Apuntes:

ANGULAR:

Es un framework, es un marco de trabajo estandarizado.

Se siguen ciertos estándares, la estructura es parecida a otros desarrolladores

Las palicaciones son mudilares. SE CREAN MODULOS

Google mantiene el framework de angular

Angular se compone de 5 bloques fundamentales: componentes, rutas, directivas, servicios, modulos.

**COMPONENTES:**

* **CONTIENE HTML**
* **CLASE TYPE SCRIPT**

**SERVICIOS:** lugares centralizados de la información

DIRECTIVAS:

Hay tres tipos:

* **De componentes**
* **Estructurales**
* **De atributos**

**RUTAS:**

Mostrar diferentes componentes basados en el Url del navegador web

MODULOS: Permite agrupar las funcionalidades y las estructuras del nuestro código

Notas de curso

**ng new <nombre de la aplicación> --standalone false**

**Modulos:**

Es una clase que tiene un decorador, el modulo en si mismo es un agrupador, encapsula una funcionalidad.

Notas sobre git:

* para regresar los cambios del ultimo commit ejecutar el comando

git checkout -- .

* agregar el repositorio que quiero conectar
* git remote add origin <https://github.com/rutademirepositorio>
* renombrar la rama git Branch -M main
* subir cambios al repositorio git push -u origin main. Cuando se ejecuta el push por primera vez ua no es necesario escribir todo el comando. Hay que escribir solo el git push

Modulos:

Cada modulo internamente tiene sus piezas independientes.

* Components
* Interfaces
* Pages
* Services.

Pensando en componentes pequeños.

El objetivo asignar responsabilidades a pequeñas piezas de código, para reutilizar fácilmente.

Comunicación entre componentes

**@input** define una propiedad que puede ser enviada desde el padre hacia el componente hijo.

**@output** Define una salida del componente para que el componente padre pueda suscribirse y escuchar

EventEmiter (Emisor de Eventos)

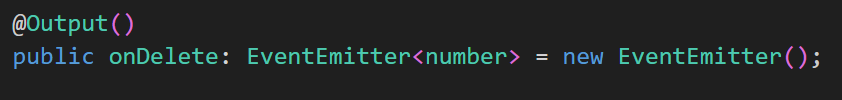
Permite que los componentes se comuniquen entre si

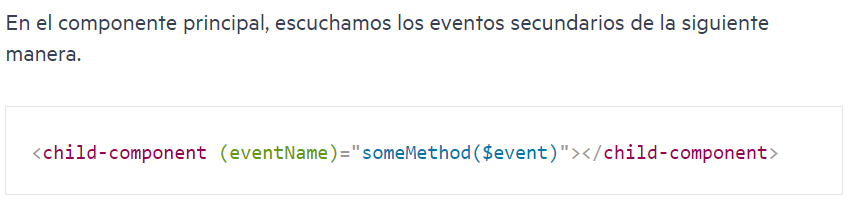
Puedes implementarlo con el uso del decorador @Output() y la clase EventEmitter. El componente padre puede escuchar el evento emitido mediante el uso del decorador @Input() . El envío de un evento se realiza a través del método EventEmitter.emit().

Formato para utilizar el emisor de eventos.



Como esta definido en el curso.





Service:

Aquí se manejan la lógica de negocio

Despliegue a producción.

Ng build

Se crea la carpeta dist. Esa carpeta la ponemos en el servidor para poder ejecutar