What is RxJs?

* RxJs (Reactive Extension for JavaScript) is a library for reactive programming using Observable, to Handle Asynchronous events
* **Angular RxJs everywhere:**
  + HTTP Request
  + Form Value Changes
  + Routing Events
  + Component Communication

SwitchMap

It will cancel previous requests and only keep the latest request (Ex. LIVE Search in an Application)

Create service:

* ng g s service/service

import { HttpClient } from '@angular/common/http';

import { Injectable } from '@angular/core';

@Injectable({

  providedIn: 'root'

})

export class UserService {

  constructor( private http:HttpClient) { }

  serachUsers(query: String){

    return this.http.get<any[]>(`https://jsonplaceholder.typicode.com/users?name\_like=${query}`)

  }

}

Inject the service in any component where we need to implement search functionality

Lets do it in app.component.html

<nav class="navbar navbar-expand-lg bg-body-tertiary">

  <div class="container-fluid">

    <a class="navbar-brand" href="#">Navbar</a>

    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

      <span class="navbar-toggler-icon"></span>

    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">

      <ul class="navbar-nav me-auto mb-2 mb-lg-0">

        <li class="nav-item">

          <a class="nav-link active" aria-current="page" href="#">Home</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" routerLink="directives" href="#">Directives</a>

        </li>

      </ul>

      <form class="d-flex" role="search">

        <input [formControl]="searchControl" class="form-control me-2" type="search" placeholder="Search" aria-label="Search"/>

        <button class="btn btn-outline-success" type="submit">Search</button>

      </form>

    </div>

  </div>

</nav>

<div class="container">

  <li \*ngFor="let u of users">{{u.name}} </li>

</div>

App.component.ts file

import { Component } from '@angular/core';

import { UserService } from './service/user.service';

import { FormControl } from '@angular/forms';

import { debounce, debounceTime, switchMap } from 'rxjs';

@Component({

  selector: 'app-root',

  templateUrl: './app.component.html',

  styleUrls: ['./app.component.css']

})

export class AppComponent {

  title = 'angular-app';

  searchControl= new FormControl('');

  users:any[]=[];

  constructor(private service:UserService){

    this.searchControl.valueChanges.pipe(

      debounceTime(300),

      switchMap(value=> this.service.serachUsers(value ?? ''))

    ).subscribe(data=>this.users=data)

  }

}

App.module.ts file

imports: [

    ReactiveFormsModule

  ],

Output:

