Aula 3 - Rotas NodeJs

Tópicos especiais em Sistemas

Prof. Juliana Costa Silva - juliana.silva@up.edu.br

O que veremos hoje

1. Revendo...
2. Introdução
3. Rotas
4. Ajustes no projeto
    4.1 Editando seu projeto

Revendo...

O que já aprendemos?

- Criamos um projeto node;
- Configuramos a porta do através do arquivo **index.js**;
- Definimos o get de nosso aplicação;

O projeto da disciplina

- Faremos um sistema de controle financeiro pessoal;
- Este sistema deve ter:
  - Registro de gastos;
  - Login de usuários;
  - Registro de renda (salários - comissões - negócios);
  - Registro de cartões de créditos;
  - Registro de contas bancárias;

Rotas

O que são rotas?

- Rotas são as configurações dadas aos caminhos percorridos no seu sistema;
- Por exemplo, você deseja ver os gastos realizados, qual seria o caminho?
- seusite/gastos

Configurando rota

Vamos configurar a rota `http://localhost:3000/gastos`.

Edite a linha 7 do arquivo **index.js**, conforme apresentado abaixo, após a edição inicie o servidor node e acesse o endereço configurado.

```
JS index.js > ...
1   const express = require('express')
2
3   const app = express()
4
5   app.listen(3000,() => console.log('servidor rodando na porta 3000'))
6
7   app.get('/gastos', (req, res) => res.send('Você está em gastos'))
```

Fonte: O autor

package.json

O arquivo **package.json** contem todas as informações do nosso projeto, desde o nome, sua versão e scripts que podem ser executados até novos pacotes que poderemos instalar.

Editando package.json

- Edite a propriedade scripts do arquivo (linha 6);
- Inclua um novo scrip, o start, e então poderá inserir o comando que quiser, como início do seu projeto.

package.json

Editando package.json

- Ao invés de obrigarmos quem está acessando o servidor a escrever node index.js e diretório de instalação, padronizaremos que o arquivo de start realizará o comando node index.js.

package.json

Editando package.json

- start realizará o comando node index.js.
- O arquivo deve ficar como abaixo;

Fonte: O autor

package.json

Iniciando o projeto com npm

- Agora seu projeto será iniciado no servidor através do comando
- npm start

Reiniciar o servidor node

- Para cada alteração que fazemos no projeto, precisamos para o servidor node com Ctrl + C;
- Iniciar o servidor novamente com o comando node index.js para que ele carregue o projeto atualizado;
- Existe uma maneira melhor de fazer isso?

Instalando o nodemon

Pare o servidor.

Instale o nodemon

Instalando o nodemon

Pare o servidor.

Instale o nodemon

- Comando: npm install –save-dev nodemon;
- Feito isso, em nosso script de start não precisamos mais ter node index.js, mas sim nodemon index.js. No console executaremos o npm start, e para restartar os servidor só precisamos ativar o comando start.

Observe as dependências criadas no arquivo **package.json**.