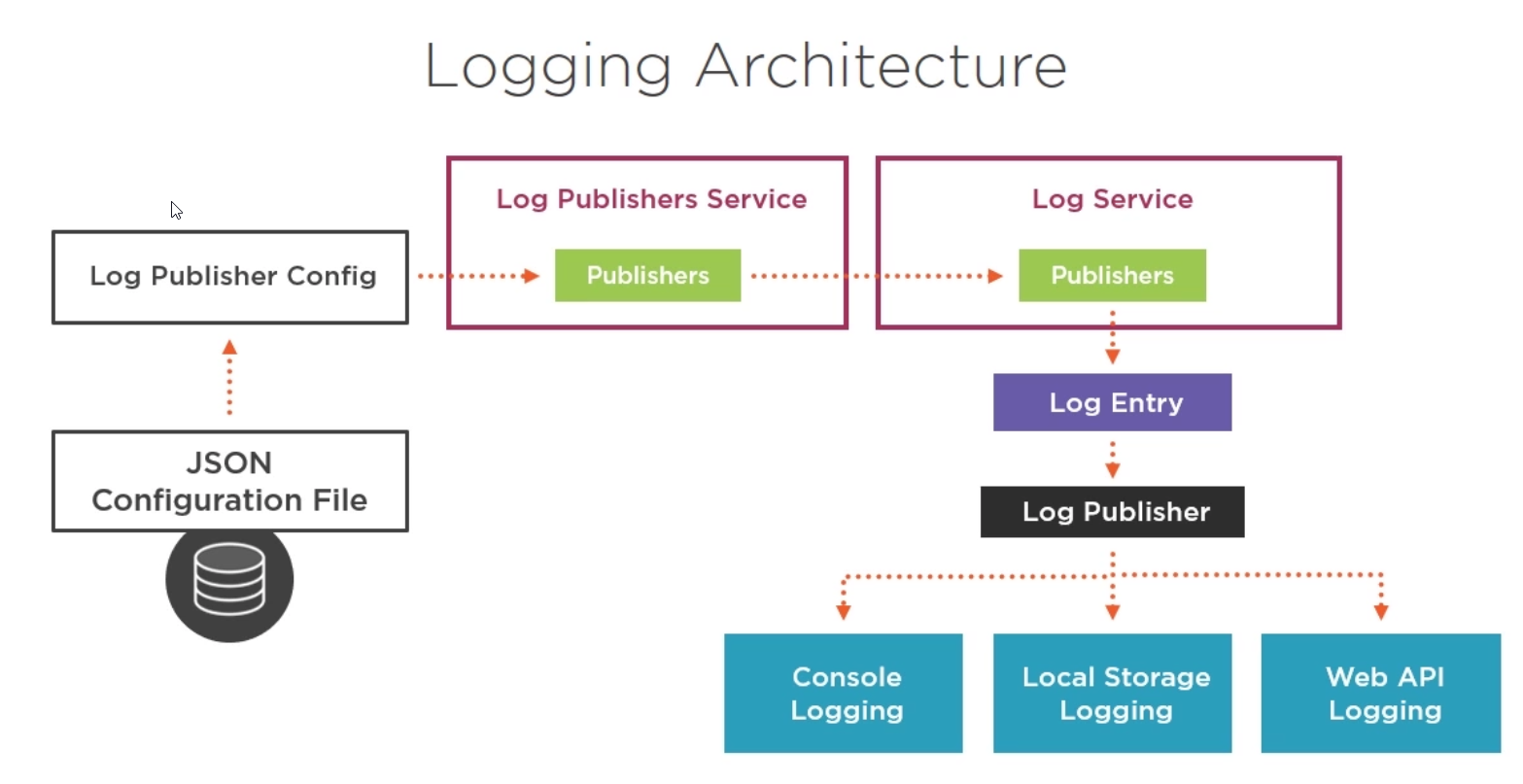
Mebo

#######################################################################  
Angular Reusable Loging Service

-------------------------------------------

دلوقتى احنا المفروض هنعمل سيرفيس لوج بيطلعلك الايرور او يعرفك الداتا الى رجعه و بتقدر تستخدها فى اماكن كتير و متقولكش بتعمل ايه يا صديقى ههههههه فمثلا انا هقدر ارجع اللوج فى   
كنسول و كذلك فى اللوكال ستوريدج و كذلك اقدر اخزنها برضوا فى ال Database   
ده هيبقى الاركتيتكشتار بتعها



طب احنا ليه بنعمل السيرفيس دى ؟؟

عشان يا صديقى ال console.log() مش بيسجل و بالتالى مش هيبقى عملك سجل بالاخطاء الى بتحصل فى الابلكيشن بتاعك و باالتالى انت محتاج حاجه تبقى بتسجل اللوج عشان تتبعها بعد كده

كذلك الكونسول لوج ده بيظهر فقط فى الكونسول و انا عاوز اظهر اللوج بتاعى فى كذا مكان

Mebo

#######################################################################  
Angular Basic Logging Service

