



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 (a) Direct Property** (b) Processed
 10. Formal Method Architecture
 11. Code Explanation
 12. Code Modification
 - Version-2 (Server Pagination)**

Client Search – (i) Using User Defined Template (a) Direct Property

Successful Code Base must be designed for Minimal Redundancy.

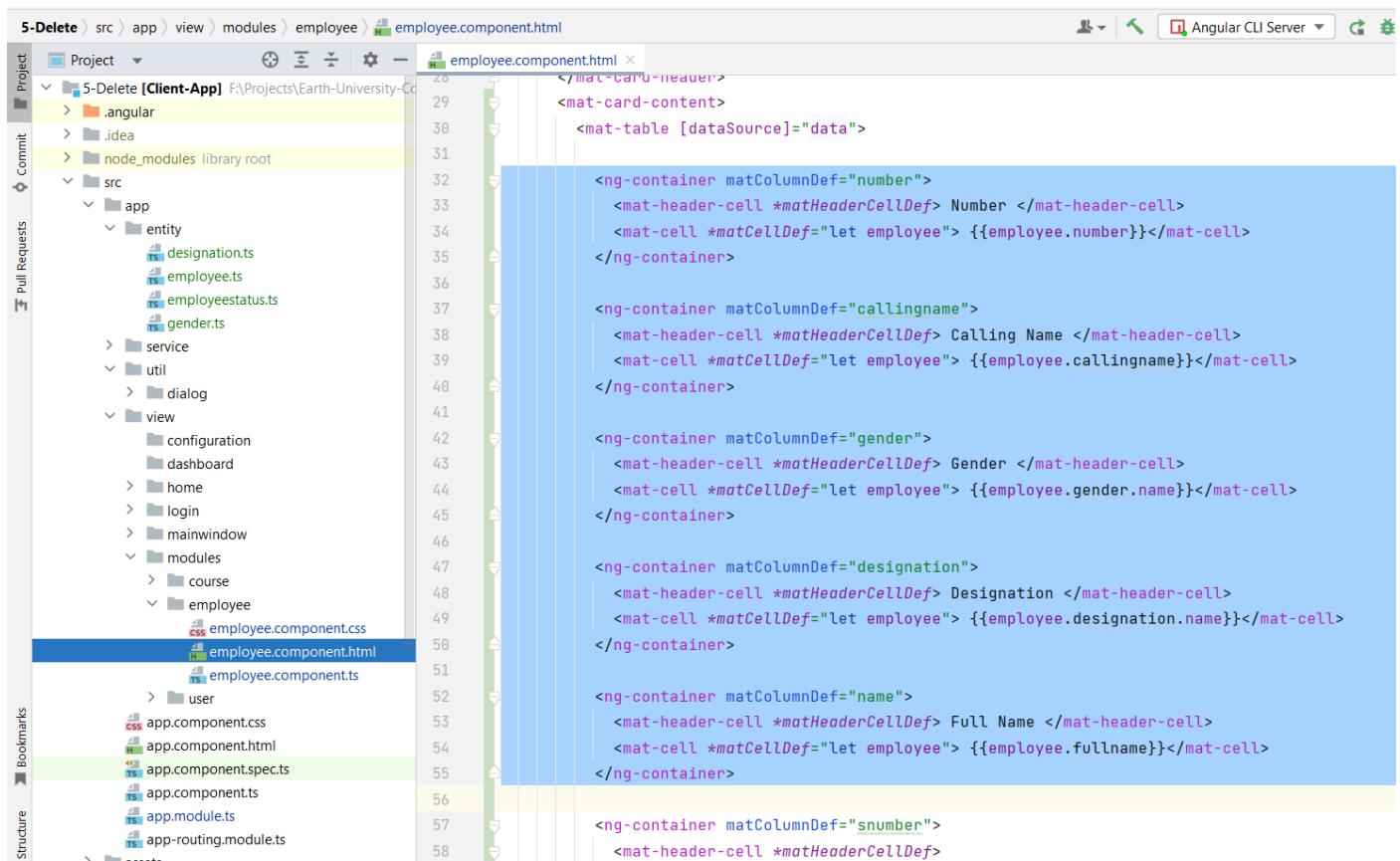
Reduced Code will make your Project more readable, reusable and maintainable.

We can reduced the repeatable codes in HTML File using Angular Structural Directives and Pipes.

Discuss popular Structural Directives used in Angular and their usage.

