

Exercício 3

Neste exercício, iremos criar uma prova de conceito de uma loja online com persistência de dados. Para esse contexto, iremos utilizar a seguinte stack: nodejs, nodemon, express, handlebars, mongoose, mongodb, knex e sqlite.

Passo 1

Clone o repositório abaixo e em seguida rode o comando ***npm install***.

<https://github.com/samwx/nodejs-knex>

Rode o comando ***npm start*** para rodar o projeto na rota 3000. Para debugar o código no VSCode, abra o projeto e aperte F5 (habemus debug!!)

Passo 2

Rode os comandos de migrations e seed para garantir que toda a estrutura está configurada conforme necessária:

1. ***npm run knex migrate:latest;***
2. ***npm run knex seed:run.***

Passo 3

Utilizando os conceitos da aula 01, crie uma rota ***/products/:id*** e liste todos os detalhes do produto clicado. Para isso, além de declarar a rota especificada, você deverá implementar o método ***getById*** no arquivo **Products.js** dentro da pasta **store**. Utilize o [método where](#) para filtrar o produto por id.

Passo 4

Abaixo dos detalhes do produto iremos listar todas as avaliações feitas via comentário. Para isso, além de listar os comentários feitos anteriormente, crie um formulário com os campos **Nome** e **Comentário** para que seja possível adicionar novos comentários. ***Obs.: Neste passo, os comentários deverão ser armazenados no mongodb via mongodb atlas.***

Caso você não tenha uma conta no mongo atlas, entre no site <https://www.mongodb.com/cloud/atlas> para criar uma conta e siga os passos abaixo:

Após criar uma conta, selecione a opção **Starter Clusters**, conforme imagem abaixo:

Starter Clusters

For teams learning MongoDB or developing small applications.




- ✓ Highly available auto-healing cluster
- ✓ End-to-end encryption
- ✓ Role-based action control
- ✗ No downtime scaling
- ✗ Network isolation
- ✗ Realtime performance metrics

Starting at
FREE

Create a cluster








Na próxima tela, selecione o cloud provider “AWS” e a region “N. Virginia”. Após isso, clique em “Create Cluster”.

Cloud Provider & Region AWS, N. Virginia (us-east-1) ▾



Create a **free tier cluster** by selecting a region with **FREE TIER AVAILABLE** and choosing the **M0** cluster tier below.

★ Recommended region ⓘ

NORTH AMERICA	EUROPE	ASIA
<div> N. Virginia (us-east-1) ★ FREE TIER AVAILABLE</div>	<div> Ireland (eu-west-1) ★ FREE TIER AVAILABLE</div>	<div> Singapore (ap-southeast-1) ★ FREE TIER AVAILABLE</div>
<div> Oregon (us-west-2) ★ FREE TIER AVAILABLE</div>	<div> Frankfurt (eu-central-1) ★ FREE TIER AVAILABLE</div>	<div> Mumbai (ap-south-1) FREE TIER AVAILABLE</div>
AUSTRALIA		
<div> Sydney (ap-southeast-2) ★ FREE TIER AVAILABLE</div>		

