

-CLASE DE HOY

- Observables para obtener datos de archivo JSON
- Servicios HttpClient
- Ejercicios básicos observables
- Parámetros en TS
- Ejercicios básicos parámetros
- Ejercicio Spotify o similar
- Grupos y peticiones específicas.

GRUPOS

DUDAS

- ENLACE AL PROYECTO DE HOY: <https://github.com/alesrin/observables>
- Comandos avanzados de búsqueda en Google: <https://negociosyestrategia.com/blog/comandos-avanzados-google/>
- ~~Manel: Que sucede si tenemos 2 constructores?? Los fusionamos? Ya que marca error si hay 2~~
- ~~Ariel: que pasa si se declara public? otros componentes se pueden suscribir a ese servicio?~~
- La importación HttpClientModulo la detecta automática?, a mi no.
- ~~Ariel: en caso de las API keys, se especifica la URL?~~
- Pablo: ¿no es buena practica hacer primero un console.log de los datos para ver su estructura?
- Sanchi: Puedes agregar los códigos ejemplo de los pasos en Notion? Sí, os lo comparto todo ahora en cuanto acabemos esta explicación

COMANDOS

- Instalar rxjs para poder utilizar observables en Angular:
 - npm install --save rxjs@6
 - npm install --save rxjs-compatible

EJERCICIO OBSERVABLE

1. Instalar rxjs para poder utilizar observables en Angular:

- **npm install --save rxjs@6**
- **npm install --save rxjs-compat**

2. Importa el módulo HTTPClientModule en el módulo raíz de tu aplicación, además inyéctalo en la propiedad import. (Si estás trabajando en una aplicación con otros módulos además del módulo raíz, deberás importarlo además también en los módulos en los cuales quieras utilizar este servicio.

3. Importa el servicio HttpClient en el constructor del componente en el que lo quieras utilizar. (Previamente deberás haber realizado el paso 2 en el módulo de este componente).

4. Inyecta el servicio en el constructor para poder utilizar el parámetro http (el cual incluye un componente de HttpClient) dentro de mi clase.

5. Crear la función del observable e inicialízala en el método ngOnInit(). Estructura observable:

- **Servicio(parámetro this.http) + método + fuente de datos + subscripción**

6. Inicializamos (y tipamos) la variable de los datos a los que nos vamos a subscribir

7. Creamos la estructura del HTML en nuestra plantilla utilizando la directiva *ngFor para iterar los objetos de nuestro array utilizando Data Binding unidireccional