

## NODEJS

Node.js é um ambiente de execução JavaScript que permite executar aplicações desenvolvidas com a linguagem de forma autônoma, sem depender de um navegador. Com ele, é possível criar praticamente qualquer tipo de aplicações web, desde servidores para sites estáticos e dinâmicos, até APIs e sistemas baseados em microserviços.

### Mãos a massa

1- Vamos baixar o NodeJs e instalar.

2- Abra o cmd

3- Digite os seguintes comandos

```
cd/  
mkdir PastaNode  
cd PastaNode
```

4 -Abra o VsCode e abra a pasta criada.

```
Crie o arquivo index.js  
Digite  
console.log("OLA MUNDO")
```

Abra o prompt de comando (cmd)

Execute o comando → node index

```
C:\PastaNode>node index  
OLA MUNDO
```

## Vamos criar nosso primeiro arquivo de servidor

```

1  /* Aqui estamos atribuindo à variável "http" as propriedades do
2  modulo http (nativo do NodeJs) */
3
4  let http = require("http");
5
6  /* Declarando as constantes como porta e o nome do servidor que vamos usar
7  vamos usar a porta 80(mas pode ser qualquer uma), já o servidor já é o famoso
8  número que conhecemos "127.0.0.1" */
9
10 const port = 80;
11 const hostname = "127.0.0.1"
12
13 /*criando o servidor, vai receber as propriedades do método createServer() o
14 req --> são as requisições que iremos processar
15 res --> são as respostas de devemos ou queremos passar*/
16
17 servidor = http.createServer((req,res)=>{
18     res.statusCode =200;
19     res.setHeader("Content-Type","text/plain");
20     res.end("OLA MUNDO - PRIMEIRO ARQUIVO DE SERVIDOR")
21 })
22
23 /* Agora vamos fazer o nosso "servidor" escutar e executar os processos no NodeJs */
24
25 servidor.listen(port,hostname,()=>{
26     console.log("SERVIDOR RODANDO")
27 })

```



Executando o Node

```

C:\PastaNode>node index
OLA MUNDO

```

← → ↻ ⓘ localhost

 Gmail  YouTube  Email – Evandro Jos...  Node Studio Treina...  Curso

OLA MUNDO - PRIMEIRO ARQUIVO DE SERVIDOR

**VAMOS CHAMAR UM ARQUIVO EXTERNO**

Crie o arquivo chamado **index.html**

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Index</title>
</head>
<body>
  <h2>ARQUIVO INDEX</h2>
</body>
</html>
```

```
 2 let http = require("http");
 3
 4 /* A variável lerArquivo vai receber as propriedades do modulo "fs" (file system - Sistema de Arquivo) */
 5
 6 let lerArquivo = require("fs");
 7
 8 const port = 80;
 9 const hostname = "127.0.0.1"
10
11 /*criando o servidor, vai receber as propriedades do método createServer() o
12 Agora estamos usando a variável lerArquivo com a propriedade readFile() */
13
14 servidor = http.createServer((req,res)=>{
15   lerArquivo.readFile("index.html",(err,data)=>{
16     res.writeHead(200,{"Content-Type":"text/html"});
17     res.write(data);// o data aqui se refere ao arquivo "index.html"
18     res.end();
19   })
20
21 })
22
23 servidor.listen(port,hostname,()=>{
24   console.log("SERVIDOR RODANDO")
25 })
26
```

Crie o arquivo chamado **usuario.html**

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Index</title>
</head>
<body>
  <h2>ARQUIVO USUÁRIO</h2>
</body>
</html>
```

Crie o arquivo chamado **admin.html**

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Index</title>
</head>
<body>
  <h2>ARQUIVO ADMIN</h2>
</body>
</html>
```

```
index.js > ...
1  let http = require("http");
2  let lerArquivo = require("fs");
3  const port = 80;
4  const hostname = "127.0.0.1"
5
6  servidor = http.createServer((req,res)=>{
7    switch(req.url){
8      case "/":
9        lerArquivo.readFile("index.html",(err,data)=>{
10         res.writeHead(200,{"Content-Type":"text/html"});
11         res.write(data);// o data aqui se refere ao arquivo "index.html"
12         res.end();
13       })
14       break
15     case "/usuario":
16       lerArquivo.readFile("usuario.html",(err,data)=>{
17         res.writeHead(200,{"Content-Type":"text/html"});
18         res.write(data);// o data aqui se refere ao arquivo "usuario.html"
19         res.end();
20       })
21       break
22     case "/usuario/admin":
23       lerArquivo.readFile("admin.html",(err,data)=>{
24         res.writeHead(200,{"Content-Type":"text/html"});
25         res.write(data);// o data aqui se refere ao arquivo "admin.html"
26         res.end();
27       })
28     }
29   })
30
31  servidor.listen(port,hostname,()=>{
32    console.log("SERVIDOR RODANDO")
33  })
```

## Exercício

1 – Crie uma pasta com o seu nome.

Exemplo: **Evandro\_Vieira** (sem espaços)

2 – Dentro dele crie o arquivo **index.html** com o seguinte código

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exercicio 01</title>
</head>
<body>
  <h2>Tela Principal</h2>
  <a href="/soma_numeros">Soma Números</a>
  <br>
  <a href="/conta_letras">Conta Letras</a>
</body>
</html>
```

3 – Crie o arquivo **soma\_numeros.html** com os seguintes códigos

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Soma Números</title>
</head>
<body>
  <p>Digite primeiro número: <input type="text" id="num1" size="5"></p>
  <p>Digite segundo número: <input type="text" id="num2" size="5"></p>
  <p>Resultado: <input type="text" id="resultado" size="5" disabled></p>
  <button id="somar">SOMAR</button>
  <button id="voltar">VOLTAR</button>
</body>
<script>
  <!-- Aqui o codigo Javascript -->
</script>
</html>
```

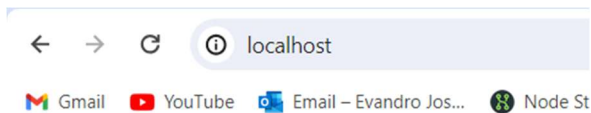
### 3 – Crie o arquivo **conta\_letras.html** com os seguintes códigos

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Conta Letras</title>
</head>
<body>
  <p>Digite uma frase: <input type="text" id="texto"></p>
  <p>Qtde Letras <input type="text" id="resultado" size="5" disabled></p>
  <button id="contar">CONTAR</button>
  <button id="voltar">VOLTAR</button>
</body>
<script>
<!-- Aqui Javascript -->
</script>
</html>
```

Agora você irá fazer as chamadas destes arquivos através do NODEJS, crie o arquivo **index.js** e faça as devidas configurações. Veja o funcionamento abaixo

**ATENÇÃO USE A PORTA 80, MAS NÃO ABRA O XAMPP**

```
C:\Evandro_Vieira>node index
SERVIDOR RODANDO
```



## Tela Principal

[Soma Números](#)  
[Conta Letras](#)

localhost/soma\_numeros

Digite primeiro número:

Digite segundo número:

Resultado:

localhost/conta\_letras

Digite uma frase:

Qtde Letras