Discuss why and how to use Pipes in Angular Applications

1. Identify the Repeatable Code Blocks and Changing Data inside these blocks
Column Definitions of the Table

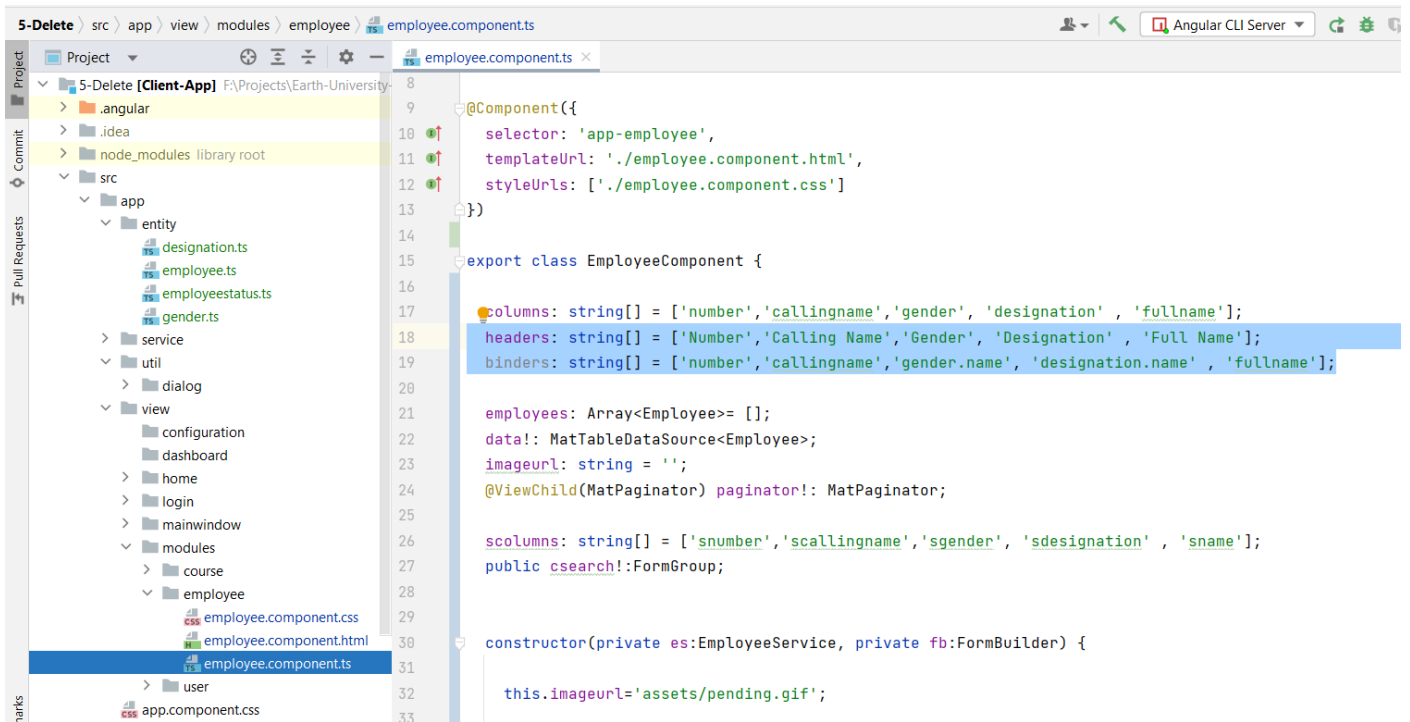


```
28 </mat-card-header>
29 <mat-card-content>
30   <mat-table [dataSource]="data">
31
32     <ng-container matColumnDef="number">
33       <mat-header-cell *matHeaderCellDef> Number </mat-header-cell>
34       <mat-cell *matCellDef="let employee"> {{employee.number}}</mat-cell>
35     </ng-container>
36
37     <ng-container matColumnDef="callingname">
38       <mat-header-cell *matHeaderCellDef> Calling Name </mat-header-cell>
39       <mat-cell *matCellDef="let employee"> {{employee.callingname}}</mat-cell>
40     </ng-container>
41
42     <ng-container matColumnDef="gender">
43       <mat-header-cell *matHeaderCellDef> Gender </mat-header-cell>
44       <mat-cell *matCellDef="let employee"> {{employee.gender.name}}</mat-cell>
45     </ng-container>
46
47     <ng-container matColumnDef="designation">
48       <mat-header-cell *matHeaderCellDef> Designation </mat-header-cell>
49       <mat-cell *matCellDef="let employee"> {{employee.designation.name}}</mat-cell>
50     </ng-container>
51
52     <ng-container matColumnDef="name">
53       <mat-header-cell *matHeaderCellDef> Full Name </mat-header-cell>
54       <mat-cell *matCellDef="let employee"> {{employee.fullname}}</mat-cell>
55     </ng-container>
56
57     <ng-container matColumnDef="snumber">
58       <mat-header-cell *matHeaderCellDef>
```

It seems there are 3 things differ from each Block to another
Column Name, Title and Data-Binding

```
<ng-container matColumnDef="number">
  <mat-header-cell *matHeaderCellDef> Number </mat-header-cell>
  <mat-cell *matCellDef="let employee"> {{employee.number}}</mat-cell>
</ng-container>
```

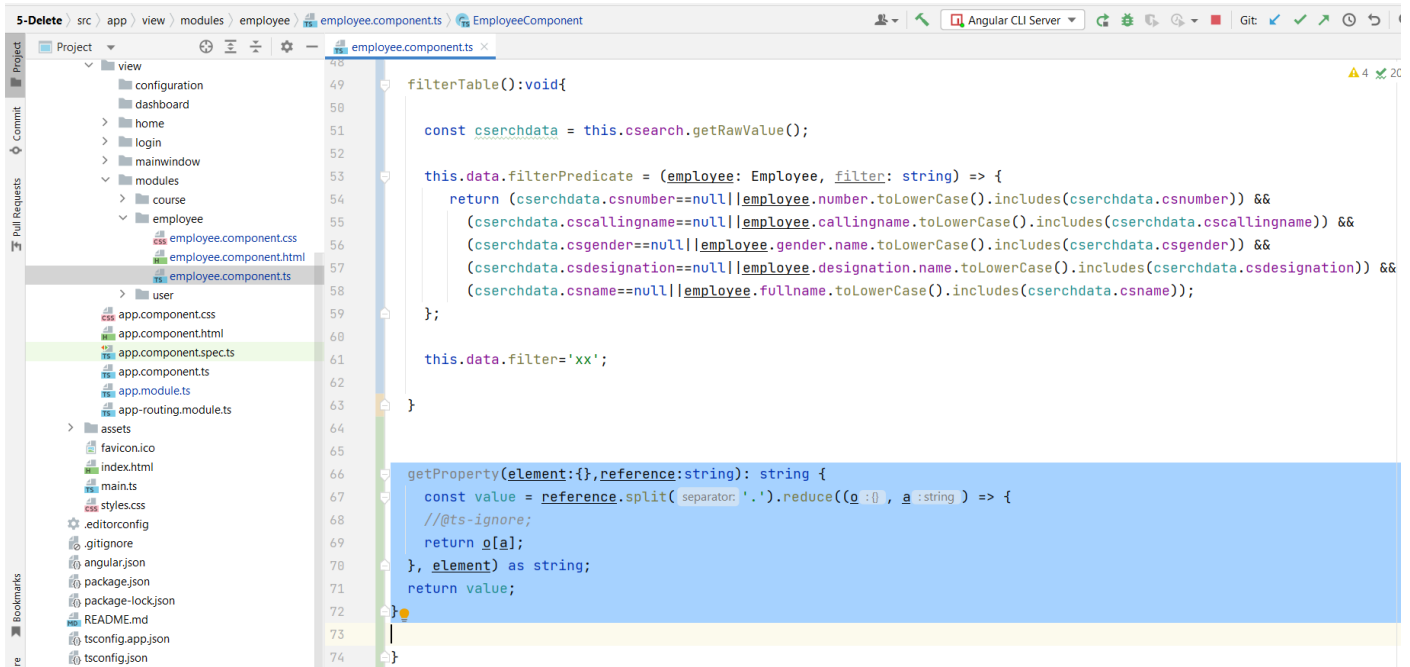
2. Create 3 Arrays to hold above details related to each column of the table



3. Create Pipe Method ("getProperty()") to get the Value to display Direct Properties (employee.callingname) and Properties available in embedded Objects (employee.gender.name)

