



If you do not pay any relevant course fee to maintain Earth University BIT platform,
Be ethical enough to contribute while you are using our Academic Contents

www.earth.lk/community
2004-2024 (20th Anniversary)

Sprint-2 Plan

Employee Detail Management

1. Establishing Project Folder Architecture
2. Server App Initialization
3. **Sprint-2 Execution** [2(a), 2(b), **2(c)(i)(iii)**(iii), 2(d), 2(e), 2(f)]
4. Completing Sprint-1 Objectives

2(a) Employee Module - Analysis, Design & DB-Preparation

2(b) Employee View

2(c) Employee Search

Design Search Criteria
Server App Modifications
Client App Modifications

2(d) Employee Insert

2(e) Employee Update

2(f) Employee Delete

2(c) Employee Search

1. Design Search Criteria
Define Search Attributes/Lists Required for Search Panels/SQL Queries Needed
2. Server-App Modification
Implement Listing Services
Implement Searching Service
 - (1) Using Controller Filtering
 - (2) Using QBE/ Named Queries/ Native Queries
3. Client-App Modification
 1. Implement Service Calls and Test
 2. Create Search Form and Observe auto binding Combo Boxes
 3. Implement Search Functionality
 4. Implement Search-Clear Functionality
 5. Create Confirm Dialog and integrate with Search Clear Confirmation

1. Content of the “5-Delete” of the “Client-App” folder at the End of Sprint-2 “View” Completion (Make sure you must start from here)

Projects > Earth-University-College > Client-App > Sprint-2 >

Name

1-View

2-Search

3-Insert

4-Update

5-Delete

Projects > Earth-University-College > Client-App > Sprint-2 > 5-Delete >

Name

.angular

.git

.idea

node_modules

src

.editorconfig

.gitignore

angular.json

package.json

package-lock.json

README.md

tsconfig.app.json

tsconfig.json

tsconfig.spec.json

Projects > Earth-University-College > Client-App > Sprint-2 > 5-Delete > src >

Name

app

assets

favicon.ico

index.html

main.ts

styles.css

> Projects > Earth-University-College > Client-App > Sprint-2 > 5-Delete > src > app >

Name

entity

service

util

view

app.component.css

app.component.html

app.component.spec.ts

app.component.ts

app.module.ts

app-routing.module.ts

Projects > Earth-University-College > Client-App > Sprint-2 > 5-Delete > src > app > view >

