

Programação Orientada e Objetos II



Anhanguera

AVALIE
SUA PROFISSÃO

QUANDO APARECER EM SEU
PORTAL UMA AVALIAÇÃO SOBRE
SEU CURSO, RESPONDA:



NOTAS

9 ou 10

SIGNIFICA QUE VOCÊ INDICA

NOTAS

7 ou 8

SIGNIFICA QUE VOCÊ NÃO INDICA



Anhanguera



Anhanguera



Existem diversas soluções e projetos que possuem os conceitos de NoSQL. Considerando o fato de ser uma tecnologia relativamente nova, com foco em cenários que necessitam armazenar grandes quantidades de dados, é necessário certo cuidado nas escolhas (GESSERT, 2016). Entre diversos projetos, o MongoDB apresenta maturidade e estabilidade, o que remete a escolha por esse projeto (HAN, E, e LE, 2011). O MongoDB tem seu nome derivado da palavra humongous que significa enorme ou monstruoso, declarando um novo sistema de gerenciamento de banco dados com as premissas do NoSQL (PLUGGE; MEMBREY e HAWKINS, 2010).



Anhanguera

O MongoDB utiliza uma estrutura do sistema NoSQL chamada document stores, em que são utilizados arquivos formatados para fazerem o armazenamento da informação. Esses arquivos utilizam o formato JSON (JavaScript Object Notation), sendo os dados gravados em modo binário. Este formato tem sido muito utilizado devido ao seu processo de leitura (por humanos e computadores) ser considerado simples quando comparado aos sistemas mais complexos, como grafos ou formatos proprietários. Além disso, já existem diversas ferramentas com processo consolidado para a conversão do Extensible Markup Language (XML) para o JSON, tornando a utilização e migração menos complexa.



Anhanguera

O MongoDB possui 3 elementos básicos para o armazenamento, que consistem no banco de dados, na coleção e no registro. Cada banco de dados possui diversas coleções, essas coleções são conjuntos de registros com a mesma estrutura, já os registros são os dados armazenados. Existem diversas versões disponíveis deste projeto, tais como a Atlas, o Community Server e o Enterprise Server. A versão Atlas é uma versão do MongoDB para ser utilizada na nuvem, composta por formas simples de criar instâncias e conjuntos do SGDB (REDMOND; WILSON, 2012). A versão que utilizaremos será a Community Edition que pode ser utilizado de forma livre. A versão Enterprise Edition envolve custos de licenciamento, porém possui algumas características a mais que a versão Community que consistem em:



- Sistema que executa diretamente na memória RAM;
- Criptografia dos dados;
- Sincronismo com o LDAP e Kerberos (sistema de controle de usuários e senhas).

Veja os passos necessários para a configuração em um ambiente Windows da versão Community.

- Fazer o download em <<https://www.mongodb.com/download-center?jmp=nav#community>> (aba Community Server). Existem versões para GNU/Linux, Microsoft Windows e Mac OSX; nesse caso se deve selecionar a versão para Microsoft Windows versão 4.0, a arquitetura do computador deve ser x86. Durante a instalação, selecione a versão “Custom” e certifique que todas os componentes estão selecionados e serão instalados.

- Após essa etapa, selecione a opção “Install MongoDB as a Service”. No item “Data Directory”, insira o seguinte caminho:

C:\data\db e no item “Log Directory” insira: C:\data\log. Utilizar a opção de instalar como serviço garante que o Microsoft Windows controla o início de forma independente.

- Após essa etapa, selecione o item que define a instalação do “MongoDB Compass”.



Anhanguera

O processo de instalação é simples e intuitivo, siga os passos que forem indicados. Depois da instalação, é necessário verificar se as pastas, onde serão armazenados os bancos de dados e os arquivos de log do MongoDB, foram criados (C:\data\db e C:\data\log). Para isso, você pode utilizar o Explorador de Arquivos do Microsoft Windows. Caso não estejam criados, você deverá criar as pastas indicadas. Também é possível fazer esse processo utilizando o prompt de comando do Windows (Command Prompt ou CMD). Para abrir o CMD você pode fazer uma pesquisa usando o campo de buscas (busque o programa prompt de comando) ou, dependendo da versão do Windows, clique com o botão direito do mouse sobre o ícone do menu iniciar e selecione prompt de comando. O CMD deve ser executado como Administrator.