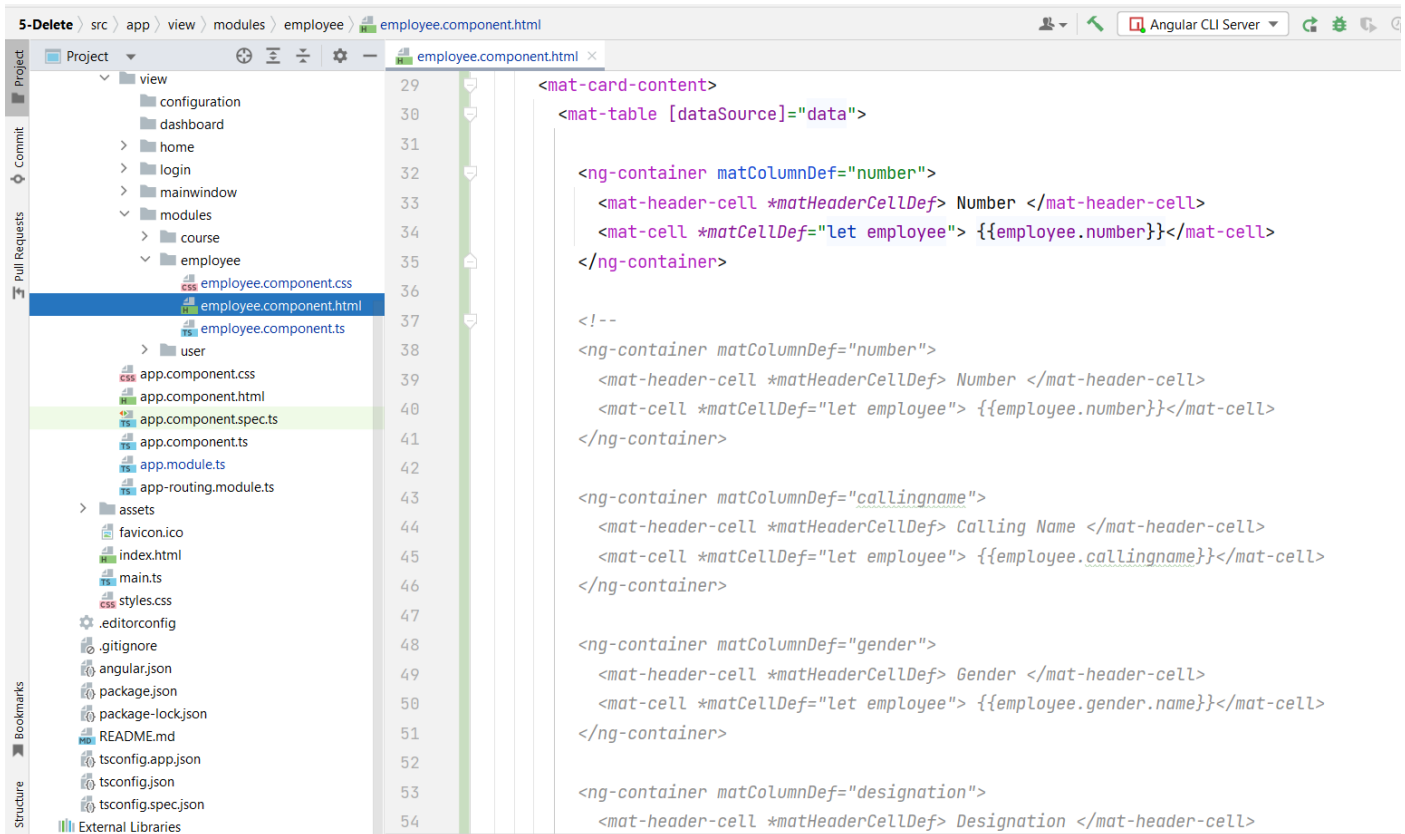
It is need to improve this Pipe Functionality of the "getProperty()" to return Complex Values based on various calculations and processing on probably several properties.

When you need to show the profit in the Column ($profit = item.saleprice - item.purchaseprice$)



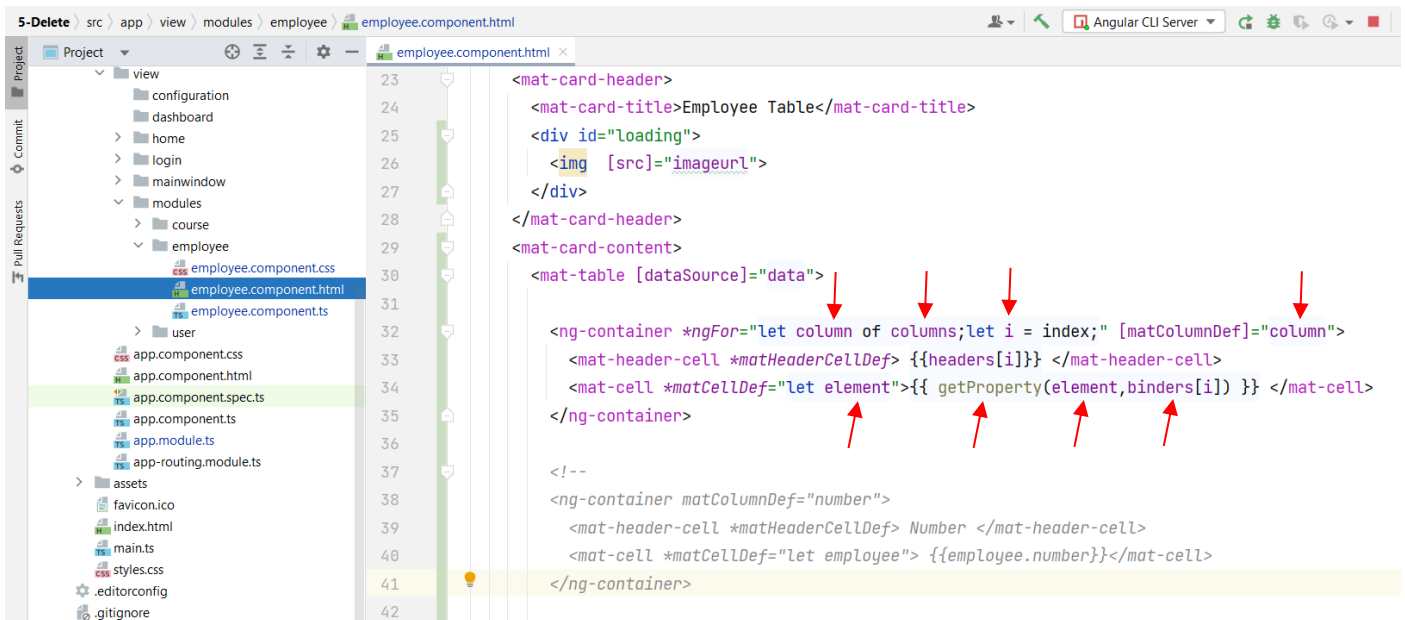
```
getProperty(element:{},reference:string): string {
    const value = reference.split('.').reduce((o, a) => {
        //@ts-ignore;
        return o[a];
    }, element) as string;
    return value;
}
```

