**RxJS**

An external library used to work with observables in Angular

**Creating a service in Angular**

* First you create a service using ng generate service “service-name” or **ng g s “service-name”**
* Next, within the service create a function that returns the information you want to be accessible for each component you want to implement it in
* Next within the component typescript file import the service from its correct location and add the name of the service to the @component section, **providers: [“name of service”]**
* Next within the constructor of the component file create a private premise with parameters and give it a name, **constructor(private “made up name” : “name of the service”)**. Adding this to the constructor allows access to everywhere within the class.
* Next create an array or variable or whichever depending on the type of information your sending across and save it empty
* Next create a function that grabs the name of the service via the made up name that was created in within the constructor and set the empty variable as the information of the service, **getInfo(){this.”empty variable” = this.”service name”. “info name or available”**

**Makining an HTTP call**

What is http. Http stands for Hypertext Transfer Protocol. It is used to communicate between browsers and web servers.

* First you have to import the http client module from **@angular/common/http** within the module along with adding http client module within imports.
* Next within the service that is already created, inject http Client form **@angular/common/http**.
* Next add a private http name within the constructor. **constructor(private “made up name” : HttpClient)**
* Next create a fuction and assign an observable to it. **getUsers():observable<”name of observable”>{return this.http.get<”name of observable>(“link to api with information)}**. The http client is added within the constructor so it can be accessible anywhere within the class. An observable can also be an array e.g, **observable<”name of observable[]”>** instead of a single information e.g, **observable<”name of observable”>**.