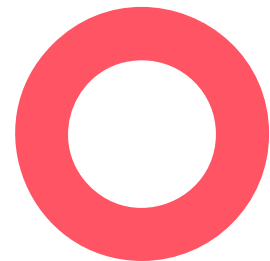


DESENVOLVIMENTO DE SISTEMAS

UC13

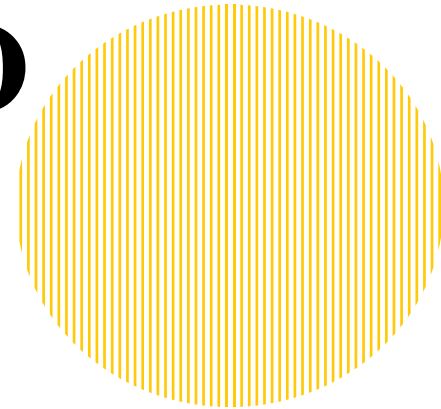
Prof. Viviane de Lima

viviane.lfrancelino@sp.senac.br

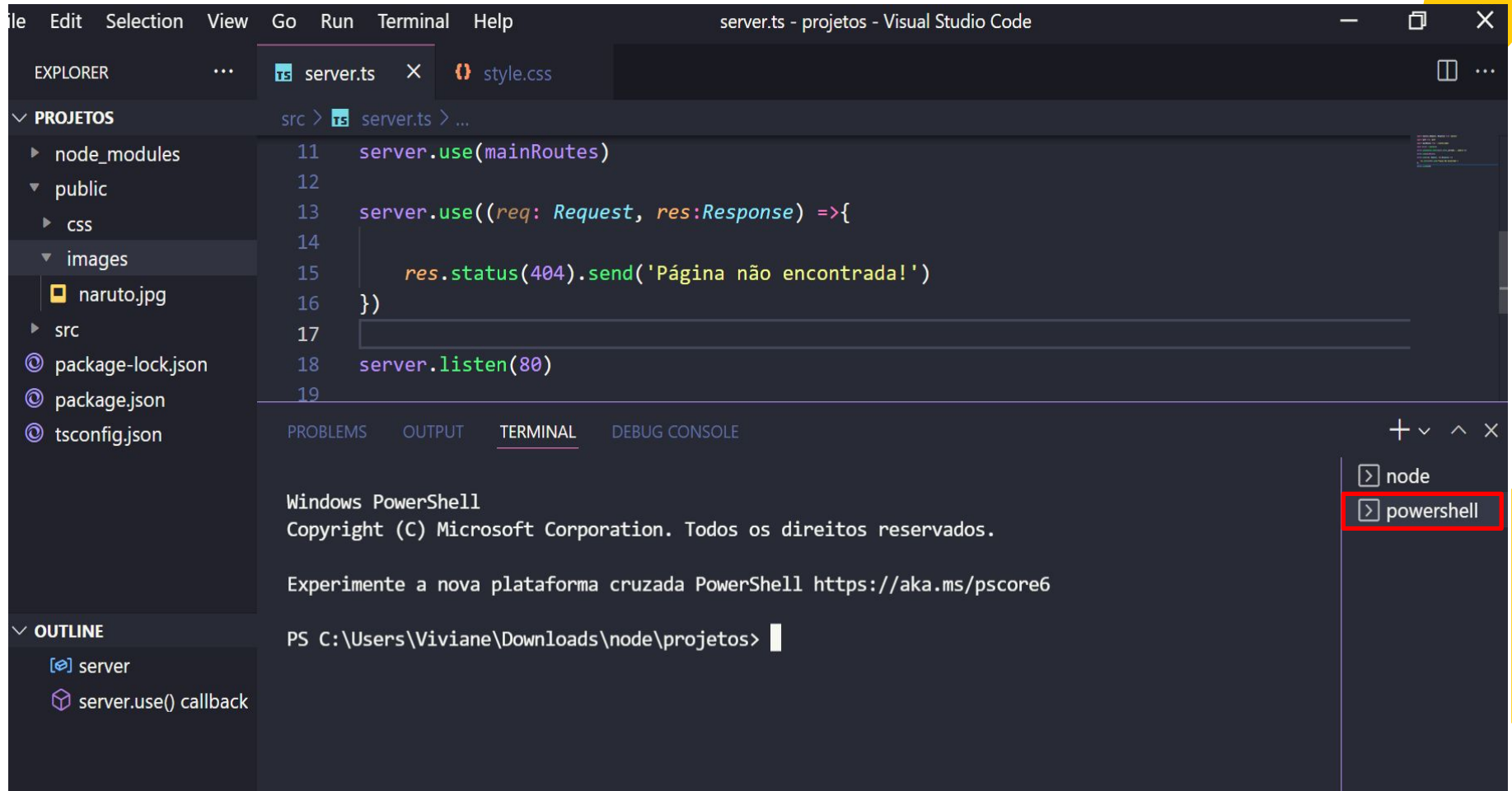


AULA 04

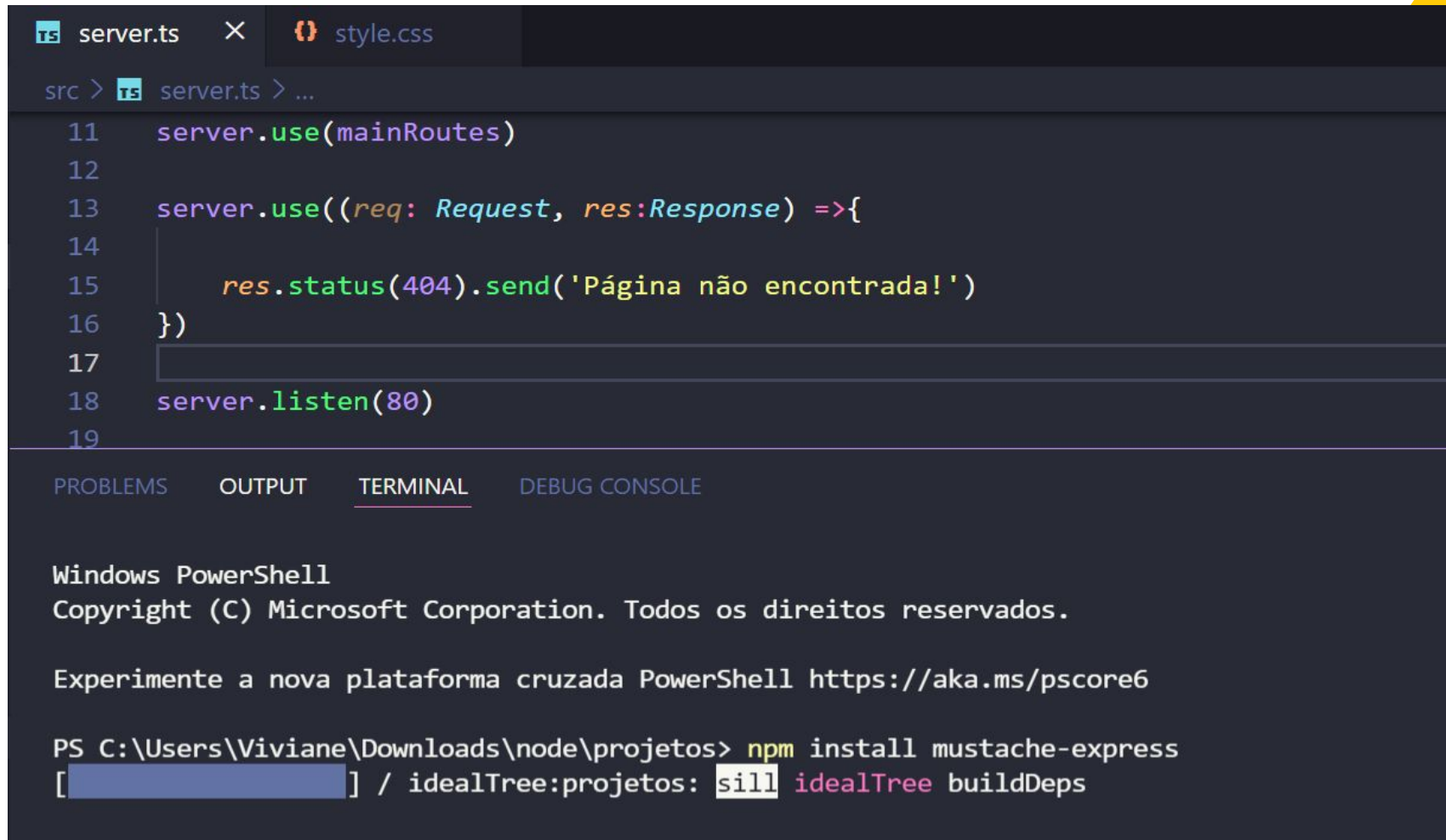
**INSTALANDO E
CONFIGURANDO O
MUSTACHE**



ABRA UM NOVO TERMINAL NO VSCODE



DIGITE: NPM INSTALL MUSTACHE-EXPRESS



The image shows a Visual Studio Code editor window with two tabs: 'server.ts' and 'style.css'. The 'server.ts' tab is active, displaying a TypeScript file with the following code:

```
11 server.use(mainRoutes)
12
13 server.use((req: Request, res: Response) =>{
14     res.status(404).send('Página não encontrada!')
15 })
16
17
18 server.listen(80)
19
```

Below the editor, the 'TERMINAL' tab is selected, showing the output of a Windows PowerShell session. The terminal displays the copyright notice for Microsoft Corporation and a prompt to experiment with the PowerShell platform. The command 'npm install mustache-express' has been executed, and the output shows the package being installed in the directory 'C:\Users\Viviane\Downloads\node\projetos'.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Users\Viviane\Downloads\node\projetos> npm install mustache-express
[ ] / idealTree:projetos: sill idealTree buildDeps
```

INSTALANDO OS TYPES DO MUSTACHE

**NPM INSTALL --SAVE-DEV
@TYPES/MUSTACHE-EXPRESS**



PROBLEMS

OUTPUT

TERMINAL

DEBUG CONSOLE

```
PS C:\Users\Viviane\Downloads\node\projetos> npm install --save-dev @types/mustache-express
```

```
added 2 packages, and audited 70 packages in 4s
```

```
2 packages are looking for funding
```

```
run `npm fund` for details
```

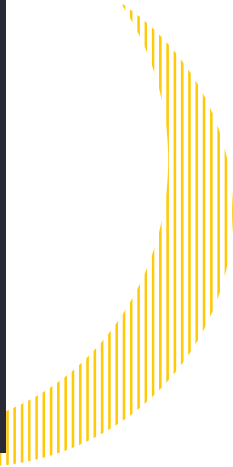
```
found 0 vulnerabilities
```

```
PS C:\Users\Viviane\Downloads\node\projetos> 
```

VEJA EM PACKAGE.JSON NOSSAS DEPENDENCIAS DE DESENVOLVIMENTO E DEPENDENCIAS NORMAIS



```
server.ts  package.json ×
package.json > ...
7   "test": "echo \"Error: no test specified\" && exit 1",
8   "start-dev": "nodemon src/server.ts"
9 },
10  "author": "",
11  "license": "ISC",
12  "devDependencies": {
13    "@types/mustache-express": "^1.2.1",
14    "@types/node": "^17.0.12",
15    "@types/validator": "^13.7.1",
16    "typescript": "^4.5.5"
17  },
18  "dependencies": {
19    "@types/express": "^4.17.13",
20    "express": "^4.17.2",
21    "mustache-express": "^1.3.2",
22    "validator": "^13.7.0"
23  }
```



AGORA VAMOS IMPORTAR O MUSTACHE EM NOSSO


SERVER.TS

server.ts × package.json

src > server.ts > ...

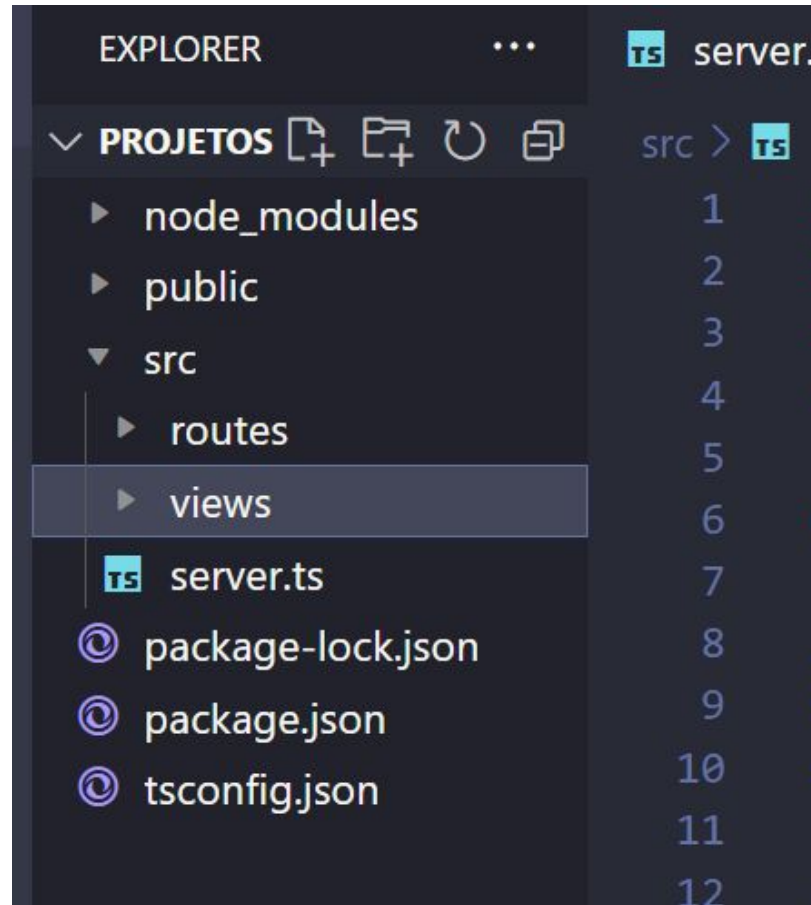
```
1  import express,{Request, Response} from 'express'
2
3  //importando o mustache
4  import mustache from 'mustache-express'
5
6  import path from 'path'
7  import mainRoutes from './routes/index'
8
9  const server = express()
10
11  server.use(express.static(path.join(__dirname, '../public')))
12
```

CONFIGURANDO O MUSTACHE NO SERVER.TS



```
server.ts  ×  @ package.json
src > TS server.ts > ...
1  import express,{Request, Response} from 'express'
2  import mustache from 'mustache-express'
3  import path from 'path'
4  import mainRoutes from './routes/index'
5
6  const server = express()
7
8  //configurando mustache
9  server.set('view engine','mustache')
10
11  server.use(express.static(path.join(__dirname,'../public')))
12
13  server.use(mainRoutes)
14
15  server.use((req: Request, res:Response) =>{
16    |
17    |   res.status(404).send('Página não encontrada!')
18  | })
19
```


**VAMOS CRIAR UMA PASTA CHAMADA VIEWS PARA COLOCARMOS
NOSSO CÓDIGO EM HTML, CRIE A PASTA VIEWS DENTRO DE SRC**



O PRÓXIMO PASSO É INFORMAR O CAMINHO DA PASTA VIEWS, VAMOS UTILIZAR PATH PARA PEGARMOS O CAMINHO COMPLETO




```
server.ts  ×  package.json

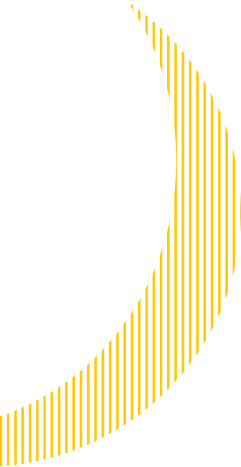
src > server.ts > ...
1  import express,{Request, Response} from 'express'
2  import mustache from 'mustache-express'
3  import path from 'path'
4  import mainRoutes from './routes/index'
5
6  const server = express()
7
8  //configurando mustache
9  server.set('view engine','mustache')
10 //informando ao mustache a minha pasta de arquivos view
11 server.set('views',path.join(__dirname,'views'))
12
13
14 server.use(express.static(path.join(__dirname,'../public')))
15 server.use(mainRoutes)
16 server.use((req: Request, res:Response) =>{
17
18     res.status(404).send('Página não encontrada!')
19 })
20
21 server.listen(80)
```



VAMOS USAR O MUSTACHE COM O COMANDO SERVER.ENGINE



