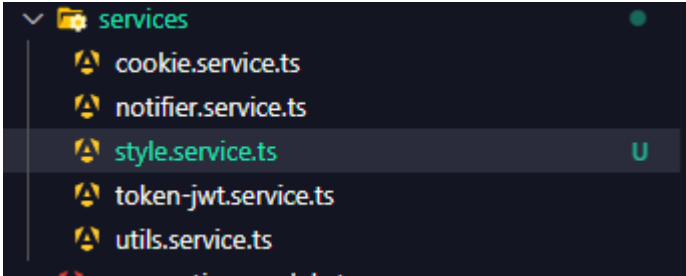


## Padrão de cores ANGULAR

Todas as tags são buscadas no banco DEFAULT

API: <https://github.com/GuilhermeBarboosa/colors-api.git>

Todo projeto terá que ter o service StyleServices (localizado INSCRICAO-FETI-FRONT)



```
1 import { Injectable } from '@angular/core';
2 import { HttpClient } from '@angular/common/http';
3
4 @Injectable({
5   providedIn: 'root',
6 })
7 export class StyleService {
8   colorVariables: { [key: string]: string } = {};
9
10  urlApi = 'http://localhost:8090';
11
12  constructor(private http: HttpClient) {
13    this.getAll().subscribe((data: any) => {
14      var response = JSON.parse(JSON.stringify(data));
15      response.forEach((element: any) => {
16        this.colorVariables[element.tipo] = element.cor;
17      });
18    });
19  }
20
21  getById(id: number) {
22    return this.http.get(`${this.urlApi}/cores/` + id);
23  }
24
25  getAll() {
26    return this.http.get(`${this.urlApi}/cores`);
27  }
28 }
```

Este código busca através da url todas os tipos e cores. Interessante colocar `console.log(response)` para saber quais estilos está sendo puxada do banco.

No banco foi denominado tipo e cor:

Tipo: `bg_primary_color`, `bg_secundary_color`, `font_color`, etc (utilizar sempre `_` no lugar de espaço)

Cor: `#26318`, `#73493` (sempre colocar `#` na frente)

Exemplo de response obtido:

Iremos sempre utilizar o tipo para implementar no component

```
▼ (3) [{...}, {...}, {...}] ⓘ  
  ▶ 0: {id: 1, tipo: 'bg_primary_color', cor: '#191970'}  
  ▶ 1: {id: 2, tipo: 'bg_secundary_color', cor: '#DC143C'}  
  ▶ 2: {id: 3, tipo: 'font_color', cor: '#4B0082'}
```

Parte de implementação

```
1 import { Component, Input, OnInit } from '@angular/core';  
2 import { CookieService } from 'src/app/services/cookie.service';  
3 import { StyleService } from 'src/app/services/style.service';  
4  
5 @Component({  
6   selector: 'app-navbar-login',  
7   templateUrl: './navbar-login.component.html',  
8   styleUrls: ['./navbar-login.component.css'],  
9 })  
10 export class NavbarLoginComponent implements OnInit {  
11   @Input() value: String | undefined;  
12   ngOnInit(): void {  
13   }  
14  
15   constructor(public styleService: StyleService) {}  
16 }  
17
```

Sempre que for precisar desses estilos será necessário inicializar o `StyleService` no constructor.

```
<div [ngStyle]="{ 'background-color': styleService.colorVariables['bg_primary_color']}" class="full-width">  
  <div *ngIf="value == 'CMS'">  
    <div class="flex items-end justify-end">  
        
    </div>  
  </div>
```

Utilizando a propriedade `[ngStyle]` vamos colocar o tipo de estilo seja `color`, `background-color`, `size`, etc. E depois definir qual tipo de variável queremos definir. Neste exemplo foi definida a cor do background.

Outros exemplos:

```
<a [ngStyle]="{ 'background-color': styleService.colorVariables['bg_secondary_color']}"
```

```
<div class="logo_name" [ngStyle]="{ 'color': styleService.colorVariables['font_color']}"
```