Utilizando HttpClient

# Script Component

import { MessageService } from "primeng/primeng";

import { Cliente } from "./../../shared/models/model";

import { ClientesService } from "./../clientes.service";

import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

@Component({

selector: 'app-listar',

templateUrl: './listar.component.html',

styleUrls: ['./listar.component.css']

})

export class ListarComponent implements OnInit {

clientes: Cliente[];

constructor(private service: ClientesService, private message: MessageService, private router:Router) {

}

ngOnInit() {

this.carregarClientes();

}

carregarClientes() {

this.service.consultar().subscribe(response => {

console.log(response);

this.clientes = response;

}, erro => {

this.message.add({ severity: 'error', summary: 'Erro ao carregar clientes', detail: erro.message });

});

}

}

# Client Service

import { map } from "rxjs/operators";

import { Observable } from "rxjs";

import { Cliente } from "src/app/shared/models/model";

import { HttpClient } from "@angular/common/http";

import { Injectable } from '@angular/core';

import { environment } from 'src/environments/environment';

@Injectable({

providedIn: 'root'

})

export class ClientesService {

BASE\_URL: string;

constructor(private http: HttpClient) {

this.BASE\_URL = environment.url;

}

salvar(cliente: Cliente): Observable<Cliente> {

return this.http.post<Cliente>(this.BASE\_URL.concat('/clientes'), cliente).pipe(map(json => { return json }));

}

buscarPorCodigo(codigo: number): Observable<Cliente> {

return this.http.get<Cliente>(this.BASE\_URL.concat(`/clientes/${codigo}`)).pipe(map(json => { return json }));

}

consultar(): Observable<Cliente[]> {

return this.http.get<Cliente[]>(this.BASE\_URL.concat('/clientes')).pipe(map(json => { return json }));

}

alterar(cliente: Cliente): Observable<Cliente> {

return this.http.patch<Cliente>(this.BASE\_URL.concat('/clientes'), cliente).pipe(map(json => { return json }));

}

deletar(codigo: number): Observable<Cliente> {

return this.http.delete<Cliente>(this.BASE\_URL.concat(`/clientes/${codigo}`)).pipe(map(json => { return json }));

}

}