-------------------------------------------

دى الطريقه العاديه تماما هنبتدى بيها اننا نعمل سيرفيس خاص باللوج يا صديقى و بالتالى ده حاجه بيسطه جدا توريك اصلا بعمل سيرفيس اذاى

أولا فى ال App.Module.ts هتحط السيرفيس بتاعت اللوجر فى ال providers

  providers:  [LoggerService,TestService],

بالنسبه للسيرفيس نفسها اهه logger.services.ts

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

@Injectable({

  providedIn: 'root',

})

export class LoggerService {

  constructor() {}

  log(msg: any) {

    console.log(new Date() + ' : ' + JSON.stringify(msg));

  }

}

بالنسبه بقه للكومبونانات الى هتسخدم فيه السيرفيس يا صديقى اهه

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

import { LoggerService } from './Services/Logger.service';

@Component({

  selector: 'app-root',

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

  styleUrls: ['./app.component.scss'],

})

export class AppComponent implements OnInit {

  constructor( private Logger : LoggerService) {

  }

  ngOnInit(): void {

    this.Logger.log("Hello from AppComponent");

  }

}

Mebo

#######################################################################  
Making Logging Levels in Services

-------------------------------------------

الليفلز دى انى اعمل للسيرفيس حاجه ذى مابينج كده اخليه يشتغل بيها بحيث ان لما اقوله انا فى الليفل الفلانى يشغل الليفل ده بس و ميشغلش غيره و لما اقوله عاوز اوقف كله يوقف الدنيا كلها

و الليفيلات دى بتبقى على شكل

Enumeration

export enum LogLevels {

  All =0 ,

  Debug =1 ,

  Info =2 ,

  Warn =3 ,

  Error =4 ,

  Fetal =5 ,

  Off =6 ,

  }

بعد كده هعمل فانكشن مهمتها انها لو اى ليفل من الى فوق دول اللوجر يشتغل بس لو الليفل بيساوى Off = 6 ساعتها ميشغلش اللوجر يا صديقى

 //this function to check if logger will be On OR Off

  level: LogLevels = LogLevels.All;

  private ShouldLog(level : LogLevels) : boolean{

    let ret:boolean= false;

    if(this.level != LogLevels.Off && level >= this.level){

      return true;

    }

    return ret;

  }

بعد كده هعمل الفانكشن الى اقوله اكتبلى بيها اللوج يا صديقى فى حاله ان الاتشيك الى فوق دى قالت ان انا فى ليفيل يسمح ليا انى اكتب لوج

  logWithDate:boolean= true;

  private WriteToLog(msg:any  , level:LogLevels){

    let value:string = "";

    if(this.ShouldLog(level)){

      if(this.logWithDate) {

        value += new Date() + " - ";

      }

        value += "Type of Log : " + LogLevels[level]  + " - Message : " + msg;

        console.log(value);

    }

  }

و هنا بقوله يا صديقى انا هبعتلك الرساله و الليفل بتعها هتخش بقه تشوف لو انا الليفل بتاعى بيتييح ان اللوج يشتغل فهتعمل تشيم تانى الى هو ارجه مع اللوج التاريخ و لا لأ لو عاوزه حطه فى الفاليو اما لو مش عاوزه فهتجيبلنا نوع اللوج و الرساله و خلاص

طيب كده انا محتاج انادى للفانكشن دى من الفانكشنز الى انا هستخدمها بشكل رئيسى بعد كده ذى فانكشن اللوج فمثال على ذلك هتبقى فانكشن اللوج مكتوبه كده

  log(msg: any) {

      this.WriteToLog( msg , LogLevels.All);

    }

و بالتالى كل السيرفيس هتبقى بالشكل ده ياباشا

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

export enum LogLevels {

  All =0 ,  Debug =1 ,  Info =2 ,  Warn =3 ,  Error =4 ,  Fetal =5 ,  Off =6 ,

  }

@Injectable({

  providedIn: 'root',

})

export class LoggerService {

  constructor() {}

  //this function to check if logger will be On OR Off

  level: LogLevels = LogLevels.All;

  private ShouldLog(level : LogLevels) : boolean{

    let ret:boolean= false;

    if(this.level != LogLevels.Off && level >= this.level){

      return true;

    }

    return ret;

  }

  logWithDate:boolean= true;

  private WriteToLog(msg:any  , level:LogLevels){

    let value:string = "";

    if(this.ShouldLog(level)){

      if(this.logWithDate) {

        value += new Date() + " - ";

      }

        value += "Type of Log : " + LogLevels[level]  + " - Message : " + msg;

        console.log(value);

    }

  }

  log(msg: any) {

      this.WriteToLog( msg , LogLevels.All);

    }

}

و الكومبونانت app.component.ts الى هناديه فيها بالشكل ده

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

import { LoggerService } from './Services/Logger.service';

@Component({

  selector: 'app-root',

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

  styleUrls: ['./app.component.scss'],

})

export class AppComponent implements OnInit {

  constructor( private Logger : LoggerService) {

  }

  ngOnInit(): void {

    this.Logger.log("Hello from AppComponent");

  }

}

لو حبيت اغير الليفل من الكومبونانت اقدر يا صديقى و هوريك اهه انى اقدر اقفل اللوج مثلا من الكومبونانت بتاعى

