

Banco de Dados NoSQL

(NS0205)

Prof. Giovane Barcelos
giovane_barcelos@uniritter.edu.br

5. Redis – Redis & NodeJS

Conteúdo Programático

- 1. Introdução ao Redis**
- 2. Primeiros Passos**
- 3. Tipos de Dados**
- 4. Persistência de Dados**
- 5. Aplicação prática com Node.js**

5. Redis – Redis & NodeJS

Como será a aplicação final?

← ⓘ localhost:3000 | 🔍 Pesquisar » ☰

Lista de Tarefas ☰

Adicionar Tarefa

Lista de Tarefas

- ☐ Finalizar o projeto Tarefas
- ☐ Assistir aula de NoSQL

Contato

Nome: Penélope Charmosa
Companhia: Uniritter
Telefone: 9999999

Editar Contato

Nome

Companhia

Telefone

URL:

<https://fathomless-wildwood-85551.herokuapp.com/>

5. Redis – Redis & Node.JS

Itens Abordados

- 1. NodeJS, Express e Configuração do Servidor***
- 2. Configuração do Cliente e Listar Tarefas***
- 3. Adicionar Tarefas***
- 4. Remover Tarefas***
- 5. Editar Contato***
- 6. Twitter, BootStrap UI***

5. Redis – Redis & Node.JS

NodeJS, Express e Configuração do Servidor

1. `sudo apt-get install redis-server`
2. `exit`
3. `mkdir -p ~/app/tarefas; cd ~/app/tarefas`
4. Criar repositório chamado de "tarefas" no <https://github.com>
5. `npm init`
description: Gerenciador de tarefas
entry point: `app.js`
git repository:
<https://github.com/giovanebarcelos/tarefas>
author: Giovane Barcelos
6. Visual Studio Code (VSC) → Open Folder → xubuntu → app → tarefas
7. VSC → Ctrl+Shift+E (Explorer) → Abrir "package.json" e incluir as dependências conforme <http://bit.do/NoSQLTarefasDependencias>
8. `cd ~/app/tarefas; npm install`

5. Redis – Redis & Node.JS

NodeJS, Express e Configuração do Servidor

9. VSC → Ctrl+Shift+G (Source Control: GIT) → Initialize Repository

10. VSC → File → Open File → ~/app/tarefas/.git/config

Inserir:

[user]

name = Giovane Barcelos

email = giovanebarcelos@gmail.com

[remote "origin"]

url =

https://github.com/giovanebarcelos/tarefas.git

fetch = +refs/heads/*:refs/remotes/origin/*

11. VSC → Ctrl+Shift+G → ... → Push to ... → Escolher Origin → Inserir Usuário e Senha

5. Redis – Redis & Node.JS

Configuração do Cliente e Listar Tarefas

- 12.** VSC → Ctrl+Shift+E → TAREFAS → New File → app.js
- 13.** Inserir no app.js as configurações de inicialização do NodeJS que inclui Criação do Cliente Redis e Configuração do renderizador de páginas (EJS), além de capturar o caminho / na URL
Código: <http://bit.do/NoSQLTarefasConfCliente>
- 14.** Para testar:
cd ~/app/tarefas; node app
- 15.** Para autocarregar sempre as alterações do NodeJS:
sudo su; npm install nodemom -g; exit
// o “nodemon” substitui o “node app”
- 16.** VSC → Ctrl+Shift+E → TAREFAS → New Folder → views
- 17.** Criar dentro do diretório views o arquivo “tarefas.ejs” conforme: <http://bit.do/NoSQLTarefasListarTarefas>

5. Redis – Redis & Node.JS

Configuração do Cliente e Listar Tarefas

- 18.** Vamos incluir algumas tarefas para serem visualizadas na aplicação
 - redis-cli
 - LPUSH tarefas "Assistir aula de NoSQL"
 - LPUSH tarefas "Finalizar o projeto Tarefas"
 - LRANGE tarefas 0 -1
- 19.** Acessar a aplicação (<http://localhost:3000>) para verificar inclusão das tarefas

5. Redis – Redis & Node.JS

Adicionar Tarefas

20. Criar captura de *post* no caminho “/tarefa/adicionar” em app.js conforme código:

<http://bit.do/NoSQLTarefasAdicionarTarefa>

21. Inserir formulário em tarefas.ejs para ler e submeter tarefa a ser adicionada.

<http://bit.do/NoSQLTarefasFormularioAdicionarTarefa>

5. Redis – Redis & Node.JS

Remover Tarefas

22. Criar captura de *post* no caminho “/tarefa/remover” em app.js conforme código:

<http://bit.do/NoSQLTarefasRemoverTarefa>

23. Criar um formulário em tarefas.ejs com possibilidade de marcar em um checkbox as tarefas que devem ser excluídas mudando a cor para vermelho das tarefas marcadas para exclusão

<http://bit.do/NoSQLTarefasFormularioRemoverTarefa>

5. Redis – Redis & Node.JS

Editar Contato

- 24.** Criar captura de *post* no caminho “/contato/editar” e envia dados do contato em app.js para ser renderizado conforme código:
<http://bit.do/NoSQLTarefasEditarContato>
- 25.** redis-cli hset contato nome “Penelope Charmosa”
- 26.** Criar um formulário em tarefas.ejs com possibilidade de enviar os dados do contato conforme o código:
<http://bit.do/NoSQLTarefasFormularioEditarContato>
- 27.** Inserir um pouco de bootstrap para ficar um pouco mais amigável: <https://bootswatch.com> (cerulean) e <http://getbootstrap.com/> conforme o código:
<http://bit.do/NoSQLTarefasBootstrap>

5. Redis – Redis & Node.JS

Deploy no Heroku e RedisLabs

28. Código fonte para conexão Redis

<http://bit.do/NoSQLTarefasConexaoRedisLabs>

29. Código fonte para execução no Heroku

<http://bit.do/NoSQLTarefasHeroku>

30. Criar uma conta em <https://redislabs.com>

31. Criar uma subscrição de banco de dados e extrair de "configuration" o "endpoint" ex: redis-13873.c13.us-east-1-3.ec2.cloud.redislabs.com:13873

5. Redis – Redis & Node.JS

Deploy no Heroku e RedisLabs

22. Criar usuário no heroku (<https://signup.heroku.com>)

23. Instalar o cliente do heroku

Executar pelo terminal

sudo su

apt-get install curl

add-apt-repository "deb <https://cli-assets.heroku.com/branches/stable/apt> ./"

curl -L <https://cli-assets.heroku.com/apt/release.key> | sudo apt-key add -

apt-get update

apt-get install heroku

exit

24. Criar repositório Heroku:

cd ~/app/tarefas; heroku create

25. Deploy do código: git push heroku master

26. Certificar-se que esta rodando: heroku ps:scale web=1

27. Abrir aplicação: heroku open

28. Para executar localmente: heroku local web

29. Para alterar: git commit -m "Mensagem";

git push heroku master; heroku open

Lembre-se

“Que a força esteja contigo.”

Star Wars

