

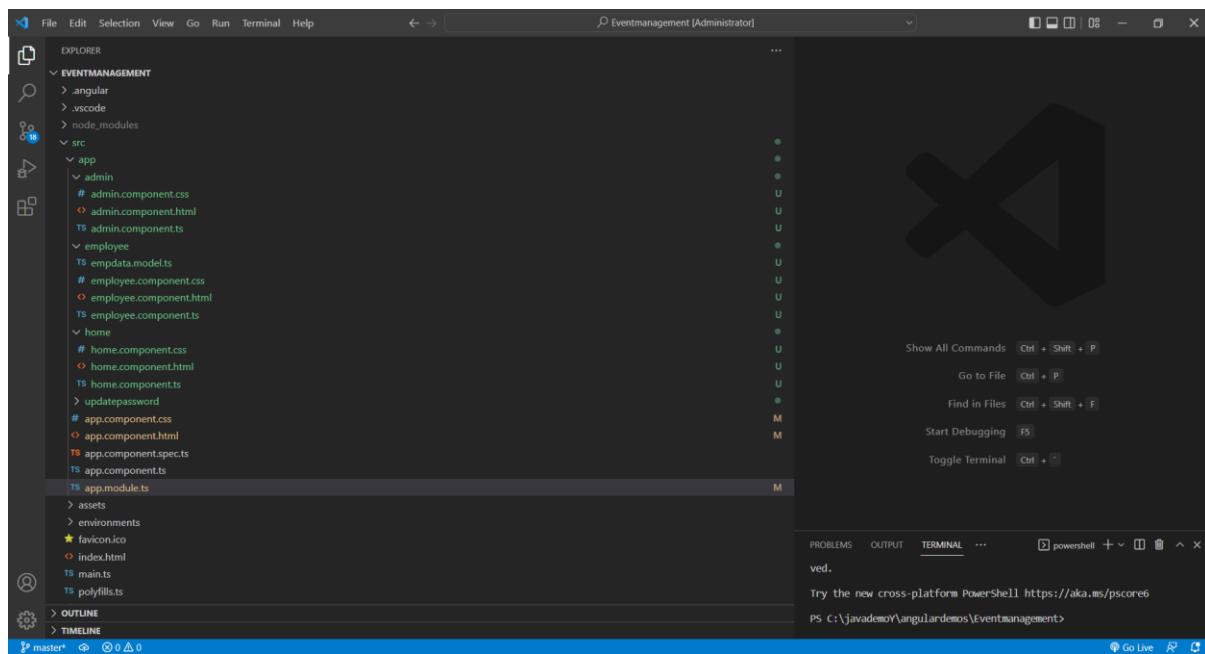
## Event Management App

Here , We used :

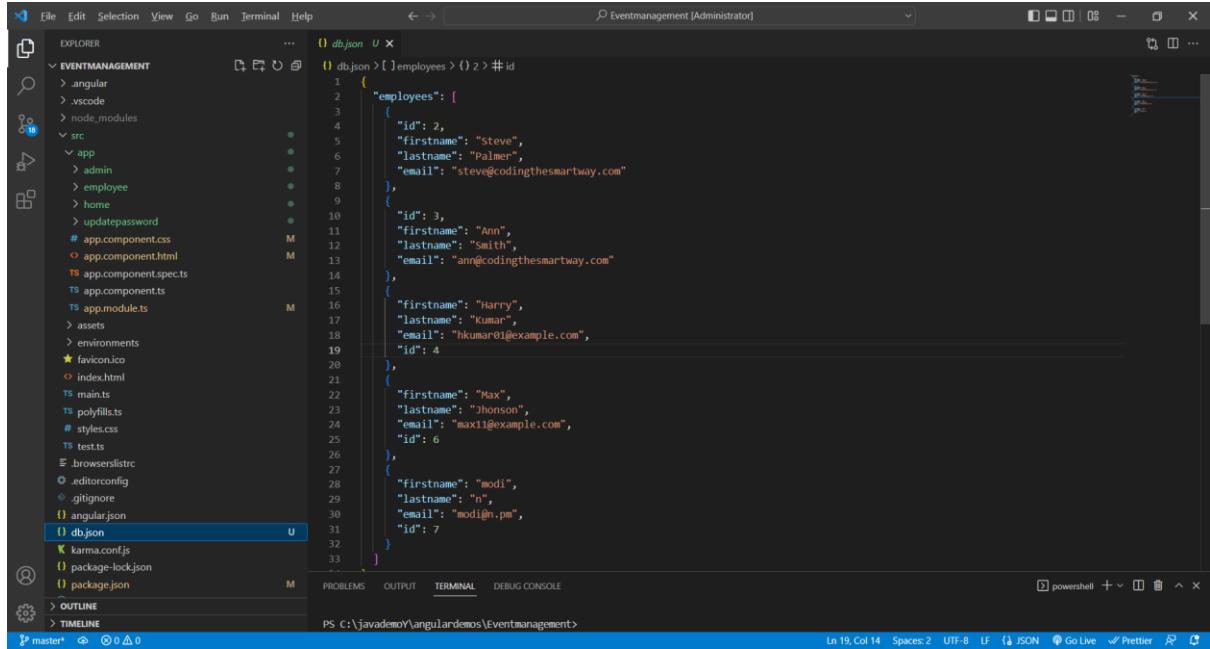
1. Angular
2. Boot-strap
3. JSON db(using Json-Server)