Caso os diretórios não estejam lá, eles devem ser criados com o CMD, para serem utilizados para armazenar o próprio arquivo de dados e de logs do MongoDB. Esse SGDB procura no diretório raiz onde ele foi instalado, porém isso pode ser alterado caso o programador deseje. Por padrão, o MongoDB é instalado no C:\, portanto, é necessário criar os diretórios C:\data\db e C:\data\log. Com o comando `md "\data\db" "\data\log"` é possível criar os dois diretórios. A Figura 4.4 apresenta o resultado dos comandos no prompt. Repare que ao usar o comando `cd \data` o diretório corrente do CMD é definido como C:\data e com o comando `dir` é possível ver os dois diretórios internos db e log.

```
C:\>md "\data\db" "\data\log"
C:\>cd \data
C:\data>dir
O volume na unidade C não tem nome.
O Número de Série do Volume é AEF3-FA5F

Pasta de C:\data
27/06/2018  15:14    <DIR>          .
27/06/2018  15:14    <DIR>          ..
27/06/2018  15:14    <DIR>          db
27/06/2018  15:14    <DIR>          log
               0 arquivo(s)          0 bytes
               4 pasta(s)      8.947.904.512 bytes disponíveis

C:\data>net start MongoDB
O serviço de MongoDB Server está sendo iniciado...
O serviço de MongoDB Server foi iniciado com êxito.

C:\data>
```




Antes de iniciar o uso do MongoDB é necessário garantir que o servidor “mongod” foi iniciado. Para isso, ainda no prompt e como administrador, é necessário utilizar o comando "C:\Program Files\MongoDB\Server\4.0\bin\mongod.exe" (sendo obrigatório utilizar as aspas). Ao “subir” o servidor, você verá na última linha, dentre várias mensagens, a instrução “waiting for connections on port 27017”. Veja que o servidor está esperando por conexões na porta 27017, que é a padrão usada pelo MongoDB. Agora, você deve abrir uma nova janela de prompt (sem fechar a que está aberta) e utilizar o comando "C:\Program Files\MongoDB\Server\4.0\bin\mongo.exe". A Figura 4.5 apresenta o resultado de executar o MongoDB shell; repare que no final da imagem a entrada de comando está apenas com o sinal de maior (>), isso significa que o MongoDB shell está esperando por comandos.



```
Administrator: Prompt de Comando - "C:\Program Files\MongoDB\Server\4.0\...
c:\>"C:\Program Files\MongoDB\Server\4.0\bin\mongo.exe"
MongoDB shell version v4.0.0
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 4.0.0
Server has startup warnings:
2018-06-27T15:14:31.707-0300 I CONTROL [initandlisten]
2018-06-27T15:14:31.707-0300 I CONTROL [initandlisten] ** WARNING: Access control
is not enabled for the database.
2018-06-27T15:14:31.707-0300 I CONTROL [initandlisten] **           Read and write
access to data and configuration is unrestricted.
2018-06-27T15:14:31.707-0300 I CONTROL [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service to collect and display
metrics about your deployment (disk utilization, CPU, operation statistics,
etc).

The monitoring data will be available on a MongoDB website with a unique
URL created for you. Anyone you share the URL with will also be able to
view this page. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command:
db.enableFreeMonitoring()
---
```



Existem diversas operações que podem ser feitas nessa interface textual do MongoDB; o Quadro 4.5 apresenta comandos que podem ser utilizados. Fique atento que o comando use <nome BD> é utilizado para criar a base de dados se ele não existir, caso exista, é definido como banco de dados ativo.

Comando	Descrição
show dbs	Apresenta os bancos de dados disponíveis.
show collections	Exibe as <i>collections</i> , que representam o conjunto de dados que estão no banco de dados.
show users	Apresenta os usuários do banco de dados corrente.
use <nome BD>	Define o banco de dados corrente ou cria o banco de dados, se puder existir diversos bancos por sistema (trocando o <nome BD> pelo nome do banco desejado).
db.<collection>.find()	Exibe os dados armazenados na coleção com o nome que será especificado no parâmetro <collection>.



