



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), 2(d), 2(e), 2(f) ]
4. Completing Sprint-1 Objectives

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

Data Design, UI Design, Sample Data, Database Preparation

### 2(b) Employee View

1. [Controller Design](#)
2. [Coding-Server App](#)
  - [Version-1 \(Client Pagination\)](#)  
[Entity Coding, DAO Coding, Controller Coding, Testing](#)
  - Version-2 (Server Pagination)  
Controller Modification, Testing
3. [Coding-Client App](#)
  - [Version-1 \(Client Pagination\)](#)  
[UI Preparation, Entity Coding, Service Coding, Controller Coding, Testing](#)
  - Version-2 (Server Pagination)  
Service Modification, Controller Modification, Testing

### 2(c) Employee Search

### 2(d) Employee Insert

### 2(e) Employee Update


### 2(f) Employee Delete

### 2(b) Employee View

1. [Controller Design](#)
2. [Coding Server-App](#)
3. [Coding Client-App](#)
  - Version-1 (Client Pagination)**
    1. Client App Preparation for Sprint-2
    2. UI Preparation with Static Data & Testing
    3. Coding Entity Classes
    4. Data Loading with Dynamic Data & Testing
    5. Coding Service Classes for Dynamic Data & Testing ( With & Without DI)
    6. Error Correction – Server App – Employee Data Service
    7. Data loading from Server & Testing, Promise-Pending/Resolve/Reject Status
    8. Client Pagination
    9. **Client Search – (i) Normal** (ii) Using Template
    10. Formal Method Architecture
    11. Code Explanation
    12. Code Modification
  - Version-2 (Server Pagination)**

## Client Search – (i) Without User Defined Template

### 1. Target Outcome

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
<div>Search by Number</div> <div>23</div>				
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda
2302	Arshad	Male	Demonstrator L1	Arshad Ahamad
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi
Items per page: 20 1 – 3 of 3 < >				

### 2. Discuss the facilities available in “MatTableDataSource”

Pagination, Sorting, Filtering, Data Manipulation

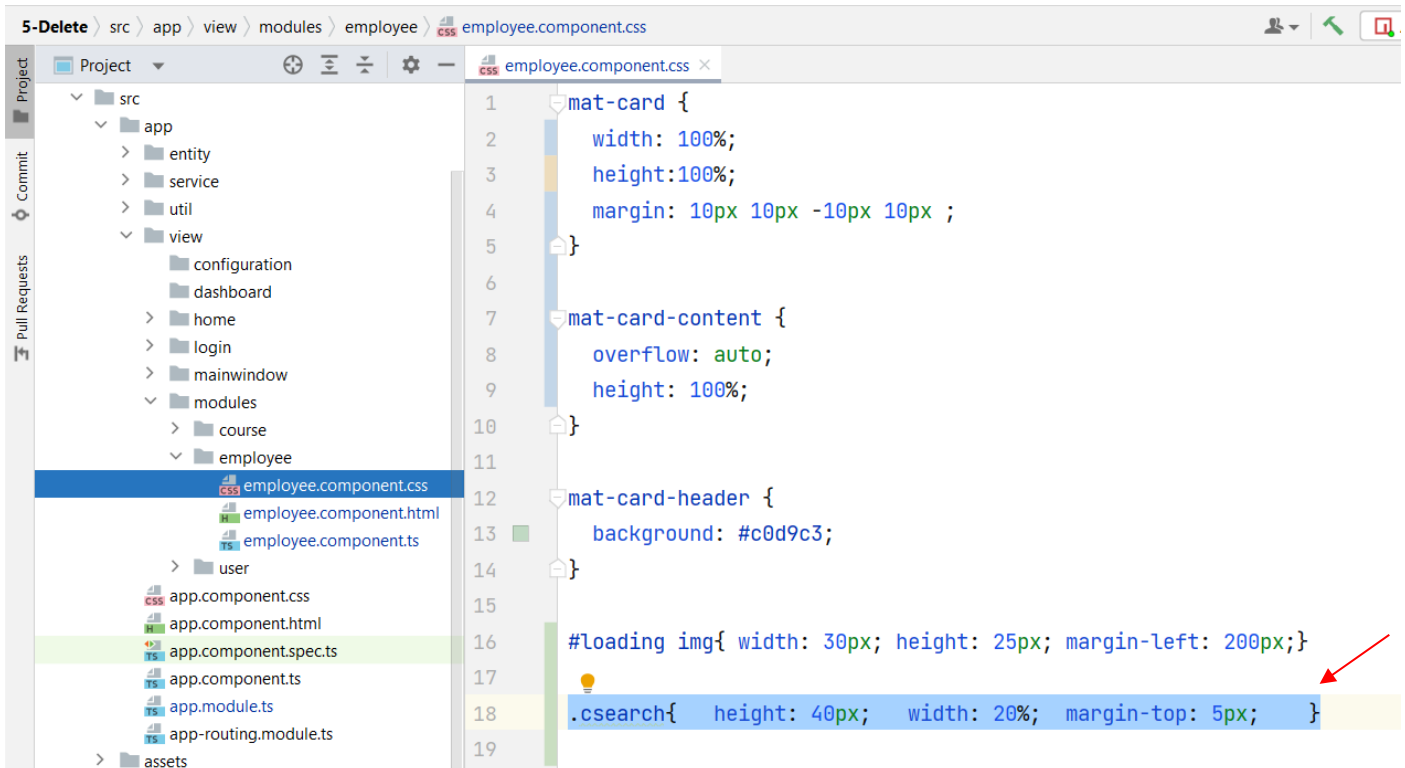
### 3. Discuss the usages of following Angular Classes in above Outcome FormBuilder, FormGroup and FormControl