Name

configuration

dashboard

home

login

mainwindow

modules

Client-App > Sprint-2 > 5-Delete > src > app > view > modules >

Name

course

employee

user

Client-App > Sprint-2 > 5-Delete > src > app > view > modules > employee

Name

employee.component.css

employee.component.html

employee.component.ts

Client-App > Sprint-2 > 5-Delete > src > app > service

Name

AuthenticateService.ts

employeeservice.ts

Client-App > Sprint-2 > 5-Delete > src > app > entity

Name

designation.ts

employee.ts

employeeestatus.ts

gender.ts

Client-App > Sprint-2 > 5-Delete > src > app > util > ui

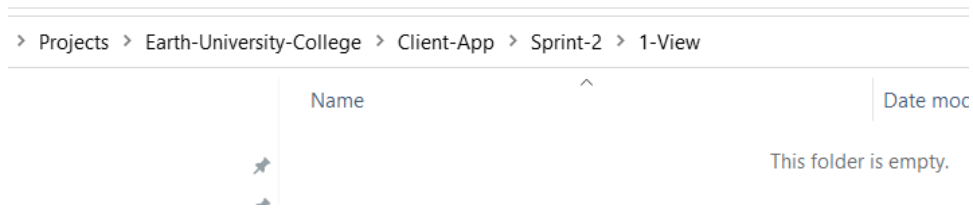
Name

ui.assist.ts

2. Copy the Content of “5-Delete” into “1-View” as Backup for Restore Point (View)

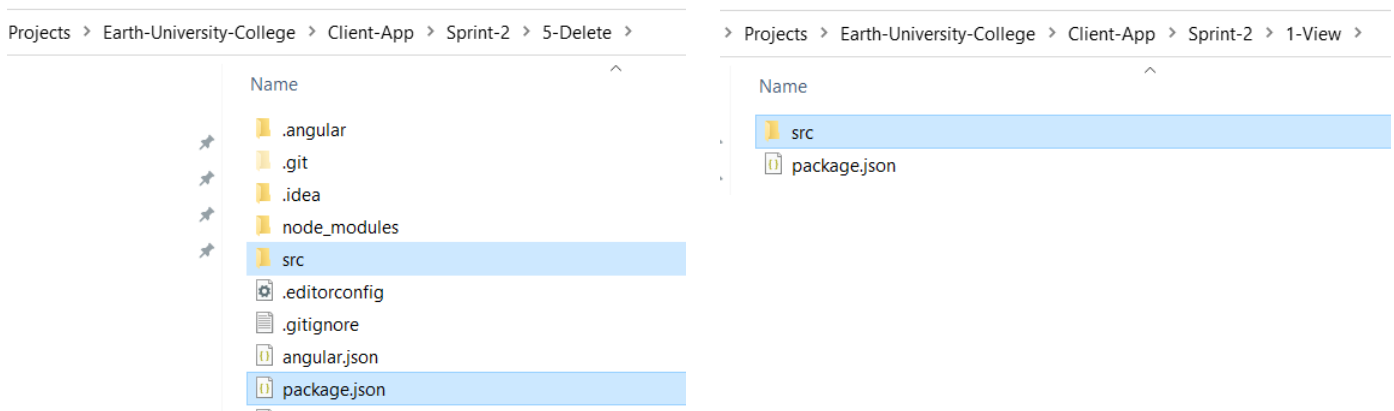
Here you do not need to copy “node-modules” as it contains lot of library files and difficult to copy. Needed dependencies are available in “package.json”

We will continue the existing project in “5-Delete” folder.



Copy the “src” folder with “package.json”

Paste as a backup



We will continue the existing project in “5-Delete” folder.

2(c) Employee Search (i) Server (ii) Client

1. Design Search Criteria

Define Search Attributes/Lists Required for Search Panels/~~SQL Queries Needed~~

2. Server-App Modification

Implement Listing Services

Implement Searching Service

(3) Using Controller Filtering

~~(4) Using QBE/ Named Queries/ Native Queries~~

3. Client-App Modification

1. **Implement Service Calls and Test**

2. **Create Search Form and Observe for auto binding Combo Boxes**

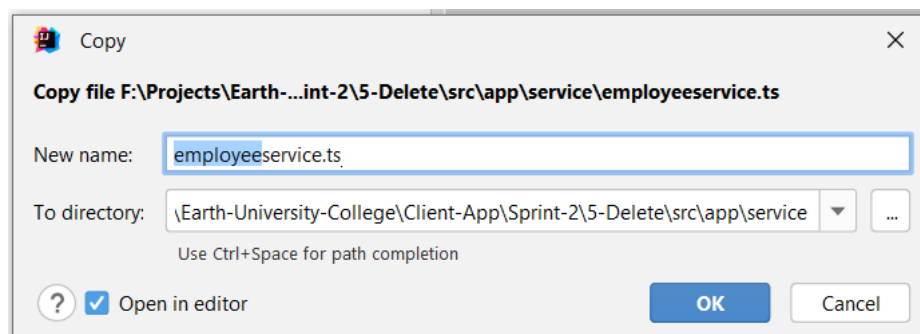
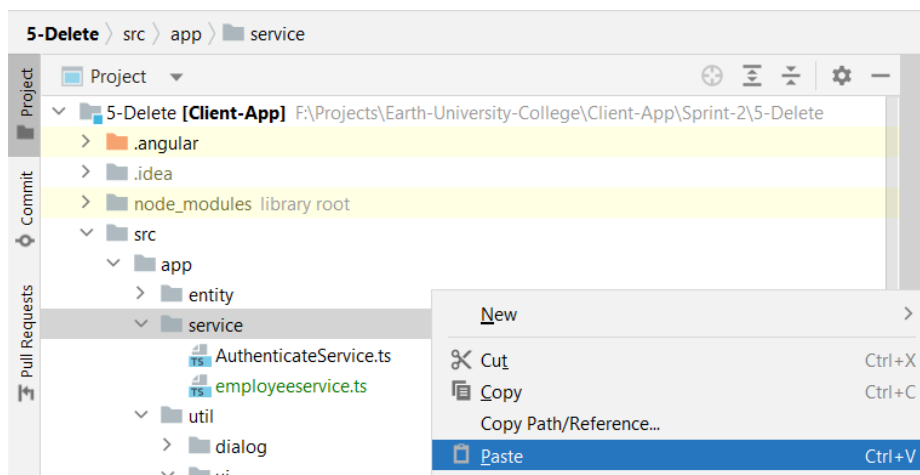
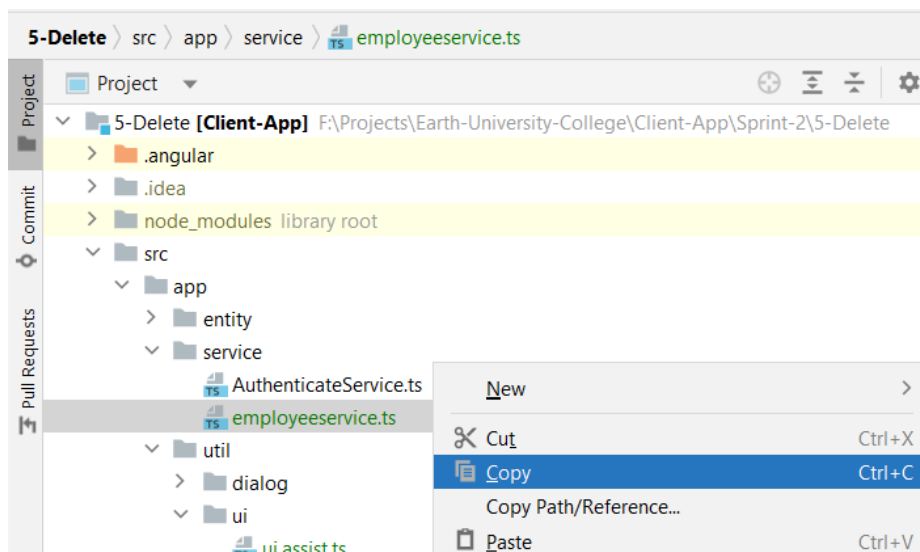
3. **Implement Search Functionality**

