

Criando uma aplicação

- No console digitar:

npm init

npm = É o gerenciador de pacotes padrão javascript para o ambiente de tempo de execução do JavaScript Node.js.

C:\Windows\system32\cmd.exe

^C

C:\Temp>

C:\Temp>npm init

This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See 'npm help json' for definitive documentation on these fields
and exactly what they do.

Use 'npm install <pkg>' afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.

package name: <temp> backend

version: <1.0.0> 1

Invalid version: "1"

version: <1.0.0> 1.0

Invalid version: "1.0"

version: <1.0.0> 1.0.0

description: projetoBD

entry point: <exemplo1.js> server.js

test command:

git repository:

keywords:

author: Marcia

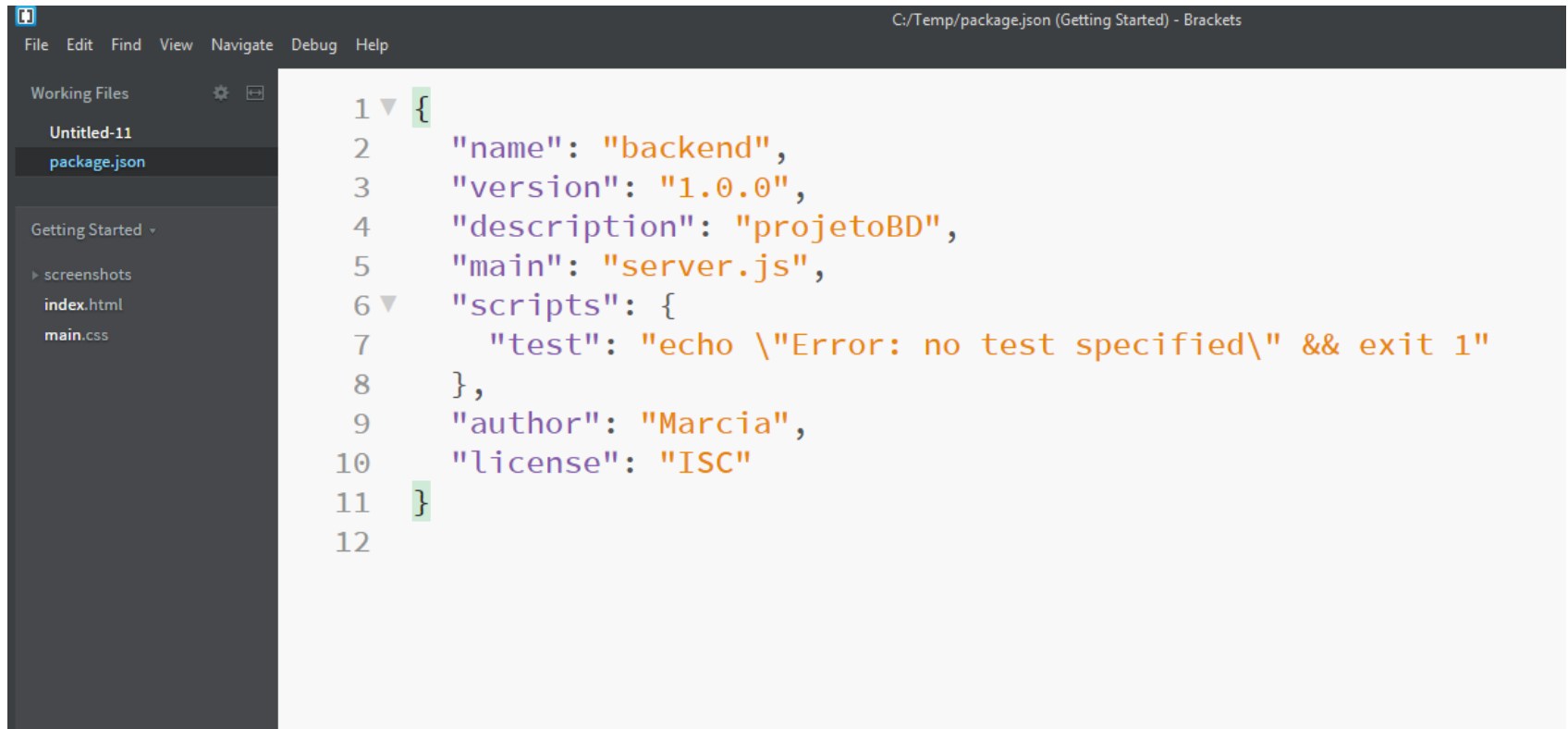
license: <ISC> 1

Sorry, license should be a valid SPDX license expression (without "LicenseRef"),
"UNLICENSED", or "SEE LICENSE IN <filename>".

