
- Powiąż te tabele za pomocą klucza obcego.

Trzecia postać normalna 3NF

Trzecia postać normalna głosi, że **kolumna informacyjna nie należąca do klucza nie zależy też od innej kolumny informacyjnej**, nie należącej do klucza. Czyli każdy niekluczowy argument jest bezpośrednio zależny tylko od klucza głównego a nie od innej kolumny.

Trzecia postać normalna

- Wyeliminuj pola, które nie zależą od klucza.

Przykład: Nieznormalizowany zbiór danych

Przedmiot	Id pracownika	Nazwisko pracownika	Id studenta	Student	Ocena	Typ oceny
TOiS	23	Bos	123	Botas	4	W
TOiS	23	Bos	123	Botas	4,5	Ć
TOiS	23	Bos	143	Moton	3,5	Ć
TOiS	23	Bos	134	Koton	4,5	W
TOiS	23	Bos	134	Koton	5	Ć
UA	23	Bos	321	Ficek	4	W
UA	23	Bos	321	Ficek	4,4	Ć
Angielski	34	Kusek	231	Bocek	5	Ć

Etapy normalizacji

Zebranie zbioru danych

Przekształcenie nieznormalizowanego zbioru danych w tabele w pierwszej postaci normalnej

Przekształcenie tabel z pierwszej postaci normalnej w drugą postać normalną

Przekształcenie tabel z drugiej postaci normalnej w trzecią postać normalną

Zależność funkcyjna

Dwa elementy danych A i B są w zależności funkcyjnej lub relacji zależnej, jeśli ta sama wartość elementu danych B pojawia się zawsze z tą samą wartością elementu danych A

- W takim przypadku mówimy, że atrybut A określa funkcyjnie atrybut B

Wszystkie atrybuty w tabeli są funkcyjnie zależne od klucza głównego tej tabeli.

Wszystkie dane osobowe są zależne funkcyjnie od numeru PESEL osoby

Np. W relacji pracownik:

- Zależność PESEL→ Nazwisko jest zależnością funkcjonalną ponieważ każdemu numerowi PESEL jest przyporządkowane dokładnie jedno nazwisko
- Zależność PESEL→Data_zwolnienia nie jest zależnością funkcjonalną, ponieważ jednemu numerowi PESEL może być przyporządkowanych wiele dat zwolnienia, jeśli pracownik przebywał na zwolnieniu kilkakrotnie

Pierwsza postać normalna

Relacja jest w pierwszej postaci normalnej wtedy i tylko wtedy, gdy każdy atrybut niekluczowy jest funkcyjnie zależny od klucza głównego

W pierwszym etapie normalizacji próbujemy znaleźć w relacji klucz główny – od którego wszystkie atrybuty niekluczowe byłyby funkcyjnie zależne. Jeśli nie można znaleźć klucza głównego, to relację należy podzielić

Nieznormalizowany zbiór danych

z usuniętymi powtarzającymi się danymi

Przedmiot	Id pracownika	Nazwisko pracownika	Id studenta	Student	Ocena	Typ oceny
TOiS	23	Bos	123	Botas	4	W
			123	Botas	4,5	Ć
			143	Moton	3,5	Ć
			134	Koton	4,5	W
			134	Koton	5	Ć
UA	23	Bos	321	Ficek	4	W
			321	Ficek	4,4	Ć
Angielski	34	Kusek	231	Bocek	5	Ć

Tabele w pierwszej postaci normalnej

Oceny

n

1

Przedmioty

Przedmiot	Id pracownika	Nazwisko pracownika
TOiS	23	Bos
UA	23	Bos
Angielski	34	Kusek

Przedmiot	Id studenta	Typ oceny	Student	Ocena
TOiS	123	W	Botas	4
TOiS	123	Ć	Botas	4,5
TOiS	143	Ć	Moton	3,5
TOiS	134	W	Koton	4,5
TOiS	134	Ć	Koton	5
UA	321	W	Ficek	4
UA	321	Ć	Ficek	4,4
Angielski	231	Ć	Bocek	5

Druga postać normalna

Relacja jest w drugiej postaci normalnej wtedy i tylko wtedy, gdy jest w pierwszej postaci normalnej i każdy atrybut niekluczowy jest w pełni funkcyjnie zależny od klucza głównego