4. Comment all Column Definitions for recovery Process.
Copy 1 Column definition as follows



```
54 <mat-card-content>
55 <mat-table [dataSource]='data'>
56
57 <ng-container matColumnDef="number">
58   <mat-header-cell *matHeaderCellDef> Number </mat-header-cell>
59   <mat-cell *matCellDef="let employee"> {{employee.number}}</mat-cell>
60 </ng-container>
61
62 <!--
63 <ng-container matColumnDef="number">
64   <mat-header-cell *matHeaderCellDef> Number </mat-header-cell>
65   <mat-cell *matCellDef="let employee"> {{employee.number}}</mat-cell>
66 </ng-container>
67
68 <ng-container matColumnDef="callingname">
69   <mat-header-cell *matHeaderCellDef> Calling Name </mat-header-cell>
70   <mat-cell *matCellDef="let employee"> {{employee.callingname}}</mat-cell>
71 </ng-container>
72
73 <ng-container matColumnDef="gender">
74   <mat-header-cell *matHeaderCellDef> Gender </mat-header-cell>
75   <mat-cell *matCellDef="let employee"> {{employee.gender.name}}</mat-cell>
76 </ng-container>
77
78 <ng-container matColumnDef="designation">
79   <mat-header-cell *matHeaderCellDef> Designation </mat-header-cell>
80 </ng-container>
81 </mat-table>
82 </mat-card-content>
```


5. Change the uncommented column definition as follows



```
30 <mat-card-header>
31 <mat-card-title>Employee Table</mat-card-title>
32 <div id="loading">
33   <img [src]="imageUrl">
34 </div>
35 </mat-card-header>
36 <mat-card-content>
37 <mat-table [dataSource]='data'>
38
39 <ng-container *ngFor="let column of columns; let i = index;" [matColumnDef]="column">
40   <mat-header-cell *matHeaderCellDef> {{headers[i]}} </mat-header-cell>
41   <mat-cell *matCellDef="let element">{{ getProperty(element,binders[i]) }} </mat-cell>
42 </ng-container>
43
44 <!--
45 <ng-container matColumnDef="number">
46   <mat-header-cell *matHeaderCellDef> Number </mat-header-cell>
47   <mat-cell *matCellDef="let employee"> {{employee.number}}</mat-cell>
48 </ng-container>
49 </mat-table>
50 </mat-card-content>
```

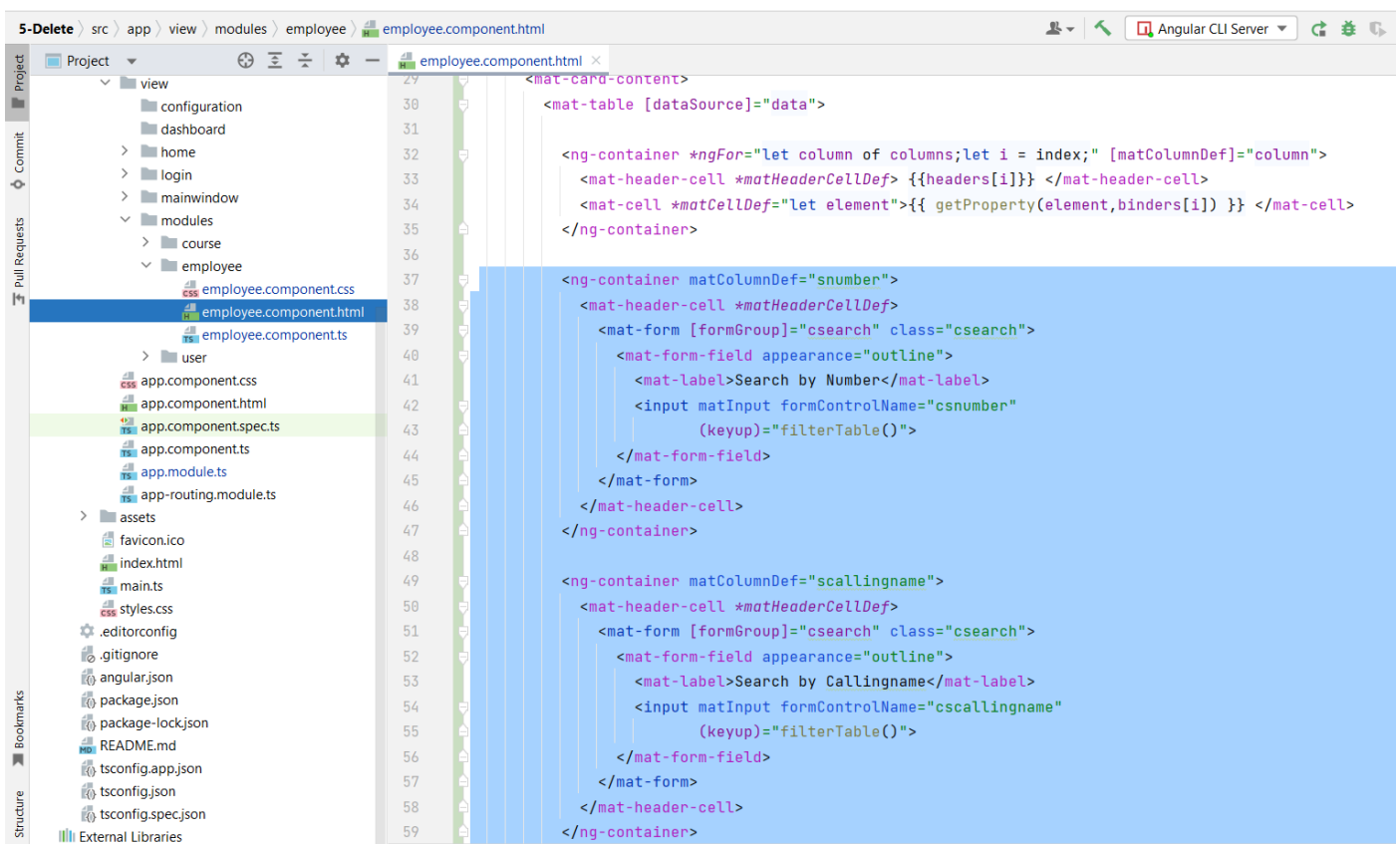
```
<ng-container *ngFor="let column of columns; let i = index;" [matColumnDef]="column">
  <mat-header-cell *matHeaderCellDef> {{headers[i]}} </mat-header-cell>
  <mat-cell *matCellDef="let element">{{ getProperty(element,binders[i]) }} </mat-cell>
</ng-container>
```

