

Iniciaremos em alguns instantes...

“As aulas disponibilizadas por meios digitais envolvem horas de estudo, pesquisa e organização dos(as) professores(as) responsável(eis) e destinam-se, única e exclusivamente, aos alunos regularmente matriculados em cursos da Universidade Cruzeiro do Sul, para fins acadêmicos

A disponibilização/ divulgação dos links de acesso às aulas síncronas e/ ou registros das aulas a colegas não matriculados e a indivíduos estranhos à comunidade acadêmica da universidade implicará responsabilização civil e criminal conforme Artigo 46, inciso IV da Lei 9610/98, combinado com o artigo 184 do Código Penal, que estabelece direito autoral ao docente.

*Assim, em atendimento à Lei e à valorização do trabalho docente, solicita-se que **não compartilhem as aulas com indivíduos que não estejam regularmente matriculados.**”*



A hand is pointing at a computer screen that displays several lines of JavaScript code. The code is color-coded, with green for text and blue for tags. The visible code includes several instances of `<script type="text/javascript">`. The background of the slide is dark gray with a white curved line separating the image from the title area.

Técnicas de Programação



Técnicas de Programação. - Prof. Msc. Luiz C M Lozano

Node.js



Técnicas de Programação. - Prof. Msc. Luiz C M Lozano

Download do Node

nodejs.org



[HOME](#) | [ABOUT](#) | [DOWNLOADS](#) | [DOCS](#) | [GET INVOLVED](#) | [SECURITY](#) | [CERTIFICATION](#) | [NEWS](#)

Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#).

[#BlackLivesMatter](#)

New security releases now available for 15.x, 14.x, 12.x and 10.x release lines

Download for Windows (x64)

14.16.0 LTS

Recommended For Most Users

15.11.0 Current

Latest Features

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

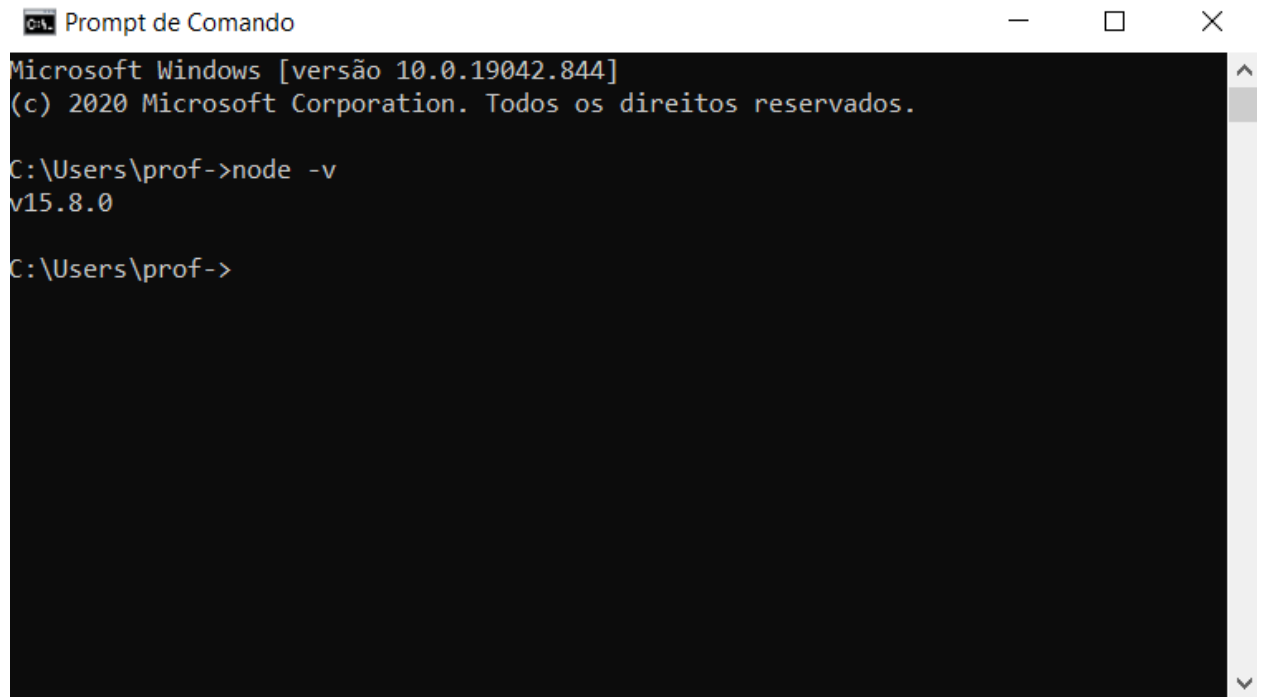
Or have a look at the [Long Term Support \(LTS\)](#) schedule.