W tabeli *oceny* atrybut *Student* zależy funkcyjne tylko od atrybutu *Id studenta*, czyli od części klucza głównego, a nie od całego klucza

Atrybut *Ocena* zależy funkcyjnie od całego klucza głównego

Tabele w drugiej postaci normalnej

1 Przedmioty n n Oceny 1 Studenci

Przedmiot	Id pracownika	Nazwisko pracownika
TOiS	23	Bos
UA	23	Bos
Angielski	34	Kusek

Przedmiot	Id studenta	Typ oceny	Ocena
TOiS	123	W	4
TOiS	123	Ć	4,5
TOiS	143	Ć	3,5
TOiS	134	W	4,5
TOiS	134	Ć	5
UA	321	W	4
UA	321	Ć	4,4
Angielski	231	Ć	5

Id studenta	Student
123	Botas
143	Moton
134	Koton
321	Ficek
231	Bocek

Trzecia postać normalna

Relacja jest w trzeciej postaci normalnej wtedy i tylko wtedy, gdy jest w drugiej postaci normalnej i każdy niekluczowy atrybut jest bezpośrednio zależny (a nie pośrednio zależny) od klucza głównego

W tabeli P*rzedmioty* atrybut *Nazwisko pracownika* jest zdeterminowany przez atrybut *Id pracownika*, a zatem atrybut *Nazwisko pracownika* jest przechodnio zależny od klucza głównego – atrybutu *Przedmiot*

Przejście do trzeciej postaci normalnej

Przedmioty

Przedmiot	Id pracownika	Nazwisko pracownika
TOiS	23	Bos
UA	23	Bos
Angielski	34	Kusek

Przedmioty

1

Przedmiot	Id pracownika
TOiS	23
UA	23
Angielski	34

n

Pracownicy

Id pracownika	Nazwisko pracownika
23	Bos
34	Kusek

Tabele w trzeciej postaci normalnej

1

Przedmioty

Przedmiot	Id pracownika
TOiS	23
UA	23
Angielski	34

n

1

Pracownicy

Id pracownika	Nazwisko pracownika
23	Bos
34	Kusek

n

n

Oceny

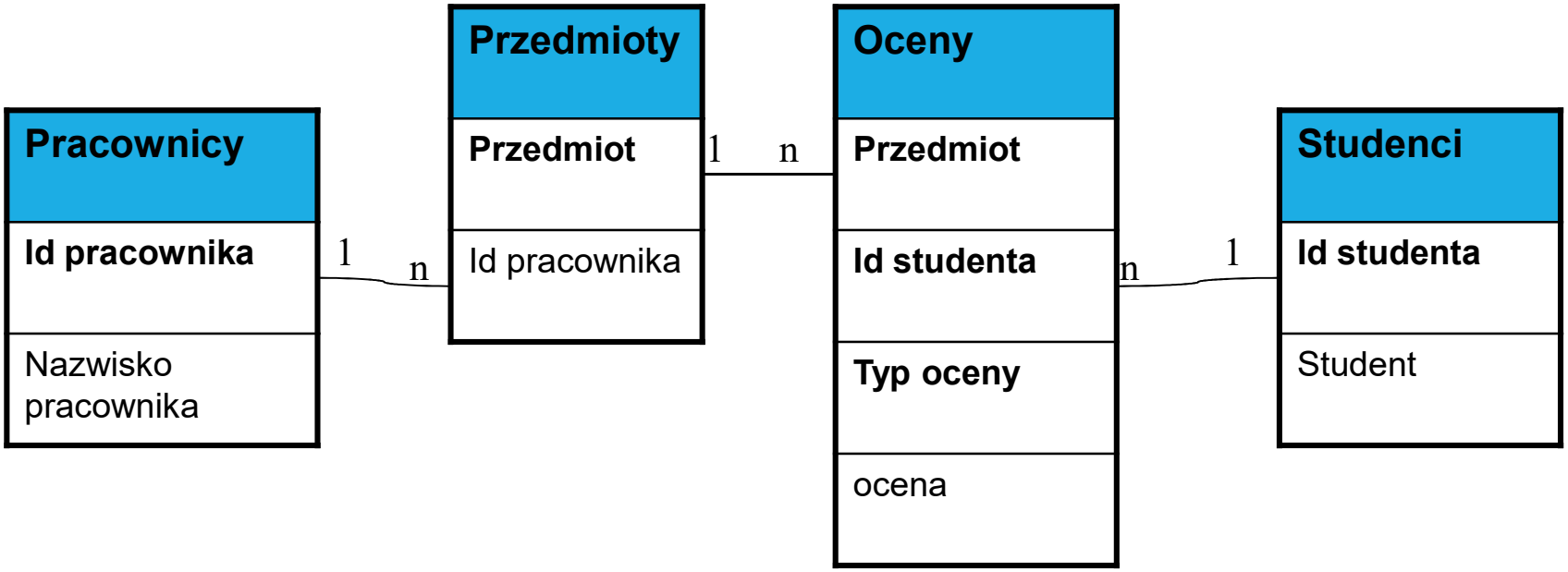
Przedmiot	Id studenta	Typ oceny	Ocena
TOiS	123	W	4
TOiS	123	Ć	4,5
TOiS	143	Ć	3,5
TOiS	134	W	4,5
TOiS	134	Ć	5
UA	321	W	4
UA	321	Ć	4,4
Angielski	231	Ć	5

