**ANGULAR Handson Solutions**

**Date:**01-07-2021

**Name:**Asritha Cherukuri

**EmpId:** 916179

**Angular Handson**

**Session1:**

*Given:*

Component Interaction - Problem Definition Following are the new features that needs to be incorporated in the angular-learning app:

· Include a new screen to display the list of employees

· This screen should also include option to search employees

· For each employee displayed there should be an option to edit

· Clicking on the edit link should lead to the reactive form edit screen

· The edit employee screen should be prepopulated with the respective employee details

*Implementation:*

*App.module.ts:*

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

import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';

import { AppComponent } from './app.component';

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

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

import { EditempComponent } from './editemp/editemp.component';

import { EmplistComponent } from './emplist/emplist.component';

import { EmpinfComponent } from './empinf/empinf.component';

import {Routes,RouterModule} from '@angular/router'

const appRoutes:Routes=[

  {path:"emplist" ,

  component:EmplistComponent}]

@NgModule({

  declarations: [

    AppComponent,

    EditempComponent,

    EmplistComponent,

    EmpinfComponent

  ],

  imports: [

    BrowserModule,

    AppRoutingModule,

    ReactiveFormsModule,

    FormsModule,

    RouterModule.forRoot(appRoutes)

  ],

  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule { }

*emplist.html*

<h3>Employee List</h3>

<div style="margin: 20px;">

    Search &nbsp;

    <input type="text" placeholder="Find by name" [(ngModel)]="searchKey"  (input)="Search()">

</div>

<div class="row">

<div class="container" \*ngFor="let emp of employeeList">

<div class="column" style="padding:5px;">

    <div class="cardblock" style="width: 18rem;border: 2px solid #000000;border-color:#0645AD;padding:5px;">

        <span style="font-size: 30px;">{{emp.name}}</span>

        <span style="float:right ;"> <a style="color: red;" [routerLink]="['/emp-info',emp.id]">emp-info</a></span>

        <br>

        {{emp.salary | currency : "INR"}}<br>

        <a class="editLink" [routerLink]="['/edit-emp-reactive',emp.id]">Edit</a>

    </div>

</div>

</div>

</div>

*Emplist.ts*

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

import { ActivatedRoute } from "@angular/router";

@Component({

  selector: 'app-emplist',

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

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

})

export class EmplistComponent implements OnInit {

  constructor() { }

  employeeList:Employee[]=[];

ngOnInit(){

  this.employeeList = [{id:1, name:"John", salary:5000, permanent:true, department:[{id:3, name:"Payroll"}],skills:[{id:1, name:"HTML"}] },

  {id:2, name:"Smith", salary:10000, permanent:true, department:[{id:6, name:"Internal"}],skills:[{id:2, name:"JAVA"}] },

  {id:3, name:"Mary", salary:55000, permanent:true, department:[{id:2, name:"Lead"}],skills:[{id:1, name:"HTML"}] },

  {id:4, name:"Sandy", salary:15000, permanent:true, department:[{id:3, name:"Payroll"}],skills:[{id:3, name:"CSS"}] },

  {id:5, name:"Martin", salary:20000, permanent:true, department:[{id:2, name:"Lead"}],skills:[{id:1, name:"HTML"}] }

]

}

public searchKey:string="";

Search()

{

  if(this.searchKey != "")

  {

    this.employeeList = this.employeeList.filter(res=>{

      return res.name.toLocaleLowerCase().match(this.searchKey.toLocaleLowerCase());

});

  }

  else if(this.searchKey == "")

  {

    this.ngOnInit();

  }

}

}

 interface Department{

  id:number;

  name:string;

}

 interface Skill{

  id:number;

  name:string;

}

 interface Employee extends Department, Skill{

  id:number;

  name:string;

  salary:number;

  permanent:boolean;

  department:Department[];

  skills:Skill[];

}

*Empinf.html:*

<h2>Employee Details</h2>

<div class="cardblock">

    <br>

    Name : {{empObj.name}}

    <br>

    <br>

    Salary : {{empObj.salary | currency : "INR"}}

    <br>

    <br>

    Permanent : {{empObj.permanent}}

    <br>

    <br>

    <a class="editLink" [routerLink]="['/edit-emp-reactive',empObj.id]">Edit</a>

</div>

*Empinf.ts:*

import { Component, Input, OnInit } from '@angular/core';

import { ActivatedRoute } from '@angular/router';

@Component({

  selector: 'app-empinf',

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

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

})

export class EmpinfComponent implements OnInit {

  ngOnInit(): void {

  }

  empObj :any={};

  @Input() employee = [{id:1, name:"John", salary:5000, permanent:true, department:[{id:3, name:"Payroll"}],skills:[{id:1, name:"HTML"}] },

  {id:2, name:"Smith", salary:10000, permanent:true, department:[{id:6, name:"Internal"}],skills:[{id:2, name:"JAVA"}] },

  {id:3, name:"Mary", salary:55000, permanent:true, department:[{id:2, name:"Lead"}],skills:[{id:1, name:"HTML"}] },

  {id:4, name:"Sandy", salary:15000, permanent:true, department:[{id:3, name:"Payroll"}],skills:[{id:3, name:"CSS"}] },

  {id:5, name:"Martin", salary:20000, permanent:true, department:[{id:2, name:"Lead"}],skills:[{id:1, name:"HTML"}] }

]

  constructor(private route:ActivatedRoute)

  {

      let eno  =  this.route.snapshot.params["id"];

      this.empObj = this.employee.find(  item => item.id == eno );

  }

}

interface Department{

  id:number;

  name:string;

}

 interface Skill{

  id:number;

  name:string;