É possível criar um banco executando o comando: “use testeBanco”, onde testeBanco é o nome do banco a ser criado (Figura 4.6). Ao confirmar o comando com “enter”, recebemos a resposta “switched to db testeBanco”, porém, ao executar o comando “show dbs” ele não aparece na lista (mesmo tendo sido criado). Para que ele apareça na lista é necessário que tenha algum registro contido nesse banco. Então, para fazer a inserção de algum registro pelo terminal, é necessário utilizar o comando `db.alunos.insertOne({ nome: "João", idade: 25, curso: "ADS" })`, onde o alunos é o nome da collection, os dados são separados por vírgula, e o texto é definido por aspas duplas.



C:\> Administrador: Prompt de Comando - "C:\Program Files\MongoDB\Server\4.0\..."

```
> use testeBanco
switched to db testeBanco
> db.alunos.insertOne( { nome: "João", idade: 25, curso: "ADS" } )
{
  "acknowledged" : true,
  "insertedId" : ObjectId("5b34279692455073b8e73c36")
}
> show dbs
admin          0.000GB
config         0.000GB
local          0.000GB
testeBanco     0.000GB
> show collections
alunos
> _
```



O comando db possui diversos outros elementos que podem ser usados. O Quadro 4.5 apresenta um conjunto dos mais utilizados. Tais comandos são muito úteis para administrar o banco, de maneira rápida, podendo inclusive gerar scripts para automatizar tarefas.

Comando	Descrição
<code>db.dropDatabase()</code>	Remove o banco de dados corrente.
<code>db.getName()</code>	Apresenta o nome do banco de dados corrente.
<code>db.stats()</code>	Apresenta estatísticas do banco de dados corrente, tais como número de <i>collections</i> , utilização de espaço em discos e outros.
<code>db.getLastError()</code>	Apresenta o último erro da base dados corrente.



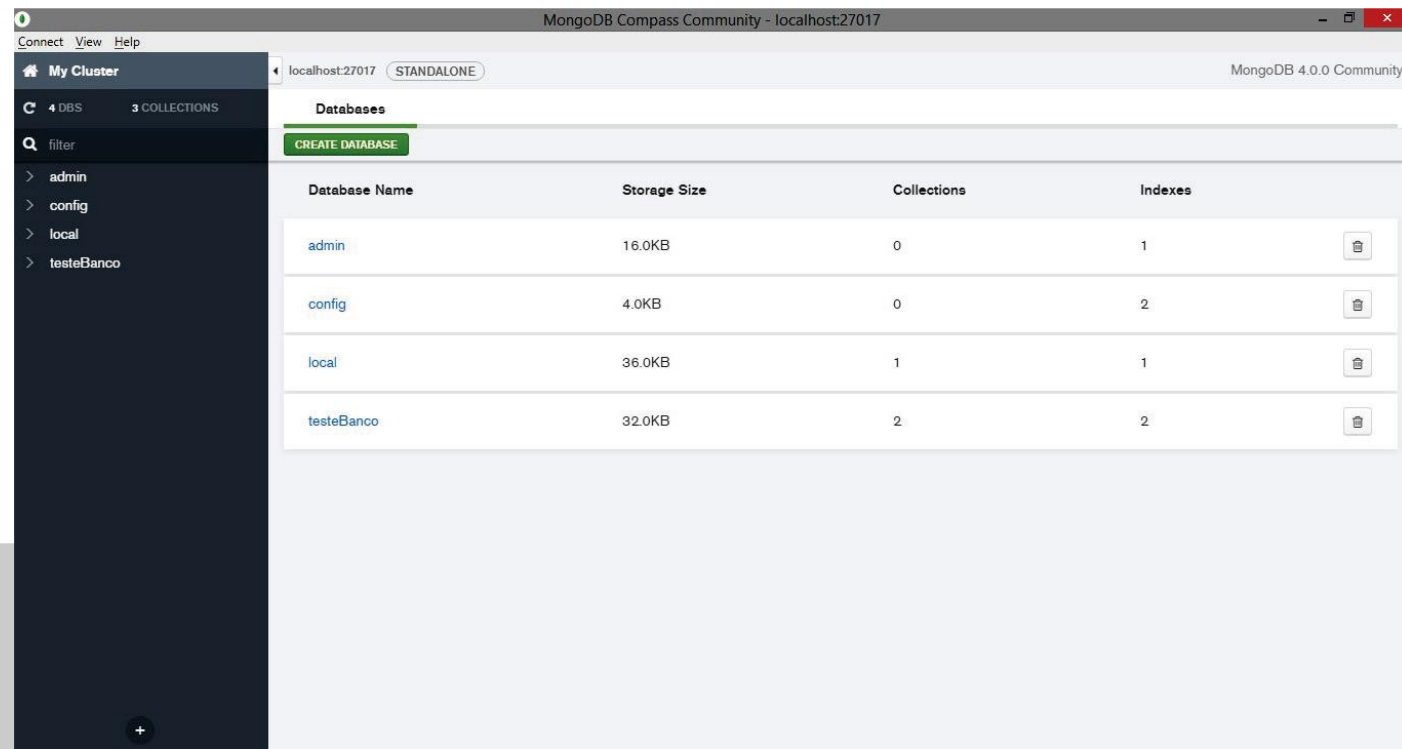
Repare que a criação e a inserção de dados no MongoDB, ocorre sem a obrigatoriedade de uma modelagem, na qual se definem tabelas, tipos de dados e relacionamentos. No MongoDB as “colunas” são criadas automaticamente, à medida que os dados são inseridos. O único passo obrigatório é definir em qual coleção os dados serão inseridos.