```
server.ts × package.json
src > server.ts > ...
1  import express,{Request, Response} from 'express'
2  import mustache from 'mustache-express'
3  import path from 'path'
4  import mainRoutes from './routes/index'
5
6  const server = express()
7
8  //configurando mustache
9  server.set('view engine','mustache')
10 server.set('views',path.join(__dirname,'views'))
11
12 //usando o mustache
13 server.engine('mustache',mustache())
14
15
16 server.use(express.static(path.join(__dirname,'../public')))
17 server.use(mainRoutes)
18 server.use((req: Request, res:Response) =>{
19   |
20   |   res.status(404).send('Página não encontrada!')
21   | })
22
23 server.listen(80)
```



PROBLEMS

OUTPUT

TERMINAL

DEBUG CONSOLE

Experimente a nova plataforma cruzada PowerShell <https://aka.ms/pscore6>

PS C:\Users\Viviane\Downloads\node\projetos> **npm** run start-dev

> projetos@1.0.0 start-dev

> nodemon src/server.ts

[nodemon] 2.0.15

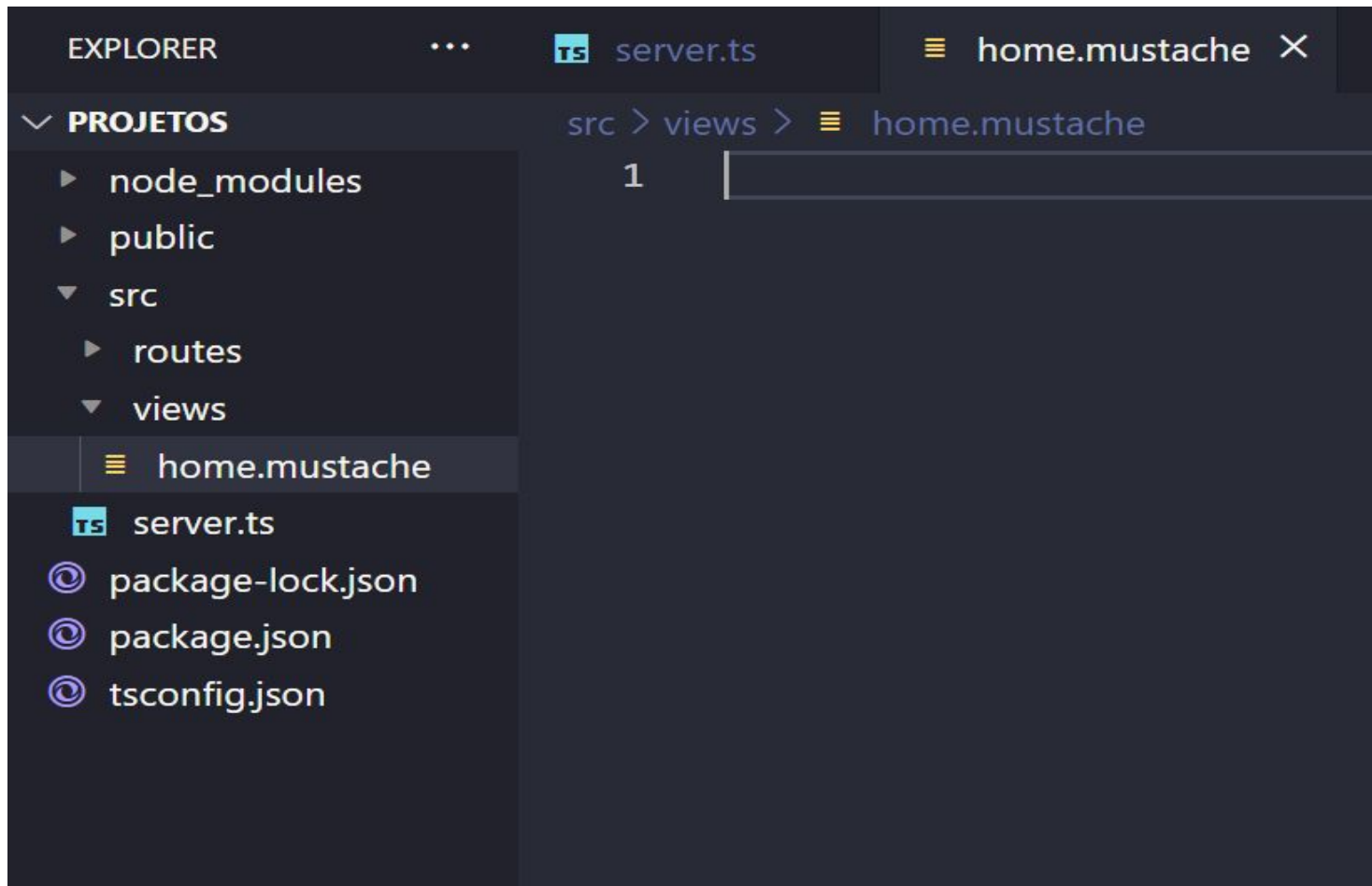
[nodemon] to restart at any time, enter `rs`

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

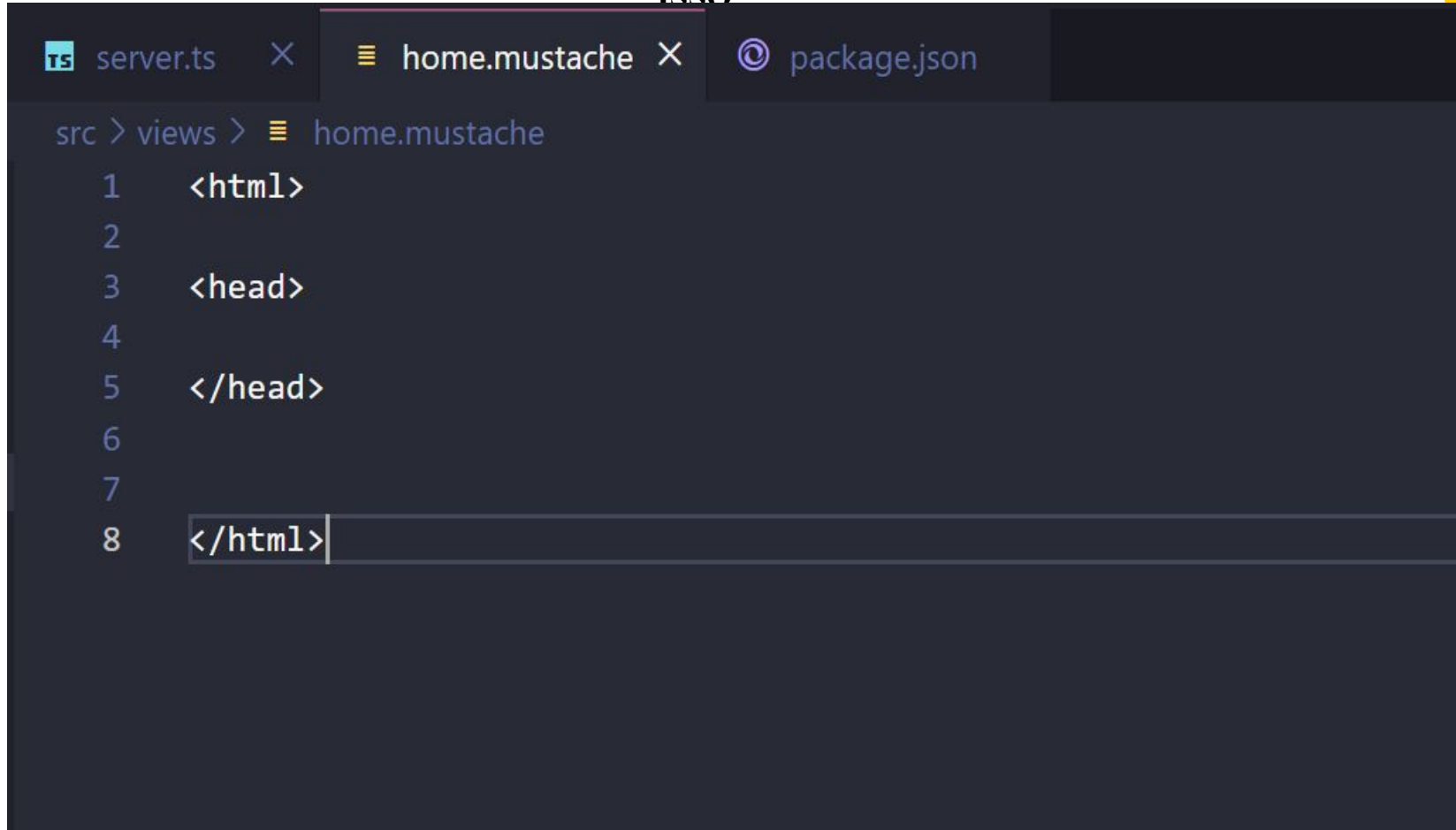
[nodemon] watching extensions: ts,json

[nodemon] starting `ts-node src/server.ts`

VAMOS CRIAR NOSSO PRIMEIRO ARQUIVO COM MUSTACHE, DENTRO DA PASTA VIEWS CRIE O ARQUIVO HOME.MUSTACHE




OBSERVE QUE QUANDO COMEÇAMOS A ESCREVER, O MUSTACHE NÃO COLORE AS TAGS EM HTML, VAMOS INSTALAR UMA EXTENSÃO PARA ISSO



```
server.ts ×  home.mustache ×  package.json
src > views > home.mustache
1  <html>
2
3  <head>
4
5  </head>
6
7
8  </html>
```

INSTALE A EXTENSÃO MUSTACHE SYNTAX

Extension: Mustache Template - Snippets & Autocomplete ✕



Mustache Template - Si

Manuel Gil | 6,693 | ★★★★★ (2)

Mustache – snippets & autocomplete for ...

[Disable](#) ▼ [Uninstall](#) ▼ ⚙️

This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Runtime Status](#)

No README available.

Categories

Programming Languages

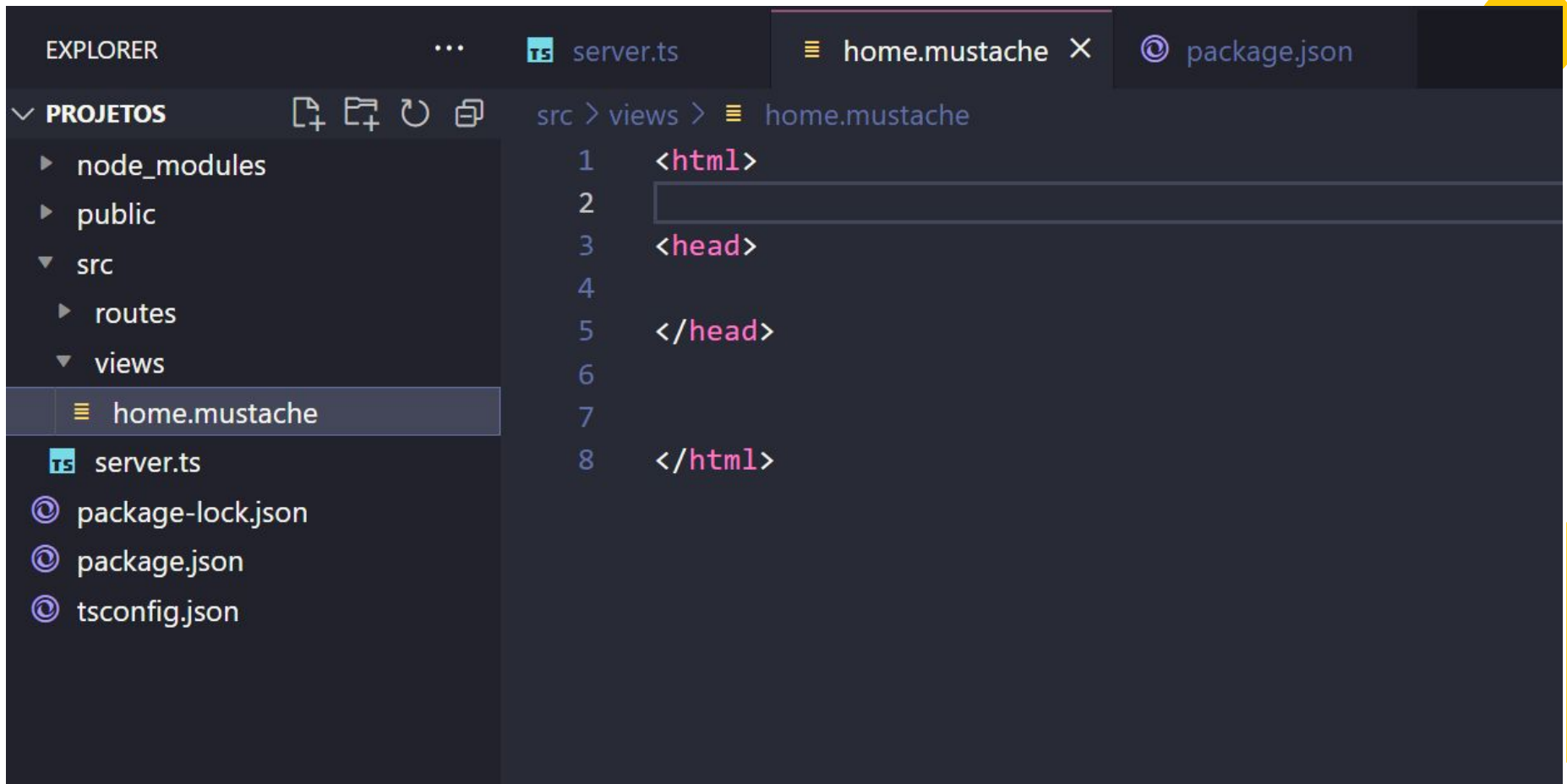
Snippets

Formatters

Keymaps

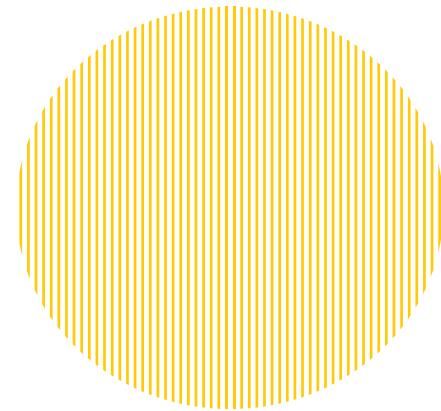
Other

VEJA QUE AGORA ELE RECONHECE AS TAGS EM HTML







EXIBINDO INFORMAÇÕES



CRIE UMA PÁGINA SIMPLES EM HTML



```
server.ts × home.mustache ×  
src > views > home.mustache  
1 <html>  
2 <head>  
3 |   <title>Página Home</title>  
4 </head>  
5  
6 <body>  
7 |   <h1>Título da Página</h1>  
8 |   <hr/>  
9 </body>  
10 <html>  COMO FAZER ESSA PÁGINA APARECER NA HOME?  
11
```

VÁ EM ROUTES>INDEX.TS



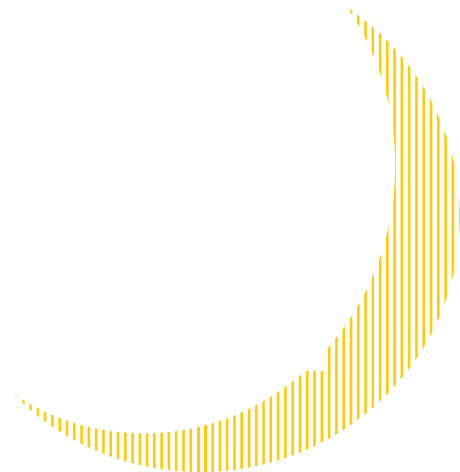
```
server.ts  home.mustache  index.ts  X
src > routes > index.ts > ...
1  import { Router, Request, Response } from "express";
2
3  const router = Router()
4
5  router.get('/', (req: Request, res: Response) =>{
6    //aqui vamos RENDERIZAR A NOSSA PÁGINA HOME.MUSTACHE
7    res.render('home')
8  })
9
10 router.get('/contato', (req: Request, res: Response) =>{
11   res.send("Formulário de contato")
12 })
13
14 router.get('/sobre', (req: Request, res: Response) =>{
15   res.send("sobre nós")
16 })
17
18
19 export default router
```

VEJA QUE A PÁGINA FOI ATUALIZADA

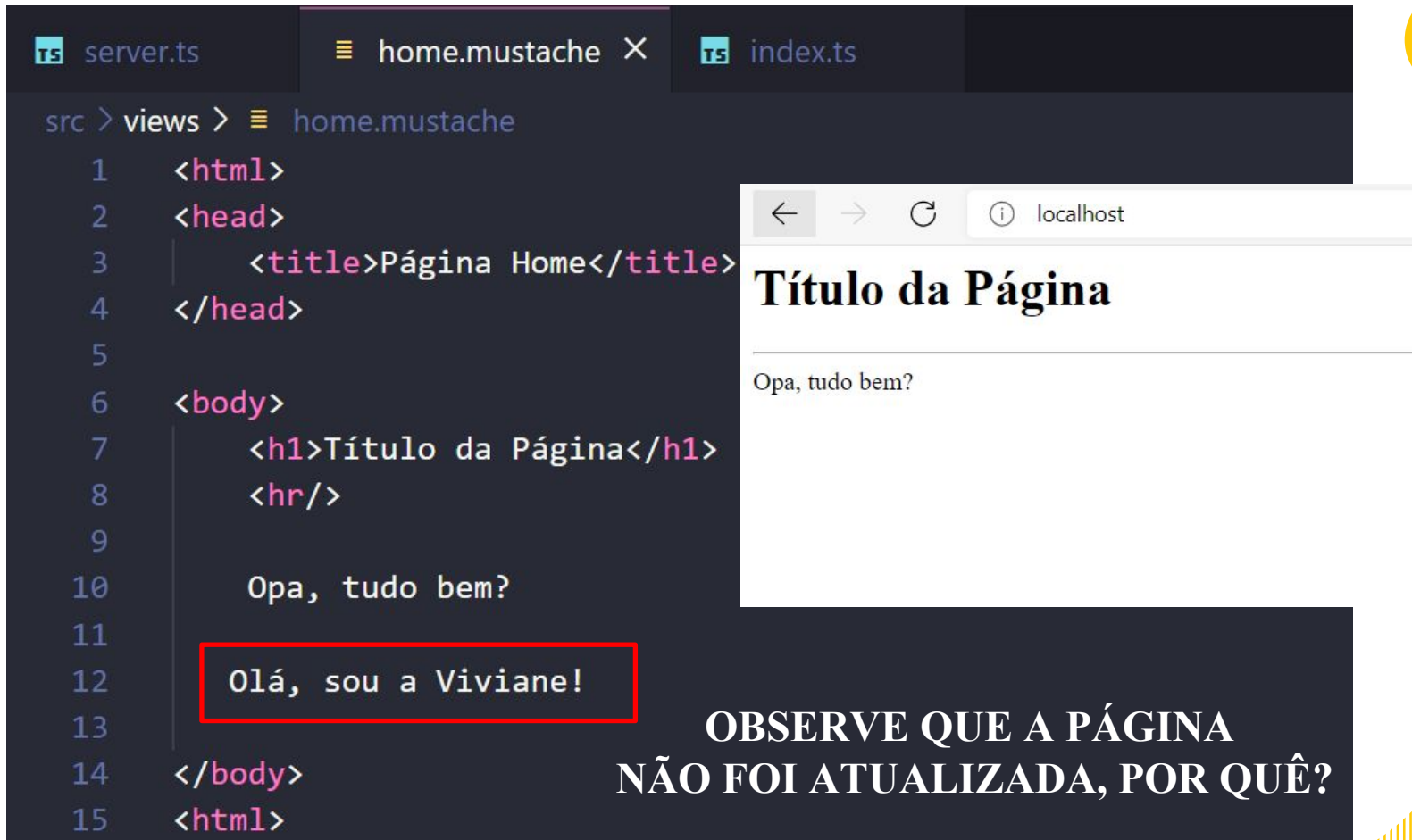


localhost

Título da Página



COLOQUE MAIS UM TEXTO NO HOME.MUSTACHE E VEJA SE A PÁGINA FOI



The image shows a code editor with a file named `home.mustache` open. The code is a Mustache template for an HTML page. It includes a title "Página Home" and a body with an `<h1>` tag containing "Título da Página" and an `<hr/>` tag. Below the header, the text "Opa, tudo bem?" is present. On line 12, the text "Olá, sou a Viviane!" is added and highlighted with a red box. To the right, a web browser window is shown at `localhost`. The browser displays the rendered page, which still shows "Título da Página" and "Opa, tudo bem?", indicating that the new text was not rendered. A yellow circle is in the top right corner, and a yellow decorative graphic is in the bottom right corner.

