**COMPONENT INTERACTION**

**Question 1**

**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

Go to subsequent hands on problems to gradually build the above requirements.  
  
**NOTE:** No output is expected as part of this hands on and this is only for problem definition.  
  
**Sample Screen Layout for Employees List**

|  |
| --- |
| https://cognizant.e-box.co.in/uploads/Image/01fseangular/ci2.png |

**Question 2**

**Component Interaction - Create Component and Display Employee List**  
  
Implement the below steps for creating a component and display employee list:

* Create a new component with name 'employee-list'
* Import Employee, Department and Skill
* Create property named 'employeeList' in this new component of type array of Employee.
* Set data for array of Employee with five employee details with all necessary details filled.
* In the HTML Template incorporate the following:
  + Page title as "Employee List"
  + Include search text box
  + Using \*ngFor iterate through employeeList and display the employee cards as per the layout specified below
  + The 'Edit' link shall currently point to a dummy link '#'
* Include a new link in header as 'Employees' routing to this new component.

**Sample Layout**

|  |
| --- |
| https://cognizant.e-box.co.in/uploads/Image/01fseangular/ci3.png |

**Answer 1 & 2**

**app.component.html**

<b>Angular Learning</b>&nbsp;&nbsp;&nbsp;<a *routerLink*="view">View Employee</a>

&nbsp;&nbsp;<a *routerLink*="employees-list">Employees</a>

&nbsp;&nbsp;<a *routerLink*="edit">Edit Employee</a>

&nbsp;&nbsp;<a *routerLink*="edit-emp-temp">Edit Employee (Template Driven)</a>

&nbsp;&nbsp;<a *routerLink*="edit-emp-react">Edit Employee (Reactive)</a>

&nbsp;&nbsp;<a *routerLink*="quant-incr">Quantity Selector</a>

&nbsp;&nbsp;<a *routerLink*="quant-sel">Quantity Selector</a>

<hr>

<router-outlet></router-outlet>

**app-routing.module.ts**

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

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

import { QuantityIncrementComponent } from './quantity-increment/quantity-increment.component';

import { QuantitySelectorComponent } from './quantity-selector/quantity-selector.component';

import { EditEmpComponent } from './edit-emp/edit-emp.component';

import { ViewEmpComponent } from './view-emp/view-emp.component';

import { EditEmpTemplateDrivenComponent } from './edit-emp-template-driven/edit-emp-template-driven.component';

import { EditEmpReactiveComponent } from './edit-emp-reactive/edit-emp-reactive.component';

const routes: **Routes** = [

{

path:'view',

component:ViewEmpComponent

},

{

path:'edit',

component:EditEmpComponent

},

{

path:'quant-incr',

component:QuantityIncrementComponent

},

{

path:'quant-sel',

component:QuantitySelectorComponent

},

{

path:'edit-emp-temp',

component:EditEmpTemplateDrivenComponent

},

{

path:'edit-emp-react',

component:EditEmpReactiveComponent

},

{

path:'employees-list',

component:EmployeesComponent

}

];

@**NgModule**({

imports: [RouterModule.**forRoot**(routes)],

exports: [RouterModule]

})

export class **AppRoutingModule** { }

**edit-emp-reactive.component.ts**

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

import { FormGroup, FormControl, Validators } from '@angular/forms';

import { Department } from '../Department';

import { Employee } from '../Employee';

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

@**Component**({

selector: 'app-edit-emp-reactive',

templateUrl: './edit-emp-reactive.component.html',

styleUrls: ['./edit-emp-reactive.component.css']

})

export class **EditEmpReactiveComponent** implements **OnInit** {

employee:**Employee**;

reactiveForm: **FormGroup**;

public employees:**Employee**[]=[{

id:1,

name:'John',

salary:10000,

permanent:true,

skills:null,

department:null,

dateofBirth:null

},

{

id:2,

name:'Dona',

salary:980000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:3,

name:'Jasmine',

salary:90000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:4,

name:'Akash',

salary:80000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

}

];

constructor(){

}

 departments: **Department**[]=[

    {id:1,name:"HelpDesk"},

    {id:2,name:"HR"},

    {id:3,name:"IT"},

    {id:4,name:"Payroll"},

    {id:5,name:"admin"}

  ];

  emp:**Employee**={

    id:null,

    name:null,

    salary:null,

    permanent:null,

    department:{

      id:1,

      name:'Payroll'

