

Nodejs + MongoDB

Programação Web 1



Objetivo de Aprendizagem

- Aplicar o MongoDB juntamente com Nodejs

Agenda

- MongoClient
- *Connection Guide*
- Cursor

MongoClient

Connection String

Exemplo de URI de conexão do MongoDB Atlas

```
1 "mongodb+srv://<username>:<password>@<your-cluster-url>/test?retryWrites=tru
```

Connection String

Connection Options

- `poolSize=20` : tamanho máximo da instância do pool de conexões
- `w=majority` : nível de confirmação de escrita pela maioria dos servidores
- `<number>` : número específico de servidores a confirmar
- `<custom write concern>` : nome de um servidor específico

Opções de Conexão - Documentação

Connection String

Connection Pool

- `MongoClient` é a classe básica para conexão com MongoDB
- Cada instância representa um *pool* (conjunto) de conexões
- Mesmo aplicações *multithread* usam apenas uma instância de `MongoClient`

MongoClient

Exemplo

```
1 const { MongoClient } = require("mongodb");
2 const uri = "mongodb+srv://sample-hostname:27017/?maxPoolSize=20&w=majority"
3 const client = new MongoClient(uri);
```

Testando a Conexão

checkconn.js

```
1  async function run() {
2    try {
3      await client.connect();
4      await client.db("admin").command({ ping: 1 });
5      console.log("Connected successfully to server");
6    } finally {
7      await client.close();
8    }
9  }
10
11 run().catch(console.dir);
```

Listando BDs

listdbs.js

```
1  async function main() {
2      const uri = "mongodb+srv://<username>:<password>@<your-cluster-url>/test?r
3      const client = new MongoClient(uri);
4      try {
5          await client.connect();
6          await listDatabases(client);
7      } catch (e) {
8          console.error(e);
9      } finally {
10         await client.close();
11     }
12   }
13 main().catch(console.error);
```

Listando BDs

Função Auxiliar

```
1  async function listDatabases(client) {  
2      const databasesList = await client.db().admin().listDatabases();  
3      console.log("Databases:");  
4      databasesList.databases.forEach(db => console.log(` - ${db.name}`));  
5  }
```

Sample Data no Atlas

"Atlas provides sample data you can load into your Atlas database deployments."

Tutorial Sample Data

Como Carregar Sample Data?

1. Acessar o Atlas
2. Clicar em "Database"
3. Clicar nos três pontos (...)
4. Selecionar "*Load Sample Dataset*"
5. Aguardar alguns minutos
6. Clicar em "*Browse Collections*"

Exemplos:

- sample_airbnb
- sample_geospatial
- sample_guides
- sample_mflix
- sample_supplies
- sample_training
- sample_weatherdata

Sample Airbnb

sample_abnb.js

```
1  async function main() {
2      const { MongoClient } = require('mongodb');
3      const uri = 'mongodb+srv://<dbUser>:<password>@cluster0.mongodb.net/?retry';
4      const client = new MongoClient(uri);
5      await client.connect();
6      await findListings(client, 3);
7      await client.close();
8  }
```

Função

findListings

```
1  async function findListings(client, resultsLimit) {  
2      const cursor = client.db('sample_airbnb')  
3          .collection('listingsAndReviews')  
4          .find()  
5          .limit(resultsLimit);  
6      const results = await cursor.toArray();  
7      if (results.length > 0) {  
8          console.log(`Found ${results.length} listing(s):`);  
9          results.forEach((result, i) => {  
10              console.log(`\n${i + 1}. Name: ${result.name}`);  
11              console.log(`    Bedrooms: ${result.bedrooms}`);  
12              console.log(`    Bathrooms: ${result.bathrooms}`);  
13          });  
14      }  
15  };
```

Cursor

- Permite percorrer o resultado de uma consulta ao BD
- Modifica documentos e campos retornados
- Operadores de comparação
- Métodos do Cursor

Perguntas

Exercícios

1

1. Reproduzir os *scripts* usando seu próprio cluster do Atlas
2. Ajustar a *connection string* usando o módulo Dotenv para evitar a exposição das credenciais de acesso ao MDB no código fonte
3. Carregar as coleções de amostra

2

checkconnV2.js

- Aprimorar a verificação de conexão a partir de `checkconn.js`
- Versão aprimorada para testar se o servidor está ativo
- Mostra os *hosts* do *cluster*
- Exibe mensagens de sucesso ou erro

3

listdbsV2.js

- Baseado no `listdbs.js`
- Versão aprimorada verifica se a consulta foi realizada corretamente
- Exibe a lista de bancos com `sizeOnDisk`
- Mostra o tamanho total do *cluster*

4

sample_abnbV2.js

- Listar 10 melhores propriedades com:
 - `review_scores_rating ≥ 80`
 - Campo `name` não vazio
 - Exibir somente: `name` , `listing_url` , `review_scores_rating`

5

`restaurants.js`

- Criar um *script* para questão 9 da aula anterior usando `restaurants.json`

Exercícios MongoDB

Referências

- Async/Await com Node.js
- Sample Data - MongoDB Atlas

José Roberto Bezerra

✉️ jbroberto@ifce.edu.br

⌚ [jbroberto76](#)

⌚ [pw1-repo](#)

