

Conectando e usando o MongoDB com Node.js

Aqui estão os passos para conectar e usar o MongoDB com Node.js:

1. Instalar o MongoDB:

- Se você ainda não possui o MongoDB instalado, faça o download e siga as instruções de instalação para o seu sistema operacional:
<https://www.mongodb.com/try/download/community>

2. Instalar o Driver MongoDB para Node.js:

- Use o gerenciador de pacotes npm para instalar o driver:

```
npm install mongodb
```

content_copyUse code [with caution](#).Bash

3. Criar um arquivo de conexão:

- Crie um arquivo JavaScript (por exemplo, db.js) com as seguintes configurações:

```
const MongoClient = require('mongodb').MongoClient;
const uri = "mongodb://localhost:27017/"; // Endereço e porta do seu servidor
MongoDB
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology:
true });

async function connectToDb() {
  try {
    await client.connect();
    console.log("Conectado ao MongoDB!");
    return client.db("nomeDoBancoDeDados"); // Substitua "nomeDoBancoDeDados" pelo
nome do seu banco de dados
  } catch (error) {
    console.error("Erro ao conectar ao MongoDB:", error);
  }
}
```

```
module.exports = { connectToDb };
```

content_copyUse code [with caution](#).JavaScript

4. Usar o driver em seu código:

- Importe o arquivo de conexão e use o driver para interagir com o banco de dados:

```
const { connectToDb } = require('./db');

async function main() {
  try {
    const db = await connectToDb(); // Conecte-se ao banco de dados

    // Inserir um novo documento na coleção "users"
    const result = await db.collection('users').insertOne({ name: 'John Doe', age:
30 });
```

```
console.log(`Documento inserido com o ID: ${result.insertedId}`);

// Buscar documentos na coleção "users"
const users = await db.collection('users').find({}).toArray();
console.log("Usuários encontrados:", users);

} catch (error) {
  console.error("Erro:", error);
} finally {
  await client.close(); // Feche a conexão com o banco de dados
}
}

main();
```

content_copyUse code [with caution](#).JavaScript

Explicação do código:

- O código acima utiliza o driver MongoDB para Node.js para realizar as seguintes operações:
 - Conectar ao banco de dados.
 - Inserir um novo documento na coleção users.
 - Buscar todos os documentos da coleção users.
 - Fechar a conexão com o banco de dados.

Dicas adicionais:

- Você pode usar o método find() com filtros para buscar documentos específicos.
- O driver MongoDB oferece uma ampla variedade de métodos para trabalhar com dados, como updateMany(), deleteMany(), aggregate() e muitos outros.
- Use o console de comandos do MongoDB (mongo) para interagir diretamente com o banco de dados e verificar os dados inseridos.

Recursos adicionais:

- **Documentação do driver MongoDB:** <https://mongodb.github.io/node-mongodb-native/>
- **Tutoriais do MongoDB:** <https://www.mongodb.com/docs/drivers/node>

Com estas informações, você poderá conectar-se ao MongoDB e realizar operações de banco de dados com Node.js.