

PROGRAMAÇÃO WEBII


1

Elaboração da Interface

Listar Produtos

Vamos iniciar...

- Nosso objetivo da aula de hoje:
- Vamos aprender como criar a nossa página de Listar Produtos.

#	Nome	Validade	Valor	Ações
1	Curso de angular	31/12/2021	R\$ 500,00	 

Isso é uma tabela

Vamos iniciar...

```
<table>
  <thead>
    <tr>
      <th>Nome</th>
      <th>Telefone</th>
      <th>CEP</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Maria </td>
      <td>11 9999 1827 </td>
      <td>02345001</td>
    </tr>
  </tbody>
</table>
```

- `<table></table>` -
- `<thead></thead>` - head (cabeçalho)
- `<tr></tr>` - table row – linha
- `<td></td>` - table data
- `<th></th>` - table head – negrito e centraliza
- `<tbody></tbody>` - corpo

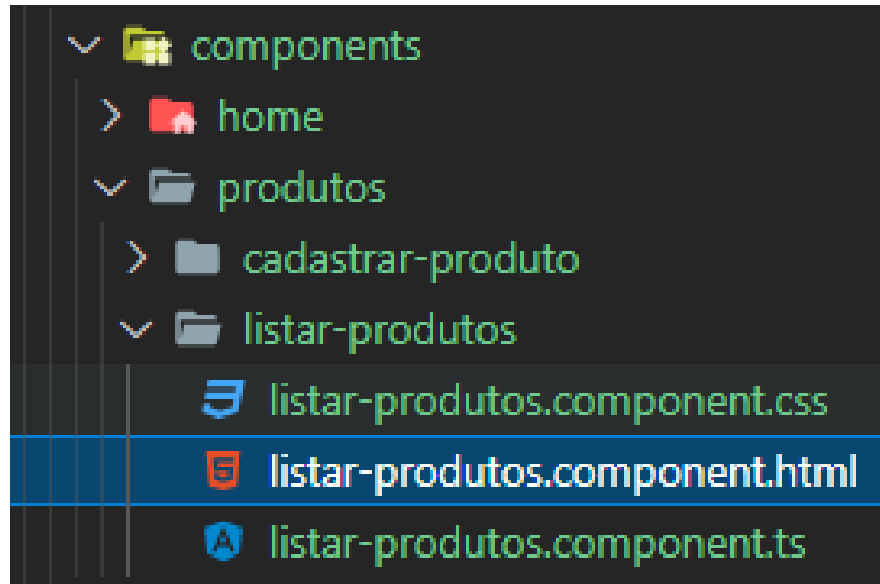
Vamos iniciar...

Por enquanto, vamos apenas trabalhar no html delas.



Abrindo o arquivo...

- Vamos entrar no site do bootstrap para deixar carregado
 - ▷ <https://getbootstrap.com/>
 - ▷ Vamos abrir na documentação 4.6
- Dentro do VsCode, vamos abrir o arquivo listar-produtos.componente.html



Configurando o listar-produtos.component.html

Vamos apagar o que está no arquivo

```
src > app > components > produtos > listar-produtos > listar-produtos.component.html > c
1  <p>listar-produtos works!</p>
2
```

Configurando o listar-produtos.component.html

Vamos digitar:

- ▶ h1, uma classe my-3 e também uma classe text-primary para ficar azul
 - ▶ h1.my-3.text-primary <TAB>
- ▶ Vamos dar um <ENTER> antes do </h1>
- ▶ Vamos escrever:
 - ▶ Listar Produtos
- ▶ Vamos criar um botão com a classe btn, btn.success e mt-2. Para isso digite:
 - ▶ button.btn.btn.success.mt-2 <TAB>

```
1 <h1 class="my-3 text-primary">
2   Listar Produtos
3   <button class="btn btn-success mt-2"></button>
4 </h1>
```


Configurando o listar-produtos.component.html

Entendendo um pouco sobre as classes como text-primary do bootstrap:

- <https://getbootstrap.com/docs/4.6/components/buttons/>
- <https://www.jquery-az.com/bootstrap-color-classes-text-background-links/>

Entendendo sobre classes mt-2 (spacing)

- <https://getbootstrap.com.br/docs/4.1/utilities/spacing/>
- <https://www.jquery-az.com/bootstrap-margin-padding-classes-spacing-explained-5-examples/>

m - for classes that set margin, like this :

- **mt** - for classes that set `margin-top`
- **mb** - for classes that set `margin-bottom`
- **ml** - for classes that set `margin-left`
- **mr** - for classes that set `margin-right`
- **mx** - for classes that set both `margin-left` and `margin-right`
- **my** - for classes that set both `margin-top` and `margin-bottom`

Where size is one of margin :

- **0** - for classes that eliminate the margin by setting it to 0, like `mt-0`
- **1** - (by default) for classes that set the margin to `$spacer * .25`, like `mt-1`
- **2** - (by default) for classes that set the margin to `$spacer * .5`, like `mt-2`
- **3** - (by default) for classes that set the margin to `$spacer`, like `mt-3`
- **4** - (by default) for classes that set the margin to `$spacer * 1.5`, like `mt-4`
- **5** - (by default) for classes that set the margin to `$spacer * 3`, like `mt-5`
- **auto** - for classes that set the margin to auto, like `mx-auto`

Configurando o listar-produtos.component.html

- Após o button, vamos digitar o tipo dele, type="button"
 - ▶ E antes do </button>, vamos digitar o nome que será Cadastrar
- Vamos salvar
- Vamos verificar como está ficando a nossa página

```
1 <h1 class="my-3 text-primary">
2   Listar Produtos
3   <button type="button" class="btn btn-success mt-2">Cadastrar</button>
4 </h1>
```

Configurando o listar-produtos.component.html

- Precisamos levar nosso botão para a direita.
- Após o mt-2, vamos digitar:
 - ▷ float-right
- Vamos salvar e verificar como ficou nossa página

```
1 <h1 class="my-3 text-primary">  
2   Listar Produtos  
3   <button type="button" class="btn btn-success mt-2 float-right">Cadastrar</button>  
4 </h1>  
5
```

Listar Produtos

Cadastrar

Configurando o listar-produtos.component.html

Próximo passo, vamos inserir um card

▶ Vamos digitar:

▶ div.card -> Estamos utilizando abreviação do Emmet, caso contrário teríamos que digitar todo o conteúdo.

▶ Teclar <TAB>

```
1 <h1 class="my-3 text-primary">
2   Listar Produtos
3   <button type="button" class="btn btn-success mt-2 float-right">Cadastrar</button>
4 </h1>
5
6 div.card
7   div.card
```

Emmet Abbreviation

<div class="card">|</div>

Configurando o listar-produtos.component.html

- Após o card, vamos digitar shadow para deixar uma sombra ao redor do card.
- Ficará assim:

```
6   <div class="card shadow">  
7  
8  
9   </div>
```

CRUD Angular Home Produtos ▾

Listar Produtos

Cadastrar

Configurando o listar-produtos.component.html

- Agora vamos criar uma tabela dentro do card:
- Temos que digitar:
 - ▶ `table.table.table-responsive.table.striped` <TAB>
 - ▶ `table-responsive` -> para ser responsiva
 - ▶ `table.striped` -> alterna a cor das linhas da tabela (linhas zebradas)

```
6 <div class="card shadow">
7   table.table.table-responsive.table.striped
8
9 </div>
```

```
6 <div class="card shadow">
7   <table class="table table-responsive table striped">
8
9
10  </table>
11
12 </div>
```

Configurando o listar-produtos.component.html

■ Na linha abaixo do table, vamos criar um thead

- ▶ Devemos digitar <thead>
- ▶ Dentro do <thead>, vamos criar uma linha <tr>
- ▶ Dentro do <tr>, vamos criar um <th>
- ▶ Dentro do <th>, vamos inserir um # (jogo da velha) que irá exibir o id do produto
- ▶ Vamos criar outro <th> onde iremos digitar Nome, que será o nome do produto
- ▶ Vamos criar outro <th> onde iremos digitar Validade, que será a validade do produto
- ▶ Vamos criar outro <th> onde iremos digitar Valor, que será o valor do produto
- ▶ Vamos criar outro <th> onde iremos digitar Ações, que iremos inserir os botões de alterar e excluir produto.

■ <ALT><SHIFT><F> -> para formatar

Configurando o listar-produtos.component.html

```
8 <div class="card shadow">
9   <table class="table table-responsive table striped">
10    <thead>
11      <tr>
12        <th>#</th>
13        <th>Nome</th>
14        <th>Validade</th>
15        <th>Valor</th>
16        <th>Ações</th>
17      </tr>
18    </thead>
19  </table>
20 </div>
```

▶ Já temos aqui a nossa tabela com o nome, validade, valor e ações, só que não está ocupando o espaço inteiro. Vamos ajustar

CRUD Angular Home Produtos ▾

Listar Produtos

Cadastrar

#	Nome	Validade	Valor	Ações
---	------	----------	-------	-------

Configurando o styles.css

- Poderíamos resolver isso, incluindo comandos no css do arquivo listar-produtos.component.css
- Mas, vamos fazer essa alteração para qualquer componente da nossa aplicação, por isso, vamos abrir o arquivo styles.css
- Vamos digitar:

```
.table {  
  display: table;  
}
```
- Vamos salvar.
- Agora está distribuído em todo o card da nossa aplicação.

```
1  /* You can add global styles to this file, and also import other style files */  
2  .table {  
3    |   display: table;  
4    |   }
```

Listar Produtos

Cadastrar

#	Nome	Validade	Valor	Ações

Configurando o listar-produtos.component.html

- Agora vamos fazer o corpo da nossa tabela.
- Após o `</thead>`, vamos digitar `<tbody>`
- Dentro do `<tbody>`, vamos inserir um `<tr>`
- Dentro do `<tr>`, vamos inserir um `<td>`
- Se temos 5 colunas, vamos copiar o `<td>` e inserir 5 colunas para a nossa tabela `<td>`
- Vamos digitar as informações conforme segue abaixo, salvar e verificar como ficou a nossa página.

```
17     </thead>
18   </thead>
19   <tbody>
20     <tr>
21       <td>1</td>
22       <td>Curso de Angular</td>
23       <td>31/12/2021</td>
24       <td>R$ 500,00</td>
25       <td></td>
26     </tr>
27   </tbody>
28 </table>
29 </div>
```

Configurando o listar-produtos.component.html

- Na última coluna, precisamos criar os botões, mas para isso precisaremos de uma fonte de ícones.
- Para isso, precisaremos instalar o font Awesome dentro da nossa aplicação.
- Acesse o site abaixo:
 - ▶ <https://www.npmjs.com/package/font-awesome>
- Copie o npm i font-awesome, só clicar no ícone



Configurando o listar-produtos.component.html

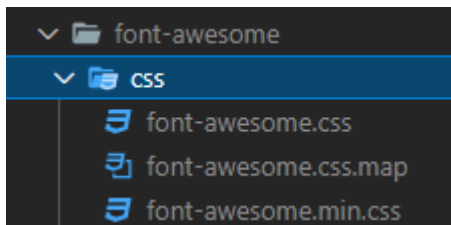
- Vamos parar o servidor do nosso site
- Vamos colar o comando para instalar a fonte.

▷ `npm i font-awesome <ENTER>`

```
Deseja finalizar o arquivo em lotes (S/N)? s
C:\angular\cursopwebII\frontend>npm i font-awesome
```

Configurando o listar-produtos.component.html

- Vamos abrir a pasta node-modules
- Vamos procurar o pacote da fonte awesome
- Abrir a pasta css
- Vamos selecionar o arquivo font-awesome.min.css
- Vamos clicar com o botão direito, selecionar copy relative path