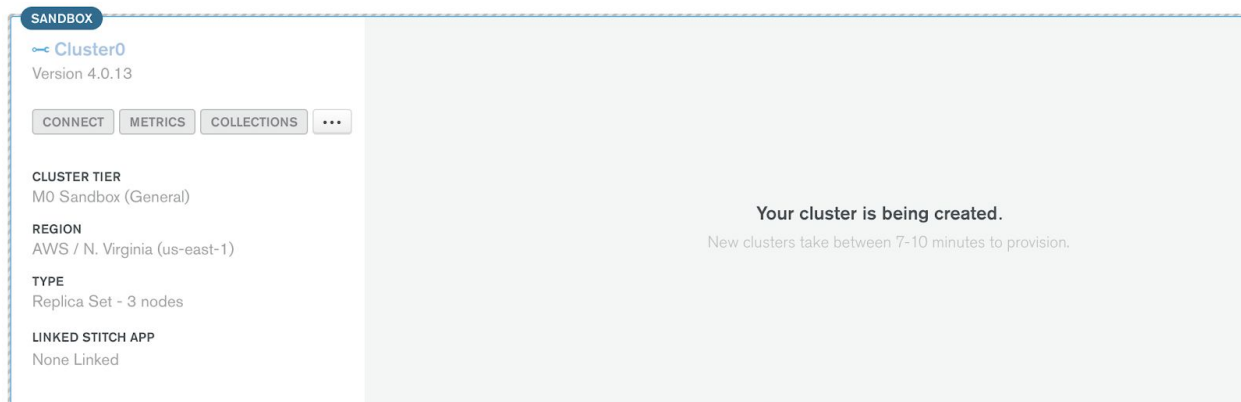
FREE

Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

Back

Create Cluster

Essa ação pode demorar alguns minutos conforme imagem abaixo:



No próximo passo, adicione o IP **0.0.0.0/0** ao whitelist e crie um usuário a seu critério (**evite utilizar a mesma senha para outros acessos, pois essa será pública. Evite caracteres especiais**).

Connect to Cluster0

Setup connection security > Choose a connection method > Connect

You need to secure your MongoDB Atlas cluster before you can use it. Set which users and IP addresses can access your cluster now. [Read more](#)

You can't connect yet. Set up your user security permission below.

1 Whitelist your connection IP address

✓ An IP address has been whitelisted. Add another whitelist entry in the [IP Whitelist tab](#).

2 Create a MongoDB User

This first user will have [atlasAdmin](#) permissions for this project.

Keep your credentials handy, you'll need them for the next step.

Username

user

Password

ex. dbUserPassword

[Autogenerate Secure Password](#)

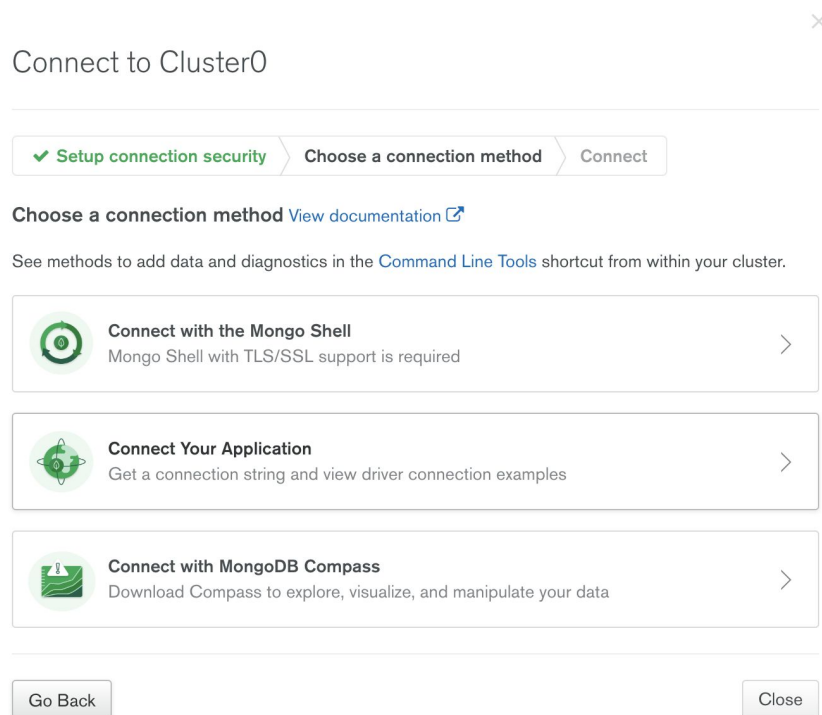
SHOW

Create MongoDB User

Close

Choose a connection method

Selecione a opção “**Connect your application**”:



Copie a connection string gerada e no arquivo **Connection.js**, atribua à variável “**connectionString**” na linha 2. Substitua a parte “<password>” pela senha escolhida e “user” pelo usuário adicionado. Exemplo:
mongodb://<dbuser>:<dbpassword>@ds213896.mlab.com:13896/puc-tutorial
mongodb://samuel:teste1234@ds213896.mlab.com:13896/puc-tutorial

Bônus

Faça um filtro de preços na listagem de produtos, de forma com que o usuário consiga selecionar os produtos em um determinado intervalo de preços.