4. **Implement Search-Clear Functionality**

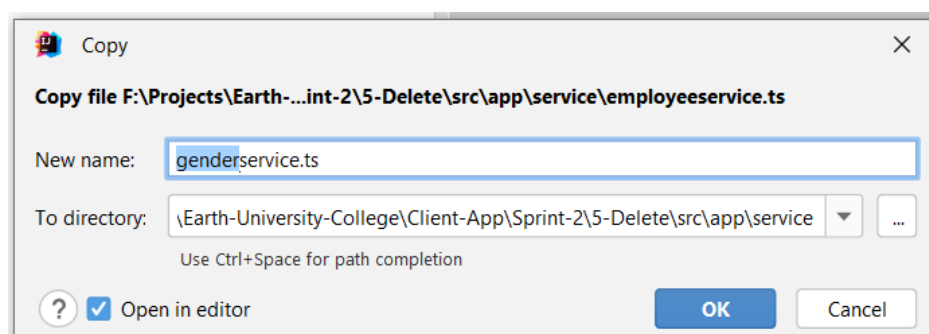
5. **Create Confirm Dialog and integrate with Search Clear Confirmation**

1. Implement Service Calles and Test

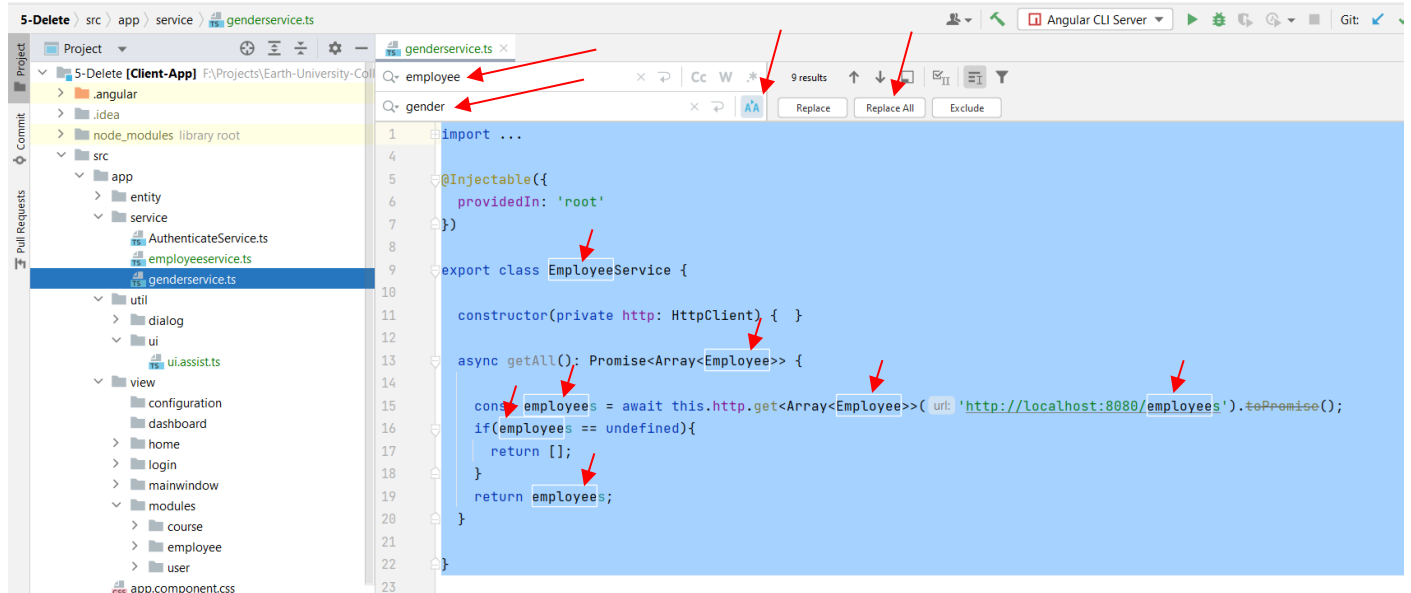
(a) Create 2 Services “GenderService” and “DesignationService”