- Testing – Load the Employee Table and test Client Search for affected components
(If test pass, remove commented Column Definitions)

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
<input type="text" value="Search by Nu..."/>	<input type="text" value="Search by Callingna
asha"/>	<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: 20 1 - 2 of 2 < >

- Observe the Client-Search Code Blocks for Redundancy



```

29 <mat-card-content>
30 <mat-table [dataSource]="data">
31
32 <ng-container *ngFor="let column of columns;let i = index;" [matColumnDef]="column">
33 <mat-header-cell *matHeaderCellDef> {{headers[i]}} </mat-header-cell>
34 <mat-cell *matCellDef="let element">{{ getProperty(element,binders[i]) }} </mat-cell>
35 </ng-container>
36
37 <ng-container matColumnDef="snumber">
38 <mat-header-cell *matHeaderCellDef>
39 <mat-form [formGroup]="csearch" class="csearch">
40 <mat-form-field appearance="outline">
41 <mat-label>Search by Number</mat-label>
42 <input matInput formControlName="csnumber"
43 (keyup)="filterTable()">
44 </mat-form-field>
45 </mat-form>
46 </mat-header-cell>
47 </ng-container>
48
49 <ng-container matColumnDef="scallingname">
50 <mat-header-cell *matHeaderCellDef>
51 <mat-form [formGroup]="csearch" class="csearch">
52 <mat-form-field appearance="outline">
53 <mat-label>Search by Callingname</mat-label>
54 <input matInput formControlName="cscallingname"
55 (keyup)="filterTable()">
56 </mat-form-field>
57 </mat-form>
58 </mat-header-cell>
59 </ng-container>

```

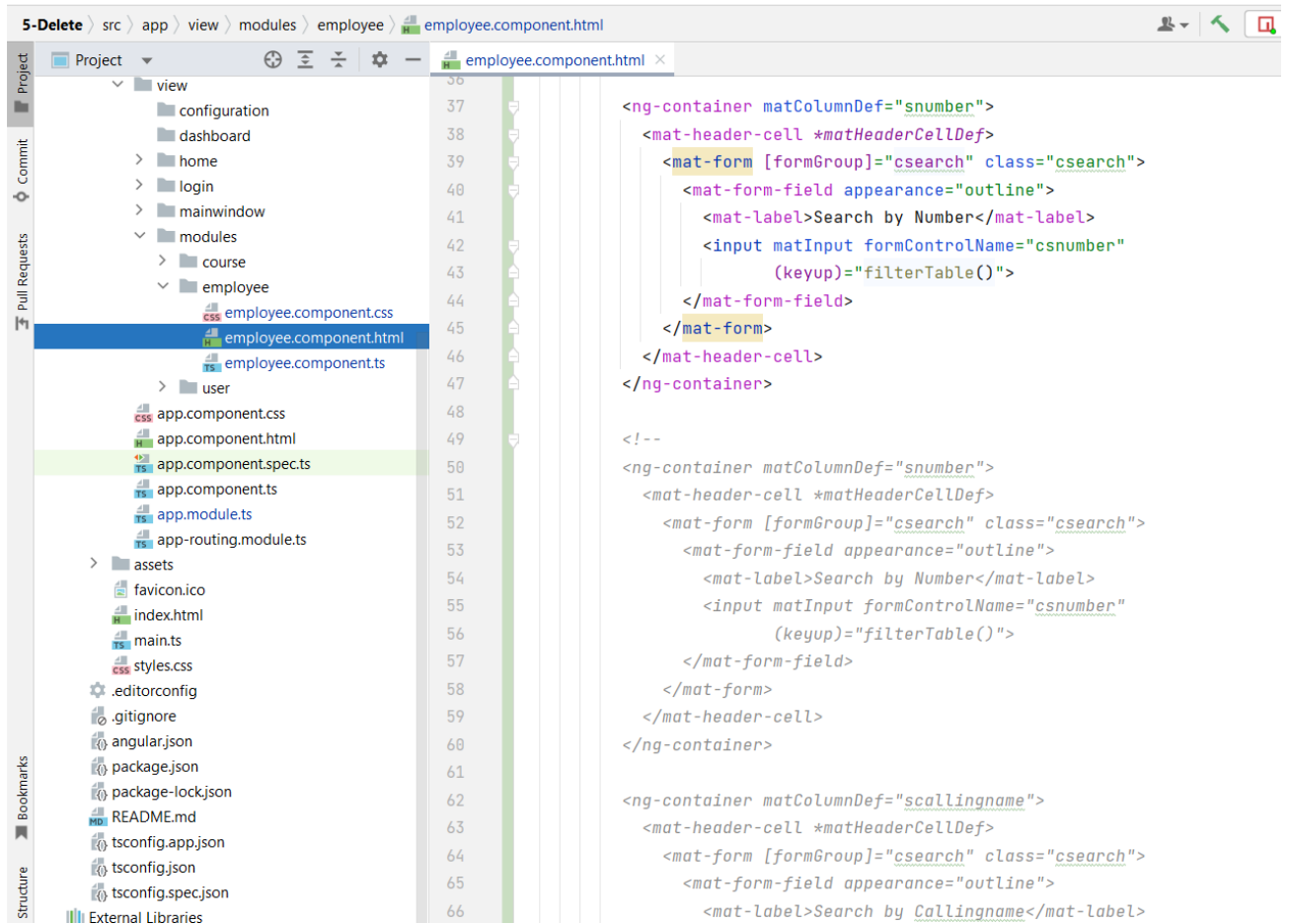
```

<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()">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>

```

Changing Values → client search column names (cscolumns) and Search Prompt Texts (prompts)