Configurando o listar-produtos.component.html

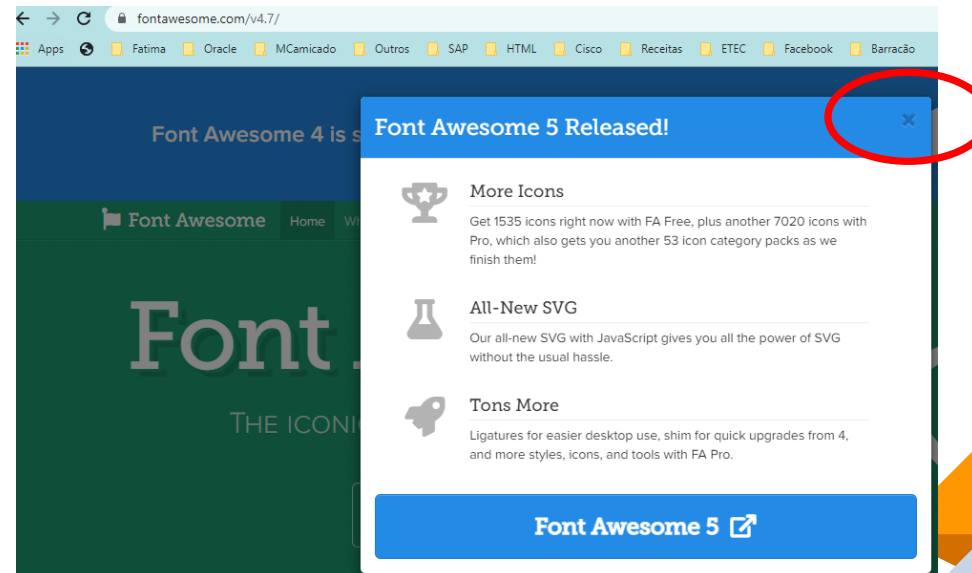
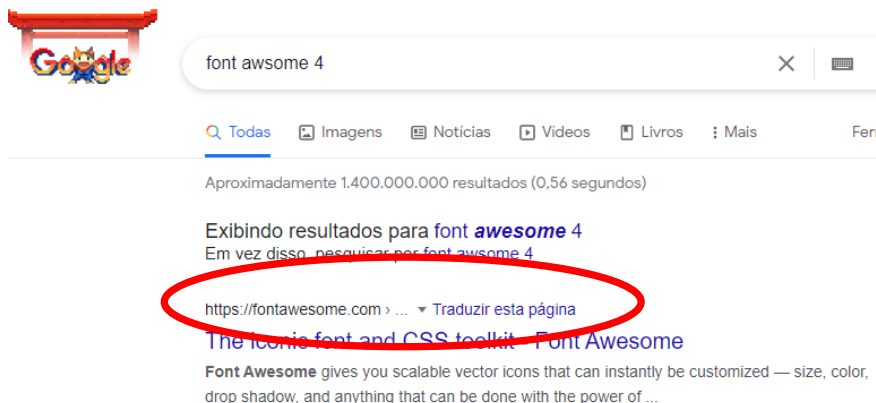
- Vamos abrir o arquivo angular.json para configurar nosso projeto
- Dentro de styles, temos o bootstrap e o style.css
- No meio deles, vamos colar a font awesome
- Vamos inserir aspas "", colar o caminho
- Inserir virgula no final
- Substituir a barra \ pela barra /
- Vamos salvar
- Vamos subir nossa aplicação novamente

➤ ng serve -o <ENTER>

```
"styles": [  
  "node_modules/bootstrap/dist/css/bootstrap.min.css",  
  "node_modules/font-awesome/css/font-awesome.min.css",  
  "src/styles.css"  
],
```

Configurando o listar-produtos.component.html

- Podemos recolher a pasta node-modules, senão ficará atrapalhando.
- Vamos voltar na nossa aplicação, mas antes de continuarmos, vamos no google e procurar por:
 - ▶ Fonte awesome 4
 - ▶ Vamos clicar no link, não precisa de registro para utilizar por isso vamos usar ela.
 - ▶ Vamos fechar essa janela que aparece



Configurando o listar-produtos.component.html

Clica na opção Icons e vamos procurar um ícone

Font Awesome 4 is so 2017. Upgrade to version 5 and get twice the icons.

Get the Latest 

 Font Awesome

[Home](#)

[What's New](#)

[Get Started](#)

[Icons](#) ▼

[Examples](#) ▼

[Accessibility](#)

[Community](#)


[License](#)

[Blog](#)

You asked, Font Awesome delivers with 41 shiny new icons in version 4.7. Want to request new icons? [Here's how](#). Need vectors or want to use on the desktop? Check the [cheatsheet](#).

shutterstock


 Aproveite a avaliação de 1 mês da Shutterstock e receba 10 imagens grátis! 
ads via Carbon

 Search icons




by Algolia


Configurando o listar-produtos.component.html

- Vamos procurar um ícone de Edit, digite Edit para procurar
- Clique no pencil

Q edit


Search for 'edit'

 pencil-square-o ([edit](#), [edit](#))


 outdent

 align-center


I italic

 pencil ([edit](#))


 list

 align-left

B bold

 pencil-square ([edit](#))

 align-justify

 text-width

A font


Configurando o listar-produtos.component.html

Vamos copiar o comando que está na tela

```
<i class="fa fa-pencil" aria-hidden="true"></i>
```

fa-pencil  - Unicode: f040 - Created: v1.0 - Categories: Web Application Icons

After you get **up and running**, you can place Font Awesome icons just about anywhere with the `<i>` tag:

 fa-pencil

```
<i class="fa fa-pencil" aria-hidden="true"></i>
```

Configurando o listar-produtos.component.html

- Vamos no nosso código, dentro do último <td> e criar um botão. Digite:
 - button.btn.btn-warning <TAB>
 - Como preenchimento desse button, colar o comando do ícone pencil
- Ficará assim:


```
19      <tbody>
20        <tr>
21          <td>1</td>
22          <td>Curso de Angular</td>
23          <td>31/12/2021</td>
24          <td>R$ 500,00</td>
25          <td>
26            <button class="btn btn-warning">
27              <i class="fa fa-pencil" aria-hidden="true"></i>
28            </button>
29          </td>
30        </tr>
31      </tbody>
```

Configurando o listar-produtos.component.html

Vamos salvar e verificar se deu certo. Já temos o botão com o ícone da caneta.