Técnicas de Programação. - Prof. Msc. Luiz C M Lozano

Verificando a
versão
instalada

node -v



```
Prompt de Comando

Microsoft Windows [versão 10.0.19042.844]
(c) 2020 Microsoft Corporation. Todos os direitos reservados.

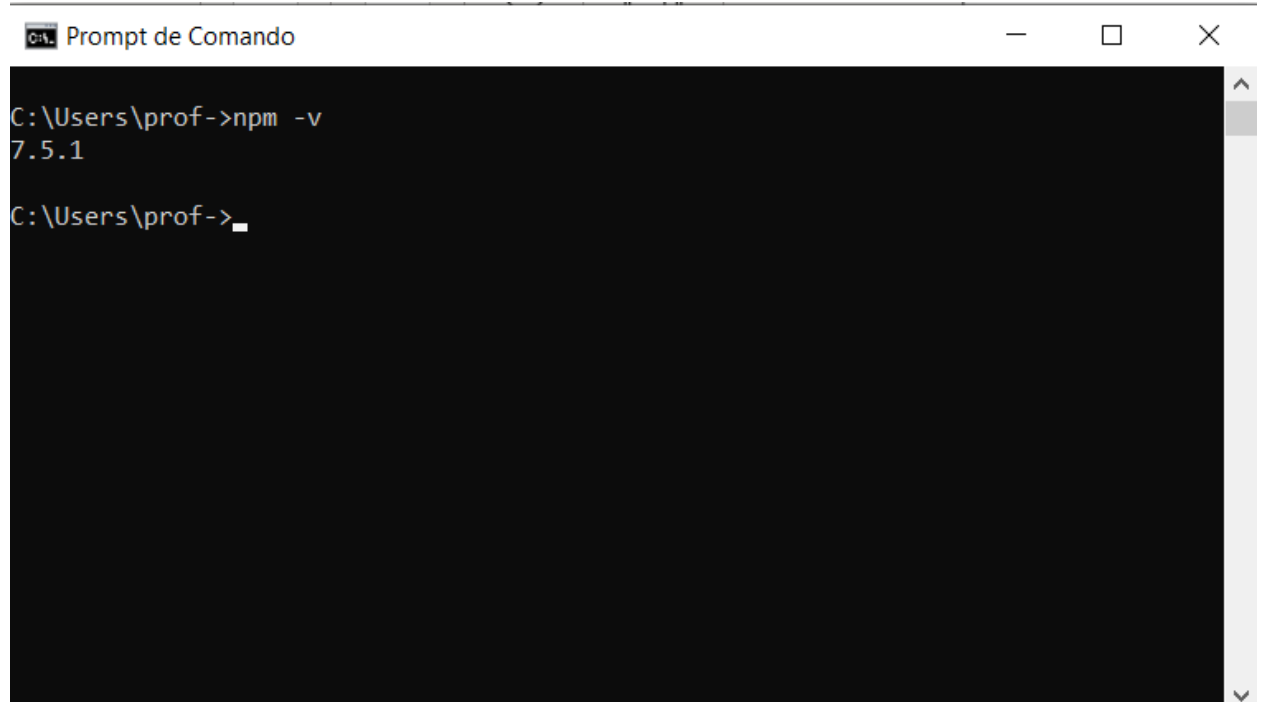
C:\Users\prof->node -v
v15.8.0

C:\Users\prof->
```