### 4. Change the HTML as follows

New <mat-header-row> with “scolumns”

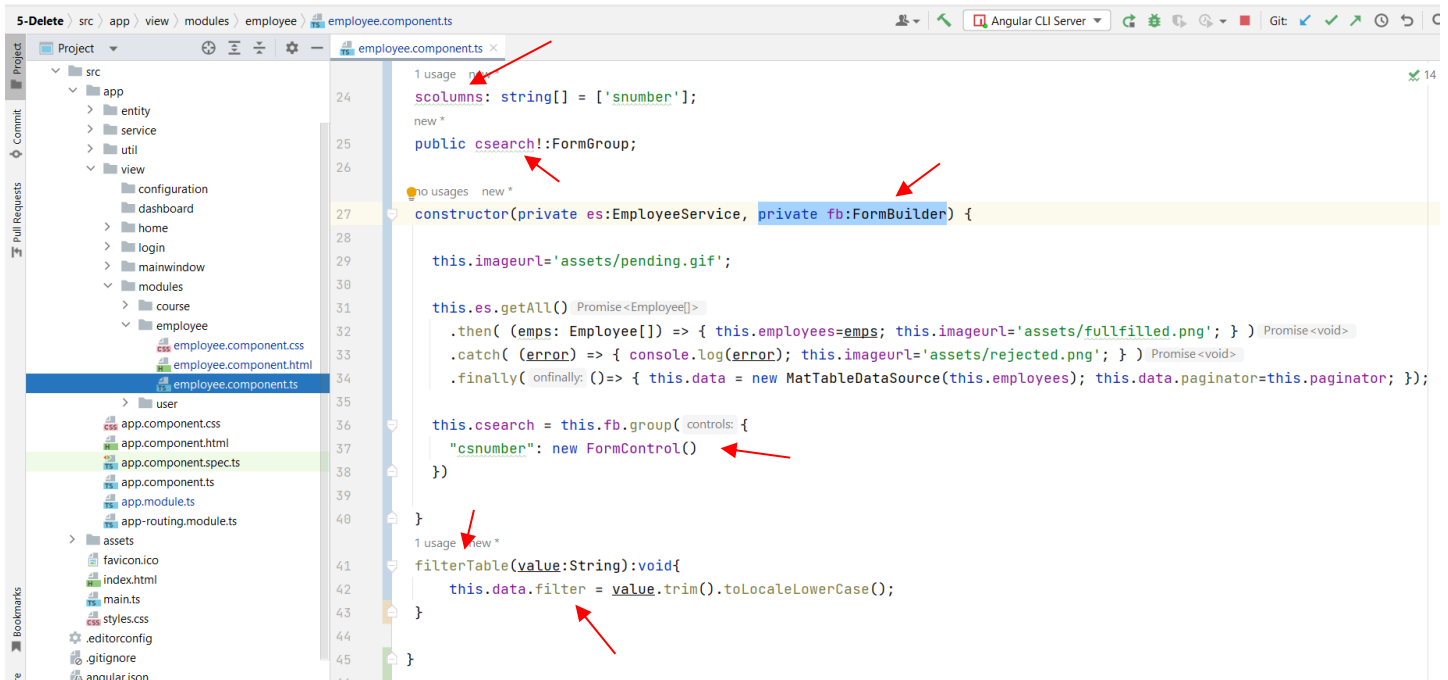
```
52
53
54 <ng-container matColumnDef="name">
55   <mat-header-cell *matHeaderCellDef> Full Name </mat-header-cell>
56   <mat-cell *matCellDef="let employee"> {{employee.fullname}}</mat-cell>
57 </ng-container>
58
59 <ng-container matColumnDef="snumber">
60   <mat-header-cell *matHeaderCellDef>
61     <mat-form [formGroup]="csearch" class="csearch">
62       <mat-form-field appearance="outline">
63         <mat-label>Search by Number</mat-label>
64         <input matInput formControlName="csnumber"
65           (keyup)="filterTable(this.csearch.controls['csnumber'].value)">
66       </mat-form-field>
67     </mat-form>
68   </mat-header-cell>
69 </ng-container>
70
71 <mat-header-row *matHeaderRowDef="columns"></mat-header-row>
72 <mat-header-row *matHeaderRowDef="scolumns"></mat-header-row>
73 <mat-row *matRowDef="let row; columns: columns;"></mat-row>
74
75 </mat-table>
76 <mat-paginator [length]="20" [pageSize]="3" [pageSizeOptions]="[3,5,10,20]"></mat-paginator>
77 </mat-card-content>
```