Here are some Screen Shots of Our app Code:

- Screen shot of App structure code:



- Screen shot of Db.json Employee( data inside the db )



The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure for "EVENTMANAGEMENT". The "db.json" file is selected and highlighted in blue.
- Editor:** Displays the contents of the "db.json" file. The JSON data represents an array of employees with fields: id, firstname, lastname, and email.
- Bottom Status Bar:** Shows the file path "C:\javademo\AngularDemos\Eventmanagement\db.json", line count "Ln 19, Col 14", and other status indicators like "Spaces: 2", "UTF-8", "LF", "JSON", "Go Live", and "Prettier".

```

db.json
[{"id": 1, "firstname": "Steve", "lastname": "Palmer", "email": "steve@codingthesmartway.com"}, {"id": 2, "firstname": "Ann", "lastname": "Smith", "email": "ann@codingthesmartway.com"}, {"id": 3, "firstname": "Harry", "lastname": "Kumar", "email": "hkumar01@example.com"}, {"id": 4, "firstname": "Max", "lastname": "Johnson", "email": "max12@example.com"}, {"id": 5, "firstname": "Modi", "lastname": "N", "email": "modi@pm"}, {"id": 6, "firstname": "John", "lastname": "Doe", "email": "john.doe@example.com"}]
  
```

- Here are screen shots of app.component code:

File Edit Selection View Go Run Terminal Help

Eventmanagement [Administrator]

```

src > app > ts app.module.ts > AppModule
  1 import { NgModule } from '@angular/core';
  2 import { FormsModule, ReactiveFormsModule } from '@angular/forms';
  3 import { BrowserModule } from '@angular/platform-browser';
  4 import { HttpClientModule } from '@angular/common/http';
  5
  6 import { AppComponent } from './app.component';
  7 import { HomeComponent } from './home/home.component';
  8 import { AdminComponent } from './admin/admin.component';
  9 import { UpdatepasswordComponent } from './updatepassword/updatepassword.component';
 10 import { EmployeeComponent } from './employee/employee.component';
 11 import { RouterModule, Routes } from '@angular/router';
 12
 13 const appRoutes: Routes = [
 14   { path: '', component: HomeComponent },
 15   { path: 'admin', component: AdminComponent },
 16   { path: 'employee', component: EmployeeComponent },
 17   { path: 'updatepassword', component: UpdatepasswordComponent }
 18 ];
 19
 20 @NgModule({
 21   declarations: [
 22     AppComponent,
 23     HomeComponent,
 24     AdminComponent,
 25     UpdatepasswordComponent,
 26     EmployeeComponent
 27   ],
 28   imports: [
 29     BrowserModule, FormsModule, ReactiveFormsModule, RouterModule.forRoot(appRoutes)
 30   ],
 31   providers: [],
 32   bootstrap: [AppComponent]
 33 })
 34 export class AppModule {}
```

Line 36, Col 27 Spaces: 2 UTF-8 LF TypeScript Go Live Prettier

File Edit Selection View Go Run Terminal Help

Eventmanagement [Administrator]

```

src > app > ts app.component.ts > ...
  1 import { Component } from '@angular/core';
  2
  3 @Component({
  4   selector: 'app-root',
  5   templateUrl: './app.component.html',
  6   styleUrls: ['./app.component.css']
  7 })
 8 export class AppComponent {
 9   title = 'Eventmanagement';
10 }
```