Para criar um conjunto de dados é necessário o comando `insertOne`. Nesse sentido, se temos um conjunto de dados com os seguintes campos:

- Nome completo;
- Logradouro;
- Número do logradouro;
- Bairro;
- CEP:

O comando necessário é: `db.cadastro.insertOne({ nomeCompleto: "João da Silva", logradouro: "Rua das Amoreiras", numeroLogradouro: 120, bairro: "Centro", cep: "12912-020" })`

Outra forma de se administrar o banco de dados é utilizar o MongoDB Compass, que é instalado junto com o pacote. Com ele é possível fazer as mesmas ações que estão disponíveis na linha de comando, mas usando uma interface gráfica. A Figura 4.7 apresenta uma visão da interface gráfica para controle do MongoDB.



Para usar essa ferramenta também é necessário carregar o servidor “mongod” antes de iniciar a interface gráfica. Uma opção muito interessante que o MongoDB Compass oferece é a visualização dos dados. Na Figura 4.7 estão sendo exibidos todos os bancos de dados disponíveis, ao selecionar um deles é possível ver quais collections ele possui. A Figura 4.8 apresenta, em detalhe, como os dados são apresentados.



The screenshot displays the MongoDB Compass web interface. On the left sidebar, the database 'testeBanco' is selected, and the 'cadastro' collection is highlighted. The main area shows a list of documents. The first document is:

```
{
  "_id": ObjectId("5b342c5992455073b8e73c37"),
  "nomeCompleto": "João da Silva",
  "logradouro": "Rua das Amoreiras",
  "numeroLogradouro": 120,
  "bairro": "Centro",
  "cep": "12912-020"
}
```

The second document is:

```
{
  "_id": ObjectId("5b34323892455073b8e73c38"),
  "nomeCompleto": "Pedro Pereira",
  "logradouro": "Rua das Macieira",
  "numeroLogradouro": 540,
  "bairro": "Centro",
  "cep": "12520-020"
}
```

At the top of the interface, there is a filter bar with the text '{ field: 'value' }', an 'OPTIONS' button, and a green 'FIND' button. Below the filter bar, there are buttons for 'INSERT DOCUMENT', 'VIEW', 'LIST', and 'TABLE'. On the right side, it says 'Displaying documents 1 - 2 of 2' with navigation arrows and a refresh icon.



