**Angular Arrays:**

Skill.ts

export class Skill

{

Sid:number;

Sname:string;

}

Employee.ts

import { Skill } from './Skill';

export class Employee

{

id:number;

name:string;

salary:number;

permanent:boolean;

skills:Skill[];

}

**App.component.ts**

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

import { Employee } from './Employee';

import {Skill} from './Skill';

@Component({

selector: 'app-root',

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

// template:`<h2> Welcome to my project </h2> <p>The title is : {{title}} </p> <div>The value of i is :{{i}}</div> <div> The message is {{message}} </div>`,

styleUrls: ['./app.component.css']

})

export class AppComponent {

public employee:Employee

={

id:0,

name:"",

permanent:false,

salary:0,

skills:[{Sid:11,Sname:"Java"},{Sid:12,Sname:"Python"}]

}

public employees:Employee[]

=[{id:1,name:"Rama",permanent:true,salary:43000,skills:[{Sid:11,Sname:"Java"}]},

{id:2,name:"Suresh",permanent:true,salary:43000,skills:[{Sid:12,Sname:"Python"}]},

{id:3,name:"Ganga",permanent:true,salary:43000,skills:[{Sid:13,Sname:"C++"}]},

{id:4,name:"Suma",permanent:true,salary:43000,skills:[{Sid:14,Sname:".NET"}]},

]

constructor()

{

this.employee.id=101;

this.employee.name="Radha";

this.employee.salary=87000;

this.employee.skills=[{Sid:1,Sname:'C++'},{Sid:2,Sname:'DBMS'}]

}

}

**App.component.html**

<!--The content below is only a placeholder and can be replaced.-->

<h2> First Basic Application </h2>

<!-- Interpolation -->

<!-- <p>The title is : {{title}} </p>

<div>The value of i is :{{i}}</div>

<div> The message is {{message}} </div>

Age: <input type=text id="age" name="age" [value]="age" [disabled]="data"/><br/>

Name:<input type=text [(ngModel)]="empname"/> <br/>

your name is {{empname}} -->

<!-- {{employee|json}} -->

<div>{{employee.id}}</div>

<div>{{employee.name}}</div>

<div>{{employee.salary}}</div>

<ul \*ngFor="let sk of employee.skills">

<li>{{sk.Sid}}</li>

<li>{{sk.Sname}}</li>

</ul>

<hr/>

<ul \*ngFor="let emps of employees">

<li>{{emps.id}}</li>

<li>{{emps.name}}</li>

<li>{{emps.salary}}</li>

<li>{{emps.permanent}}</li>

<ul \*ngFor="let s of emps.skills">

<li>{{s.Sid}}</li>

<li>{{s.Sname}}</li>

</ul>

</ul>

<!-- {{emp.name}}

{{emp.salary}}

{{emp.permanent}} -->

**Angular HTTP and Observables:**

EmpDetail

Response

Request

Observable

Get

HTTP

EmpService

EmpList

DB

**Observables:**

Source

House 1

House 3

House 2

Subscribe

Convert

NewsPaper Company

An Observable is a sequence of items that arrives **asynchronously** over a time.

With a HTTP Call sends as a single item which is a HTTP Response.

An observable is a HTTP Response which arrives asynchronously.

This is not a format which we can readily use in our application. So we have to convert into an array so that it can be used in our components. We provide this data to only to the components which have subscribed to our service.

**4 Steps:**

1. Http Get Request from EmpService.
2. Receive the observable and cast it in to Employee Array.
3. Subscribe to the Observable from EmpList and EmpDetails
4. Assign the Employee Array to the Local Variable.

**RxJS:**

RxJS is a library to enable us to work with Observables in Angular Applications. It is Reactive extensions for Javascript. It is an external library to work with observables.

**Error Handling in Angular:**

**Employee.Service.ts**

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

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

import { IEmployee } from './employee';

import {Observable} from 'rxjs/Observable';

import { catchError, map } from 'rxjs/operators';

import {throwError} from 'rxjs';

import 'rxjs/add/operator/catch';

import 'rxjs/add/observable/throw';

@Injectable({

providedIn: 'root'

})

export class EmployeeService {

private \_url:string="/assets/Data/employees1.json";

constructor(private http:HttpClient) { }

getEmployees():Observable<IEmployee[]>

{

return this.http.get<IEmployee[]>(this.\_url)

.pipe(

map(data=>{

return data

}),

catchError(error=>{

return Observable.throw(error.message|| "server Error");

}

//catchError((res: Response)=>this.OnError(res))

));

}

errorHandler(error:HttpErrorResponse)

{

return Observable.throw(error.message|| "server Error");

//return throwError(error);

}

}

**EmpList.Component.ts**

@Component({

selector: 'app-employee-list',

templateUrl: './employee-list.component.html',

styleUrls: ['./employee-list.component.css']

})

export class EmployeeListComponent implements OnInit {

public employees=[];

public errorMsg;

constructor(private \_employeeService:EmployeeService) { }

ngOnInit() {

this.\_employeeService.getEmployees()

.subscribe(data=>this.employees=data,

error=>this.errorMsg=error)

}

}

**Emplist.Component.html**

<p>employee-list works!</p>

{{errorMsg}}

<ul \*ngFor="let emp of employees">

<li>{{emp.name}}</li>

</ul>