## 5. Change the CSS as follows




```
1 mat-card {
2   width: 100%;
3   height:100%;
4   margin: 10px 10px -10px 10px ;
5 }
6
7 mat-card-content {
8   overflow: auto;
9   height: 100%;
10 }
11
12 mat-card-header {
13   background: #c0d9c3;
14 }
15
16 #loading img{ width: 30px; height: 25px; margin-left: 200px;}
17
18 .csearch{ height: 40px; width: 20%; margin-top: 5px; }
```


## 6. Change the TS as follows



```
24 scolumns: string[] = ['snumber'];
25 new *
26 public csearch!: FormGroup;
27
28 constructor(private es: EmployeeService, private fb: FormBuilder) {
29
30   this.imageUrl='assets/pending.gif';
31
32   this.es.getAll() Promise<Employee[]>
33   .then( (emps: Employee[]) => { this.employees=emps; this.imageUrl='assets/fullfilled.png'; } ) Promise<void>
34   .catch( (error) => { console.log(error); this.imageUrl='assets/rejected.png'; } ) Promise<void>
35   .finally( (onfinally: ()=> { this.data = new MatTableDataSource(this.employees); this.data.paginator=this.paginator; });
36
37   this.csearch = this.fb.group( controls: {
38     "csnumber": new FormControl()
39   })
40 }
41
42 filterTable(value:String):void{
43   this.data.filter = value.trim().toLocaleLowerCase();
44 }
45 }
```

## 7. Testing with different values

Employee Table 				
<input type="text" value="Search by Number"/>				
2201	Ashan	Male	Demonstrator L1	Ashan Pathum
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal
2203	Imran	Male	Demonstrator L1	Thahir Imran
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda
2302	Arshad	Male	Demonstrator L1	Arshad Ahamad
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
<input type="text" value="Search by Number"/>				
2301	Lakshan	Male	Demonstrator L1	Lakshan Ruwinda
2302	Arshad	Male	Demonstrator L1	Arshad Ahamad
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi
Items per page: <input type="text" value="20"/> 1 - 3 of 3 < >				

