

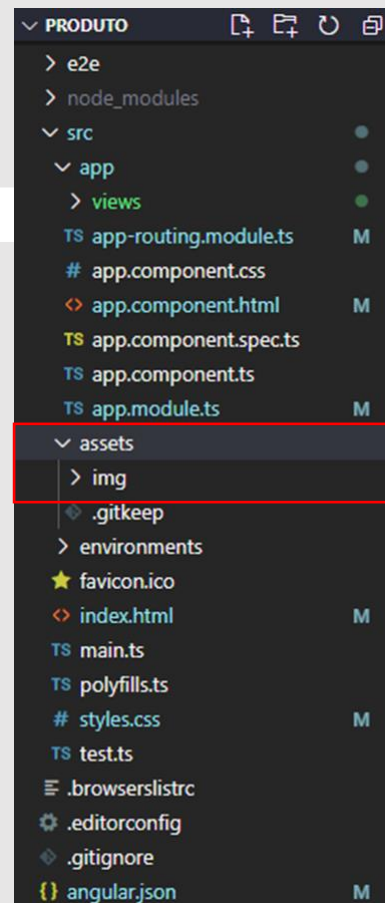
CRIANDO COMPONENTES

Imagem

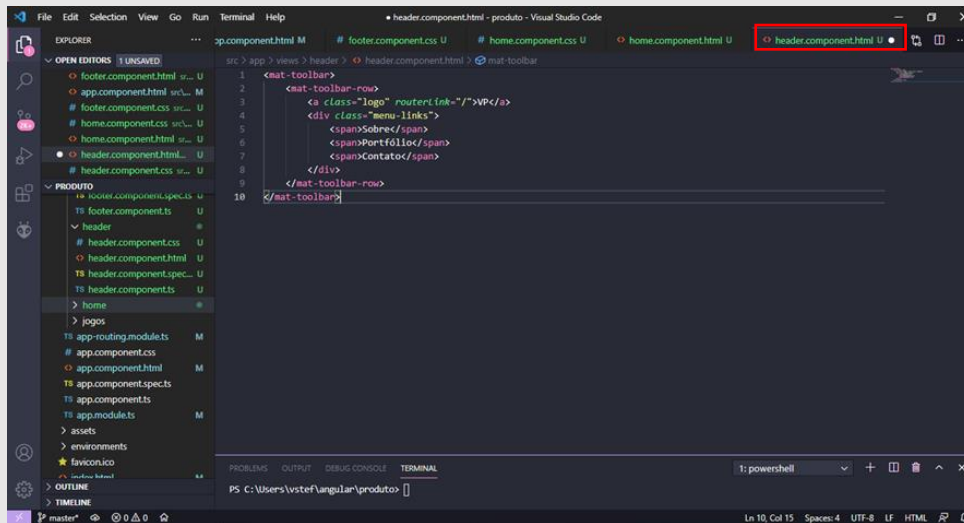
1. Localize a pasta **assets**, que é gerada automaticamente quando um projeto é criado no Angular.

2. Clique sobre ela com o botão direito do mouse, selecione a opção “**new folder**” e crie uma pasta chamada **img**, na qual você adicionará todas as imagens que serão utilizadas em seu site.

Para exemplificar, adicionaremos a imagem do logo do SENAI.