1

Studenci

Id studenta	Student
123	Botas
143	Moton
134	Koton
321	Ficek
231	Bocek

Schemat


Pracownicy
Id pracownika
Nazwisko pracownika
1
n
Przedmioty
Przedmiot
Id pracownika
1
n
Oceny
Przedmiot
Id studenta
Typ oceny
ocena
n
1
Studenci
Id studenta
Student

Przysięga normalizacji ☺

Bez powtórzeń

Pola zależą od klucza

Od całego klucza

I niczego innego, tylko klucza

Przykład pola nieelementarnego

Pole ***adres*** w tabeli **OBYWATEL** może być w niektórych zastosowaniach polem atomowym lub nieatomowym (elementarnym, nieelementarnym) w zależności od zastosowań bazy danych np. w ewidencji obywateli dla celów sporządzania listy wyborców w okręgach wyborczych lub list poborowych pole to może być nieelemetarnym – może wymagać dostępu do części atrybutu adres, t.j. np. ***adres_ulica***, ***adres_nr_domu, adres_nr_mieszkania***

ZADANIE

1. Mając dane w tabeli, doprowadź do 3 NF:

A)

IMIE I NAZIWKO LEKARZA	IMIE I NAZWISKO PACJENTA	ADRES PACJENTA	PESEL PACJENTA	SPECJALIZACJA LEKARZA	DATA I GODZINA WIZYTY	NR_GABINETU

1. B)

NAZWA PRODUKTU	CENA PRODUKTU	GATUNEK PRODUKTU	IMIE I NAZWISKO KLIENTA	ADRES KLIENTA	DATA_ZAMOWIENIA	ILOSC ZAMOWIENIA

Rozwiązanie A)

Rozwiązanie B -samodzielnie

Normalizacja a wydajność

Normalizacja baz danych dostarcza **mechanizmu** pozwalającego unikać anomalii. Ma to jednak swoją cenę: Dostęp do danych w bazie znormalizowanej może być wolniejszy, gdyż RDBMS musi wykonywać złączenia.

Dlatego w wielkich bazach danych zoptymalizowanych na odczyt (na przykład w hurtowniach danych) często rezygnuje się z wyższych postaci normalnych, przechowując dane w tabelach 1PN. *To* także ma swoją cenę: Wprowadzając dane do takich tabel lub modyfikując istniejące dane należy dołożyć szczególnej staranności, aby nie dopuścić do anomalii usuwania lub dołączania, a szczególnie do anomalii modyfikacji (redundancja jest w tego typu bazy niejako wbudowana).

Relational Database Management System, RDBMS

Źródło:

https://support.microsoft.com/pl-pl/kb/283878

Umiejętności z lekcji:

- wymienić i wyjaśnić anomalie w bazach danych (może być na przykładzie)
- Ile jest wszystkich postaci normalnych?
- Wyjaśnić na czym polega 1NF, 2NF i 3NF
- Mając tabelę nieznormalizowaną przejść do 3 NF.