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

import { LoggerService, LogLevels } from './Services/Logger.service';

@Component({

  selector: 'app-root',

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

  styleUrls: ['./app.component.scss'],

})

export class AppComponent implements OnInit {

  constructor( private Logger : LoggerService) {

  }

  ngOnInit(): void {

**this.Logger.level = LogLevels.Off;**

    this.Logger.log("Hello from AppComponent");

  }

}

Mebo

#######################################################################  
Add Method for Each Level in Enum

--------------------------------------------------

دلوقتى جه الوقت الى نطبق فيه ال Singularity و هو ان كل فانكشن يبقى ليها مهمه خصه بيها هى و بالتالى هنخلى لكل ليفل من ليفيلات اللوج فانكشن خاصه بيه بتاخد الرساله

    log(msg: any ) {

      this.WriteToLog( msg , LogLevels.All);

    }

    Debug(msg: any ) {

      this.WriteToLog( msg , LogLevels.Debug);

    }

    Info(msg: any ) {

      this.WriteToLog( msg , LogLevels.Info);

    }

    Warn(msg: any ) {

      this.WriteToLog( msg , LogLevels.Warn);

    }

    Error(msg: any ) {

      this.WriteToLog( msg , LogLevels.Error);

    }

    Fetal(msg: any ) {

      this.WriteToLog( msg , LogLevels.Fetal);

    }

طب عاوز ابعت للوج بقه شويه حاجات Optional معاها تكتبهم لو اتبعتوا يكتبهم لو متبعتوش خلاص دى تتنفذ اذاى ؟؟

دى يا سيدى ابعتلها ArrayOfParam نسميه OptionalParameter

و هكيون التغيير فى الفانكشنز الجديده دى و فى الفانكشن الى اسمها writeToLog

لتصبح السيرفيس

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

export enum LogLevels {

  All = 0,

  Debug = 1,

  Info = 2,

  Warn = 3,

  Error = 4,

  Fetal = 5,

  Off = 6,

}

@Injectable({

  providedIn: 'root',

})

export class LoggerService {

  constructor() {}

  //this function to check if logger will be On OR Off

  level: LogLevels = LogLevels.All;

  private ShouldLog(level: LogLevels): boolean {

    let ret: boolean = false;

    if (this.level != LogLevels.Off && level >= this.level) {

      return true;

    }

    return ret;

  }

  logWithDate: boolean = true;

  private WriteToLog(msg: any, level: LogLevels, Params: any[]) {

    let value: string = '';

    if (this.ShouldLog(level)) {

      if (this.logWithDate) {

        value += new Date() + ' - ';

      }

      value += 'Type of Log : ' + LogLevels[level] + ' - Message : ' + msg;

      console.log(value);

    }

  }

  log(msg: any, ...OptionalParams: any[]) {

    this.WriteToLog(msg, LogLevels.All, OptionalParams);

  }

  Debug(msg: any, ...OptionalParams: any[]) {

    this.WriteToLog(msg, LogLevels.Debug, OptionalParams);

  }

  Info(msg: any, ...OptionalParams: any[]) {

    this.WriteToLog(msg, LogLevels.Info, OptionalParams);

  }

  Warn(msg: any, ...OptionalParams: any[]) {

    this.WriteToLog(msg, LogLevels.Warn, OptionalParams);

  }

  Error(msg: any, ...OptionalParams: any[]) {

    this.WriteToLog(msg, LogLevels.Error, OptionalParams);

  }

  Fetal(msg: any, ...OptionalParams: any[]) {

    this.WriteToLog(msg, LogLevels.Fetal, OptionalParams);

  }

}

و هنديها فى الكومبونانت كده

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

import { LoggerService, LogLevels } from './Services/Logger.service';

@Component({

  selector: 'app-root',

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

  styleUrls: ['./app.component.scss'],

})

export class AppComponent implements OnInit {

  constructor(private Logger: LoggerService) {}

  ngOnInit(): void {

    //  this.Logger.level = LogLevels.Off;

  this.Logger.level = LogLevels.Warn;

    this.Logger.log('Hello from AppComponent');

    this.Logger.Debug('Hello from AppComponent');

    this.Logger.Info('Hello from AppComponent');

    this.Logger.Warn('Hello from AppComponent');

    this.Logger.Error('Hello from AppComponent');

    this.Logger.Fetal('Hello from AppComponent');

  }

}

شوف بقه السنتراليزيسن انك لما تفتح ليفل الاوف ده هههههه بالسلامه انت كل ده ميظهرش و لو ليفل ذى warn شغلته كل الى قبله يقفوا و الى بعده بس هما الى يشتغلوا

Mebo

#######################################################################  
Format Optional Parameters

--------------------------------------------------

المره الى فاتت شغلنا السيرفيس بس مهتمناش بال OptionalParameters فالمره ده بقه المفروض هبعت برامتيارز و المفروض انى اخدهم بشكل معين احطهم فى اللوج و بالتالى تعالى نعملهم فانكشن خاصه بيهم اسمها FormatBuilderForOptParam