



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 with Processing Pipe Functions

1. Code Modification – Add a Column at the End of the table to show “Number(Calling Name)”
Column Name – modi
Column Header – Modification
Binding Values – Processed Output with “number” & “callingname” as 2201(Ashan), 2202(Rukmal)

Employee Table 				
Number	Calling Name	Gender	Designation	Full Name
2201	Ashan	Male	Demonstrator L1	Ashan Pathum
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal
2203	Imran	Male	Demonstrator L1	Thahir Imran
Items per page: 3 1 – 3 of 6 < >				

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
<input type="text" value="Search by Nu..."/>	<input type="text" value="Search by Na..."/>	<input type="text" value="Search by Ge..."/>	<input type="text" value="Search by Des..."/>	<input type="text" value="Search by Full..."/>	<input type="text" value="Search by Modi"/>
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 < >					

Here Binding Value is not just a property reference.

Hence we need to use another Pipe function to get the processed value to display

Observe the importance of the Maintainability and the strength of a design in Code Modification.

(a) Test Driven Development will start with Addition of a simple Column to Display Number Column Again

- (i) Add data to the 5 Arrays for Both Display and Client Search

```
14
15 export class EmployeeComponent {
16
17   columns: string[] = ['number', 'callingname', 'gender', 'designation', 'fullname', 'modi'];
18   headers: string[] = ['Number', 'Calling Name', 'Gender', 'Designation', 'Full Name', 'Modification'];
19   binders: string[] = ['number', 'callingname', 'gender.name', 'designation.name', 'fullname', 'number'];
20
21   cscolumns: string[] = ['csnumber', 'cscallingname', 'csgender', 'csdesignation', 'csname', 'csmodi'];
22   csprompts: string[] = ['Search by Number', 'Search by Name', 'Search by Gender', 'Search by Designation', 'Search by Full Name', 'Search by Modi'];
23   public csearch!: FormGroup;
24
25   employees: Array<Employee> = [];
26   data!: MatTableDataSource<Employee>;
27   imageUrl: string = '';
28   @ViewChild(MatPaginator) paginator!: MatPaginator;
29
```

(ii) Add Form Control(csmodi) into From Group to support Client Search (csearch)

```
30
31 constructor(private es: EmployeeService, private fb: FormBuilder) {
32
33     this.imageUrl = 'assets/pending.gif';
34
35     this.es.getAll() Promise<Employee[]>
36     .then((emps: Employee[]) => {
37         this.employees = emps;
38         this.imageUrl = 'assets/fullfilled.png';
39     }) Promise<void>
40     .catch((error) => {
41         console.log(error);
42         this.imageUrl = 'assets/rejected.png';
43     }) Promise<void>
44     .finally( onfinally: () => {
45         this.data = new MatTableDataSource(this.employees);
46         this.data.paginator = this.paginator;
47     });
48
49     this.csearch = this.fb.group( controls: {
50         "csnumber": new FormControl(),
51         "cscallingname": new FormControl(),
52         "csgender": new FormControl(),
53         "csdesignation": new FormControl(),
54         "csname": new FormControl(),
55         "csmodi": new FormControl(),
56     });
57
58 }
59
```


(iii) Add a Filter Criteria to support Client Search (Copy from Number Column and Change the "csmodi"

```
60
61 filterTable(): void {
62
63     const cserchdata = this.csearch.getRawValue();
64
65     this.data.filterPredicate = (employee: Employee, filter: string) => {
66         return (cserchdata.csnumber == null || employee.number.toLowerCase().includes(cserchdata.csnumber)) &&
67             (cserchdata.cscallingname == null || employee.callingname.toLowerCase().includes(cserchdata.cscallingname)) &&
68             (cserchdata.csgender == null || employee.gender.name.toLowerCase().includes(cserchdata.csgender)) &&
69             (cserchdata.csdesignation == null || employee.designation.name.toLowerCase().includes(cserchdata.csdesignation)) &&
70             (cserchdata.csname == null || employee.fullname.toLowerCase().includes(cserchdata.csname)) &&
71             (cserchdata.csmodi == null || employee.number.toLowerCase().includes(cserchdata.csmodi));
72     };
73
74     this.data.filter = 'xx';
75
76 }
```