### Applying Client Search for Other Columns

## 8. Changes in TS File

- (i) Add 4 another columns into the "scolumns" array
- (ii) Add Form-Control into FormGroup

```
scolumns: string[] = ['snumber', 'scallingname', 'sgender', 'sdesignatation' , 'sname'];
```

```
this.csearch = this.fb.group( controls: {  
  "csnumber": new FormControl(),  
  "cscallingname": new FormControl(),  
  "csgender": new FormControl(),  
  "csdesignatation": new FormControl(),  
  "csname": new FormControl()  
});
```

## 9. Changes in CSS File

```
.csearch{ height: 40px; width: 100%; margin-top: 5px; }
```

20% → 100%

## 10. Changes in HTML File

Add Header Cell Definitions to other columns, Change the column names and prompt text as required

If you correctly follow the Design Patterns and Process Steps and Tools, this task could be completed in 3 minutes

Open TS to get Column Names

```
scolumns: string[] = ['snumber', 'scallingname', 'sgender', 'sdesignation', 'sname'];
```

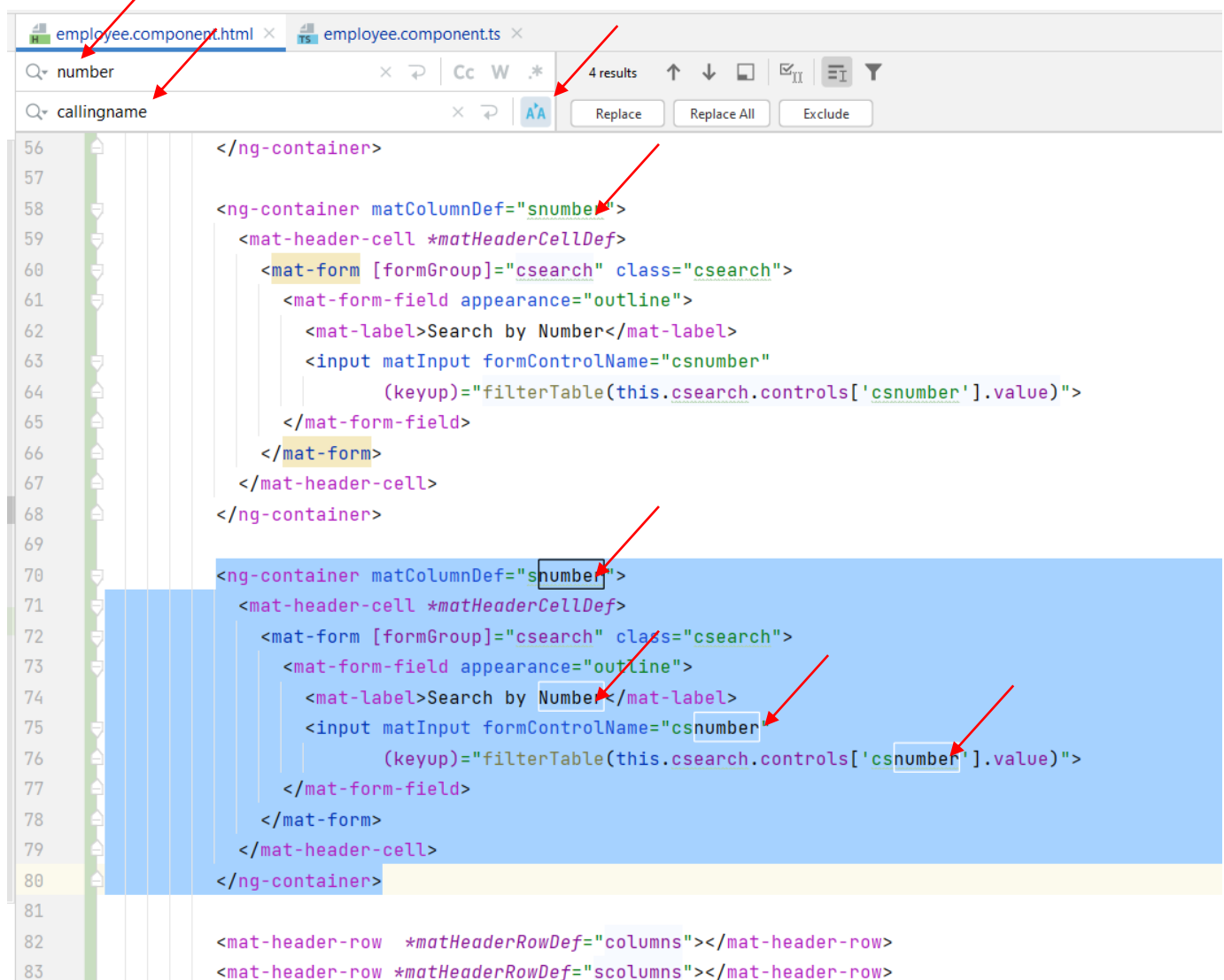
Open HTML document, select and copy the following Code segment related with 1<sup>st</sup> Column ("snumber")

```
<ng-container matColumnDef="snumber">
  <mat-header-cell *matHeaderCellDef>
    <mat-form [formGroup]="csearch" class="csearch">
      <mat-form-field appearance="outline">
        <mat-label>Search by Number</mat-label>
        <input matInput formControlName="csnumber"
          (keyup)="filterTable(this.csearch.controls['csnumber'].value)">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>
```

