products-controller

'use strict'

const mongoose = require('mongoose');

const Product = mongoose.model('Product');

exports.post = (req, res, next) => {

var product = new Product(req.body);

product

.save()

.then(x => {

res.status(201).send({

message: 'Produto cadastrado com sucesso!'

});

}).catch(e => {

res.status(400).send({

message: 'Falha ao cadastrar o produto!',

data: e

}); //var res = product.save();

});

};

exports.put = (req, res, next) => {

const id = req.params.id;

res.status(200).send({

id: id,

item: req.body

});

};

exports.delete = (req, res, next) => {

res.status(200).send(req.body);

};

Product

'use strict'

const mongoose = require('mongoose');

const Schema = mongoose.Schema;

const schema = new Schema({

title: {

type: String,

required: true,

trim: true

},

slug: {

type: String,

required: [true, 'O slug e obrigatorio'],

trim: true,

index: true,

unique: true

},

description: {

type: String,

required: true,

},

price: {

type: Number,

required: true

},

active: {

type: Boolean,

required: true,

default: true

},

tags: [{

type: String,

required: true,

}]

});

module.exports = mongoose.model('Product', schema);

index-route

'use strict'

const express = require('express');

const router = express.Router();

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

res.status(200).send({

title: "Node Store API",

version: "0.0.2"

})

})

module.exports = router;

products-route

'use strict'

const express = require('express');

const router = express.Router();

const controller = require('../controllers/products-controller');

router.post('/', controller.post);

router.put('/:id', controller.put);

router.delete('/', controller.delete);

module.exports = router;

app

'use strict'

const express = require('express');

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

const mongoose = require('mongoose');

const app = express();

const router = express.Router();

//Connecta ao banco

mongoose.connect('mongodb://EdsonSobrinho:120647R@ds018268.mlab.com:18268/ndstr')

//Carrega os Models

const Product = require('./models/product');

//Carrega as Rotas

const indexRoute = require('./routes/index-route');

const productRoute = require('./routes/products-route');

app.use(bodyParser.json());

app.use(bodyParser.urlencoded({

extended: false

}));

app.use('/', indexRoute);

app.use('/products', productRoute);

module.exports = app;

server

'use strict'

const http = require('http');

const debug = require('debug')('nodestr:server');

const express = require('express');

const app = express();

const port = normalizePort (process.env.PORT ||'3000');

app.set('port', port);

const server = http.createServer(app);

const router = express.Router();

const route = router.get('/', (req, res, next) => {

res.status(200).send({

title: "Node Store API",

version: "0.0.2"

});

});

app.use('/', route);

server.listen(port);

server.on('error', onError);

server.on('listening', onListening;

console.log('API rodando na porta ' + port);

function normalizePort(val) {

const port = parseInt(val, 10);

if (isNaN(port)) {

return val;

}

if (port >= 0) {

return port;

}

return false;

}

function onError(error) {

if (error.syscall !== 'listen') {

throw error;

}

const bind = typeof port === 'string' ?

'Pipe' + port :

'Port' + port;

switch (error.code) {

case 'EACCES':

console.error(bind + ' requires elevated privileges');

process.exit(1);

break;

case 'EADDRINUSE':

console.error(bind + ' is already in use');

process.exit(1);

break;

default:

throw error;

}

}

function onListening() {

const addr = server.address();

const bind = typeof addr === 'string'

? 'pipe' + addr

: 'port' + addr.port;

debug ('Listenig on ' + bind)

}