8. Comment all Client Search (csearch) Definitions for recovery Process.
Copy 1 definition as follows



```
5-Delete src > app > view > modules > employee > employee.component.html
Project 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.spect.ts app.component.ts app.module.ts app-routing.module.ts assets favicon.ico index.html main.ts styles.css .editorconfig .gitignore angular.json package.json package-lock.json README.md tsconfig.app.json tsconfig.json tsconfig.spec.json External Libraries

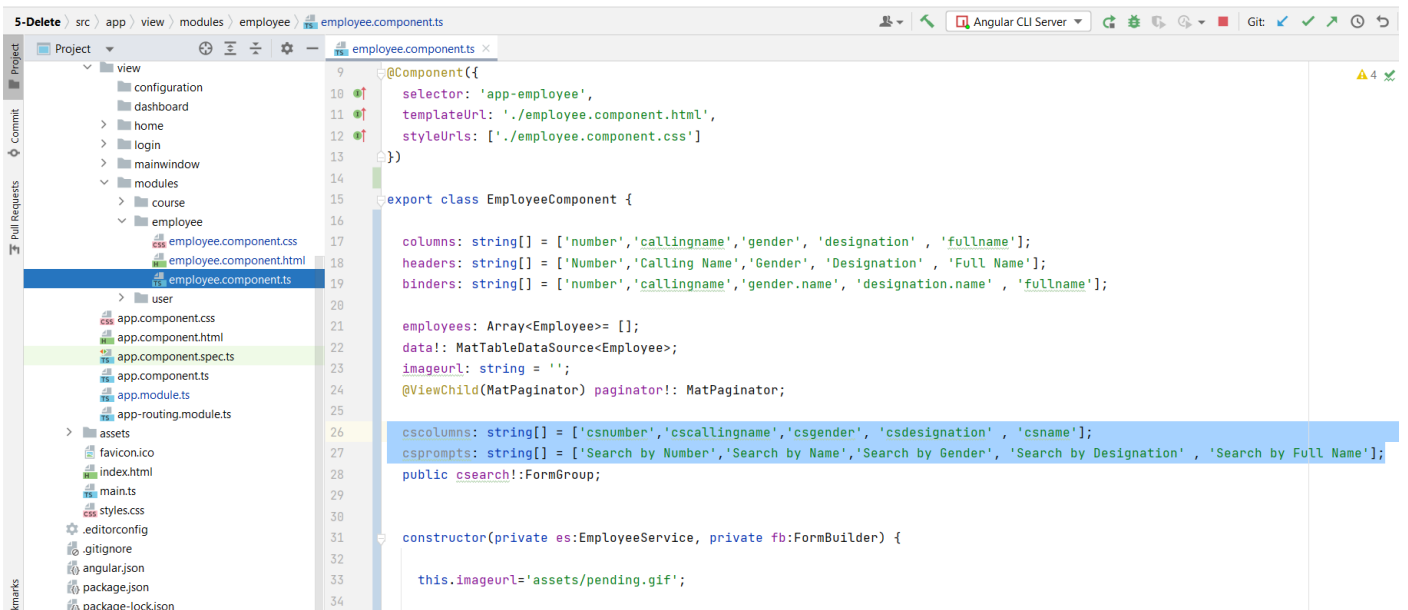
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66

<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()">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>

<!--
<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()">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>

<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>
```

9. Define needed data arrays for Client Search Code Blocks



```
5-Delete src > app > view > modules > employee > employee.component.ts
Project 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.spect.ts app.component.ts app.module.ts app-routing.module.ts assets favicon.ico index.html main.ts styles.css .editorconfig .gitignore angular.json package.json package-lock.json
Angular CLI Server Git

9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34

@Component({
  selector: 'app-employee',
  templateUrl: './employee.component.html',
  styleUrls: ['./employee.component.css']
})
export class EmployeeComponent {
  columns: string[] = ['number', 'callingname', 'gender', 'designation', 'fullname'];
  headers: string[] = ['Number', 'Calling Name', 'Gender', 'Designation', 'Full Name'];
  binders: string[] = ['number', 'callingname', 'gender.name', 'designation.name', 'fullname'];

  employees: Array<Employee> = [];
  data!: MatTableDataSource<Employee>;
  imageUrl: string = '';
  @ViewChild(MatPaginator) paginator!: MatPaginator;

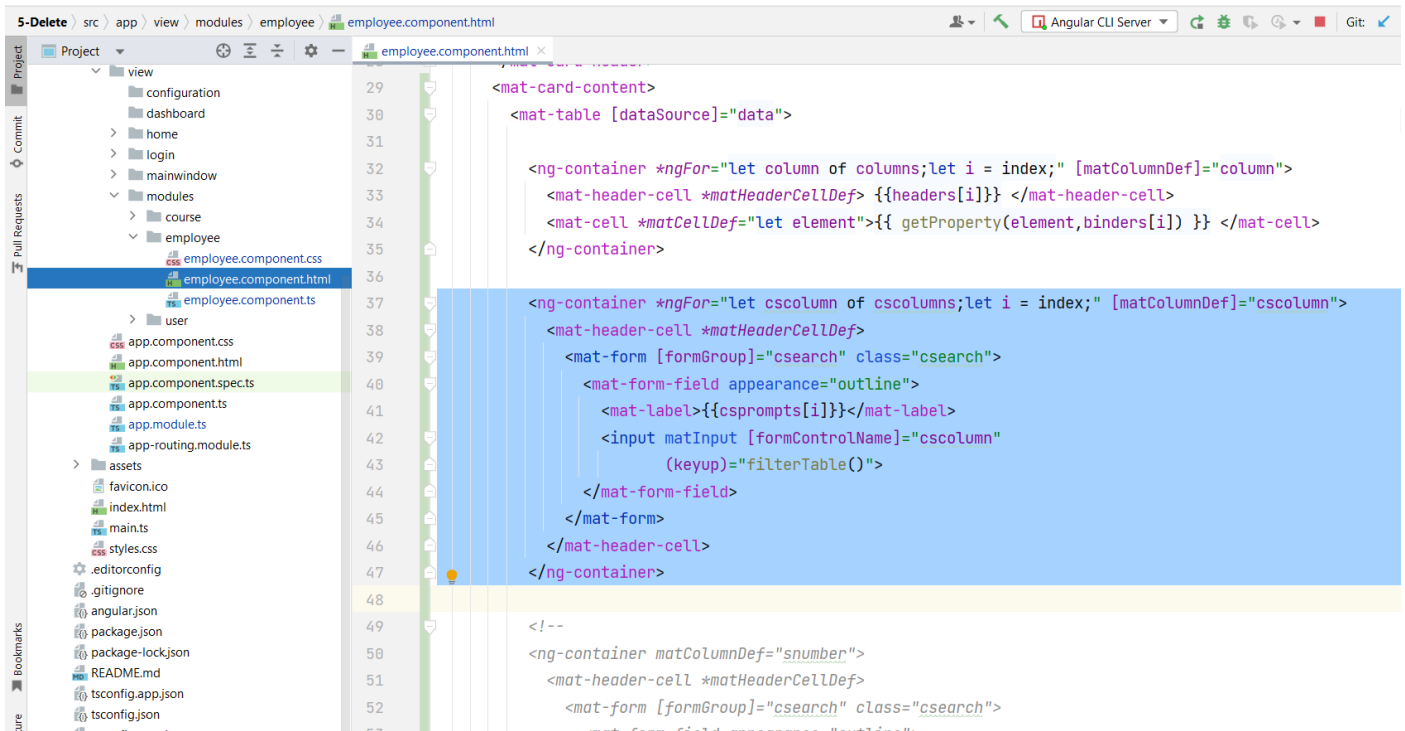
  cscolumns: string[] = ['csnumber', 'scallingname', 'csgender', 'csdesignation', 'csname'];
  csprompts: string[] = ['Search by Number', 'Search by Name', 'Search by Gender', 'Search by Designation', 'Search by Full Name'];
  public csearch!: FormGroup;

  constructor(private es: EmployeeService, private fb: FormBuilder) {
    this.imageUrl = 'assets/pending.gif';
  }
}
```

```
cscolumns: string[] = ['csnumber', 'scallingname', 'csgender', 'csdesignation', 'csname'];

csprompts: string[] = ['Search by Number', 'Search by Name', 'Search by Gender', 'Search by Designation', 'Search by Full Name'];
```