Change “employeeservice” into “genderservice”



Ctrl + R (Find & Replace)

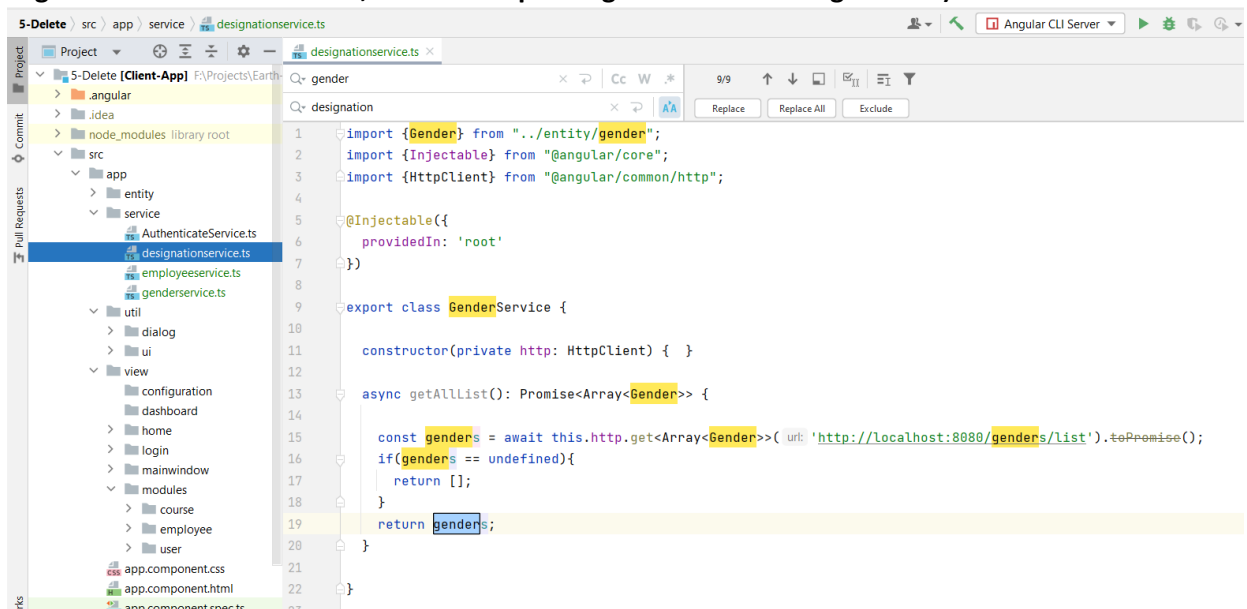


Change the “getAll()” into “getAllList()”



Repeat the Process to create Designation Service

(Copy the genderservice.ts and Paste, Find and Replace “gender” with “designation”)



Testing Service Calls

Create 2 Arrays

Instruct the Angular Framework (Routing Module who Call the Constructor) to Inject 2 Services(gs,ds).

```
5-Delete > src > app > view > modules > employee > employee.component.ts EmployeeComponent

Project
  5-Delete [Client-App] F:\Projects\Earth-University-Co
    > .angular
    > .idea
    > node_modules library root
    > src
      > app
        > entity
        > service
        > util
        > view
          > configuration
          > dashboard
          > home
          > login
          > mainwindow
          > modules
            > course
            > employee
              > employee.component.css
              > employee.component.html
              > employee.component.ts
            > user
              > app.component.css
              > app.component.html
              > app.component.spec.ts
              > app.component.ts
              > app.module.ts
              > app-routing.module.ts
          > assets
            > favicon.ico
            > index.html
            > main.ts
            > styles.css

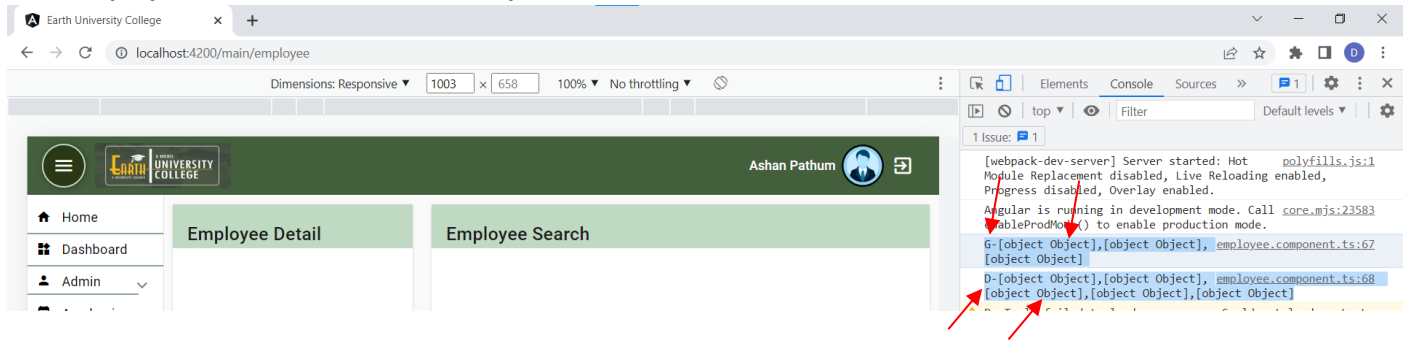
Commit
Pull Requests
Bookmarks
Structure

19
20 export class EmployeeComponent {
21
22   columns: string[] = ['number', 'callingname', 'gender'];
23   headers: string[] = ['Number', 'Calling Name', 'Gender'];
24   binders: string[] = ['number', 'callingname', 'gender'];
25
26   cscolumns: string[] = ['csnumber', 'cscallingname', 'csgender'];
27   csprompts: string[] = ['Search by Number', 'Search by Calling Name', 'Search by Gender'];
28
29   public csearch!: FormGroup;
30
31   employees: Array<Employee> = [];
32   data!: MatTableDataSource<Employee>;
33   imageurl: string = '';
34   @ViewChild(MatPaginator) paginator!: MatPaginator;
35
36   genders: Array<Gender> = [];
37   designations: Array<Designation> = [];
38
39   uiassist: UiAssist;
40
41   constructor(
42     private es: EmployeeService,
43     private gs: GenderService,
44     private ds: DesignationService,
45     private fb: FormBuilder) {
46
47     this.uiassist = new UiAssist(this);
48   }
```

Request Data using “getAllList()” methods of each injected Service(gs,ds) within the “initialize()” method
Console.log() outputs to test Data Arrival. Use “then()” or “finally()” of the Promise returned by each service

```
employee.component.ts x
57
58 }
59
60 ngOnInit() {
61   this.initialize();
62 }
63
64 initialize() {
65   this.createView();
66
67   this.gs.getAllList().then( (gens : Gender[]) => { this.genders = gens; console.log("G-"+this.genders); } );
68   this.ds.getAllList().then( (dess : Designation[]) => { this.designations = dess; console.log("D-"+this.designations); } );
69
70   this.createSearch();
71   this.createForm();
72 }
```

Run the project and observe Console Output



localhost:8080/genders/list

```
[{"id":1,"name":"Male"}, {"id":2,"name":"Female"}, {"id":3,"name":"Other"}]
```

← 3 objects

localhost:8080/designations/list

```
[{"id":1,"name":"Demonstrator L1"}, {"id":2,"name":"Assistent G1"}, {"id":3,"name":"Assistent G2"}, {"id":4,"name":"Demonstrator L2"}, {"id":5,"name":"Demonstrator L3"}]
```

← 5 Objects

2. Implement Form Controls needed for Search Form

- (i) Form for Client Search (csearch) ← C-Search
- (ii) **Form for Server Search (ssearch) ← S-Search**
- (iii) Form for CRUD Operation (crud) ← Main Form used to (Add/Update/Delete/View Details)

```
employee.component.ts
24 binders: string[] = ['number', 'callingname', 'gender.name']
25
26 cscolumns: string[] = ['csnumber', 'cscallingname', 'csgender']
27 csprompts: string[] = ['Search by Number', 'Search by Name', 'Search by Designation']
28
29 public csearch!: FormGroup;
30 public ssearch!: FormGroup;
31
32 employees: Array<Employee> = [];
```

```
employee.component.ts
42 constructor(
43   private es: EmployeeService,
44   private gs: GenderService,
45   private ds: DesignationService,
46   private fb: FormBuilder) {
47
48   this.uiassist = new UiAssist(this);
49
50   this.csearch = this.fb.group( controls: {
51     "csnumber": new FormControl(),
52     "cscallingname": new FormControl(),
53     "csgender": new FormControl(),
54     "csdesignation": new FormControl(),
55     "csname": new FormControl(),
56     "csmodi": new FormControl(),
57   });
58
59   this.ssearch = this.fb.group( controls: {
60     "ssnumber": new FormControl(),
61     "ssfullname": new FormControl(),
62     "ssgender": new FormControl(),
63     "ssdesignation": new FormControl(),
64     "ssnic": new FormControl()
65   });
66
67 }
```

3. Implement Search Form (Data Lists will be automatically bind by the Angular framework)

Employee Search

Gender

Designation

Search

Clear Search

Gender

Male

Not Selected

Male

Female

Other

Designation

Not Selected

Demonstrator L1

Assistant G1

Assistant G2

Demonstrator L2

Demonstrator L3

```
5-Delete src > app > view > modules > employee > employee.component.html
Project 5-Delete [Client-App] F:\Projects\Earth-University-Colle
> .angular
> .idea
> node_modules library root
src
  app
    entity
    service
    util
    dialog
    ui
      ui.assist.ts
  view
    configuration
    dashboard
    home
    login
    mainwindow
    modules
      course
      employee
        employee.component.css
        employee.component.html
        employee.component.ts
      user
    app.component.css
    app.component.html
```

```
1 <mat-grid-list [cols]="12" [rowHeight]="4rem">
2
3 <mat-grid-tile [colspan]="4" [rowspan]="10">
4   <mat-card>
5     <mat-card-header>
6       <mat-card-title>Employee Detail</mat-card-title>
7     </mat-card-header>
8     <mat-card-content> </mat-card-content>
9   </mat-card>
10 </mat-grid-tile>
11
12 <mat-grid-tile [colspan]="8" [rowspan]="4">
13   <mat-card>
14     <mat-card-header>
15       <mat-card-title>Employee Search</mat-card-title>
16     </mat-card-header>
17     <mat-card-content> </mat-card-content>
18   </mat-card>
19 </mat-grid-tile>
20
21 <mat-grid-tile [colspan]="8" [rowspan]="6">
22   <mat-card>
23     <mat-card-header>
24       <mat-card-title>Employee Table</mat-card-title>
```

Need to add "MatSelectModule" into the imports of the "app.module.ts" file

```
app.module.ts
44 imports: [
45   MatSelectModule,
46   MatTableModule,
47   MatPaginatorModule,
```



```

12
13 <mat-grid-tile [colspan]="8" [rowspan]="4">
14   <mat-card id="ssearch">
15     <mat-card-header>
16       <mat-card-title>Employee Search</mat-card-title>
17     </mat-card-header>
18
19     <mat-card-content>
20
21       <mat-form [formGroup]="ssearch">
22
23         <mat-form-field appearance="outline">
24           <mat-label>Search by Number</mat-label>
25           <input matInput formControlName="ssnumber">
26         </mat-form-field>
27
28         <mat-form-field appearance="outline">
29           <mat-label>Search by Fullname</mat-label>
30           <input matInput formControlName="ssfullname">
31         </mat-form-field>
32
33         <mat-form-field appearance="outline">
34           <mat-label>Search by NIC</mat-label>
35           <input matInput formControlName="ssnic">
36         </mat-form-field>
37
38         <mat-form-field appearance="outline">
39           <mat-label>Gender</mat-label>
40           <mat-select formControlName="ssgender">
41             <mat-option [value]="null" selected>Not Selected</mat-option>
42             <mat-option *ngFor="let gender of genders" [value]="gender.id">{{gender.name}}</mat-option>
43           </mat-select>
44         </mat-form-field>
45
46         <mat-form-field appearance="outline">
47           <mat-label>Designation</mat-label>
48           <mat-select formControlName="ssdesignation">
49             <mat-option [value]="null" selected>Not Selected</mat-option>
50             <mat-option *ngFor="let designation of designations" [value]="designation.id">{{designation.name}}</mat-option>
51           </mat-select>
52         </mat-form-field>
53
54         <div>
55           <button mat-raised-button color="primary"><mat-icon>search</mat-icon> Search </button>
56           <button mat-raised-button color="primary"><mat-icon>reorder</mat-icon> Clear Search </button>
57         </div>
58
59       </mat-form>
60
61     </mat-card-content>
62
63   </mat-card>
64 </mat-grid-tile>

```

CSS Modifications

```

1  employee.component.css
2  mat-card { width: 100%; height: 100%; margin: 10px 10px -10px 10px ; }
3
4  mat-card-content { overflow: auto; height: 100%; }
5
6  mat-card-header { background: #c0d9c3; }
7
8  #loading img{ width: 30px; height: 25px; margin-left: 200px;}
9
10 .csearch{ height: 40px; width: 100%; margin-top: 5px; }
11
12 #ssearch mat-card-content{ padding-top: 30px; }
13
14 #ssearch mat-form-field { padding-left: 15px; width: 200px; }
15
16 #ssearch button{ width: 195px; margin-left: 10px; margin-top: 5px; background-color: #435f3b; }

```

4. Event Handling with Search Button

- (a) Declaring the Function and show values from the Search Form
- (b) Registering the Function with the Button and Test the Query String as a Console output
- (c) Updating data from the Server via Service Call of the Employee Service

(a) Declaring the Function and show values from the Search Form

```
employee.component.ts
138
139 btnSearchMc(): void {
140
141     const sserchdata = this.ssearch.getRawValue();
142
143     let number = sserchdata.ssnumber;
144     let fullname = sserchdata.ssfullname;
145     let nic = sserchdata.ssnic;
146     let genderid = sserchdata.ssgender;
147     let designationid = sserchdata.ssdesignation;
148
149     let query="";
150
151     if(number!=null && number.trim()!="") query=query+"&number="+number;
152     if(fullname!=null && fullname.trim()!="") query=query+"&fullname="+fullname;
153     if(nic!=null && nic.trim()!="") query=query+"&nic="+nic;
154     if(genderid!=null) query=query+"&genderid="+genderid;
155     if(designationid!=null) query=query+"&designationid="+designationid;
156
157     if(query!="") query = query.replace( searchValue: /^.\/, replaceValue: "?")
158
159     console.log(query);
160
161
162 }
```

(b) Registering the Function with the Button and Test the Query String as a Console output

```
employee.component.html
51
52 <div>
53     <button mat-raised-button color="primary" (click)="btnSearchMc()"><mat-icon>search</mat-icon> Search </button>
54     <button mat-raised-button color="primary"><mat-icon>reorder</mat-icon> Clear Search </button>
55 </div>
```

The image shows a series of five overlapping screenshots of an 'Employee Search' form, illustrating the step-by-step construction of a search query. Red arrows connect the form fields to the corresponding query parameters in the browser's console log.

Form Fields and Query Parameters:

- Search by Number:** 2201 → `?number=2201`
- Search by Fullname:** asha → `?number=2201&fullname=asha`
- Search by NIC:** 2001 → `?number=2201&fullname=asha&nic=2001`
- Gender:** Male → `?number=2201&fullname=asha&nic=2001&genderid=1`
- Designation:** Demonstrator L2 → `?number=2201&fullname=asha&nic=2001&genderid=1&designationid=4`

Browser Console Log:

Query	File
<code>?number=2201</code>	<code>employee.component.ts:152</code>
<code>?number=2201&fullname=asha</code>	<code>employee.component.ts:152</code>
<code>?number=2201&fullname=asha&nic=2001</code>	<code>employee.component.ts:152</code>
<code>?number=2201&fullname=asha&nic=2001&genderid=1</code>	<code>employee.component.ts:152</code>
<code>?number=2201&fullname=asha&nic=2001&genderid=1&designationid=4</code>	<code>employee.component.ts:152</code>

(c) Updating data from the Server via Service Call of the Employee Service

(i) Modify Employee Service

```

1 import {Employee} from "../entity/employee";
2 import {Injectable} from "@angular/core";
3 import {HttpClient} from "@angular/common/http";
4
5 @Injectable({
6   providedIn: 'root'
7 })
8
9 export class EmployeeService {
10
11   constructor(private http: HttpClient) { }
12
13   async getALL(): Promise<Array<Employee>> {
14
15     const employees = await this.http.get<Array<Employee>>(url: 'http://localhost:8080/employees').toPromise();
16     if(employees == undefined){
17       return [];
18     }
19     return employees;
20   }
21 }

```

```

async getALL(query:string): Promise<Array<Employee>> {
  const employees = await this.http.get<Array<Employee>>(url: 'http://localhost:8080/employees'+query).toPromise();
  if(employees == undefined){
    return [];
  }
  return employees;
}

```


(ii) Modify loadTable(), createView() functions to reflect the changes in “getALL()” in employeeservice

```

82
83 createView() {
84   this.imageUrl = 'assets/pending.gif';
85   this.loadTable( query: "");
86 }
87
88 createSearch(){ }
89
90 createForm(){ }
91
92 loadForm(){ }
93
94
95 loadTable(query:string) {
96
97   this.es.getALL(query) Promise<Employee[]>
98   .then((emps: Employee[]) => {
99     this.employees = emps;
100    this.imageUrl = 'assets/fullfilled.png';
101   }) Promise<void>
102   .catch((error) => {
103     console.log(error);
104     this.imageUrl = 'assets/rejected.png';
105   }) Promise<void>
106   .finally( onfinally: () => {
107     this.data = new MatTableDataSource(this.employees);
108     this.data.paginator = this.paginator;
109   });
110 }
111 }

```

Re run the project ad test Employee Table for loaded Data ← Regression Testing

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
<input type="text" value="Search by ..."/>	<input type="text" value="Search by ..."/>	<input type="text" value="Search by ..."/>	<input type="text" value="Search by ..."/>	<input type="text" value="Search by ..."/>	<input type="text" value="Search by ..."/>
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	2202(Rukmal)
2203	Imran	Male	Demonstrator L1	Thahir Imran	2203(Imran)
Items per page: <input type="text" value="3"/> 1 – 3 of 6 < >					

(iii) Modify btnSerchMc() function as follows

console.log(query) → this.loadTable(query);

```
139 btnSearchMc(): void {
140
141     const sserchdata = this.ssearch.getRawValue();
142
143     let number = sserchdata.ssnumber;
144     let fullname = sserchdata.ssfullname;
145     let nic = sserchdata.ssnic;
146     let genderid = sserchdata.ssgender;
147     let designationid = sserchdata.ssdesignation;
148
149     let query="";
150
151     if(number!=null && number.trim()!="") query=query+"&number="+number;
152     if(fullname!=null && fullname.trim()!="") query=query+"&fullname="+fullname;
153     if(nic!=null && nic.trim()!="") query=query+"&nic="+nic;
154     if(genderid!=null) query=query+"&genderid="+genderid;
155     if(designationid!=null) query=query+"&designationid="+designationid;
156
157     if(query!="") query = query.replace( searchValue: /\^./, replaceValue: "?")
158
159     this.loadTable(query);
160
161
162 }
```

(iv) Test – All possible Tests

You do not need to test the following. You can test based on your own data

Employee Search

Employee Table



2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	2202(Rukmal)
2203	Imran	Male	Demonstrator L1	Thahir Imran	2203(Imran)
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda	2301(Lakshan)
2302	Arshad	Male	Demonstrator L1	Arshad Ahamad	2302(Arshad)
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi	2303(Imasha)

Items per page:

10

1 - 6 of 6



All Demonstrators in the Search

Employee Search

Search by Fullname

sha

Employee Table



2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	2202(Rukmal)
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda	2301(Lakshan)
2302	Arshad	Male	Demonstrator L1	Arshad Ahamad	2302(Arshad)
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi	2303(Imasha)

Items per page:

10

1 - 5 of 5



Except Thahir Imran ← No "sha" in the name

Employee Search

sha

Male

Employee Table



Search by...	Search by...	Search by...	Search by...	Search by...	Search by...
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	2202(Rukmal)
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda	2301(Lakshan)
2302	Arshad	Male	Demonstrator L1	Arshad Ahamad	2302(Arshad)

Items per page: 10 1 - 4 of 4

Imasha Removed ← She is Female

Employee Search

shan

Male

Employee Table



Number	Calling Name	Gender	Designation	Full Name	Modification
Search by ...	Search by ...	Search by ...	Search by ...	Search by ...	Search by ...
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda	2301(Lakshan)

Items per page: 10 1 - 2 of 2

Arshad Removed ← Arshad contin "sha" but not "shan"

Employee Search

Search by Number
2201

Search by Fullname
shan

Search by NIC

Gender
Male

Designation

Search

Clear Search

Employee Table

Number	Calling Name	Gender	Designation	Full Name	Modification
Search by ...	Search by ...	Search by ...	Search by ...	Search by ...	Search by ...
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)

Items per page: 10
1 - 1 of 1

Lakshan Removed ← 2201 only match with Ashan

Clear Search Facility

- Define the “btnSearchClearMc()” event handling function
- Register the function with the Button
- Test

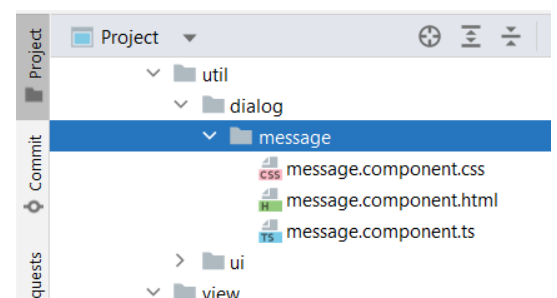
```
btnSearchClearMc(): void {
  this.ssearch.reset();
  this.loadTable( query: "");
}
```

```
<div>
<button mat-raised-button color="primary" (click)="btnSearchMc()"><mat-icon>search</mat-icon> Search </button>
<button mat-raised-button color="primary" (click)="btnSearchClearMc()"><mat-icon>reorder</mat-icon> Clear Search </button>
</div>
```



Exercise:

Review the codes in “login.ts” to revise how to use Dialog Boxes
Implement new Dialog Component for Confirm Dialog

Get User Consent, before clearing the form
By modifying “btnSearchClearMc()”



Employee Search

Search by Number	Search by Fullname shan	Search by NIC
Gender Male	Designation	
 Search		 Clear Search



Employee Table



Number	Calling Name	Gender	Designation	Full Name	Modification
Search by ...	Search by ...	Search by ...	Search by ...	Search by ...	Search by ...
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda	2301(Lakshan)
Items per page: 3 1 - 2 of 2 < >					

When Click on “Clear Search”

Employee Search

Search by Number	Search by Fullname	Search by NIC
Gender	Designation	
 Search		 Clear Search

Employee Table



Number	Calling Name	Gender	Designation	Full Name	Modification
Search by ...	Search by ...	Search by ...	Search by ...	Search by ...	Search by ...
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	2202(Rukmal)
2203	Imran	Male	Demonstrator L1	Thahir Imran	2203(Imran)
Items per page: 3 1 - 3 of 6 < >					