Paste 4 Times below this for rest of the 4 columns

Select the 2<sup>nd</sup> Code Set and Use Ctrl+R (Find & Replace Tool of the IDEA, Preserve Case)

Copy the name from "scolumns" without prefix "s"



```

<ng-container matColumnDef="scallingname">
  <mat-header-cell *matHeaderCellDef>
    <mat-form [formGroup]="csearch" class="csearch">
      <mat-form-field appearance="outline">
        <mat-label>Search by Callingname</mat-label>
        <input matInput formControlName="cscallingname"
          (keyup)="filterTable(this.csearch.controls['cscallingname'].value)">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>

```

This will improved not only the speed but also error prevention and encourage standardization

```

<ng-container matColumnDef="sgender">
  <mat-header-cell *matHeaderCellDef>
    <mat-form [formGroup]="csearch" class="csearch">
      <mat-form-field appearance="outline">
        <mat-label>Search by Gender</mat-label>
        <input matInput formControlName="csgender"
          (keyup)="filterTable(this.csearch.controls['csgender'].value)">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>

<ng-container matColumnDef="sdesignation">
  <mat-header-cell *matHeaderCellDef>
    <mat-form [formGroup]="csearch" class="csearch">
      <mat-form-field appearance="outline">
        <mat-label>Search by Designation</mat-label>
        <input matInput formControlName="csdesignation"
          (keyup)="filterTable(this.csearch.controls['csdesignation'].value)">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>

```

```

<ng-container matColumnDef="sname">
  <mat-header-cell *matHeaderCellDef>
    <mat-form [formGroup]="csearch" class="csearch">
      <mat-form-field appearance="outline">
        <mat-label>Search by Name</mat-label>
        <input matInput formControlName="csname"
          (keyup)="filterTable(this.csearch.controls['csname'].value)">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>

```

```

<mat-header-row *matHeaderRowDef="columns"></mat-header-row>
<mat-header-row *matHeaderRowDef="scolumns"></mat-header-row>
<mat-row *matRowDef="let row; columns: columns;"></mat-row>

```

```

</mat-table>

```

```

<mat-paginator [length]="20" [pageSize]="3" [pageSizeOptions]="[3,5,10,20]"></mat-paginator>

```

```

</mat-card-content>

```

```

</mat-card>


```


```

</mat-grid-tile>

```

## 11. Testing

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
<input type="text" value="Search by Nu..."/>	<input type="text" value="Search by Callingna&lt;br/&gt;Ash"/>	<input type="text" value="Search by Gen..."/>	<input type="text" value="Search by Des..."/>	<input type="text" value="Search by Na..."/>
2201	Ashan	Male	Demonstrator L1	Ashan Pathum
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi
Items per page: <input type="text" value="3"/> 1 - 2 of 2 < >				

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
<input type="text" value="Search by Number&lt;br/&gt;22"/>	<input type="text" value="Search by Callingna&lt;br/&gt;Ash"/>	<input type="text" value="Search by Ge..."/>	<input type="text" value="Search by De..."/>	<input type="text" value="Search by Na..."/>
2201	Ashan	Male	Demonstrator L1	Ashan Pathum
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal
2203	Imran	Male	Demonstrator L1	Thahir Imran

### Test Failed...Why?

Ash → Ashan(22) , Imasha(23)

Ash(22) → Ash(22)

← Expected Output

But it seems that Client Search applied here not combined result, but filtering separately.

### Correct the Failure.....

This solution will use Entity Classes.

Hence make sure that you created Entity Classes with relevant attributes.

Constructor method are not essential.

## 12. Entity Classes

```
5-Delete > src > app > entity > designation.ts
1 export class Designation {
2
3   new *
4   public id!: number;
5   public name!: string;
6
7   no usages new *
8   constructor(id:number,name:string) {
9     this.id=id;
10    this.name=name;
11  }
```

```
5-Delete > src > app > entity > gender.ts
1
2 export class Gender {
3
4   new *
5   public id!: number;
6   public name!: string;
7
8   no usages new *
9   constructor(id:number,name:string) {
10    this.id=id;
11    this.name=name;
12  }
```



```

1 import {Gender} from "../gender";
2 import {Designation} from "../designation";
3 import {EmployeeStatus} from "../employeeStatus";
4
5 + usages new *
6 export class Employee{
7
8   new *
9   public id !: number;
10
11   new *
12   public fullname !: string;
13
14   new *
15   public number !: string;
16
17   new *
18   public callingname !: string;
19
20   new *
21   public photo !: string;
22
23   new *
24   public dob !: string;
25
26   new *
27   public nic !: string;
28
29   new *
30   public address !: string;
31
32   new *
33   public mobile !: string;
34
35   new *
36   public land !: string;
37
38   new *
39   public doassignment !: string;
40
41   new *
42   public description !: string;
43
44   new *
45   public gender !: Gender;
46
47   new *
48   public designation !: Designation;
49
50   new *
51   public employeeStatus !: EmployeeStatus;
52 }

```

```

1 public designation !: Designation;
2
3 new *
4 public employeeStatus !: EmployeeStatus;
5
6 no usages new *
7 constructor(id:number, fullname:string, number:string,
8   callingname:string, photo:string, dob:string,
9   nic:string, address:string, mobile:string,
10  land:string, doassignment:string, description:string,
11  gender:Gender, designation:Designation,
12  employeeStatus:EmployeeStatus) {
13
14   this.id=id;
15   this.fullname=fullname;
16   this.number=number;
17   this.callingname=callingname;
18   this.photo=photo;
19   this.dob=dob;
20   this.nic=nic;
21   this.address=address;
22   this.mobile=mobile;
23   this.land=land;
24   this.doassignment=doassignment;
25   this.description=description;
26   this.gender=gender;
27   this.designation=designation;
28   this.employeeStatus=employeeStatus;
29 }

```

Attributes for Filtering on Search Criteria

### 13. Changes in TS File

```

36 this.csearch = this.fb.group( controls: {
37   "csnumber": new FormControl(),
38   "cscallingname": new FormControl(),
39   "csgender": new FormControl(),
40   "csdesignation": new FormControl(),
41   "csname": new FormControl()
42 })
43
44
45 + usages new *
46 filterTable():void{
47
48   const cserchdata = this.csearch.getRawValue();
49
50   this.data.filterPredicate = (employee: Employee, filter: string) => {
51
52     return (cserchdata.csnumber==null||employee.number.toLowerCase().includes(cserchdata.csnumber)) &&
53     (cserchdata.cscallingname==null||employee.callingname.toLowerCase().includes(cserchdata.cscallingname)) &&
54     (cserchdata.csgender==null||employee.gender.name.toLowerCase().includes(cserchdata.csgender)) &&
55     (cserchdata.csdesignation==null||employee.designation.name.toLowerCase().includes(cserchdata.csdesignation)) &&
56     (cserchdata.csname==null||employee.fullname.toLowerCase().includes(cserchdata.csname));
57
58   }
59
60   this.data.filter='xx';
61 }
62 }