license: <ISC> ISC

About to write to C:\Temp\package.json:

```
{
  "name": "backend",
  "version": "1.0.0",
  "description": "projetoBD",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Marcia",
  "license": "ISC"
}
```

A screenshot of the Brackets code editor interface. The top menu bar includes 'File', 'Edit', 'Find', 'View', 'Navigate', 'Debug', and 'Help'. The title bar shows the file path 'C:/Temp/package.json (Getting Started) - Brackets'. On the left, the 'Working Files' panel lists 'Untitled-11' and 'package.json'. Below it, the 'Getting Started' panel shows 'screenshots', 'index.html', and 'main.css'. The main editor area displays the content of 'package.json' with line numbers 1 through 12. The JSON is formatted with syntax highlighting: keys are in purple, string values in orange, and the test command in a mix of orange and purple. The code defines a project named 'backend' with version '1.0.0', description 'projetoBD', main file 'server.js', and a test script that echoes an error message and exits with code 1. The author is 'Marcia' and the license is 'ISC'.

```
1 {  
2   "name": "backend",  
3   "version": "1.0.0",  
4   "description": "projetoBD",  
5   "main": "server.js",  
6   "scripts": {  
7     "test": "echo \"Error: no test specified\" && exit 1"  
8   },  
9   "author": "Marcia",  
10  "license": "ISC"  
11 }  
12
```

Agora sim, existe um projeto NODE!

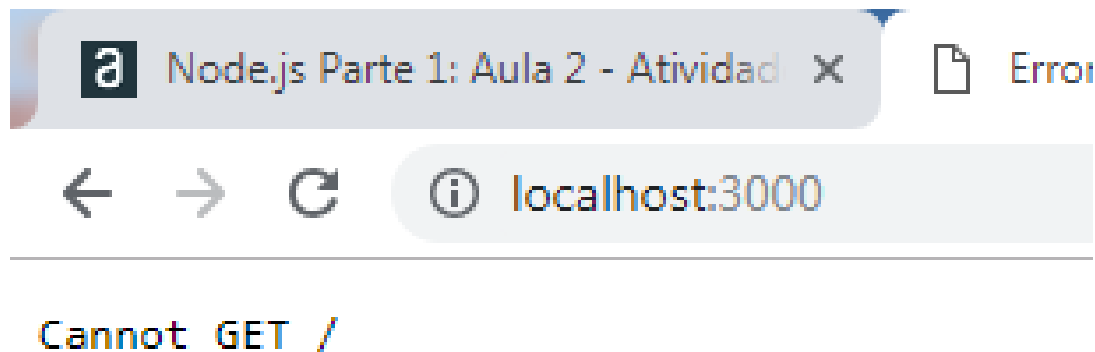
Instalando o módulo express:

```
C:\Temp>
C:\Temp>npm install express@4.16.3 --save-exact
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN backend@1.0.0 No repository field.
+ express@4.16.3
added 50 packages in 11.548s
C:\Temp>
```

```
1  {
2    "name": "backend",
3    "version": "1.0.0",
4    "description": "projetoBD",
5    "main": "server.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1"
8    },
9    "author": "Marcia",
10   "license": "ISC",
11   "dependencies": {
12     "express": "4.16.3"
13   }
14 }
15
```

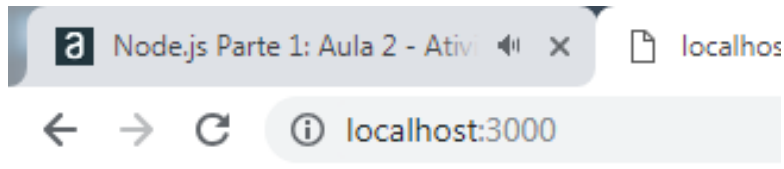
Colocando o servidor no ar:

```
C:\Temp>node server.js  
Servidor rodando na porta 3000
```

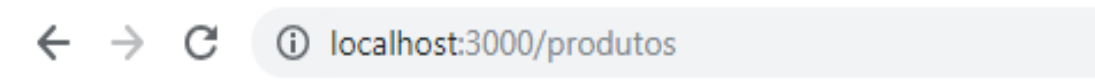


O servidor está “falando” que não está conseguindo retornar alguma resposta usando o GET. Isso ocorreu porque não criamos nenhuma ROTA para uma aplicação.