```
src > views > home.mustache
1  <html>
2  <head>
3    <title>Página Home</title>
4  </head>
5
6  <body>
7    <h1>Título da Página</h1>
8    <hr/>
9
10   Opa, tudo bem?
11
12   Olá, sou a Viviane!
13
14 </body>
15 </html>
```

localhost

Título da Página

Opa, tudo bem?

**OBSERVE QUE A PÁGINA
NÃO FOI ATUALIZADA, POR QUÊ?**

PRECISAMOS ADICIONAR A EXTENSÃO MUSTACHE NO NODEMON

PROBLEMS

OUTPUT

TERMINAL

DEBUG CONSOLE

Experimente a nova plataforma cruzada PowerShell <https://aka.ms/pscore6>

```
PS C:\Users\Viviane\Downloads\node\projetos> npm run start-dev
```

```
> projetos@1.0.0 start-dev
```

```
> nodemon src/server.ts
```

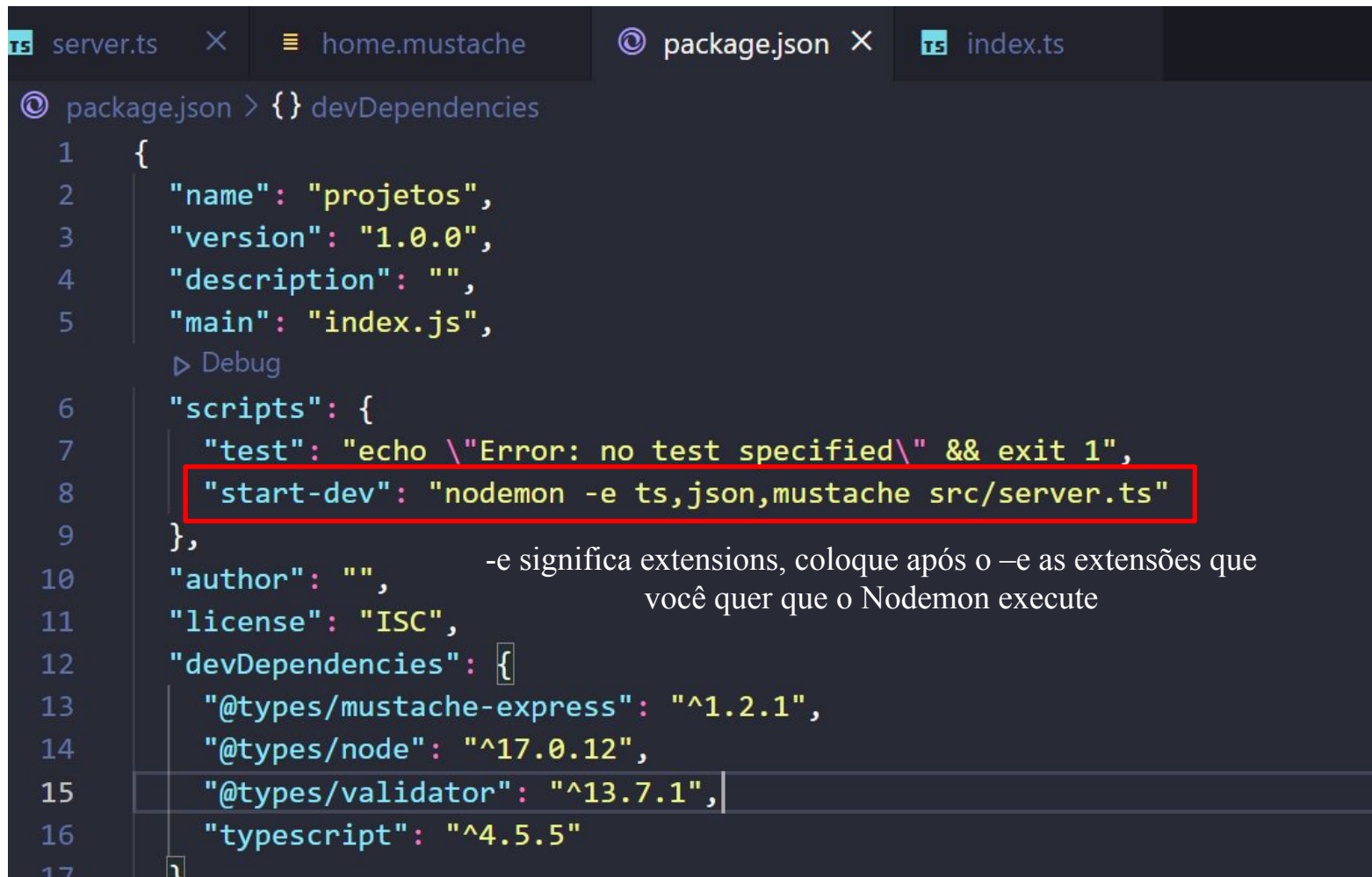
```
[nodemon] watching path(s): *.*
```