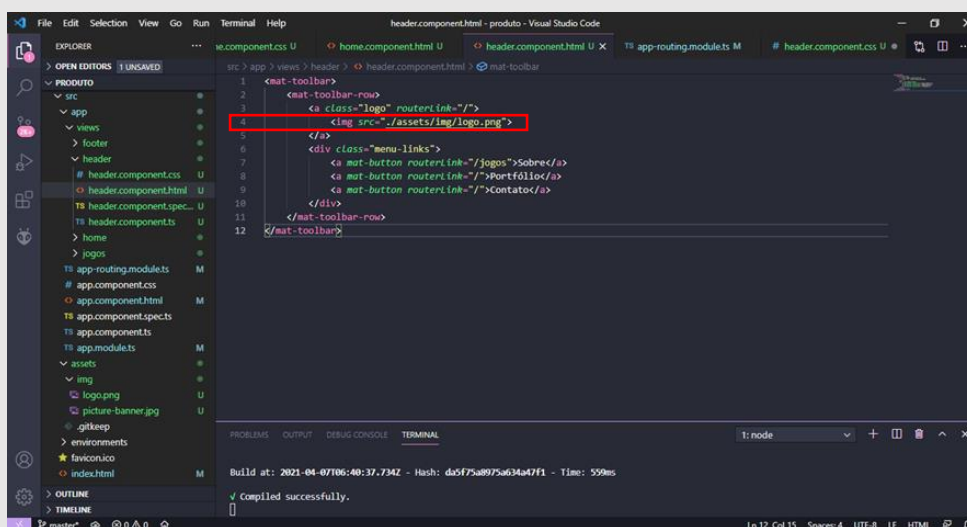
3. Acesse o arquivo **header.components.html**. Faremos algumas alterações no menu criado.



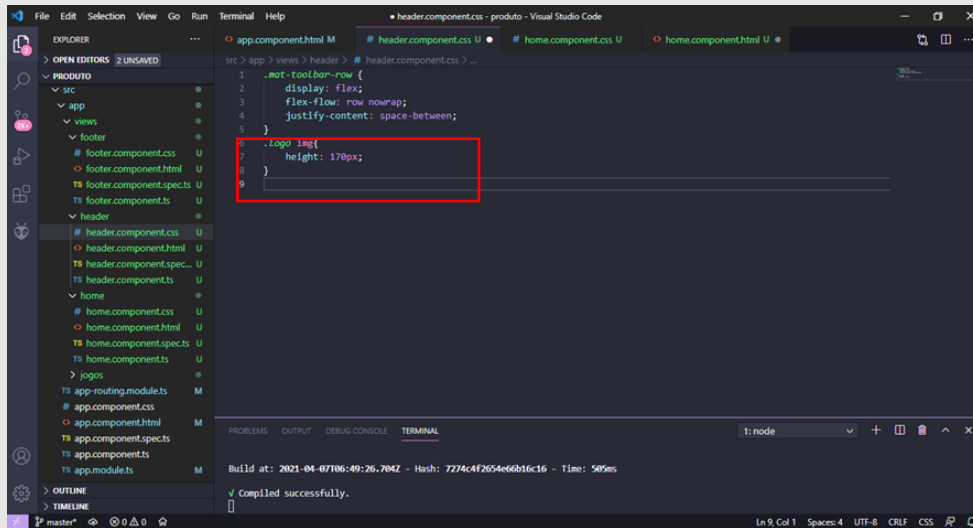
HTML.

Note que, na linha 3 da imagem acima, estamos passando no logo as letras **VP**. Iremos substituí-las pelo logo do SENAI.

Observe a linha 4:

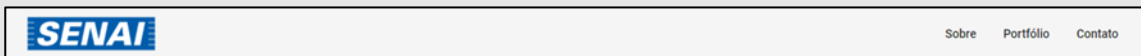


4. No arquivo **header.component.css**, observe as linhas de 6 a 9:



```
1 .mat-toolbar-row {
2   display: flex;
3   flex-flow: row nowrap;
4   justify-content: space-between;
5 }
6 .logo img {
7   height: 170px;
8 }
9
```