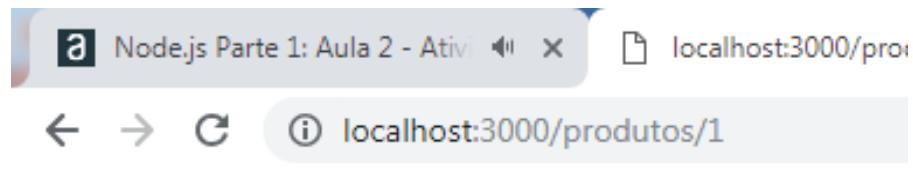
Agora, com a resposta “programada”:



Ola GENTE!!!

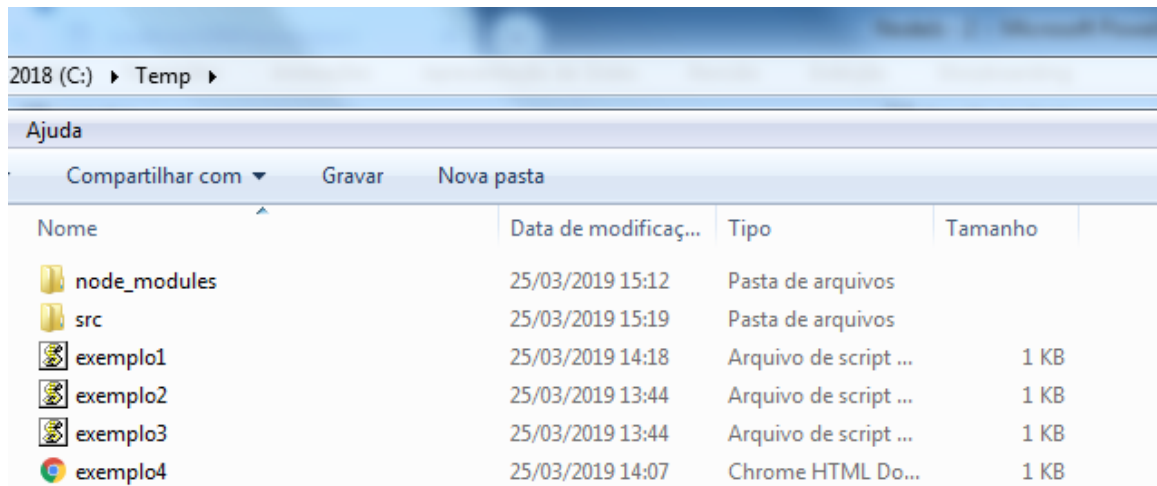


LISTAGEM DE PRODUTOS!!!



Dados do Produto 1!!

Pasta **NODE_MODULES**: conter todas as dependências da aplicação. Tudo o que for necessário para “rodar” a aplicação serão baixadas dentro desta pasta. Todas as bibliotecas baixadas estarão de acordo com o SO. Para compartilhar arquivos node, remover essa pasta! Nesse caso, o package json já saberá quais são as dependências.

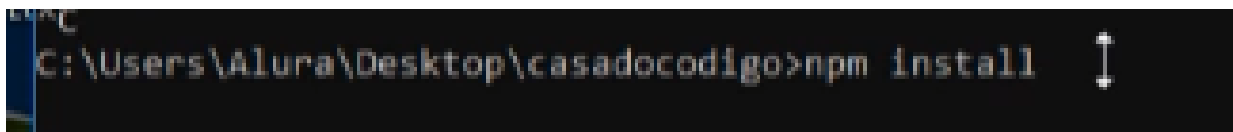


2018 (C:) > Temp

Ajuda

Compartilhar com ▾ Gravar Nova pasta

Nome	Data de modificaç...	Tipo	Tamanho
node_modules	25/03/2019 15:12	Pasta de arquivos	
src	25/03/2019 15:19	Pasta de arquivos	
exemplo1	25/03/2019 14:18	Arquivo de script ...	1 KB
exemplo2	25/03/2019 13:44	Arquivo de script ...	1 KB
exemplo3	25/03/2019 13:44	Arquivo de script ...	1 KB
exemplo4	25/03/2019 14:07	Chrome HTML Do...	1 KB



```
C:\Users\Alura\Desktop\casadocodigo>npm install
```

Encapsulando o express...

```
debug Help C:/Temp/server-v2.js (Getting Started) - Brackets

1  const app = require('./src/config/express-customizado');
2
3  app.listen(3000, function() {
4    console.log('Servidor rodando na porta 3000');
5  });
6
7  // 1o rota
8  app.get('/', function(req, resp) {
9    resp.send('<html><head><title></title></head><body><h1>Ola GENTE!!!</h1></body><html>');
10  });
11
12  // 2o rota
13  app.get('/produtos', function(req, resp) {
14    resp.send('<html><head><title></title></head><body><h1>LISTAGEM DE PRODUTOS!!!</h1></body><html>');
15  });
```

```
debug Help C:/Temp/src/config/express-customizado.js (Getting Started) - Brackets

1  const express = require('express');
2  const app = express();
3
4  // dizendo para o node que meu modulo vai exportar app
5  // estou encapsulando o express
6  module.exports = app;
```


Encapsulando as rotas...

No diretório: src/app/rotas:

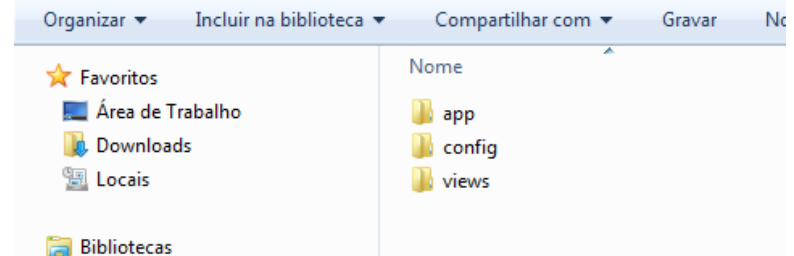
```
Debug Help C:/Temp/src/app/rotas/rotas.js (Getting Started) - Brackets
1 // estou exportando uma função capaz de receber um parametro
2 module.exports = (app) => {
3   // 1o rota
4   app.get('/', function(req,resp) {
5     resp.send('<html><head><title></title></head><body><h1>Ola GENTE!!!</h1></body></html>');
6   });
7
8   // 2o rota
9   app.get('/produtos', function(req,resp) {
10    resp.send('<html><head><title></title></head><body><h1>LISTAGEM DE PRODUTOS!!!</h1></body></html>');
11  });
12
13  // 3o rota
14  app.get('/produtos/1', function(req,resp) {
15    resp.send('<html><head><title></title></head><body><h1>Dados do Produto 1!!</h1></body></html>');
16  });
17  };
18
```

```
1 const express = require('express');
2 const app = express();
3
4 const rotas = require('../app/rotas/rotas');
5 rotas(app);
6
7 // dizendo para o node que meu modulo vai exportar app
8 // estou encapsulando o express
9 module.exports = app;
```

Express + Marko...

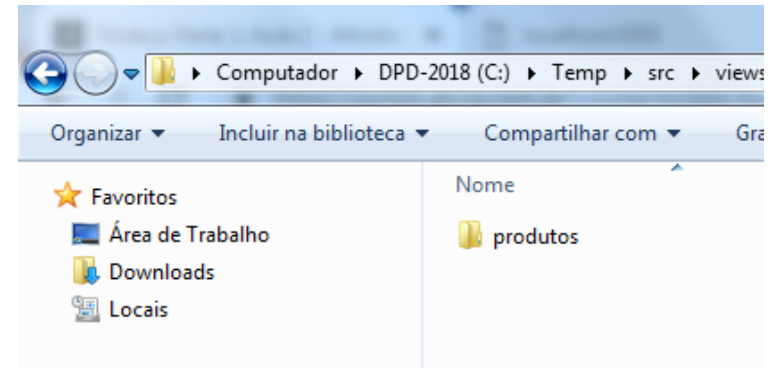
<https://markojs.com/docs/express/>

```
C:\Temp>npm install marko --save
npm WARN backend@1.0.0 No repository field.
+ marko@4.16.3
added 59 packages in 12.062s
C:\Temp>
```



```
• C:/Temp/src/config/express-customizado.js (Getting Started) - Brackets
bug Help

1 // para habilitar o marko
2 require('marko/node-require').install();
3 require('marko/express');
4
5 const express = require('express');
6 const app = express();
7
8 const rotas = require('../app/rotas/rotas');
9 rotas(app);
10
11 // dizendo para o node que meu modulo vai exportar app
12 // estou encapsulando o express
13 module.exports = app;
```



Acesso ao BD

Informações importantes a observar:

- O nome do BD reflete o nome do projeto
- Não usar hífen
- Separa-se palavras com underlines
- O nome das tabelas que representam entidades será no plural
- Todas as palavras são em letras minúsculas

Acesso ao BD

Para nosso exemplo inicial, vamos criar o BD chamado LOJA.
Criando as tabelas CLIENTES, CATEGORIAPROD e PRODUTOS:

- CREATE DATABASE LOJA;
- USE LOJA;

CLIENTES				
Campo	Tipo	Null?	PK	Info extra
id_cliente	Int	Não	Sim	Auto incremento
nome_cliente	Varchar(40)	Não	Não	
data_niver	smallDateTime	Não	Não	
email	Varchar(60)	Não	Não	

Acesso ao BD

CATEGORIAPRODUTOS				
Campo	Tipo	Null?	PK	Info extra
id_categ_prod	Int	Não	Sim	Auto incremento
desc_categ_prod	Varchar(20)	Não	Não	

PRODUTOS				
Campo	Tipo	Null?	PK	Info extra
id_prod	Int	Não	Sim	Auto incremento
desc_produto	Varchar(40)	Não	Não	
preco_produto	Float	Não	Não	
id_categ_prod	Int	Não	Não	

Acesso ao BD

```
CREATE TABLE CLIENTES (  
  id_cliente int NOT NULL AUTO_INCREMENT,  
  nome_cliente varchar(40) NOT NULL,  
  data_niver smalldatetime NOT NULL,  
  email varchar(60) NOT NULL,  
  PRIMARY KEY(id_cliente));
```

Acesso ao BD

```
CREATE TABLE CATEGORIAPRODUTOS (  
  id_categ_prod int NOT NULL AUTO_INCREMENT,  
  desc_categ_prod varchar(40) NOT NULL,  
  PRIMARY KEY(id_categ_prod));
```

```
CREATE TABLE PRODUTOS (  
  id_prod int NOT NULL AUTO_INCREMENT,  
  desc_produto varchar(40) NOT NULL,  
  preco_produto float NOT NULL,  
  id_categ_prod int NOT NULL,  
  PRIMARY KEY(id_prod)  
  FOREIGN KEY(id_categ_prod) REFERENCES  
  CATEGORIAPRODUTOS(id_categ_prod));
```

Acesso ao BD

```
INSERT INTO CLIENTES VALUES (NULL,'Monica Ferraz','12/04/1990','ferraz@ig.com.br')
INSERT INTO CLIENTES VALUES (NULL,'Luiz Castro','10/01/1972','castro@gmail.com')
INSERT INTO CLIENTES VALUES (NULL,'Lucia Gomes','27/07/1972','lucia@hotmail.com')
INSERT INTO CLIENTES VALUES (NULL,'Paulo Vidal','23/07/2002','pvidal@ig.com.br')
```

```
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Esportivos')
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Festa')
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Confort')
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Crocs')
```

```
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Tenis NIKE X',678.90,1)
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Tenis Adidas Y',566.99,1)
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Viviana',351.90,2)
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Arezzo',432.87,2)
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Picadilly',223.87,3)
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Tradicional',432.87,4)
INSERT INTO CATEGORIAPRODUTOS VALUES (NULL,'Sapatilha',289.90,4)
```


Acesso ao BD

```
INSERT INTO CLIENTES VALUES (NULL,'Monica Ferraz','12/04/1990','ferraz@ig.com.br')
```

```
INSERT INTO CLIENTES VALUES (NULL,'Beatriz Segal','12/04/1990','bsegal@cotuca.br')
```

```
INSERT INTO CLIENTES VALUES (NULL,'Luciana Vasconcelos','30/03/2000','luvaz@ig.com.br')
```

```
INSERT INTO CLIENTES VALUES (NULL,'Monica Ferraz','21/04/1972','ferraz@ig.com.br')
```

```
INSERT INTO CLIENTES VALUES (NULL,'Anderson Nascimento','19/08/1990','nascimento@ig.com.br')
```