    },

    skills:[

      {

        id:1,

        name:'HTML'

      },

      {

        id:2,

        name:'CSS'

      },

      {

        id:3,

        name:'Javascript'

      }

    ],

    dateofBirth: new **Date**('04/20/2019')

}

**getEmp**():void{

}

**ngOnInit**() {

this.**initForm**();

}

get **empName**() {

return this.reactiveForm.**get**('empName');

}

**onSubmit**() {

**console**.**log**("Name is : " + this.reactiveForm.value.empName);

**console**.**log**("Salary is : " + this.reactiveForm.value.empSalary);

**console**.**log**("Permananent or Not : " + this.reactiveForm.value.empPermananent);

**console**.**log**("Department is : " + this.reactiveForm.value.empDept);

}

private **initForm**() {

this.reactiveForm = new **FormGroup**({

'empName': new **FormControl**(this.employee.name,[

Validators.required,

Validators.**minLength**(4),

Validators.**maxLength**(20)

]),

'empSalary':new **FormControl**(this.empl.salary),

'empPermananent':new **FormControl**(this.empl.permanent),

'empDept':new **FormControl**(this.emp.department)

}

);

}

}

**edit-emp-reactive.component.html**

<div *style*="float: left;width: 33%;">

<h1> Edit Employee </h1>

<form *[formGroup]*="reactiveForm" *(ngSubmit)*="onSubmit()">

<div *class*="form-group">

<label *for*="Name">Name : </label>

<br/>

<input *type*="text" *id*="empname" *formControlName*="empName">

<span *class*="help-block" *\*ngIf*="empName.errors?.required && empName.touched">

Name is Required

</span>

<span *class*="help-block" *\*ngIf*="empName.errors?.minlength">

Name Must be at least 2 characters long

</span>

<span *class*="help-block" *\*ngIf*="empName.errors?.maxlength">

Name can have maximum of 10 characters

</span>

</div>

<div *class*="form-group">

<label *for*="salary">Salary</label>

<br/>

<input *type*="number" *name*="salary" *id*="salary" *formControlName*="empSalary"/>

</div>

<div *class*="form-group">

<div>

<label *for*="control-label">Is Permanent or not ?</label>

<br/>

<label *class*="radio-inline">

<input *type*="radio" *name*="empPermananent" *id*="permanent" *value*="true" *formControlName*="empPermananent" /> Yes

</label>

<label *class*="radio-inline">

<input *type*="radio" *name*="empPermananent" *id*="permanent" *val*="false" *formControlName*="empPermananent" /> No

</label>

</div>

</div>

<div *class*="form-group">

<label *for*="department">Department : </label>

<br/>

<select *id*="department" *name*="empDept" *class*="form-group" *formControlName*="empDept" >

<option *\*ngFor*="let dept of departments" *[value]*="dept.id">

{{dept.name}}

</option>

</select>

</div>

<br/>

<div *style*="margin-left:150px;margin-bottom: 20px;">

<button *type*="submit" *class*="btn" *[disabled]*="!reactiveForm.valid">Save</button>

</div>

</form>

</div>

<div *style*="float:left;width: 33%" >

<strong>Employee Model</strong>

<br/>

<pre>{{emp|json}}</pre>

</div>

<div *style*="float: left;width:33%">

<strong>Angular Genrated Form Model</strong>

<br/>

<pre>{{reactiveForm.value|json}}</pre>

</div>

**employees.component.ts**

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

import { Department } from '../Department';

import { Employee } from '../Employee';

import { Skill } from '../Skill';

@**Component**({

selector: 'app-employees',

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

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

})

export class **EmployeesComponent** implements **OnInit** {

**ngOnInit**()

{

}

constructor()

{

}

public employees:**Employee**[]=[{

id:1,

name:'John',

salary:10000,

permanent:true,

skills:null,

department:null,

dateofBirth:null

},

{

id:2,

name:'Dona',

salary:980000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:3,

name:'Jasmine',

salary:90000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:4,

name:'Akash',

salary:80000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

}

];

}

**employees.component.html**

<h2>Employees List</h2>

Search : <input *type*="text" *id*="Search" *name*="Search">

<br/>

<br/>

<div *class*="container">

<div *class*="card-columns" >

<div *class*="card border border-primary" *\*ngFor*="let emp of employees">

<div *class*="card-body text-center">

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

<br/>

{{emp.salary}}

<br/>

<a *routerLink*="#>Edit</a>

</div>

</div>

</div>

</div>

**Question 3**

**Component Interaction - Master / Detail Component**  
  
The employee-list component needs to be split into two components:

* Parent Component will be employee-list - Contains the page heading and search text box.
* Child Component will be employee-info - Contains the view for displaying a single employee

Refer screen layout below that contains details about the component details included in red.

|  |
| --- |
| https://cognizant.e-box.co.in/uploads/Image/01fseangular/ci4.png |

Follow steps below to incorporate above mentioned changes:

* Create a new component employee-info
* Include property 'employee' of type Employee
* Assign static data to employee property
* Define the Edit link in employee-info HTML as specified below

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

* Include this component app selector in employee-list component HTML. This will display a single employee in employee-list component.
* Modify the app selector in previous step to be included into \*ngFor and pass the employee reference as input to employee-info component
* Include @Input decorator in employee-info component for the employee property

​​​​​​​**Answer 3**

**app-routing.module.ts**

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

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

import { QuantityIncrementComponent } from './quantity-increment/quantity-increment.component';

import { QuantitySelectorComponent } from './quantity-selector/quantity-selector.component';

import { EditEmpComponent } from './edit-emp/edit-emp.component';

import { ViewEmpComponent } from './view-emp/view-emp.component';

import { EditEmpTemplateDrivenComponent } from './edit-emp-template-driven/edit-emp-template-driven.component';

import { EditEmpReactiveComponent } from './edit-emp-reactive/edit-emp-reactive.component';

const routes: **Routes** = [

{

path:'view',

component:ViewEmpComponent

},

{

path:'edit',

component:EditEmpComponent

},

{

path:'quant-incr',

component:QuantityIncrementComponent

},

{

path:'quant-sel',

component:QuantitySelectorComponent

},

{

path:'edit-emp-temp',

component:EditEmpTemplateDrivenComponent

},

{

path:'edit-emp-react',

component:EditEmpReactiveComponent

},

{

path:'employees-list',

component:EmployeesComponent

},

{

path:'employees-info',

component: **EmployeesInfoComponent**

}

];

@**NgModule**({

imports: [RouterModule.**forRoot**(routes)],

exports: [RouterModule]

})

export class **AppRoutingModule** { }

**employees.component.ts**

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

import { Department } from '../Department';

import { Employee } from '../Employee';

import { Skill } from '../Skill';

@**Component**({

selector: 'app-employees',

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

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

})

export class **EmployeesComponent** implements **OnInit** {

public employees:**Employee**[]=[{

id:1,

name:'John',

salary:10000,

permanent:true,

skills:null,

department:null,

dateofBirth:null

},

{

id:2,

name:'Dona',

salary:980000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:3,

name:'Jasmine',

salary:90000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:4,

name:'Akash',

salary:80000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

}

];

searchKey:string;

**ngOnInit**()

{

}

constructor()

{

}

}

**employees.component.html**

<h2>Employees List</h2>

Search : <input *type*="text" *id*="Search" *name*="Search" >

<br/>

<br/>

<div *class*="panel panel-primary" *\*ngFor*="let employee of employees">

<app-employees-info *[employee]*='employee'> </app-employees-info>

</div>

**employees-info.component.ts**

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

import { Employee } from 'src/app/Employee';

@**Component**({

selector: 'app-employees-info',

templateUrl: './employees-info.component.html',

styleUrls: ['./employees-info.component.css']

})

export class **EmployeesInfoComponent** implements **OnInit** {

constructor() { }

**ngOnInit**() {

}

@**Input**() employee: **Employee**;

}

**employees-info.component.html**

<div *class*="container">

<div *class*="card-columns" >

<div *class*="card border border-primary" >

<div *class*="card-body text-center">

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

<br/>

{{employee.salary}}

<br/>

<a *routerLink*="../edit-emp-react/{{employee.id}}" >Edit</a>

</div>

</div>

</div>

</div>

**Question 4**

**Component Interaction - Implement Search**  
  
Keying in a name in the search text box should modify the list of cards immediately.  
  
Implement this search feature based on the steps below:

* Create a property named searchKey and incorporate two way binding with search text box
* Incorporate keyup event in the search text box point to a function named search()
* Implement search() method that will filter using the filter() method of employeeList array with case insensitive comparision of the searchKey and name in the array.
* The result of the filter() method should be assigned to a component property named 'filteredEmployees'
* The \*ngFor in HTML template should be using filteredEmployees instead of employeeList

**Sample Layout**

|  |
| --- |
| https://cognizant.e-box.co.in/uploads/Image/01fseangular/ci5.PNG |

**Answer 4**

*Rest all same as previous*

**employees.component.ts**

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

import { Department } from '../Department';

import { Employee } from '../Employee';

import { Skill } from '../Skill';

@**Component**({

selector: 'app-employees',

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

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

})

export class **EmployeesComponent** implements **OnInit** {

public filteredEmployees:**Employee**[];

public employees:**Employee**[]=[{

id:1,

name:'John',

salary:10000,

permanent:true,

skills:null,

department:null,

dateofBirth:null

},

{

id:2,

name:'Dona',

salary:980000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:3,

name:'Jasmine',

salary:90000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:4,

name:'Akash',

salary:80000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

}

];

searchKey:string=null;

**ngOnInit**()

{

this.filteredEmployees=this.employees;

}

constructor()

{

}

**search**():void{

this.filteredEmployees= this.employees.**filter**(x=>x.name.**toLowerCase**().**indexOf**(this.searchKey.**toLowerCase**())!==-1);

}

}

**employees.component.html**

<h2>Employees List</h2>

Search : <input *type*="text" *id*="Search" *name*="Search" *(keyup)*="search()" *[(ngModel)]*=searchKey>

<br/>

<br/>

<div *class*="panel panel-primary" *\*ngFor*="let employee of filteredEmployees">

<app-employees-info *[employee]*='employee'> </app-employees-info>

</div>

**Question 5**

**Component Interaction - Link Employee List with Edit Employee**  
  
The edit link in Employee List should open Edit Employee component with respective employee details.  
  
Follow steps below to implement this:

* In app-routing.module.ts modify edit-emp-reactive path to have id parameter for representing employee id. Refer code below.

{ path: 'edit-emp-reactive/:id', component: EditEmpReactiveComponent }

* In employee-info.component.html modify edit link with router link. Refer code below.

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

* In edit-emp-reactive.component.ts incorporate the following changes
  + Import ActivatedRoute

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

* Initiliaze the route using constructor. Refer code below.

constructor(private route : ActivatedRoute) { }

* Get the employee id parameter from the router link. Include the below code in ngOnInit() method.

const employeeId = this.route.snapshot.paramMap.get('id');

* Log the employeeId using console.log() to verify if we have got the employeeId appropriately.

**IMPORTANT NOTE**  
Now we need to retrieve the specific employee from the employee list based on employeeId, so that we can populate edit employee form. Again we are supposed to copy the hardcoded employee array from employee-list component. Instead we will use Service feature available in angular to achieve this. Using services we can use this data cross components. The following practice problems will help us implement Services after which we will complete the implementation of Edit Employee component.

**Answer 5**

**app.component.html**

<b>Angular Learning</b>&nbsp;&nbsp;&nbsp;<a *routerLink*="view">View Employee</a>

&nbsp;&nbsp;<a *routerLink*="employees-list">Employees</a>

&nbsp;&nbsp;<a *routerLink*="edit">Edit Employee</a>

&nbsp;&nbsp;<a *routerLink*="edit-emp-temp">Edit Employee (Template Driven)</a>

&nbsp;&nbsp;<a *routerLink*="edit-emp-react">Edit Employee (Reactive)</a>

&nbsp;&nbsp;<a *routerLink*="quant-incr">Quantity Selector</a>

&nbsp;&nbsp;<a *routerLink*="quant-sel">Quantity Selector</a>

<hr>

<router-outlet></router-outlet>

**app-routing.module.ts**

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

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

import { QuantityIncrementComponent } from './quantity-increment/quantity-increment.component';

import { QuantitySelectorComponent } from './quantity-selector/quantity-selector.component';

import { EditEmpComponent } from './edit-emp/edit-emp.component';

import { ViewEmpComponent } from './view-emp/view-emp.component';

import { EditEmpTemplateDrivenComponent } from './edit-emp-template-driven/edit-emp-template-driven.component';

import { EditEmpReactiveComponent } from './edit-emp-reactive/edit-emp-reactive.component';

const routes: **Routes** = [

{

path:'view',

component:ViewEmpComponent

},

{

path:'edit',

component:EditEmpComponent

},

{

path:'quant-incr',

component:QuantityIncrementComponent

},

{

path:'quant-sel',

component:QuantitySelectorComponent

},

{

path:'edit-emp-temp',

component:EditEmpTemplateDrivenComponent

},

{

path:'edit-emp-react\:id',

component:EditEmpReactiveComponent

},

{

path:'employees-list',

component:EmployeesComponent

}

];

@**NgModule**({

imports: [RouterModule.**forRoot**(routes)],

exports: [RouterModule]

})

export class **AppRoutingModule** { }

**edit-emp-reactive.component.ts**

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

import { FormGroup, FormControl, Validators } from '@angular/forms';

import { Department } from '../Department';

import { Employee } from '../Employee';

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

@**Component**({

selector: 'app-edit-emp-reactive',

templateUrl: './edit-emp-reactive.component.html',

styleUrls: ['./edit-emp-reactive.component.css']

})

export class **EditEmpReactiveComponent** implements **OnInit** {

employee:**Employee**;

reactiveForm: **FormGroup**;

public employees:**Employee**[]=[{

id:1,

name:'John',

salary:10000,

permanent:true,

skills:null,

department:null,

dateofBirth:null

},

{

id:2,

name:'Dona',

salary:980000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:3,

name:'Jasmine',

salary:90000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:4,

name:'Akash',

salary:80000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

}

];

constructor(private route:**ActivatedRoute**) {

}

 departments: **Department**[]=[

    {id:1,name:"HelpDesk"},

    {id:2,name:"HR"},

    {id:3,name:"IT"},

    {id:4,name:"Payroll"},

    {id:5,name:"admin"}

  ];

  emp:**Employee**={

    id:null,

    name:null,

    salary:null,

    permanent:null,

    department:{

      id:1,

      name:'Payroll'

    },

    skills:[

      {

        id:1,

        name:'HTML'

      },

      {

        id:2,

        name:'CSS'

      },

      {

        id:3,

        name:'Javascript'

      }

    ],

    dateofBirth: new **Date**('04/20/2019')

}

**getEmp**():void{

}

**ngOnInit**() {

this.**initForm**();

}

get **empName**() {

return this.reactiveForm.**get**('empName');

}

**onSubmit**() {

**console**.**log**("Name is : " + this.reactiveForm.value.empName);

**console**.**log**("Salary is : " + this.reactiveForm.value.empSalary);

**console**.**log**("Permananent or Not : " + this.reactiveForm.value.empPermananent);

**console**.**log**("Department is : " + this.reactiveForm.value.empDept);

}

private **initForm**() {

const eid = +this.route.snapshot.paramMap.**get**('id');

this.employee=this.employees.**find**(e=>e.id==eid);

this.reactiveForm = new **FormGroup**({

'empName': new **FormControl**(this.employee.name,[

Validators.required,

Validators.**minLength**(4),

Validators.**maxLength**(20)

]),

'empSalary':new **FormControl**(this.employee.salary),

'empPermananent':new **FormControl**(this.employee.permanent),

'empDept':new **FormControl**(this.employee.department)

}

);

}

}

**employees.component.ts**

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

import { Department } from '../Department';

import { Employee } from '../Employee';

import { Skill } from '../Skill';

@**Component**({

selector: 'app-employees',

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

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

})

export class **EmployeesComponent** implements **OnInit** {

public filteredEmployees:**Employee**[];

public employees:**Employee**[]=[{

id:1,

name:'John',

salary:10000,

permanent:true,

skills:null,

department:null,

dateofBirth:null

},

{

id:2,

name:'Dona',

salary:980000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:3,

name:'Jasmine',

salary:90000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

},

{

id:4,

name:'Akash',

salary:80000,

permanent:true,

dateofBirth:null,

skills:null,

department:null

}

];

searchKey:string=null;

**ngOnInit**()

{

this.filteredEmployees=this.employees;

}

constructor()

{

}

**search**():void{

this.filteredEmployees= this.employees.**filter**(x=>x.name.**toLowerCase**().**indexOf**(this.searchKey.**toLowerCase**())!==-1);

}

}

**employees.component.html**

<h2>Employees List</h2>

Search : <input *type*="text" *id*="Search" *name*="Search" *(keyup)*="search()" *[(ngModel)]*=searchKey>

<br/>

<br/>

<div *class*="panel panel-primary" *\*ngFor*="let employee of filteredEmployees">

<app-employees-info *[employee]*='employee'> </app-employees-info>

</div>