Construção de api em nodeJs

1 – Criar uma pasta e entra na pasta com cmd  
2 – Criar um projeto: npm ini (preencher as informações)  
3 – Abrir o projeto no Vs Code  
4 - fazer a instalação do expresso para geração de API : npm install express --save  
5 – fazer a criação do servidor: criar um arquivo “server.js” com os comandos importando o express:  
  
*const express = require('express');*

*const app = express();*

*app.listen(3000,() =>{*

*console.log('Servidor Api NoFood iniciado na porta 3000.');*

*});*  
6 – fazer o start do servidor abrindo o localhost na porta indicada (node server)  
7 – caso o projeto só tenha o package.json e o package-lock.json, podemos baixar o node\_modules usando: npm i

8 – instalar o nodemon : npm install –g nodemon

9 – executar o projeto como : nodemon server

10 – Ctrl + C sempre para o serviço

11 – fazer a instalação do body-parse para conversão json : npm install body-parser –save  
12 – fazer a importação do body-parser no projeto

const express = require('express');

const bodyParser = require('body-parser');

const app = express();

app.use(bodyParser.json());

app.use(bodyParser.urlencoded({extended: false}));

Código Exemplo completo:

const express = require('express');

const bodyParser = require('body-parser');

const app = express();

app.use(bodyParser.json());

app.use(bodyParser.urlencoded({extended: false}));

//CRUD

//verbs http (verbos http)

let pessoas = [];

//R - READ

app.get('/',(req, res)=>{

res.status(200).send(pessoas);

});

//C - CREATE

app.post('/', (requisicao, resposta)=>{

console.log('Corpo:', requisicao.body);

pessoas.push(requisicao.body);

resposta.status(201).send(requisicao.body);

});

//U - UPDATE

app.put('/:id', (req, res) =>{

let pessoaencontrada = pessoas.filter(pes => {return pes.id == req.params.id});

pessoaencontrada = req.body;

res.status(202).send(pessoaencontrada);

});

//d - DELETE

app.delete('/:id', (req, res) =>{

for (let index = 0; index < pessoas.length; index++){

const pessoa = pessoas[index];

if(pessoa.id == req.params.id){

pessoas.splice(index, 1);

}

}

res.status(204).send();

});

//400 - bad request

//401 - Unauthorized

//500 - internal server error

app.listen(3000,() =>{

console.log('Servidor Api NoFood iniciado na porta 3000.');

});

13 – Usando o banco de dados mLab.com : login: itaprojetos , senha: @itaprojetos1, banco :itaprojetos, dados no banco: usuário administrador, senha: nofood123456

14- Vamos usar o mongoose para conexão : npm install mongoose –save