

Aula 2 - Modelagem do banco de dados

Docupedia Export

Author:Ferro Alisson (CtP/ETS)

Date:14-Aug-2023 14:19

Table of Contents

1 Banco de dados orientado a documentos

3

2 Estrutura de dados

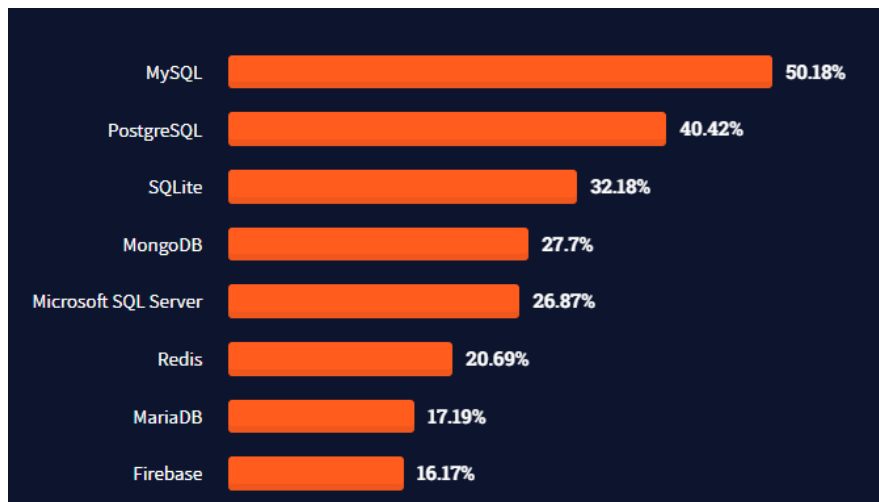
4

3 Modelagem

5

1 Banco de dados orientado a documentos

O MongoDB é orientado a documentos, isso significa que são documentos JSON binário (BSON), em uma linguagem NOSQL, significa que seu relacionamento é entre documentos, segundo pesquisa do Stack Overflow, o MongoDB está entre os bancos de dados mais utilizados de 2021, sendo o mais utilizado dentre os NOSQL, [clique aqui para ver mais](#)



O MongoDB, é um banco criado pensando em escalabilidade, tornando-o uma opção muito interessante para grandes volumes de dados, especialmente os desestruturados.

Dados desestruturados são um problema para a imensa maioria dos bancos de dados relacionais, mas não tanto para o MongoDB. Quando o seu schema é variável, é livre, usar MongoDB vem muito bem a calhar. Os documentos BSON (JSON binário) do Mongo são schemaless e aceitam quase qualquer coisa que você quiser armazenar, sendo um mecanismo de persistência perfeito para uso com tecnologias que trabalham com JSON nativamente, como JavaScript (e consequentemente Node.js).

Não se deve utilizar MongoDB quando relacionamentos entre diversas entidades são importantes para o seu sistema.

2 Estrutura de dados

A estrutura de dados do mongoDB se assemelha a dos bancos relacionais contendo suas particularidades



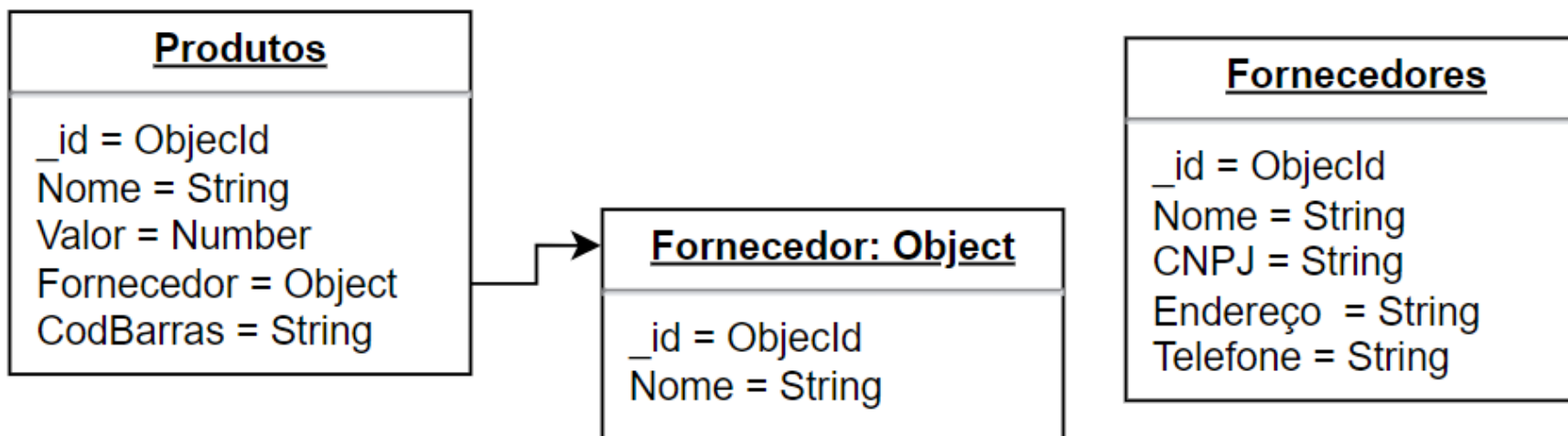
Onde no SQL temos as tabelas, no mongo chamamos de collection, as linhas são documentos e as colunas são campos ou fields.

3 Modelagem

O MongoDB trabalha com uma estrutura orientada a objetos, diferentemente da forma em que bancos relacionais trabalham.

Ao modelar um banco orientado a documentos, a ideia é de que cada documento seja suficiente para trazer as informações necessárias para a *collection*.

Sendo assim, ao modelar uma *database* de produtos, o id e nome do fornecedor pode conter na *collection*.



Como para o produto, conter informações do fornecedor como por exemplo endereço, pode não ser relevante, então criamos uma nova *collection* e deixamos somente o que é relevante para produtos e iremos ter outra *collection* somente de fornecedor contendo todas as informações.

Vale resaltar que na *collection* de Fornecedor e de Produtos tem que ter o mesmo `_id`.

Desafio: Faça uma modelagem de um banco de dados para uma escola, onde temos Alunos, Disciplina, Sala de Aula e Professor