```

Parameter passed into the filterTable() function removed.  
Required Values will be taken from the FromGroup (csearch)

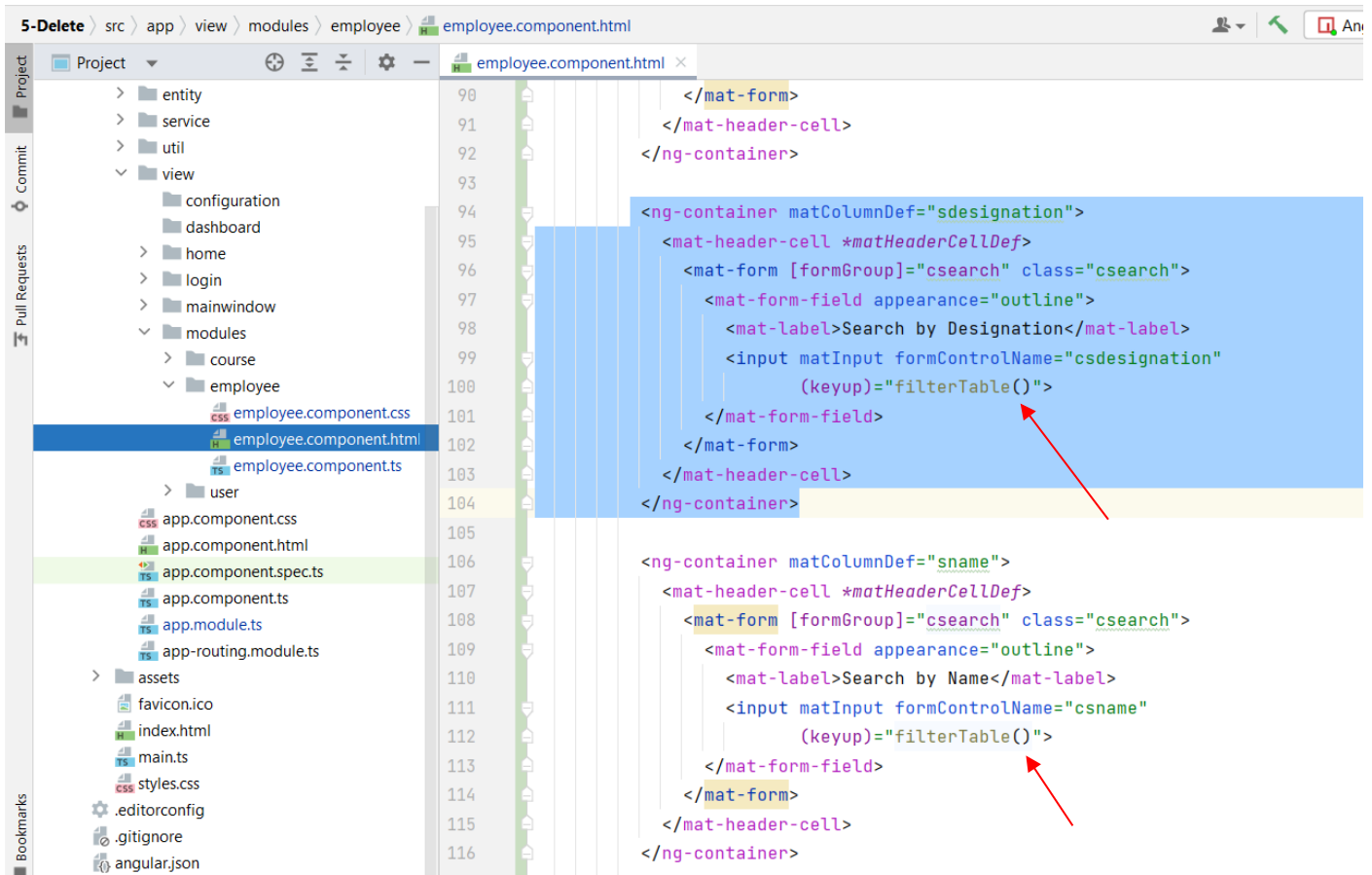
Discuss `const cserchdata = this.csearch.getRawValue();`