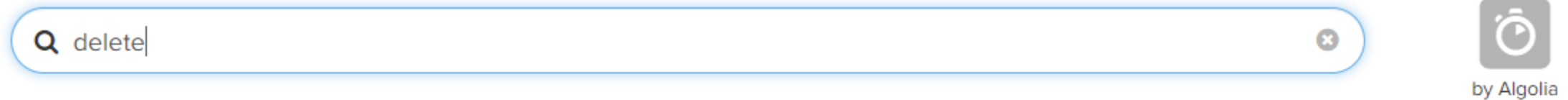
Listar Produtos

[Cadastrar](#)

#	Nome	Validade	Valor	Ações
1	Curso de Angular	31/12/2021	R\$ 500,00	

Configurando o listar-produtos.component.html

- Vamos fazer o próximo botão. Para isso, vamos voltar na documentação do font awesome e vamos procurar um ícone de delete. Tem vários, vamos utilizar o do lixo (vazio).
- Vamos clicar na seleção abaixo.




Search for 'delete'



Configurando o listar-produtos.component.html

Vamos copiar o código

fa-trash-o  - Unicode: f014 - Created: v1.0 - Categories: Web Application Icons

After you get **up and running**, you can place Font Awesome icons just about anywhere with the `<i>` tag:

 fa-trash-o

```
<i class="fa fa-trash-o" aria-hidden="true"></i>
```

Configurando o listar-produtos.component.html

- Após o botão de editar, vamos criar um novo botão. Digite o comando abaixo:
 - ▶ button.btn.btn-danger <TAB>
 - ▶ Como preenchimento desse button, colar o comando do ícone pencil
- Ficará assim:



```
19      <tbody>
20      <tr>
21      <td>1</td>
22      <td>Curso de Angular</td>
23      <td>31/12/2021</td>
24      <td>R$ 500,00</td>
25      <td>
26      <button class="btn btn-warning">
27      | <i class="fa fa-pencil" aria-hidden="true"></i>
28      </button>
29      <button class="btn btn-danger">
30      | <i class="fa fa-trash-o" aria-hidden="true"></i>
31      </button>
32      </td>
33      </tr>
34      </tbody>
```

Configurando o listar-produtos.component.html

Vamos salvar e verificar como ficou. Os dois ícones ficaram muito juntos, vamos melhor no nosso código.

Listar Produtos

[Cadastrar](#)

#	Nome	Validade	Valor	Ações
1	Curso de Angular	31/12/2021	R\$ 500,00	 



Configurando o listar-produtos.component.html

Vamos inserir uma margem left de 2 no botão excluir, ficará assim:

```
<button class="btn btn-danger ml-2">  
  <i class="fa fa-trash-o" aria-hidden="true"></i>  
</button>
```

Listar Produtos

Cadastrar

#	Nome	Validade	Valor	Ações
1	Curso de Angular	31/12/2021	R\$ 500,00	 

Dessa maneira já temos uma página de lista de produtos, onde temos uma tabela que irá exibir nossos produtos, onde simulamos a primeira informação para depois preencheremos com os dados.

Falta ativar o botão Cadastrar para abrir nossa tela de cadastrar produtos.

Configurando o listar-produtos.component.html

- Retornando na nossa aplicação, vamos procurar pelo botão cadastrar e inserir a propriedade routerLink (L deve ser maiúsculo).
- Após o type="button", digitar:
 - ▶ routerLink="" -> Dentro das aspas, deveremos inserir a rota. Vamos procurar para ver qual é a rota. Vamos no arquivo app.routing.module.ts e copiamos a rota: produtos/cadastrar. Vamos copiar e colar no routerLink, conforme segue abaixo:
 - ▶ routerLink="/produtos/cadastrar"
 - ▶ <ALT><SHIFT><F> para formatar o arquivo

```
<h1 class="my-3 text-primary">
  Listar Produtos
  <button type="button" routerLink="/produtos/cadastrar" class="btn btn-success mt-2 float-right">
    Cadastrar
  </button>
</h1>
```

Configurando o listar-produtos.component.html

- Vamos testar....
- Clicando no botão cadastrar, deverá abrir a página para cadastrarmos os nossos produtos, conforme segue abaixo.