Após a instalação do MongoDB e dos testes iniciais, é necessário instalar as dependências necessárias para utilizar esse SGBD junto ao Java e ao Eclipse. Uma das opções é baixar o arquivo Jar e incluí-lo diretamente no projeto, o que está cada vez menos aconselhado, pois todo o tratamento de dependência é feito pelo programador. Uma forma mais dinâmica é utilizar o Maven, que faz o tratamento e controle das dependências das bibliotecas em uma aplicação Java de maneira documentada. Com esse componente é possível apenas indicar o software que será necessário e ele trata todas as dependências. No Eclipse é possível criar um projeto já com o suporte ao Maven seguindo o menu File >> New >> Other... No menu para criação de novo item, buscar Maven e selecionar o Maven Project. A Figura 4.9 apresenta a primeira interface para a criação do projeto com suporte ao Maven. Nessa etapa é importante selecionar a opção “Create a simple project (skip archetype selection)” e com isso não será necessário escolher entre as arquiteturas que o projeto se encaixa na formatação do Maven.



Anhanguera

New Maven Project

Select project name and location

☒ Create a simple project (skip archetype selection)

☒ Use default Workspace location

Location: Browse...

☐ Add project(s) to working set

Working set: More...

► Advanced

? < Back Next > Finish Cancel



Na segunda interface é preciso definir o Group Id e Artifact Id, onde o primeiro representa um nome único dos projetos e o segundo definirá o nome para a distribuição do projeto (jar). Por exemplo, podemos definir o Group Id como exemplo.mongodb e o Artifact Id como JavaMongoDBTeste. As outras opções podem ser deixadas como padrão.

Repare que no projeto criado, existe um arquivo chamado pom.xml. Ele controla diversos aspectos no projeto, tais como as dependências, e ao abri-lo é possível ver um editor específico no Eclipse. Na parte inferior dessa tela, procure por uma aba escrito “pom.xml” e selecione essa opção para editar diretamente no arquivo, adicionando o conteúdo XML fornecido pelo projeto do MongoDB em `<https://mongodb.github.io/mongo-java-driver/>` (acesso em: 02 jul. 2018). O arquivo deve ficar semelhante ao Quadro 4.6, no qual você pode notar que a parte de “dependencies” foi inserida conforme recomendações.



```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>exemplo.mongodb</groupId>
  <artifactId>JavaMongoDBTeste</artifactId>
  <version>0.0.1-SNAPSHOT</version>

  <dependencies>
    <dependency>
      <groupId>org.mongodb</groupId>
      <artifactId>mongodb-driver</artifactId>
      <version>3.6.4</version>
    </dependency>
  </dependencies>

</project>
```



Anhanguera

Com isso o projeto está pronto para fazer acesso ao MongoDB pelo Java; dessa forma não é necessário tratar diretamente as dependências e o programador pode focar no desenvolvimento. A utilização do MongoDB é mais simples e direta que de um SGDB relacional, sendo necessário menos tempo para configurações e ajustes na estrutura do banco. Essa utilização vai de acordo com as metodologias ágeis, pois a ideia é gastar mais tempo no desenvolvimento do sistema do que planejar algo que pode nem ser aceito pelo cliente.



Anhanguera

Exercícios pág. 211 à 214



Anhanguera

O MongoDB é um ótimo exemplo de um sistema NoSQL, no qual é possível fazer a instalação e controle dos dados de maneira intuitiva. Existem duas formas de fazer essa administração, uma dela é feita com uma interface gráfica e outra utilizando a linha de comando.

Qual das opções apresenta apenas comandos de linha do MongoDB?

- a) use, drop, select, set.
- b) show, db, use, db.dropDatabase(), view.
- c) select, view, set, create.
- d) show, db, use, show collections.
- e) set, create, show, release.



A administração de um sistema de gerenciamento de banco de dados (SGDB) do tipo NoSQL tem menos opções que um sistema de banco de dados relacional. Essa característica vem diretamente da própria filosofia que os sistemas NoSQL apresentam.

Dentre os comandos abaixo que podem ser usados no MongoDB shell, escolha a opção que descreve a sequência correta para fazer a criação de um banco, a criação de uma collection, e suas respectivas visualizações (banco e collection).

1 – show collections.

2 - db.pessoas.insertOne({ info: "dados", numero: 70 }).

3 - show dbs.

4 - use nomeBanco.

a) 1 – 2 – 3 – 4.

b) 3 – 1 – 2 – 4.

c) 1 – 2 – 2 – 4.

d) 2 – 4 – 3 – 1.

e) 4 – 2 – 3 – 1.