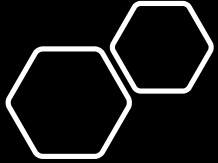
Verificando a
instalação do
NPM

npm -v



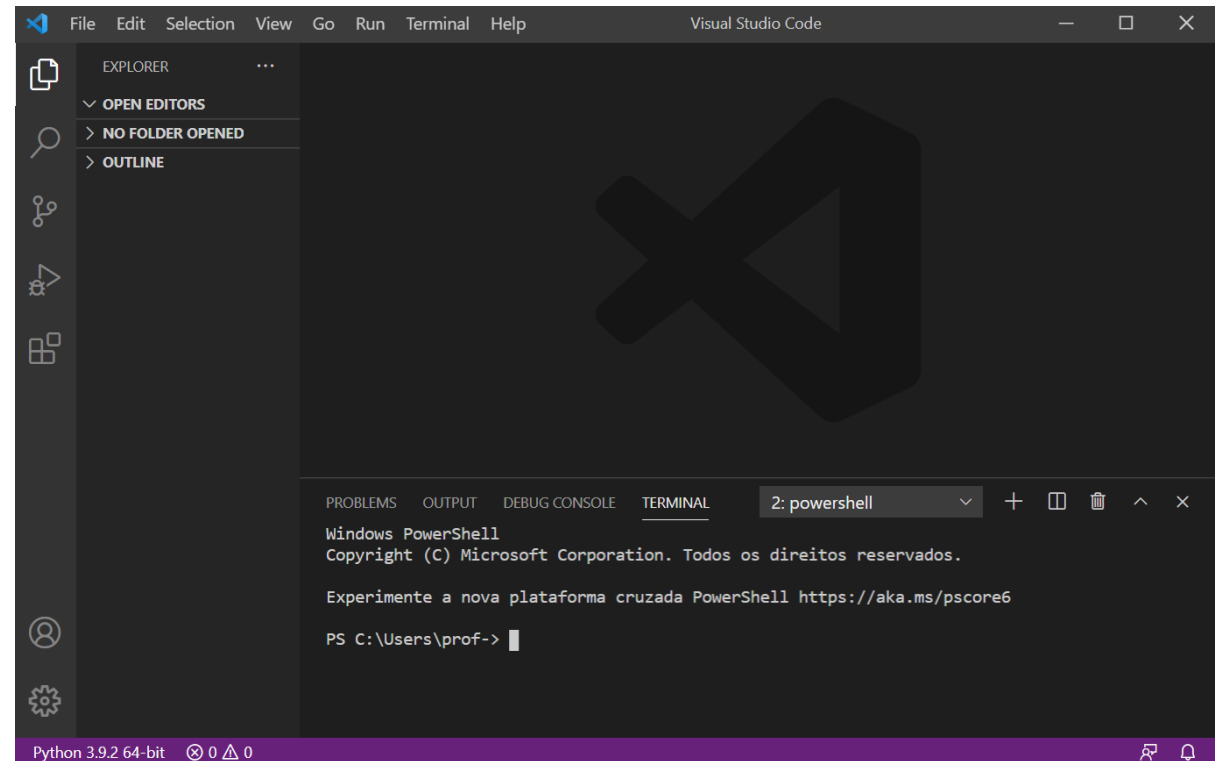
```
C:\Users\prof->npm -v
7.5.1
C:\Users\prof->
```





Iniciando um primeiro projeto...

Dentro do VSCode
acesse a opção
terminal/new
terminal.