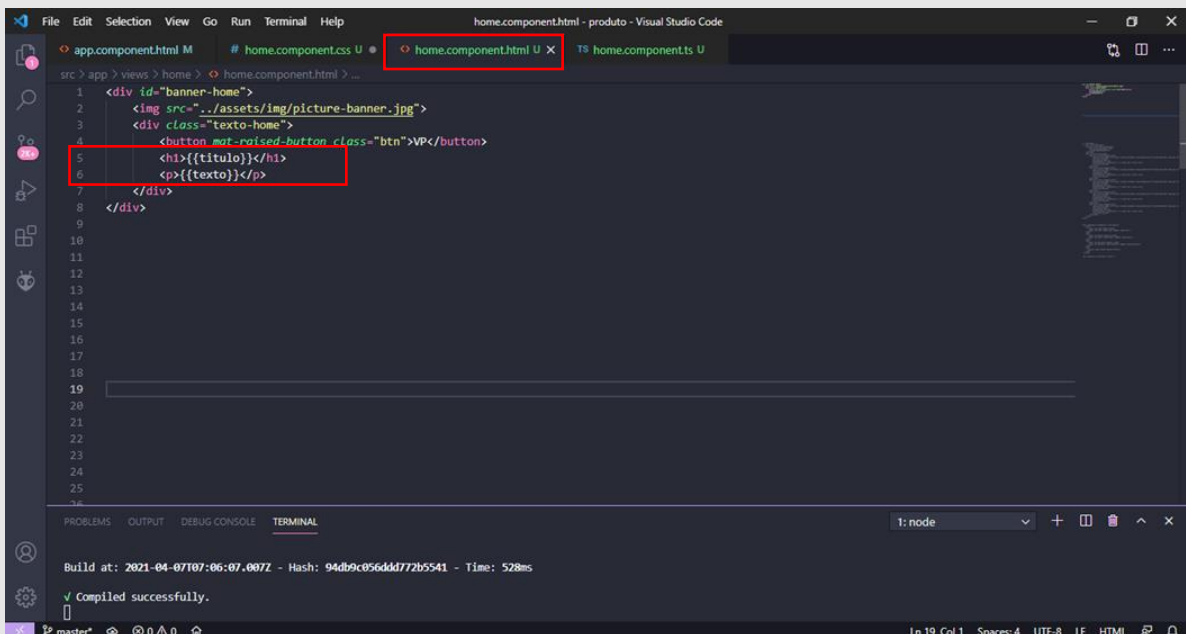
5. Confira o resultado no navegador.



Banner

1. No terminal do Visual Studio Code, crie uma pasta chamada **home ng g c views/home**. Não se esqueça de salvar na pasta **img** a imagem que você deseja inserir.

2. Adicione as seguintes linhas de código destacadas a seu arquivo **home.components.html**.



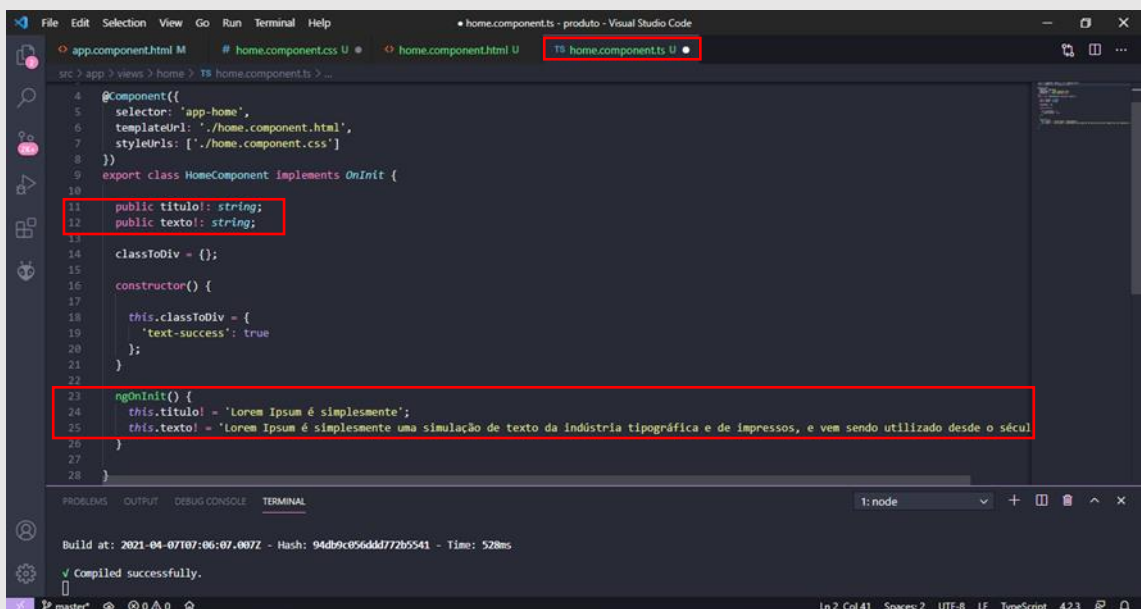
```
1 <div id="banner-home">
2   
3   <div class="texto-home">
4     <button mat-raised-button class="btn">VP</button>
5     <h1>{{titulo}}</h1>
6     <p>{{texto}}</p>
7   </div>
8 </div>
```