(iv) Keep the "getProperty()" same as the previous

```
78
79 getProperty(element: {}, reference: string): string {
80
81     const value = reference.split( separator: '.' ).reduce((o: {}, a: string) => {
82         //@ts-ignore;
83         return o[a];
84     }, element) as string;
85     return value;
86
87 }
```

- (v) Test the Output for all outcome of the affected component (Regression Testing)
View All, Client Search

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
Search by Nu...	Search by Na...	Search by Ge...	Search by Des...	Search by Full...	Search by Modi
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	2202
2203	Imran	Male	Demonstrator L1	Thahir Imran	2203

Items per page: 3 1 - 3 of 6

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
Search by Nu...	Search by Na...	Search by Ge...	Search by Des...	Search by Full...	Search by Modi 2201
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201

Items per page: 3 1 - 1 of 1

(b) Test Driven Development – Step 2

Convert the above column to Support Processed Output as 2201(Ashan) instead of 2201

- (i) Change the binding value to a Function name as “getModi()”

You can imagine this with “getProfit()” in Item Table, where Profit will be calculated using Purchase-Price and Sale-Price

```

14
15 export class EmployeeComponent {
16
17     columns: string[] = ['number', 'callingname', 'gender', 'designation', 'fullname', 'modi'];
18     headers: string[] = ['Number', 'Calling Name', 'Gender', 'Designation', 'Full Name', 'Modification'];
19     binders: string[] = ['number', 'callingname', 'gender.name', 'designation.name', 'fullname', 'getModi()'];
20

```


- (ii) Change the “getProperty()” to support references of pipe functions in addition to property reference

```

79 getProperty(element: {}, reference: string): string {
80
81     if (!reference.endsWith("(")) {
82         const value = reference.split( separator: '.' ).reduce((o: {}, a: string) => {
83             //@ts-ignore;
84             return o[a];
85         }, element) as string;
86         return value;
87     }
88     else { return 'Ha Ha'; }
89
90 }

```

(iii) Test the Output

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
<div>Search by Nu... Search by Na... Search by Ge... Search by Des... Search by Full... Search by Modi</div>					
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	Ha Ha
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	Ha Ha
2203	Imran	Male	Demonstrator L1	Thahir Imran	Ha Ha
Items per page: 3 1 - 3 of 6 < >					

- (iv) Add a Custom Cell-Value-Factory (As in the JavaFX), here a Pipe Function in Angular to build the content to display


```
92
93   getModi(element: Employee) {
94     return 'La La';
95   }
```

- (v) Modify the getProperty() to call this function to build the cell values

```
78
79   getProperty(element: {}, reference: string): string {
80
81     if (!reference.endsWith("(")) {
82       const value = reference.split(separator: '.').reduce((o: {}, a: string) => {
83         //@ts-ignore;
84         return o[a];
85       }, element) as string;
86       return value;
87     }
88     else {
89       //@ts-ignore;
90       return this.getModi(element);
91     }
92   }
93 }
94
95
96   getModi(element: Employee) {
97     return 'La La';
98   }
```

Discuss true/false – Any Column specified in columns array with “()” will get this Value
Discuss – Why we use `//@ts-ignore;` here.

(vi) Observe the Output

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
Search by Nu...	Search by Na...	Search by Ge...	Search by Des...	Search by Full...	Search by Modi
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	La La
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	La La
2203	Imran	Male	Demonstrator L1	Thahir Imran	La La
Items per page: 3 1 - 3 of 6 < >					

(vii) Change the `getProperty()` function to specific Pipe Function for each Column as specified in the Column Definition Array

```
79  getProperty(element: {}, reference: string): string {
80
81      if (!reference.endsWith("(")) {
82          const value = reference.split(separator: '.').reduce((o: {}, a: string) => {
83              //@ts-ignore;
84              return o[a];
85          }, element) as string;
86          return value;
87      }
88      else {
89          reference = reference.substring(0, reference.indexOf('('));
90          //@ts-ignore;
91          return this[reference](element);
92      }
93  }
94 }
```

Discuss the importance of the following line

- (1) `reference = reference.substring(0, reference.indexOf('('));`
- (2) `return this[reference](element);`

Discuss alternative approaches in converting String reference of a Function into real executable function call in Type Script

Discuss the Importance of Java Reflection API in referencing Methods and Attributes in Java

(viii) Change the “getModi()” as follows,

```
97  getModi(element: Employee) {  
98      return 'Na Na';  
99  }
```