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\> mkdir projetos
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\projetos> mkdir projetonode
```

Armazenando o Projeto



Técnicas de Programação. - Prof. Msc. Luiz C M Lozano

Iniciando
nosso
projeto...

Dentro da pasta projetonode
(nossa pasta do projeto atual,
digite o comando npm init).

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  
PS C:\projetos\projetonode> npm init
```

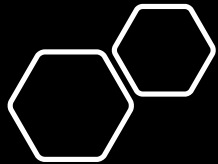


Configuração do arquivo JSON

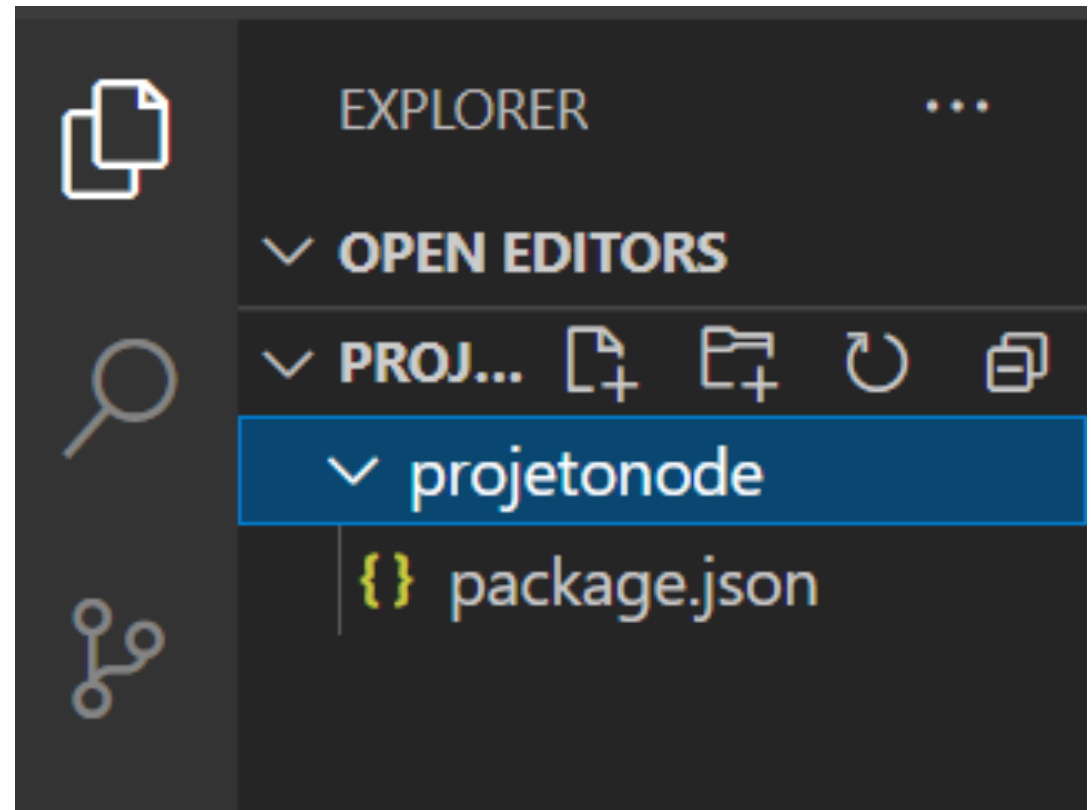
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  2: node  ▾  +  
  
It only covers the most common items, and tries to guess sensible defaults.  
  
See `npm help init` for definitive documentation on these fields  
and exactly what they do.  
  
Use `npm install <pkg>` afterwards to install a package and  
save it as a dependency in the package.json file.  
  
Press ^C at any time to quit.  
package name: (projetonode) |
```

Dar enter em todas as opções nesse momento,
posteriormente iremos alterar essas configurações.



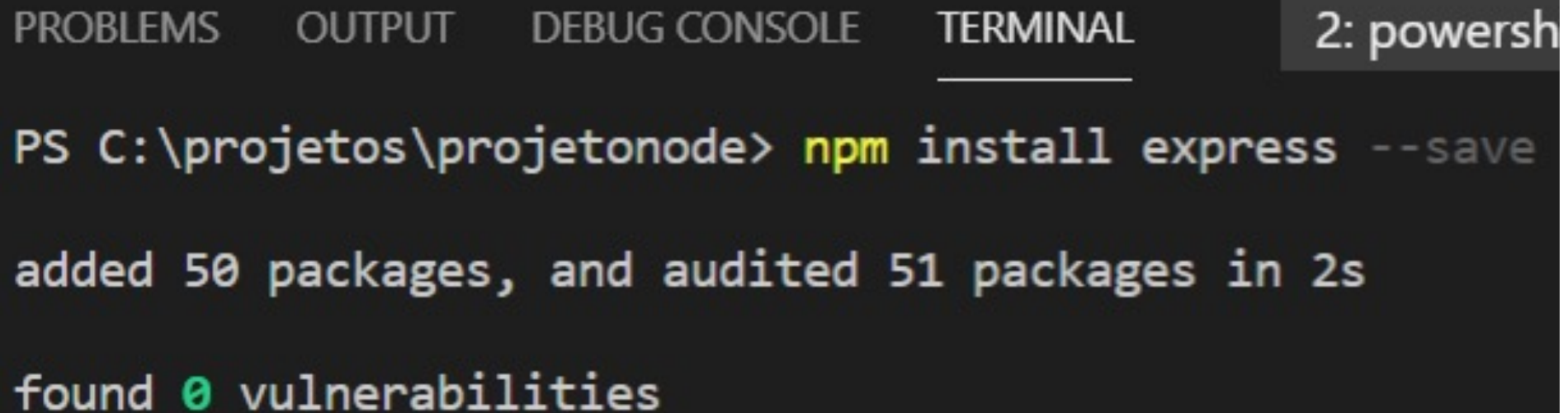


Arraste a
pasta do
projeto
p/ dentro
do Editor



Instalação de dependências no projeto

No terminal digite `npm install express -- save`



The screenshot shows a terminal window with tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The TERMINAL tab is active, showing a PowerShell prompt at 'C:\projetos\projetonode'. The command 'npm install express --save' has been executed. The output indicates that 50 packages were added and 51 packages were audited in 2 seconds, with 0 vulnerabilities found.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  2: powersh  
PS C:\projetos\projetonode> npm install express --save  
  
added 50 packages, and audited 51 packages in 2s  
  
found 0 vulnerabilities
```



Instalação de dependências no projeto

No terminal digite `npm install nodemon --save-dev`

```
PS C:\projetos\projetonode> npm install nodemon --save-dev
up to date, audited 169 packages in 851ms
11 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```



Abra o arquivo
package.json

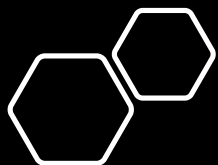
`{}` package.json ✕

projetonode > `{}` package.json > ...

```
1  {
2    "name": "projetonode",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1"
8    },
9    "author": "",
10   "license": "ISC",
11   "dependencies": {
12     "express": "^4.17.1"
13   },
14   "devDependencies": {
15     "nodemon": "^2.0.7"
16   }
17 }
```

Veja que as
dependências
foram adicionadas
ao projeto.





Ajeitando o arquivo package.json

Para que o arquivo fique mais conciso, iremos fazer algumas alterações:

`{}` package.json X

projetonode > `{}` package.json > ...

```
1  {
2    "name": "projetonode",
3    "version": "1.0.0",
4    "scripts": {
5
6    },
7
8    "dependencies": {
9      "express": "^4.17.1"
10   },
11   "devDependencies": {
12     "nodemon": "^2.0.7"
13   }
14 }
```

No scripts inserimos atalhos





Ajeitando o arquivo
package.json

Criando
atalho do
start:

`{}` package.json ✕

projetonode > `{}` package.json > ...

