

Angular

Tahaluf Training Center 2021



Chapter 7

- 1 RXJS
- 2 Hits API
- 3 Toastr



RXJS

RxJS is Reactive Extension for JavaScript. It is a JavaScript library that **uses** observables to work with reactive programming that deals with asynchronous data calls, callbacks and event-based programs.

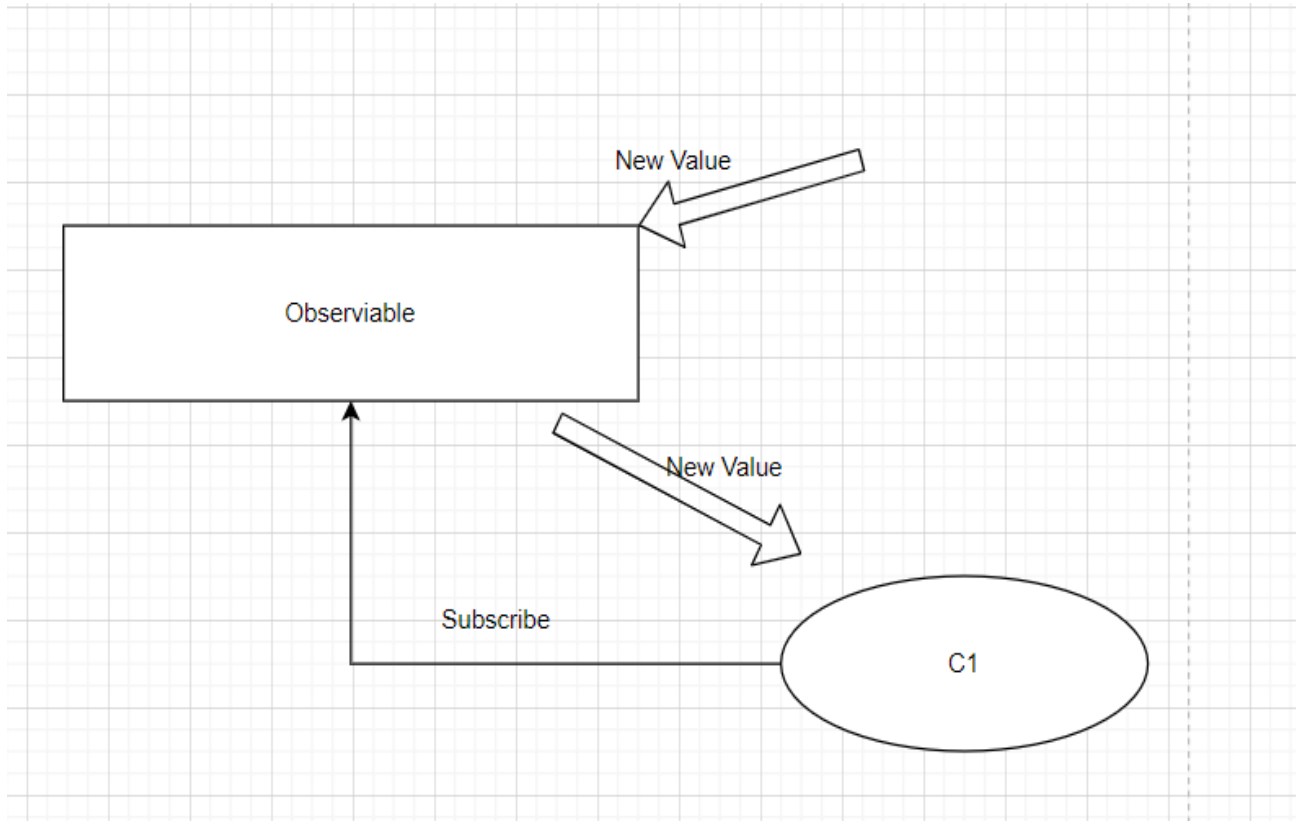


RXJS

An **Observable** is a Producer of multiple values, "pushing" them to Observers (Consumers). A Function is a lazily evaluated computation that synchronously returns a single value on invocation



RXJS



RXJS

In homeServices : declare an object of behavior subject and give the default value =0

```
numberOfActiveCourse = new BehaviorSubject(0);
```



RXJS

Inside OnInit function:

```
ngOnInit(): void {  
  console.log("Strated !!!");  
  this.AlertActiveCourse()  
  setInterval(()=>{
```

```
    this.homeServices.numberOfActiveCourse.next(this.  
    homeServices.numberOfActiveCourse .value+1)  
  },3000)
```



RXJS

And use this object in home component:

```
AlertActiveCourse() {  
    this  
    .homeServices.numberOfActiveCourse.subscribe((v  
alue)=>{  
        alert('the new value is '+value);  
    })  
}
```



Exercise:

Add other variable in home.component.ts and make the value for this variable is the value for number of active user and display it in the home page.

In the same way for profile page.



Exercise Solution:

In `home.component.ts`

```
CurrentActiveCourse:number =0;
```

```
AlertActiveCourse() {  
    this  
    .homeServices.numberOfActiveCourse.subscribe((value)=  
>{  
        //alert('the new value is '+value);  
        this.CurrentActiveCourse=value;  
    })  
}
```



Exercise Solution:

In `home.component.html`

```
<h2>Welcome to our portal</h2>
<h2>Current Active Course is
:{{CurrentActiveCourse}}</h2>
    <h2>{{homeServices.message}}</h2>
```



Exercise Solution:

In profile.component.ts

```
currentCourse:number=0;
```

```
ngOnInit(): void {  
    this  
    .home.numberOfActiveCourse.subscribe(value=>{  
        this.currentCourse=value;  
    })  
}
```



Exercise Solution:

In profile.component.html

```
<h2>Current Active Course {{currentCourse}}</h2>  
<h1>{{home.selctorCours.description}}</h1>  
<h1>  
    {{home.selctorCours.subTitle}}  
</h1>  
<h1>{{home.selctorCours.type}}</h1>
```



Chapter 7

1 RXJS

2 **Hits API**

3 Toastr



Hits API

Angular provide a built in package called http.

To use this package , first you must add the module for this package on shared module for our project.

```
import{HttpClientModule}from '@angular/common/http'
```

And add **HttpClientModule** in import and export section



Hits API

Create an object from httpClient to use API request methods like delete, update, put ,get.

```
constructor(private http:HttpClient)
```



Hits API

Example for create a get function using http object.

```
getCourse(){  
  return this.http.get("https://localhost:44320/api/course/GetAllCourse",{  
    headers:{  
      Authorization:'Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bm1xdWVfbmFtZSI6  
    }  
  })  
}
```



Hits API

Test in postman

GET

▼

https://localhost:44320/api/course/GetAllCourse

Params

Authorization ●

Headers (9)

Body ●

Pre-request Script

Tests

Settings

Type

Bearer Token ▼

The authorization header will be automatically generated when you send the request.
[Learn more about authorization](#) ↗

! Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. [Learn more about variables](#) ↗

Token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...



Hits API

In home.component.ts

```
getAllCourses(){  
    //call services  
  
    this.homeServices.getCourse().subscribe(result  
=> { console.log(result) },  
    err => { console.log(err) })  
}
```



Hits API

In home.component.ts
because the Oninit function call by default I will call the
function inside it.

```
ngOnInit(): void {  
    this.getAllCourses();  
}
```



Hits API

The process is:

- 1) Show the loader.
- 2) Hits api.
- 3) After response: hide the loader.
- 4) Show toaster “small massage”.



Chapter 7

- 1 RXJS
- 2 Hits API
- 3 **Toastr**



Toastr

To use toaster in you project:

First: Install the toastr package using this command:

```
npm install ngx-toastr
```

Then, Install angular animations:

```
npm install @angular/animations --save
```



Toastr

After that, Import the style in angular.json.

```
"styles": [  
  "node_modules/ngx-toastr/toastr.css"  
],
```



Toastr

Add ToastrModule to app NgModule, make sure you have BrowserModule as well.

```
import {  
    ToastrModule, ToastNoAnimation,  
    ToastNoAnimationModule} from 'ngx-toastr';
```

And in import section

```
ToastNoAnimationModule.forRoot(),  
ToastrModule.forRoot()
```



Toastr

In **home.component.ts** define an object for this services in the constructor.

```
private toast:ToastrService
```

And use this object on OnInit function

```
ngOnInit(): void {  
    this.getAllCourse();  
    this.toast.success("Welcome to first toast ");  
}
```



Toastr

The result is:

