Observables, Subject y relativos.

Imaginemos una aplicación sencilla con un formulario para registrar personas.  
Creamos un componente que sería nuestro formulario, un componente que seria la lista de personas y un service.

***En el formulario podemos utilizar formGroup…***

En el service, como vamos a trabajar con personas creamos una interfaz:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tenemos que crear en el service también un array con las personas y queremos que en el constructor se inicialice a vacio.

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

OBSERVABLES: Te suscribes, y puede ser un array o lo que sea, si hay algún cambio en ese valor, te lo vuelve a mandar

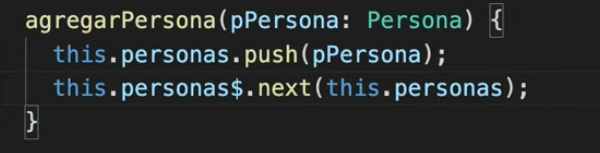
El Subject es como una lanzadera, le mandas con el método **next()** el array de personas o lo que sea y a la gente suscrita le llegara.

Texto

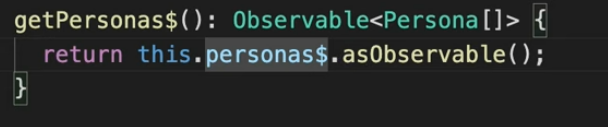
Descripción generada automáticamente

En nuestro caso tenemos el array de personas y el subject que nos va a devolver un array de personas con cada modificación (Lo que va entre <> es lo que devuelve). **Importante inicializar cada elemento.**

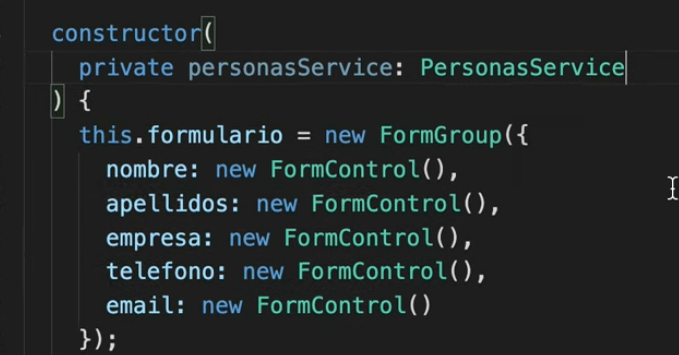
Luego si en algún momento queremos añadir una persona a **nuestro array** es tan simple como un método que haga push. EL **SUBJECT** entra en juego aquí, puesto que podemos usarlo para que devuelva el array con la modificación a todos aquellos suscritos, y así reciban la última versión con toda la info de las personas. Esto se hace con el método next() del subject, y de parámetros mandamos lo que queremos que reciban todos aquellos suscritos, en este caso el array de personas.

****

Sin embargo, para poder suscribirte a el subject necesitas un método que te lo devuelva como **observable**.



Si hacemos un formulario y queremos usar este observable (definido en el service). Lo inyectamos en el constructor (), y luego podemos llamar a cualquier método del service.

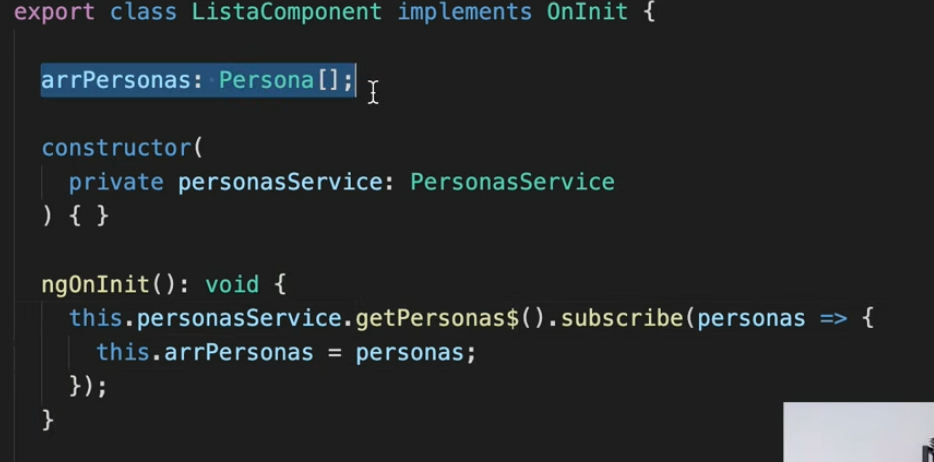


Con las cosas propias del formulario que podemos ignorar.

Luego que cuando se haga submit del formulario se llama al método de agregar persona.

EN EL COMPONENTE LISTA.

Vamos a mostrar las personas, para ello necesitamos estar bien actualizados de las personas que hay en cada momento, para ello nos podemos suscribir desde el componente de la lista y pasarle los valores de la suscripción.



Importante, suscribirse y marcar que personas (la variable que devuelve la suscripción) sea guardada en nuestro array