```
1  {
2    "name": "projetonode",
3    "version": "1.0.0",
4    "scripts": {
5      "start": "nodemon ./server.js"
6    },
7
8    "dependencies": {
9      "express": "^4.17.1"
10   },
11   "devDependencies": {
12     "nodemon": "^2.0.7"
13   }
14 }
```

Endereço do servidor

Para iniciar o servidor (quando ele for criado), digitaremos no terminal `npm start`



Técnicas de Programação. - Prof. Msc. Luiz C M Lozano

Criando nossa aplicação...

Antes de trabalhar no servidor, vamos criar nossa aplicação;

Crie um arquivo chamado app.js;

```
JS app.js > ...
1  //fazer a importação do express
2  const express = require('express');
3
4  //Rotas (quando o servidor for acessado, pra onde ele vai?)
5  const router = express.Router();
6  router.get('/',(req, res)=>{ //primeira rota //função anônima//dois parametros
7    res.send('Olá Mundo!');
8  });
9
10 //configurações básicas do aplicativo
11 const app = express();
12 app.use('/',router); //foi passado 1 rota pois criamos apenas 1
13
14 module.exports = app; //exportamos o app, pois vamos importa-lo no servidor
```



Vamos criar o nosso servidor...

Crie um novo arquivo chamado server.js;

Nesse arquivo iremos puxar nossa aplicação usando o express;

Fazer a aplicação rodar em uma porta específica;

Iniciar o servidor;

```
JS server.js > ...
1  const app = require('./app');
2
3  app.set('port',7777); //escolha uma porta padrão
4  const server = app.listen(app.get('port'),()=>{
5    console.log("Servidor rodando na porta: "+server.address().port);
6  });
7
```



Iniciando o servidor...

Abra o terminal do VSCode, se certifique de estar dentro da pasta do projeto e digite npm start.

Com o nodemon, toda vez que houver uma alteração no aplicativo ele irá reiniciar o servidor de forma automática.

```
PS C:\projetos\projetonode> npm start
```

```
> projetonode@1.0.0 start
```

```
> nodemon ./server.js
```

Veja que o servidor esta sendo invocado pelo nodemon

```
[nodemon] 2.0.7
```

```
[nodemon] to restart at any time, enter `rs`
```

```
[nodemon] watching path(s): *.*
```

```
[nodemon] watching extensions: js,mjs,json
```

```
[nodemon] starting `node ./server.js`
```

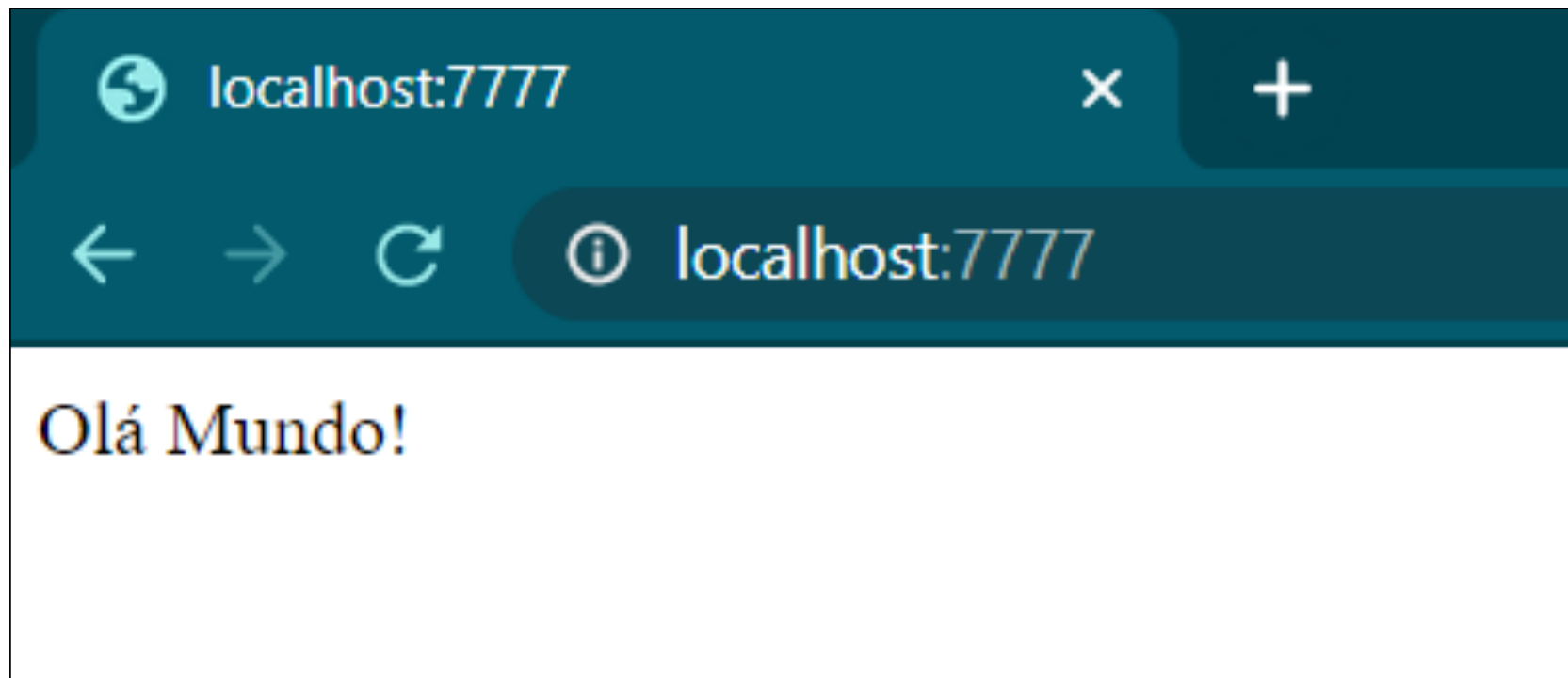
```
Servidor rodando na porta: 7777
```

Nossa mensagem no console.log



Testando nosso projeto...

Acesse o navegador e entre no endereço localhost:porta



Fazendo alteração no projeto...

Altere no arquivo app.js o escrito para Olá Mundo 2021;

Ao salvar o arquivo você irá notar no terminal que o servidor foi reiniciado automaticamente, isso se deve ao Nodemon.

```
res.send('Olá Mundo 2021!');
```

```
PS C:\projetos\projetonode> npm start
```

```
> projetonode@1.0.0 start
```

```
> nodemon ./server.js
```

```
[nodemon] 2.0.7
```

```
[nodemon] to restart at any time, enter `rs`
```

```
[nodemon] watching path(s): *.*
```

```
[nodemon] watching extensions: js,mjs,json
```

```
[nodemon] starting `node ./server.js`
```

```
Servidor rodando na porta: 7777
```

```
[nodemon] restarting due to changes...
```

```
[nodemon] starting `node ./server.js`
```

```
Servidor rodando na porta: 7777
```

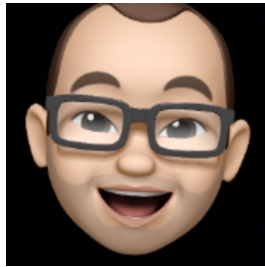
```
[nodemon] restarting due to changes...
```

```
[nodemon] starting `node ./server.js`
```

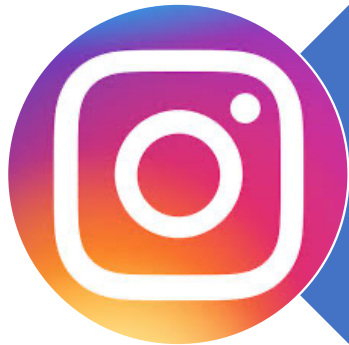
```
Servidor rodando na porta: 7777
```



Siga-me...



<https://www.facebook.com/luizmlozano>



<https://www.instagram.com/luizmlozano>



Técnicas de Programação. - Prof. Msc. Luiz C M Lozano

Créditos:

Material parcialmente elaborado
pelo Prof. Alexander Gobatto e
adaptado pelo Prof. Luiz Lozano



Técnicas de Programação. - Prof. Msc. Luiz C M Lozano