(ix) Observe the Output

Employee Table					
Number	Calling Name	Gender	Designation	Full Name	Modification
Search by Nu...	Search by Na...	Search by Ge...	Search by Des...	Search by Full...	Search by Modi
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	Na Na
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	Na Na
2203	Imran	Male	Demonstrator L1	Thahir Imran	Na Na
Items per page: 3 1 - 3 of 6					

(x) Change the “getModi()” as follows

```
96  
97  getModi(element: Employee) {  
98  
99      return element.number + '(' + element.callingname + ')';  
100  
101  }
```

(xi) Observe the Output




Employee Table					
Number	Calling Name	Gender	Designation	Full Name	Modification
Search by Nu...	Search by Na...	Search by Ge...	Search by Des...	Search by Full...	Search by Modi
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					


(xii) Correcting in Client Search

```
60
61 filterTable(): void {
62
63     const cserchdata = this.csearch.getRawValue();
64
65     this.data.filterPredicate = (employee: Employee, filter: string) => {
66         return (cserchdata.csnumber == null || employee.number.toLowerCase().includes(cserchdata.csnumber)) &&
67             (cserchdata.cscallingname == null || employee.callingname.toLowerCase().includes(cserchdata.cscallingname)) &&
68             (cserchdata.csgender == null || employee.gender.name.toLowerCase().includes(cserchdata.csgender)) &&
69             (cserchdata.csdesignation == null || employee.designation.name.toLowerCase().includes(cserchdata.csdesignation)) &&
70             (cserchdata.csname == null || employee.fullname.toLowerCase().includes(cserchdata.csname)) &&
71             (cserchdata.csmodi == null || employee.number.toLowerCase().includes(cserchdata.csmodi));
72     };
73
74     this.data.filter = 'xx';
75
76 }
```

```
61 filterTable(): void {
62
63     const cserchdata = this.csearch.getRawValue();
64
65     this.data.filterPredicate = (employee: Employee, filter: string) => {
66         return (cserchdata.csnumber == null || employee.number.toLowerCase().includes(cserchdata.csnumber)) &&
67             (cserchdata.cscallingname == null || employee.callingname.toLowerCase().includes(cserchdata.cscallingname)) &&
68             (cserchdata.csgender == null || employee.gender.name.toLowerCase().includes(cserchdata.csgender)) &&
69             (cserchdata.csdesignation == null || employee.designation.name.toLowerCase().includes(cserchdata.csdesignation)) &&
70             (cserchdata.csname == null || employee.fullname.toLowerCase().includes(cserchdata.csname)) &&
71             (cserchdata.csmodi == null || this.getModi(employee).toLowerCase().includes(cserchdata.csmodi));
72     };
73
74     this.data.filter = 'xx';
75
76 }
```

(xiii) Test Client Search on newly add Column

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
<input data-bbox="245 1435 402 1485" type="text" value="Search by Nu..."/>	<input data-bbox="443 1435 600 1485" type="text" value="Search by Na..."/>	<input data-bbox="641 1435 798 1485" type="text" value="Search by Ge..."/>	<input data-bbox="839 1435 995 1485" type="text" value="Search by Des..."/>	<input data-bbox="1037 1435 1193 1485" type="text" value="Search by Full..."/>	<input data-bbox="1235 1424 1391 1485" type="text" value="Search by Modi 2201"/>
2201	Ashan	Male	Demonstrator L1	Ashan Pathum	2201(Ashan)
Items per page: <input data-bbox="1059 1570 1161 1619" type="text" value="3"/> 1 - 1 of 1  					

Employee Table 					
Number	Calling Name	Gender	Designation	Full Name	Modification
<input data-bbox="245 1800 402 1850" type="text" value="Search by Nu..."/>	<input data-bbox="443 1800 600 1850" type="text" value="Search by Na..."/>	<input data-bbox="641 1800 798 1850" type="text" value="Search by Ge..."/>	<input data-bbox="839 1800 995 1850" type="text" value="Search by Des..."/>	<input data-bbox="1037 1800 1193 1850" type="text" value="Search by Full..."/>	<input data-bbox="1235 1789 1391 1850" type="text" value="Search by Modi ruk"/>
2202	Rukmal	Male	Demonstrator L1	Dileesha Rukmal	2202(Rukmal)
Items per page: <input data-bbox="1059 1935 1161 1984" type="text" value="3"/> 1 - 1 of 1 