Observe que, nas linhas 5 e 6, estamos passando um dado **{{titulo}}** entre duas chaves. Esse título significa uma variável, que configuramos no arquivo **home.components.ts**. Em vez de passar um texto longo na tag **<h1>**, podemos passar uma variável que conterá o texto. Desse modo, o **HTML** interpretará e o **browser** apresentará o valor que estiver dentro dessa variável.

3. Para criar e configurar as variáveis, acesse o arquivo **home.components.ts**.

Note que, nas linhas 11 e 12, criamos uma variável **public** + **nome da variável** e o **tipo** dela.

Nas linhas 23, 24 e 25, dentro do escopo de **ngOnInit**, estamos chamando as variáveis com **this.variável** e estamos passando o valor (tipo **string**) entre aspas, ou seja, tudo que estiver entre aspas será interpretado como texto.



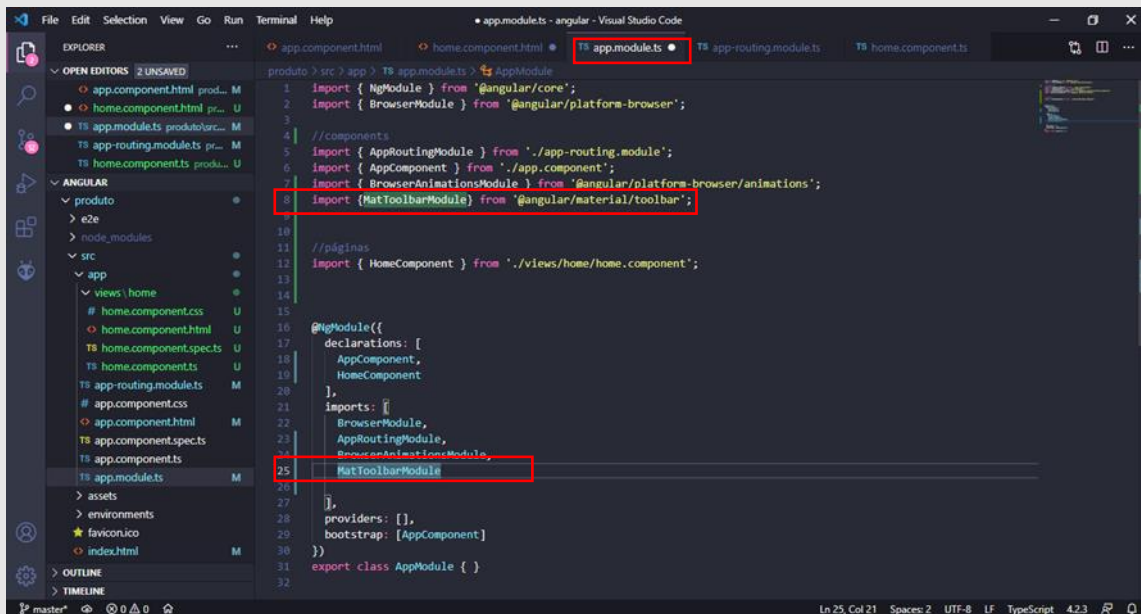
```
4 @Component({
5   selector: 'app-home',
6   templateUrl: './home.component.html',
7   styleUrls: ['./home.component.css']
8 })
9 export class HomeComponent implements OnInit {
10
11   public titulo!: string;
12   public texto!: string;
13
14   classToDiv = {};
15
16   constructor() {
17
18     this.classToDiv = {
19       'text-success': true
20     };
21   }
22
23   ngOnInit() {
24     this.titulo = 'Lorem Ipsum é simplesmente';
25     this.texto = 'Lorem Ipsum é simplesmente uma simulação de texto da indústria tipográfica e de impressos, e vem sendo utilizado desde o século';
26   }
27
28 }
```

Dica!