```
[nodemon] watching extensions: ts,json
```

```
[nodemon] starting `ts-node src/server.ts`
```

**VEJA QUE NOSSO NODEMON SÓ ESTÁ
MONITORANDO EXTENSÕES TS,JSON**

VÁ EM PACKAGE.JSON E VAMOS COLOCAR O MUSTACHE NO NODEMON

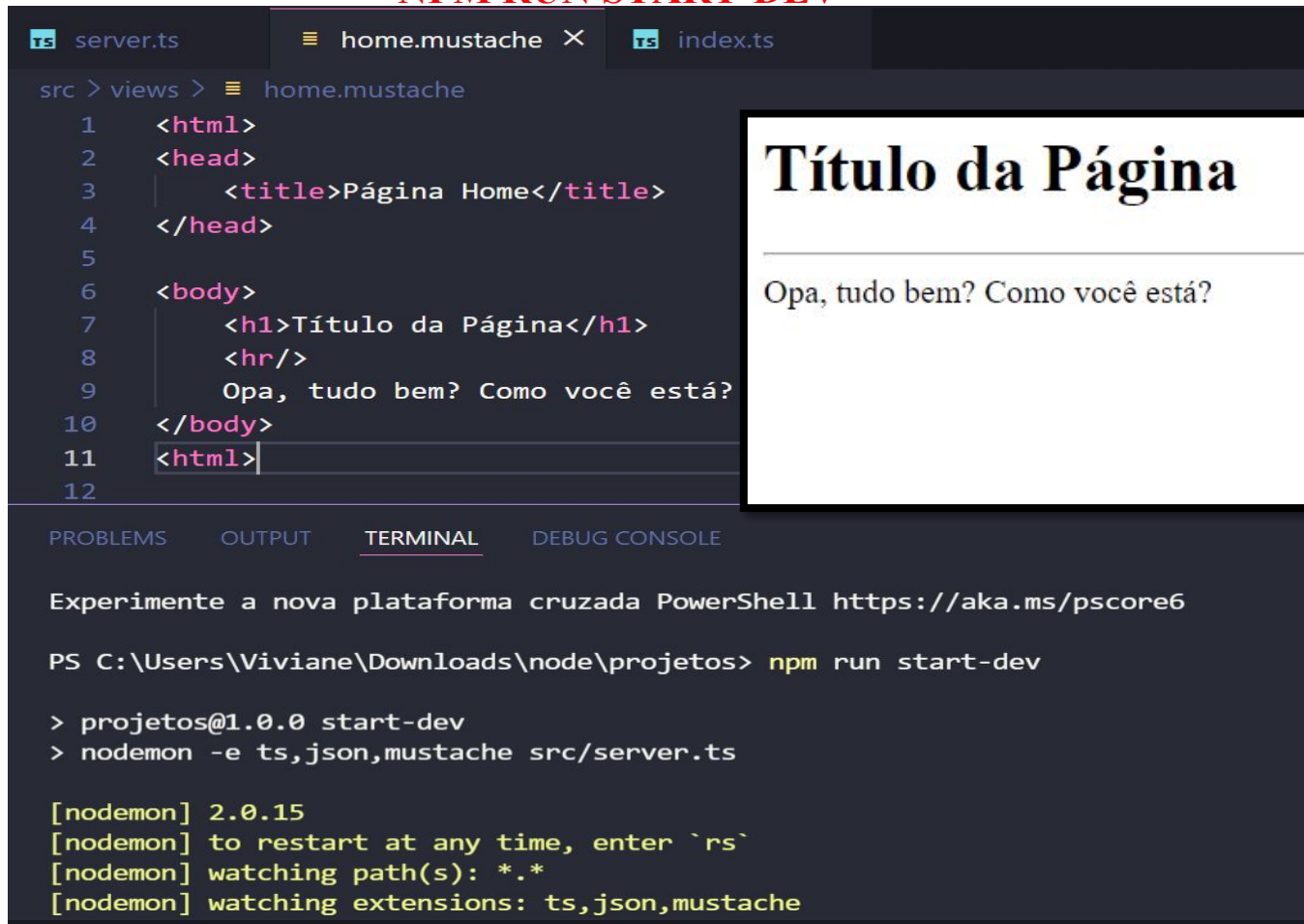


```
server.ts × home.mustache package.json × index.ts
package.json > {} devDependencies
1 {
2   "name": "projetos",
3   "version": "1.0.0",
4   "description": "",
5   "main": "index.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1",
8     "start-dev": "nodemon -e ts,json,mustache src/server.ts"
9   },
10  "author": "",
11  "license": "ISC",
12  "devDependencies": {
13    "@types/mustache-express": "^1.2.1",
14    "@types/node": "^17.0.12",
15    "@types/validator": "^13.7.1",
16    "typescript": "^4.5.5"
17  }
```

-e significa extensions, coloque após o -e as extensões que você quer que o Nodemon execute

FECHE O TERMINAL E EXECUTE O NODEMON NOVAMENTE

NPM RUN START-DEV



The image shows a VS Code editor window with a file named `server.ts` open. The file content is as follows:

```
src > views > home.mustache
1  <html>
2  <head>
3    <title>Página Home</title>
4  </head>
5
6  <body>
7    <h1>Título da Página</h1>
8    <hr/>
9    Opa, tudo bem? Como você está?
10 </body>
11 <html>
12
```

Below the editor, the **TERMINAL** tab is active, displaying the following output:

```
Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Users\Viviane\Downloads\node\projetos> npm run start-dev

> projetos@1.0.0 start-dev
> nodemon -e ts,json,mustache src/server.ts

[nodemon] 2.0.15
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: ts,json,mustache
```

Overlaid on the right side of the terminal is a preview of the rendered HTML page:

Título da Página

Opa, tudo bem? Como você está?

EXERCÍCIO:

CRIE DUAS PÁGINAS PARA AS ROTAS CONTATO E SOBRE

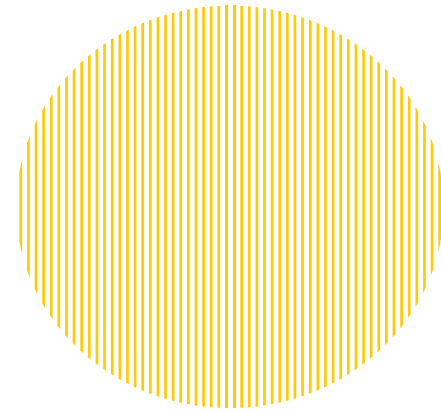


```
TS index.ts  X  home.mustache  {} package.json

src > routes > TS index.ts > ...

1  import { Router, Request, Response } from 'express'
2
3  const router = Router()
4
5  router.get('/', (req: Request, res: Response) => {
6    res.render('home')
7  })
8
9  router.get('/contato', (req: Request, res: Response) => {
10    res.send("Está é a página de contato")
11  })
12
13  router.get('/sobre', (req: Request, res: Response) => {
14    res.send("Sobre Nós")
15  })
16
17
18  export default router
```

**MUSTACHE:
EXIBINDO
INFORMAÇÕES**

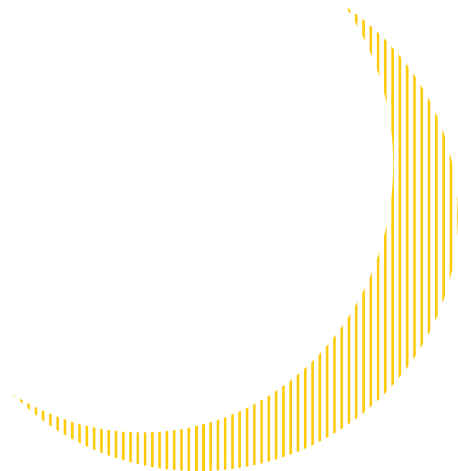


OBSERVE QUE NOSSA PÁGINA ESTÁ ESTÁTICA, VAMOS TORNA-LÁ DINÂMICA




Título da Página

Opa, tudo bem? Como você está?



VÁ EM INDEX.TS E COLOQUE UMA VARIÁVEL NA ROTA HOME



```
server.ts  home.mustache  index.ts  X
src > routes > index.ts > ...
1  import { Router, Request, Response } from "express";
2
3  const router = Router()
4
5  router.get('/', (req: Request, res: Response) =>{
6      let user: string = 'Pioran'
7
8      res.render('home')
9  })
10
11
12  router.get('/contato', (req: Request, res: Response) =>{
13      res.send("Formulário de contato")
14  })
15
16  router.get('/sobre', (req: Request, res: Response) =>{
17      res.send("sobre nós")
18  })
19
20  export default router
```

Como podemos mandar essa variável para o nosso arquivo HTML?

PRECISAMOS COLOCAR A VARIÁVEL DENTRO DO NOSSO RES.RENDER('HOME')

```
server.ts  home.mustache  index.ts  X
src > routes > index.ts > ...
1  import { Router, Request, Response } from "express";
2
3  const router = Router()
4
5  router.get('/', (req: Request, res: Response) =>{
6
7
8      res.render('home', {
9          user: 'Pioran'
10      })
11  })
12
13  router.get('/contato', (req: Request, res: Response) =>{
14      res.send("Formulário de contato")
15  })
16
17  router.get('/sobre', (req: Request, res: Response) =>{
18      res.send("sobre nós")
19  })
```

USANDO O USER NA NOSSA VIEW

home.mustache X

TS index.ts

src > views > home.mustache

```
1  <html>
2  <head>
3    <title>Página Home
4  </head>
5
6  <body>
7    <h1>Título da Página</h1>
8    <hr/>
9    Opa, tudo bem? {{user}} Como você está?
10 </body>
11 </html>
12
```

Título da Página

Opa, tudo bem? Pioran Como você está?

TAMBÉM PODEMOS USAR OBJETOS

```
home.mustache index.ts ×
src > routes > index.ts > ...
1  import { Router, Request, Response } from "express";
2
3  const router = Router()
4
5  router.get('/', (req: Request, res: Response) => {
6      //criando o objeto user
7      let user = {
8          name: 'Viviane',
9          age: 26
10     }
11     //mandando o objeto user para minha view
12     res.render('home', {
13         user
14     })
15 })
16
17 router.get('/contato', (req: Request, res: Response) => {
18     res.send("Formulário de contato")
19 })
```

USANDO UM ATRIBUTO ESPECÍFICO DO OBJETO NA VIEW

home.mustache X

index.ts

Título da Página

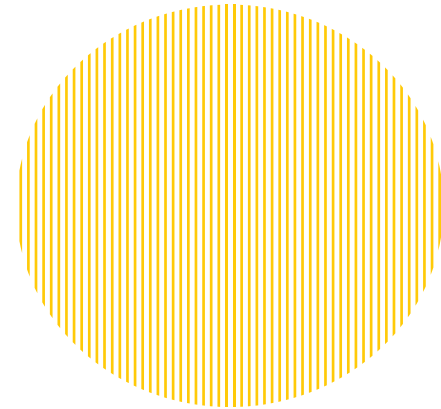
Opa, tudo bem? Viviane Como você está? você já tem 26 anos?

Título da Página


Opa, tudo bem? Viviane Como você está?

```
7      <n1>Título da Página</n1>
8      <hr/>
9      Opa, tudo bem? {{user.name}} Como você está?
10 </body>
11 <html>
```


**MUSTACHE:
EXIBIÇÃO
CONDICIONAL**



AGORA QUE JÁ APRENDEMOS A EXIBIR INFORMAÇÕES NA PÁGINA, VAMOS EXIBIR AS MENSAGENS SOMENTE SE NÓS QUISERMOS, ISSO É ÚTIL POIS QUANDO VOCÊ ESTIVER CRIANDO O SITE PODE SER QUE VOCÊ QUEIRA MOSTRAR MENSAGENS EM UM MOMENTO ESPECÍFICO.



EM QUAIS SITUAÇÕES VOCÊS ACHAM QUE PODEMOS EXIBIR MENSAGENS CONDICIONAIS?

- **USUÁRIO FEZ UMA COMPRA E VOCÊ GOSTARIA DE EXIBIR A MENSAGEM “COMPRA REALIZADA COM SUCESSO”**
- **USUÁRIO FEZ O LOGIN E ERROU A SENHA, E VOCÊ PRECISA MOSTRAR A MENSAGEM “EMAIL OU SENHA INVÁLIDA”**

OBSERVE QUE NESSES EXEMPLOS A MENSAGEM NÃO IRÁ APARECER TODA HORA, ERA IRÁ APARECER SOMENTE EM UM DETERMINADO MOMENTO



EXIBINDO MENSAGEM DE BOAS-VINDAS NO MUSTACHE VÁ EM ROUTES>INDEX.TS



```
home.mustache  TS index.ts  X
src > routes > TS index.ts > ...
1  import { Router, Request, Response } from "express";
2
3  const router = Router()
4
5  router.get('/',(req:Request, res: Response) =>{
6
7      res.render('home',{
8          name:'Viviane',
9          lastName: 'de Lima',
10         /*aqui nesse exemplo eu não
11         gostaria de exibir a mensagem
12         de boas vindas */
13         showWelcome:false
14     })
15 })
16
17 router.get('/contato',(req:Request, res:Response) =>{
18     res.send("Formulário de contato")
```

Lembrando que tudo que usamos no Mustache deve ser enviado no render

Quando eu coloco a variável como tipo false, eu tenho acesso a essa variável dentro do Mustache

VOLTE PARA HOME.MUSTACHE E VAMOS COLCOAR NOSSA CONDICIONAL

```
<html>
<head>
  <title>Página Home</title>
</head>
```

```
<body>
```

```
  <h1>Título da Página</h1>
```

```
  <hr/>
```

```
  {{#showWelcome}}
```

```
  Opa, {{name}} {{lastName}} seja bem vinda!
```

```
  {{/showWelcome}}
```

```
</body>
```

```
</html>
```

Abrimos showWelcome

Fechamos showWelcome

ATUALIZE A PÁGINA E VEJA QUE A NOSSA MENSAGEM DESAPARECEU

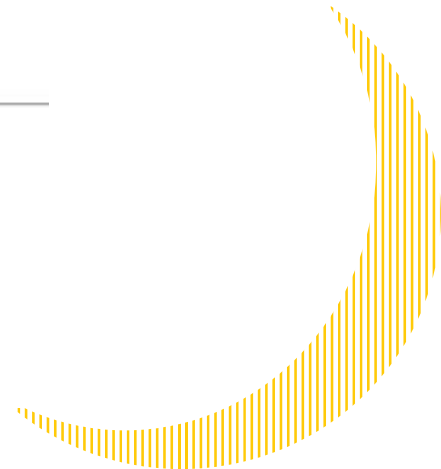


Título da Página

**COLOQUE SHOWWELCOME COMO TRUE EM INDEX.TS E VEJA QUE IRÁ APARECER A
MENSAGEM**

Título da Página

Opa, Viviane de Lima seja bem vinda!



EXERCÍCIO

USANDO A EXIBIÇÃO CONDICIONAL DO MUSTACHE, SE A PESSOA FOR MAIOR DE 18 ANOS EXIBA A MENSAGEM 'VOCÊ É MAIOR DE IDADE' OU SE A PESSOA FOR MENOR DE 18 ANOS NÃO EXIBE NENHUMA MENSAGEM

Título da Página

Opa, Viviane de Lima seja bem vinda! Você é maior de idade!

RESOLUÇÃO

INDEX.TS

```
import { Router, Request, Response } from "express";

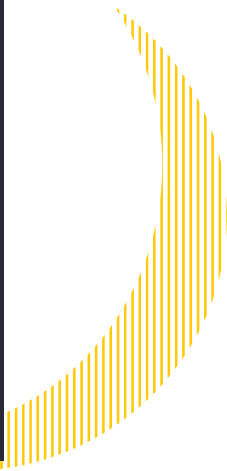
const router = Router()

router.get('/', (req: Request, res: Response) => {

  let age: number = 18
  let showOld: boolean = false

  if (age > 18) {
    showOld = true
  }

  res.render('home', {
    name: 'Viviane',
    lastName: 'de Lima',
    showOld
  })
})
```



RESOLUÇÃO

HOME.MUSTACHE



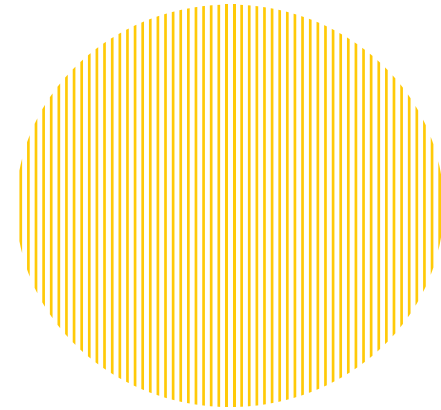
```
<html>
<head>
  <title>Página Home</title>
</head>

<body>
  <h1>Título da Página</h1>
  <hr/>

  Opa, {{name}} {{LastName}} seja bem vinda!
  {{#showOld}}
    Você é maior de idade!
  {{/showOld}}
</body>
</html>
```



**MUSTACHE:
EXIBINDO LISTAS
(LOOP)**



VAMOS CRIAR UMA LISTA DE PRODUTOS COM MUSTACHE

```
<html>
<head>
  <title>Página Home</title>
</head>

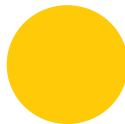
<body>
  <h1>Título da Página</h1>
  <hr/>

  Opa, {{name}} {{lastName}} seja bem vinda!
  {{#showOld}}
    Você é maior de idade!
  {{/showOld}}

  <hr/>

  <h2>Produtos</h2>

</body>
</html>
```

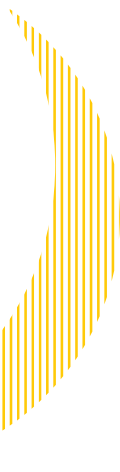


A NOSSA LISTA SERÁ DENTRO DO RENDER, VAMOS COLCOCAR NOME E PREÇO



```
home.mustache × TS index.ts ×
src > routes > TS index.ts > ...

4
5 router.get('/', (req: Request, res: Response) =>{
6
7     let age: number = 20
8     let showOld: boolean = false
9
10    if(age > 18){
11        showOld = true
12    }
13
14    res.render('home', {
15        name: 'Viviane',
16        lastName: 'de Lima',
17        showOld,
18        products: [
19            {title: 'Produto x', price: 10 },
20            {title: 'Produto y', price: 30 },
21            {title: 'Produto w', price: 50 }
22        ]
23    })
24 })
```

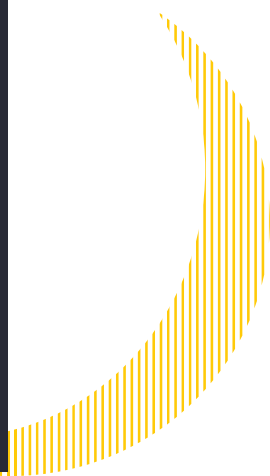


EXIBINDO A LISTA NO HOME.MUSTACHE



```
home.mustache X index.ts
src > views > home.mustache

5
6 <body>
7   <h1>Título da Página</h1>
8   <hr/>
9
10  Opa, {{name}} {{lastName}} seja bem vinda!
11  {{#showOld}}
12    Você é maior de idade!
13  {{/showOld}}
14
15  <hr/>
16
17  <h2>Produtos</h2>
18  <ul>
19    {{#products}}
20
21      <li> {{title}} - R$ {{price}} </li>
22
23    {{/products}}
24
25  </ul>
26
27 </body>
28 <html>
```



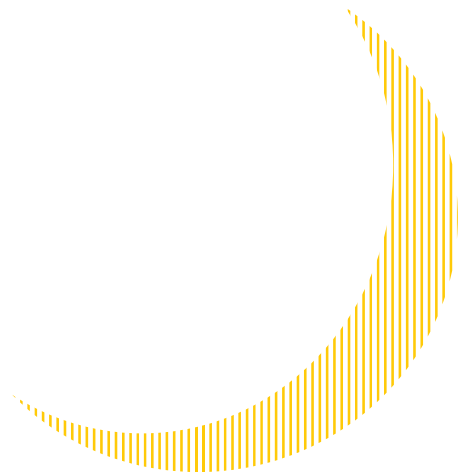


Título da Página

Opa, Viviane de Lima seja bem vinda! Você é maior de idade!

Produtos

- Produto x - R\$ 10
- Produto y - R\$ 30
- Produto w - R\$ 50



EXIBINDO UMA LISTA SIMPLES

```
res.render('home',{
  name:'Viviane',
  lastName: 'de Lima',
  showOld,
  products:[
    {title:'Produto x', price:10 },
    {title:'Produto y', price:30 },
    {title:'Produto w', price:50 }
  ],
  frases:[
    'frase do dia 1',
    'frase do dia 2',
    'frase do dia 3'
  ]
})
})
```

EXIBINDO UMA LISTA SIMPLES



```
home.mustache X index.ts
src > views > home.mustache
18     <ul>
19         {{#products}}
20
21         <li> {{title}} - R$ {{price}} </li>
22
23         {{/products}}
24
25     </ul>
26
27     <hr/>
28
29     {{#frases}}
30
31     <li>{{.}}</li>
32
33     {{/frases}}
34
35 </body>
36 <html>
```



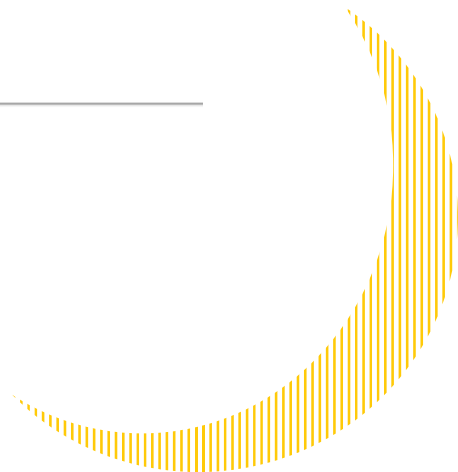
Título da Página

Opa, Viviane de Lima seja bem vinda! Você é maior de idade!

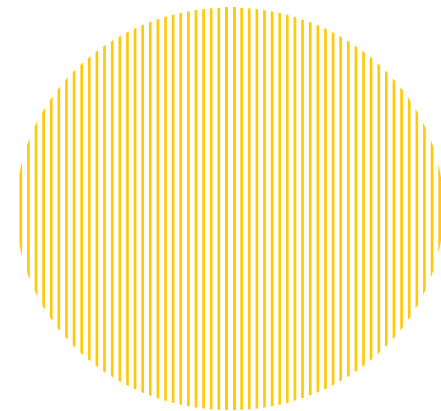
Produtos

- Produto x - R\$ 10
- Produto y - R\$ 30
- Produto w - R\$ 50

-
- frase do dia 1
 - frase do dia 2
 - frase do dia 3



MUSTACHE: ELSE DE LISTAS



E SE VOCÊ NÃO TIVER NENHUM PRODUTO PARA SER EXIBIDO?

VAMOS SUPOR QUE O USUÁRIO BUSCOU POR UM PRODUTO QUE VOCÊ NÃO TEM NA LOJA E VOCÊ PRECISA EXIBIR A MENSAGEM 'NENHUM PRODUTO CADASTRADO'. VAMOS USAR O EXEMPLO DAS NOSSAS FRASES:

```
<hr/>
```

```
{{#frases}}
```

```
<li>{{.}}</li>
```

```
{{/frases}}
```

```
{{^frases}}
```

```
    Não temos frases motivacionais hoje
```

```
{{/frases}}
```

```
</body>
```

```
<html>
```

Basta usar o [^] para fazer o else de listas, ou seja, se não tiver nenhuma frase, ele irá aparecer essa frase



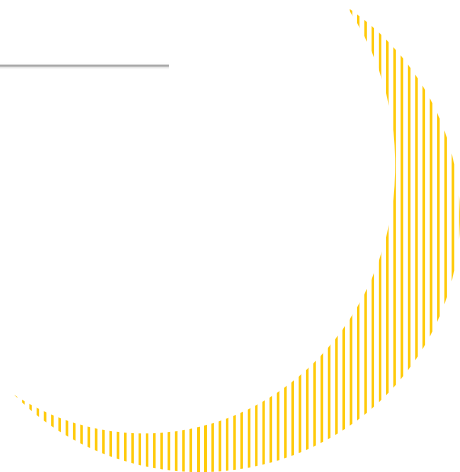
Título da Página

Opa, Viviane de Lima seja bem vinda! Você é maior de idade!

Produtos

- Produto x - R\$ 10
- Produto y - R\$ 30
- Produto w - R\$ 50

-
- frase do dia 1
 - frase do dia 2
 - frase do dia 3



REMOVA A LISTA DE FRASES NO INDEX.TS E ATUALIZE O NAVEGADOR NOVAMENTE

```
res.render('home', {  
  name: 'Viviane',  
  lastName: 'de Lima',  
  showOld,  
  products: [  
    {title: 'Produto x', price: 10},  
    {title: 'Produto y', price: 30},  
    {title: 'Produto z', price: 50},  
  ],  
  frases: [  
    'Frase motivacional 1',  
    'Frase motivacional 2',  
    'Frase motivacional 3',  
    'Frase motivacional 4',  
    'Frase motivacional 5',  
  ],  
})  
})
```

Título da Página

Opa, Viviane de Lima seja bem vinda! Você é maior de idade!

Produtos

- Produto x - R\$ 10
- Produto y - R\$ 30
- Produto w - R\$ 50

Não temos frases motivacionais hoje

PODEMOS USAR O ELSE DE LISTAS NO SHOWOLD TAMBÉM

```
<body>
  <h1>Título da Página</h1>
  <hr/>

  Opa, {{name}} {{lastName}} seja bem vinda!
  {{#showOld}}
    Você é maior de idade!
  {{/showOld}}

  {{^showOld}}
    Você é menor de idade!
  {{/showOld}}

  <hr/>

  <h2>Produtos</h2>
  <ul>
    {{#products}}
      <li> {{title}} - R$ {{price}} </li>
```

Título da Página

Opa, Viviane de Lima seja bem vinda! Você é menor de idade!

Produtos

- Produto x - R\$ 10
- Produto y - R\$ 30
- Produto w - R\$ 50

Não temos frases motivacionais hoje