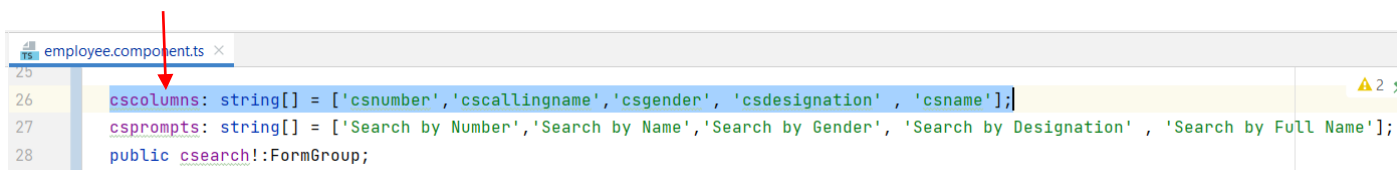

10. Modify the Template Code Block with required variables from the Controller



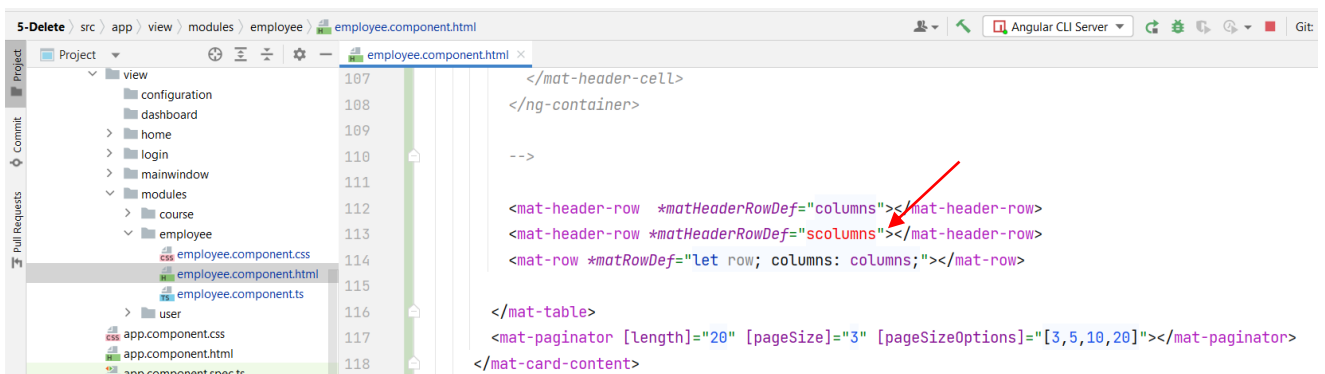
```
5-Delete | src | app | view | modules | employee | employee.component.html
Project | Commit | Pull Requests | Bookmarks | employee.component.html
view
├── configuration
├── dashboard
├── home
├── login
├── mainwindow
├── modules
│   ├── course
│   └── employee
│       ├── employee.component.css
│       ├── 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
│   ├── package.json
│   ├── package-lock.json
│   ├── README.md
│   ├── tsconfig.app.json
│   └── tsconfig.json
└── ...
29 <mat-card-content>
30 <mat-table [dataSource]="data">
31
32 <ng-container *ngFor="let column of columns;let i = index;" [matColumnDef]="column">
33 <mat-header-cell *matHeaderCellDef> {{headers[i]}} </mat-header-cell>
34 <mat-cell *matCellDef="let element">{{ getProperty(element,binders[i]) }} </mat-cell>
35 </ng-container>
36
37 <ng-container *ngFor="let cscolumn of cscolumns;let i = index;" [matColumnDef]="cscolumn">
38 <mat-header-cell *matHeaderCellDef>
39 <mat-form [formGroup]="csearch" class="csearch">
40 <mat-form-field appearance="outline">
41 <mat-label>{{csprompts[i]}}</mat-label>
42 <input matInput [formControlName]="cscolumn"
43 (keyup)="filterTable()">
44 </mat-form-field>
45 </mat-form>
46 </mat-header-cell>
47 </ng-container>
48
49 <!--
50 <ng-container matColumnDef="snumber">
51 <mat-header-cell *matHeaderCellDef>
52 <mat-form [formGroup]="csearch" class="csearch">
```

```
<ng-container *ngFor="let cscolumn of cscolumns;let i = index;" [matColumnDef]="cscolumn">
  <mat-header-cell *matHeaderCellDef>
    <mat-form [formGroup]="csearch" class="csearch">
      <mat-form-field appearance="outline">
        <mat-label>{{csprompts[i]}}</mat-label>
        <input matInput [formControlName]="cscolumn"
          (keyup)="filterTable()">
      </mat-form-field>
    </mat-form>
  </mat-header-cell>
</ng-container>
```