No código em HTML, passamos apenas o **nome da variável** entre **chaves duplas**. Caso deseje **mudar o conteúdo do texto ou do título**, basta alterar o **valor da variável**.



4. Abra o arquivo **app.module.ts** e adicione o *import* que você copiou do site do Material à linha 8, na sequência dos demais componentes. Note que, dentro das chaves, há um nome. É necessário copiar esse nome e colá-lo dentro da tag **<imports>**, conforme consta na linha 25. Feito isso, salve (**ctrl + s**) as informações. Agora, adicione o HTML ao seu projeto. Veja abaixo:



```
1 import { NgModule } from '@angular/core';
2 import { BrowserModule } from '@angular/platform-browser';
3
4 //components
5 import { AppRoutingModule } from './app-routing.module';
6 import { AppComponent } from './app.component';
7 import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
8 import { MatToolbarModule } from '@angular/material/toolbar';
9
10
11 //páginas
12 import { HomeComponent } from '../views/home/home.component';
13
14
15 @NgModule({
16   declarations: [
17     AppComponent,
18     HomeComponent
19   ],
20   imports: [
21     BrowserModule,
22     AppRoutingModule,
23     BrowserAnimationsModule,
24     MatToolbarModule
25   ],
26   providers: [],
27   bootstrap: [AppComponent]
28 })
29 export class AppModule { }
```

Importante

Será necessário repetir esse procedimento para todos os componentes do Angular Material.



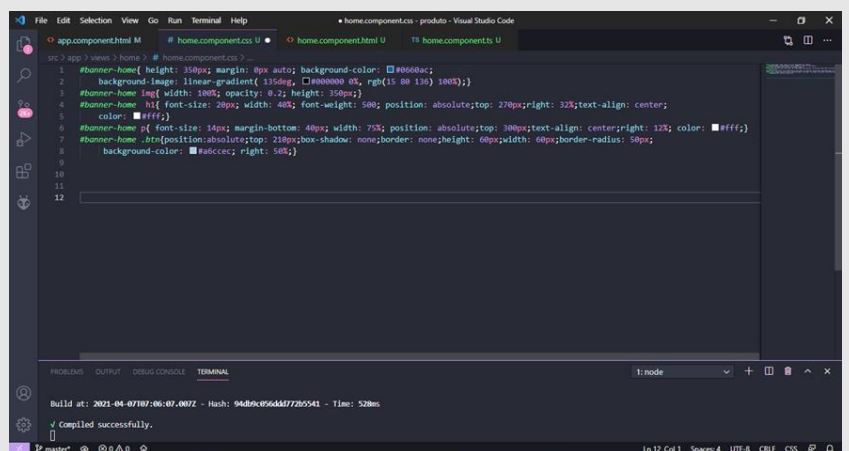
Você sabia?

Modificadores de acesso são palavras-chaves utilizadas para especificar a acessibilidade de uma declaração no **código ts**. Podem ser:



- **Public:** qualquer código externo terá acesso a essas informações.
- **Private:** o código será acessado apenas no local onde foi declarado.
- **Protected:** o código será acessado pela classe que foi criada e pelos derivados.
- **Internal:** o código tem um acesso limitado, sendo acessado apenas pelo *assembly* atual.
- **Protected internal:** o código será acessado pelo *assembly* atual e pelos derivados da classe.

5. Agora, faça a estilização do banner.



6. Salve (**ctrl + s**) as informações. Caso o *server* esteja rodando, ele atualizará automaticamente; caso tenha parado, lembre-se de subir as informações novamente (**ng serve --port 333 -o**).

Veja o resultado:

