Amadeusz Nowak

Sebastian Mikoś

Opis przeniesienia nierelacyjnej bazy danych Do MongoDB

**Przeniesienie bazy relacyjnej do MongoDB**

Analiza struktury bazy relacyjnej

1. Przejrzenie schematu bazy danych:
   1. Zdefiniowanie wszystkich tabel, kolumn oraz kluczy głównych i obcych bazy danych.
   2. Określanie relacji między tabelami:
      1. Jeden do wielu
      2. Wiele do wielu
2. Klasyfikacja tabel:
   1. Tabele główne (np. „klienci”, „egzemplarze”).
   2. Tabele relacyjne (np. „rezerwacje\_egzemplarze”, połączenie „wiele do wielu”).

**Projektowanie struktury w MongoDB**

MongoDB opiera się na kolekcjach i dokumentach JSON, które przechowują dane w elastycznej strukturze. Należy wprowadzić odpowiednie zmiany:

1. Tabele główne → Kolekcje.
2. Rekordy w tabelach → dokumenty JSON.
3. Każda tabela główna w SQL stanie się kolekcją w MongoDB:
   1. Tabela „egzemplarze” w PostgreSQL staje się kolekcją „egzemplarze”.
   2. Tabela „klienci” w PostgreSQL staje się kolekcją „klienci”.
   3. Tabela „modele” w PostgreSQL staje się kolekcją „modele”.
   4. Tabela „pracownicy” w PostgreSQL staje się kolekcją „pracownicy”.
   5. Tabela „rezerwacje” w PostgreSQL staje się kolekcją „rezerwacje”.
   6. Tabela „wypozyczone” w PostgreSQL staje się kolekcją „wypozyczone”.
4. Przykład: Dla pierwszego rekordu w tabeli „egzemplarze”:

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

Dokumentami zostaną wszystkie rekordy. Postać pierwszego rekordu:

{

„\_id”: ObjectId („ABC123”),

„kolor”: „Czarny”,

„ID\_modelu”: ObjectId(“2”),

„rok\_produkcji”: „2019-07-05”,

„cena”: 1000,

„przebieg”: 47000,

„rezerwacje”: [ObjectId(“1”), ObjectId(“2”)]

}

Przykładowe relacje „jeden do wielu” → Zagnieżdżenie dokumentów:

1. Relacja modele → egzemplarze
2. Opis relacji: Jeden model (kolekcja modele) może mieć wiele egzemplarzy (kolekcja egzemplarze).
3. Struktura MongoDB przy użyciu referencji:
   1. Dokumenty modele zawierają unikalne identyfikatory (\_id).
   2. Dokumenty egzemplarze przechowują referencje (id\_modelu) do odpowiedniego dokumentu w kolekcji modele.
4. klienci → rezerwacje
   1. Opis relacji: Jeden klient (kolekcja klienci) może mieć wiele rezerwacji (kolekcja rezerwacje).
   2. Struktura MongoDB przy użyciu referencji:
      1. Dokumenty klienci zawierają unikalne identyfikatory (\_id lub pesel).
      2. Dokumenty rezerwacje przechowują referencje (pesel) do odpowiedniego klienta w kolekcji klienci
5. Relacja pracownicy → modele (opcjonalna)
   1. Opis relacji: Jeden pracownik (kolekcja pracownicy) może być przypisany jako opiekun do wielu modeli (kolekcja modele).
   2. Struktura MongoDB przy użyciu referencji:
      1. Dokumenty pracownicy zawierają unikalne identyfikatory (\_id).
      2. Dokumenty modele przechowują referencje (opiekun) do odpowiedniego pracownika w kolekcji pracownicy.

Przykład dla tabeli „egzemplarze”, „modele”:

{

"\_id": ObjectId("2"),

"marka": "Ford",

"model": "Focus",

"pojemnosc\_silnika": 2000,

"moc\_silnika": 150,

„opiekun”: ObjectId("2")

}

{

„\_id”: ObjectId („ABC123”),

„kolor”: „Czarny”,

„ID\_modelu”: ObjectId(“2”),

„rok\_produkcji”, „2019-07-05”,

„cena”: 1000,

„przebieg”: 47000

}

**Relacja wiele do wielu:**

Rezerwacje → egzemplarze:

1. Wiele egzemplarzy (kolekcja egzemplarze) może być powiązanych z wieloma rezerwacjami (kolekcja rezerwacje), np. egzemplarz może być wynajmowany wielokrotnie, a jedna rezerwacja może obejmować wiele egzemplarzy.
2. Struktura MongoDB przy użyciu referencji:
   1. Dokumenty rezerwacje zawierają tablice referencji do egzemplarzy.
   2. Dokumenty egzemplarze zawierają tablice referencji do rezerwacji.

Przykład:

Kolekcja: egzemplarze

{

„\_id”: ObjectId („ABC123”),

„kolor”: „Czarny”,

„ID\_modelu”: ObjectId(“2”),

„rok\_produkcji”, „2019-07-05”,

„cena”: 1000,

„przebieg”: 47000

„rezerwacje”: [ObjectId(“1”), ObjectId(“2”)]

}

{

„\_id”: ObjectId („XYZ456”),

„kolor”: „Szary”,

„ID\_modelu”: ObjectId(“3”),

„rok\_produkcji”, „2019-07-05”,

„cena”: 1000,

„przebieg”: 50000

„rezerwacje”: [ObjectId(“1”)]

}

Kolekcja: rezerwacje

{

"\_id": ObjectId(“1”),

"pesel": 12345678901,

"data\_rozpoczecia": "2025-01-01",

"data\_zakonczenia": "2025-01-10",

"egzemplarze": ["ABC123", "XYZ456"]

}

{

"\_id": ObjectId(“2”),

"pesel": 11111111111,

"data\_rozpoczecia": "2025-02-01",

"data\_zakonczenia": "2025-02-10",

"egzemplarze": ["ABC123"]

}

Zalety użycia referencji:

1. Dane są przechowywane osobno, co pozwala uniknąć duplikacji danych w przypadku powtarzających się.
2. Umożliwia przechowywanie dużej liczby danych, niezależnie od rozmiaru pojedynczego dokumentu.