11. Change the name of "scolumns" into "cscolumns"



```
26 cscolumns: string[] = ['csnumber','cscallingname','csgender','csdesignation','csname'];
27 csprompts: string[] = ['Search by Number','Search by Name','Search by Gender','Search by Designation','Search by Full Name'];
28 public csearch!: FormGroup;
```




```
107 </mat-header-cell>
108 </ng-container>
109
110 -->
111
112 <mat-header-row *matHeaderRowDef="columns"></mat-header-row>
113 <mat-header-row *matHeaderRowDef="scolumns"></mat-header-row>
114 <mat-row *matRowDef="let row; columns: columns;"></mat-row>
115
116 </mat-table>
117 <mat-paginator [length]="20" [pageSize]="3" [pageSizeOptions]="[3,5,10,20]"></mat-paginator>
118 </mat-card-content>
```

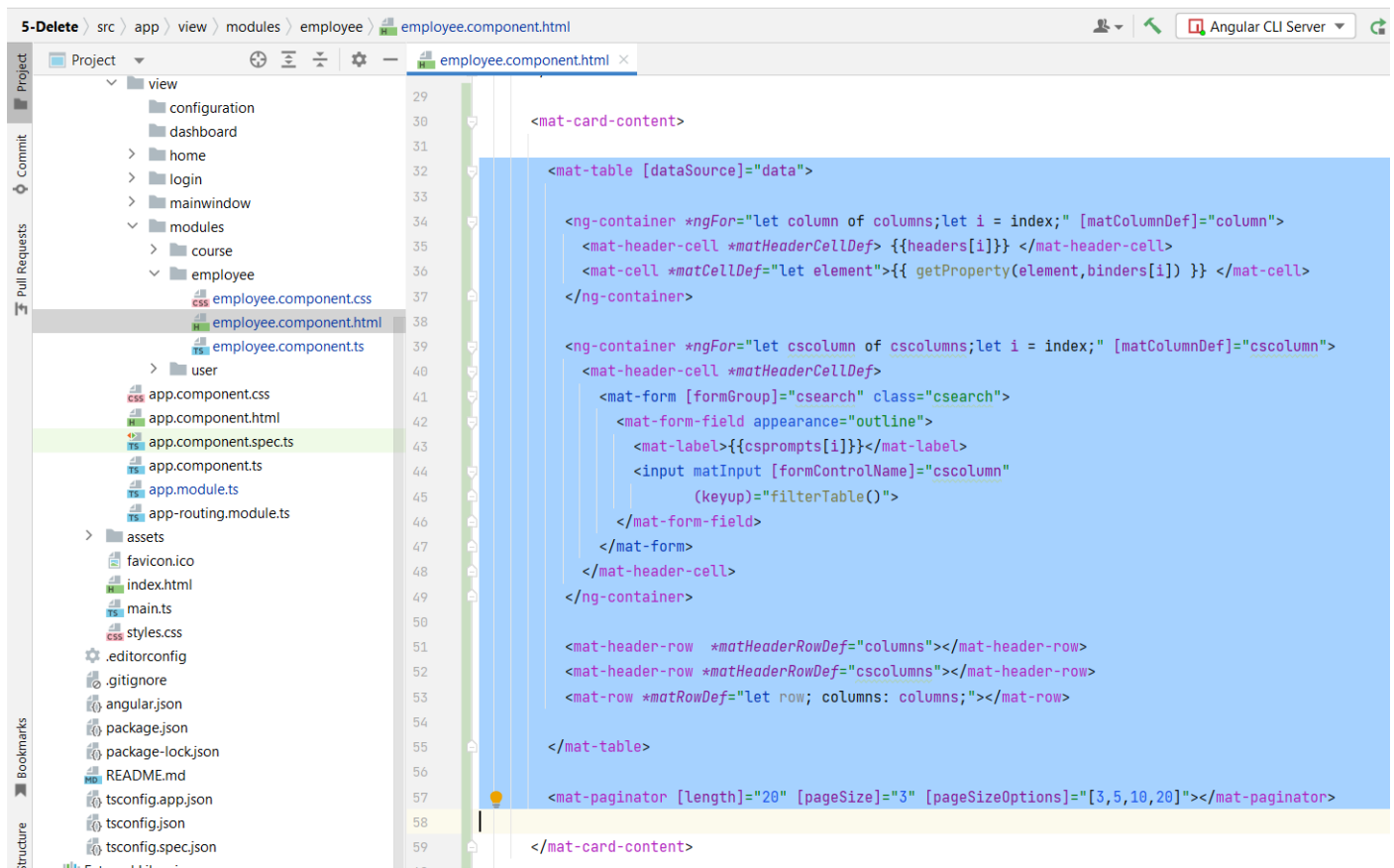
<mat-header-row *matHeaderRowDef="scolumns"></mat-header-row> ← Old Array Name

<mat-header-row *matHeaderRowDef="cscolumns"></mat-header-row> ← New Array Name

12. Testing – Load the Employee Table and test Client Search for affected components
(If test pass, remove commented Column Definitions)

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
<input type="text" value="22"/>	<input type="text" value="Search by Name"/>	<input type="text" value="Search by Gen..."/>	<input type="text" value="Search by Des..."/>	<input type="text" value="Search by Full..."/>
2201	Ashan	Male	Demonstrator L1	Ashan Pathum
Items per page: <input type="text" value="3"/> 1 – 1 of 1 < >				

13. Template Coding for the Table and Client Search



```
5-Delete src > app > view > modules > employee > employee.component.html
Project view
configuration
dashboard
home
login
mainwindow
modules
course
employee
  employee.component.css
  employee.component.html
  employee.components
  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
package.json
package-lock.json
README.md
tsconfig.app.json
tsconfig.json
tsconfig.spec.json
External Libraries

29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

<mat-card-content>

<mat-table [dataSource]="data">

  <ng-container *ngFor="let column of columns;let i = index;" [matColumnDef]="column">
    <mat-header-cell *matHeaderCellDef> {{headers[i]}} </mat-header-cell>
    <mat-cell *matCellDef="let element">{{ getProperty(element,binders[i]) }} </mat-cell>
  </ng-container>

  <ng-container *ngFor="let cscolumn of cscolumns;let i = index;" [matColumnDef]="cscolumn">
    <mat-header-cell *matHeaderCellDef>
      <mat-form [formGroup]="csearch" class="csearch">
        <mat-form-field appearance="outline">
          <mat-label>{{csprompts[i]}}</mat-label>
          <input matInput [formControlName]="cscolumn" (keyup)="filterTable()">
        </mat-form-field>
      </mat-form>
    </mat-header-cell>
  </ng-container>

  <mat-header-row *matHeaderRowDef="columns"></mat-header-row>
  <mat-header-row *matHeaderRowDef="cscolumns"></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>
```