}

 interface Employee extends Department, Skill{

  id:number;

  name:string;

  salary:number;

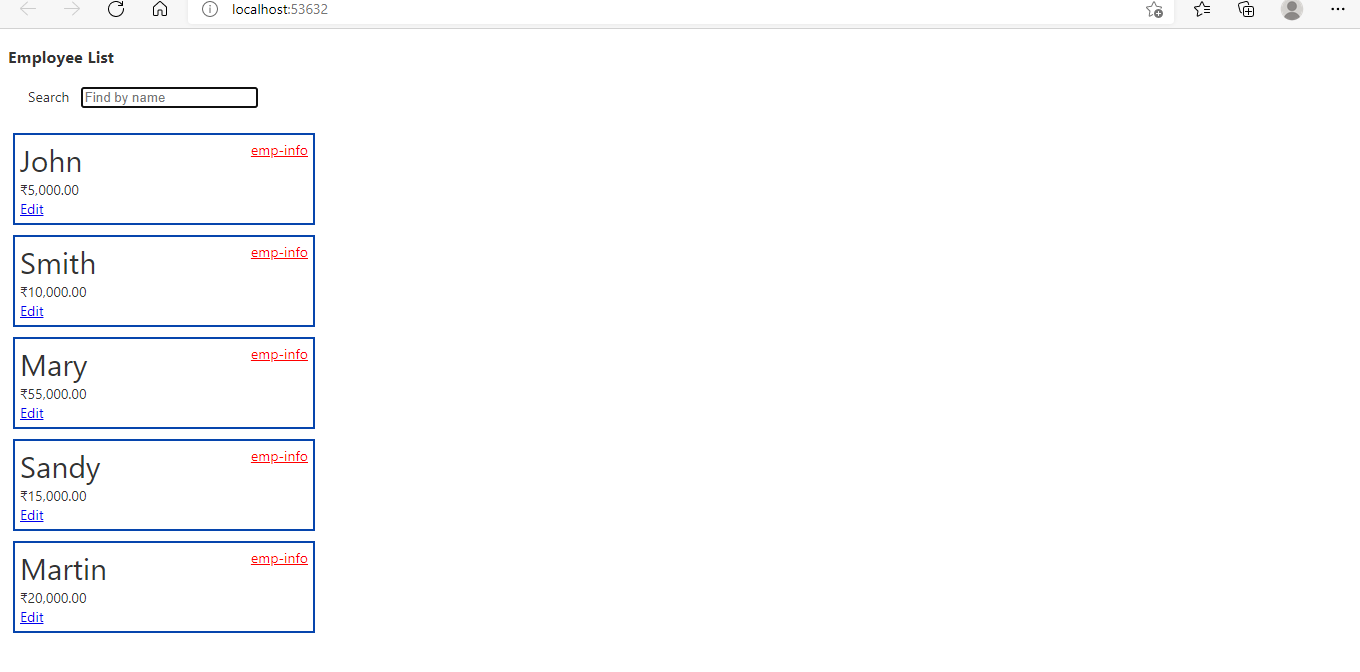
  permanent:boolean;

  department:Department[];

  skills:Skill[];

}

*Output:*

**

*===================================================================* **Session2:**

*Given:*

Service - Create service for handling employee data

*Implementation:*

*Employee.service*

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

@Injectable({

  providedIn: 'root'

})

export class EmployeeService {

  constructor() { }

  employeeList:Employee[]= [{id:1, name:"John", salary:5000, permanent:true, department:[{id:3, name:"Payroll"}],skills:[{id:1, name:"HTML"}] },

  {id:2, name:"Smith", salary:10000, permanent:true, department:[{id:6, name:"Internal"}],skills:[{id:2, name:"JAVA"}] },

  {id:3, name:"Mary", salary:55000, permanent:true, department:[{id:2, name:"Lead"}],skills:[{id:1, name:"HTML"}] },

  {id:4, name:"Sandy", salary:15000, permanent:true, department:[{id:3, name:"Payroll"}],skills:[{id:3, name:"CSS"}] },

  {id:5, name:"Martin", salary:20000, permanent:true, department:[{id:2, name:"Lead"}],skills:[{id:1, name:"HTML"}] }]

  getAllEmployees() : Employee[]{

      return this.employeeList;

    }

  }

  export interface Department{

    id:number;

    name:string;

  }

   interface Skill{

    id:number;

    name:string;

  }

   interface Employee extends Department, Skill{

    id:number;

    name:string;

    salary:number;

    permanent:boolean;

    department:Department[];

    skills:Skill[];

  }

*Auth.service*

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

@Injectable({

  providedIn: 'root'

})

export class AuthService {

  constructor() { }

  loggedIn!: boolean;

  login()

  {

    this.loggedIn=true;

  }

  logout()

  {

    this.loggedIn=false;

  }

  isLoggedIn():boolean

  {

    return this.loggedIn;

  }

}

*Authguard.service*

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

import { ActivatedRouteSnapshot, CanActivate, RouterStateSnapshot, UrlTree } from '@angular/router';

import { Observable } from 'rxjs';

import { AuthService } from './auth.service';

import { Router, ActivatedRoute, ParamMap } from '@angular/router';

@Injectable({

  providedIn: 'root'

})

export class AuthguardService {

  constructor(private authService: AuthService,

    private router: Router) { }

    canActivate(

      next: ActivatedRouteSnapshot,

      state: RouterStateSnapshot): Observable<boolean | UrlTree> | Promise<boolean | UrlTree> | boolean | UrlTree {

        if (this.authService.isLoggedIn()) {

          return true;

           }

           else{

            this.router.navigate(['/login']);

           }

        return false;

    }}