`this.data.filterPredicate = (employee: Employee, filter: string) => { } ;`  
`this.data.filter='xx';`

Logic used to implement the Filtering. ==null, toLowerCase(), includes(), &&, || usages



## 14. Changes in HTML File



```
5-Delete > src > app > view > modules > employee > employee.component.html
Project
Commit
Pull Requests
Bookmarks
> entity
> service
> util
> view
  > configuration
  > dashboard
  > home
  > login
  > mainwindow
  > modules
    > course
    > employee
      employee.component.css
      employee.component.html
      employee.components.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
.editorconfig
.gitignore
angular.json

90 </mat-form>
91 </mat-header-cell>
92 </ng-container>
93
94 <ng-container matColumnDef="sdesignation">
95 <mat-header-cell *matHeaderCellDef>
96 <mat-form [formGroup]="csearch" class="csearch">
97 <mat-form-field appearance="outline">
98 <mat-label>Search by Designation</mat-label>
99 <input matInput formControlName="csdesignation"
100 (keyup)="filterTable()">
101 </mat-form-field>
102 </mat-form>
103 </mat-header-cell>
104 </ng-container>
105
106 <ng-container matColumnDef="sname">
107 <mat-header-cell *matHeaderCellDef>
108 <mat-form [formGroup]="csearch" class="csearch">
109 <mat-form-field appearance="outline">
110 <mat-label>Search by Name</mat-label>
111 <input matInput formControlName="csname"
112 (keyup)="filterTable()">
113 </mat-form-field>
114 </mat-form>
115 </mat-header-cell>
116 </ng-container>
```


Remove the parameter from all the filterTable() function calls in event handler registration.

Now filterTable() function will collect all the search value from the Search Form Group instead of passing individually.

## 15. Changes in CSS File

Nothing Changed

## 16. Testing

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
<input type="text" value="Search by Nu..."/>	<div><div>Search by Callingna</div><input type="text" value="asha"/></div>	<input type="text" value="Search by Gen..."/>	<input type="text" value="Search by Des..."/>	<input type="text" value="Search by Na..."/>
2201	Ashan	Male	Demonstrator L1	Ashan Pathum
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi
Items per page: <input type="text" value="20"/> 1 - 2 of 2 < >				

Employee Table ☒

Number	Calling Name	Gender	Designation	Full Name
<div>Search by Number</div> <input type="text" value="22"/>	<div>Search by Callingna</div> <input type="text" value="asha"/>	<div>Search by Gen...</div>	<div>Search by Des...</div>	<div>Search by Na...</div>
2201	Ashan	Male	Demonstrator L1	Ashan Pathum

Items per page: 20 1 - 1 of 1

Employee Table ☒

Number	Calling Name	Gender	Designation	Full Name
<div>Search by Nu...</div>	<div>Search by Callingna</div> <input type="text" value="asha"/>	<div>Search by Gender</div> <input type="text" value="f"/>	<div>Search by Des...</div>	<div>Search by Na...</div>
2303	Imasha	Female	Demonstrator L1	Imasha Gaupadi

Items per page: 20 1 - 1 of 1

Discuss, why is it? How to correct this failure?

Employee Table ☒

Number	Calling Name	Gender
<div>Search by Nu...</div>	<div>Search by Callingna</div> <input type="text" value="ash"/>	<div>Search by C</div>
2201	Ashan	Male
2303	Imasha	Female

Employee Table ☒

Number	Calling Name	Gender	Designation	Full N:
<div>Search by Nu...</div>	<div>Search by Callingna</div> <input type="text" value="Ash"/>	<div>Search by Gen...</div>	<div>Search by Des...</div>	<div>Seai</div>

Items per page: 3 0 of 0

Discuss 4 Types of Binding Methods supported by Angular.  
Use possible example from above coding

1	Interpolation	{{ }}	
2	Property	[ ]	
3	Event	( )	
4	Two-Way	[( ngModel )]	
5	Class Binding – CSS	[class.active]="isActive"	
6	Style Binding	[style.color]="textColor" [style.font-size.px]="fontSize"	