Line 1, Col 1 Spaces: 2 UTF-8 LF TypeScript Go Live Prettier

File Edit Selection View Go Run Terminal Help

src > app > app.component.html > div.container > div.row > div.col1

```

1 <div class="container">
2   <div class="row">
3     <div class="col1">
4       <ul class="nav nav-tabs">
5         <li role="presentation" class="nav-item" routerLinkActive="active" [routerLinkActiveOptions] = { exact: true }>
6           <a class="nav-link" routerLink="/" style="text-decoration: none">
7             <button>Home</button></a>
8         </li>
9         <br />
10        <li role="presentation" class="nav-item" routerLinkActive="active">
11          <a class="nav-link" routerLink="admin" style="text-decoration: none">
12            <button>Admin Login</button></a>
13        </li>
14        <br />
15        <li role="presentation" class="nav-item" routerLinkActive="active">
16          <a class="nav-link" routerLink="updatepassword" style="text-decoration: none">
17            <button>Change/Update Password</button></a>
18        </li>
19      </ul>
20    </div>
21  </div>
22  <div class="row">
23    <div class="col2">
24      <router-outlet></router-outlet>
25    </div>
26  </div>
27
```

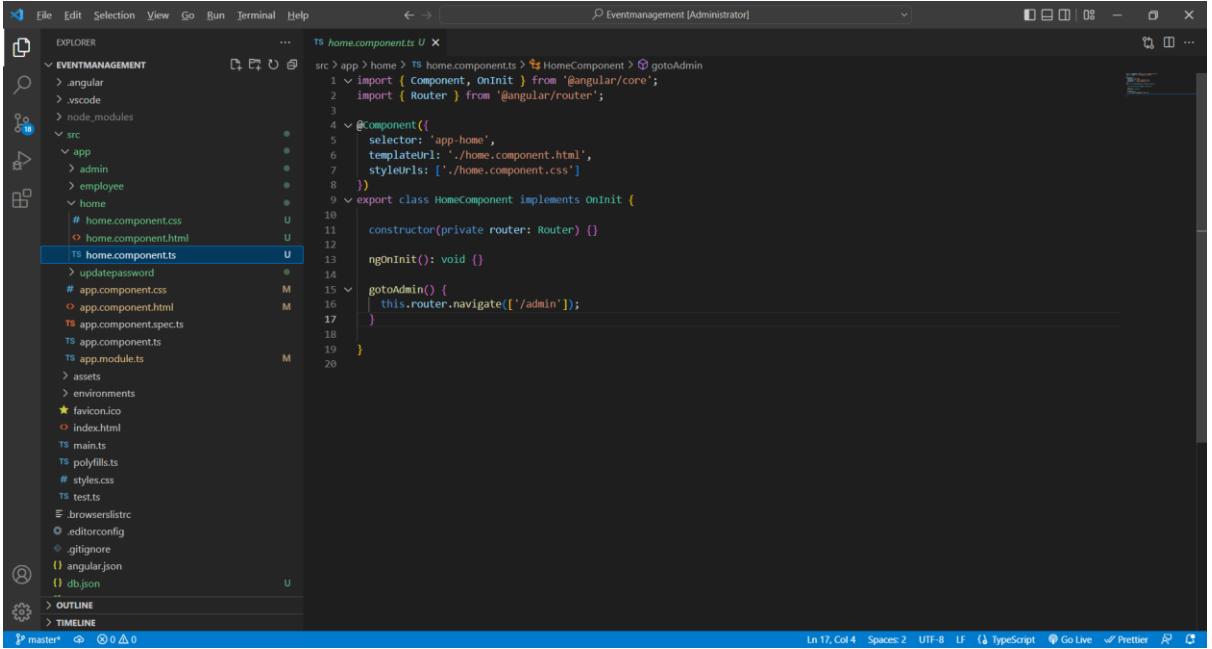
File Edit Selection View Go Run Terminal Help

src > app > # app.component.css > col1

```

1 ul {
2   list-style-type: none;
3 }
4 .container{
5   background-color: #rgb(213, 225, 232);
6   width: 100%;
7   margin: 5px;
8   padding: 5px;
9 }
10 }
11 .col1{
12   background-color: #rgb(167, 234, 239);
13   padding: 5px;
14   margin: 5px;
15 }
```

- Here screen Shots of home Component code:

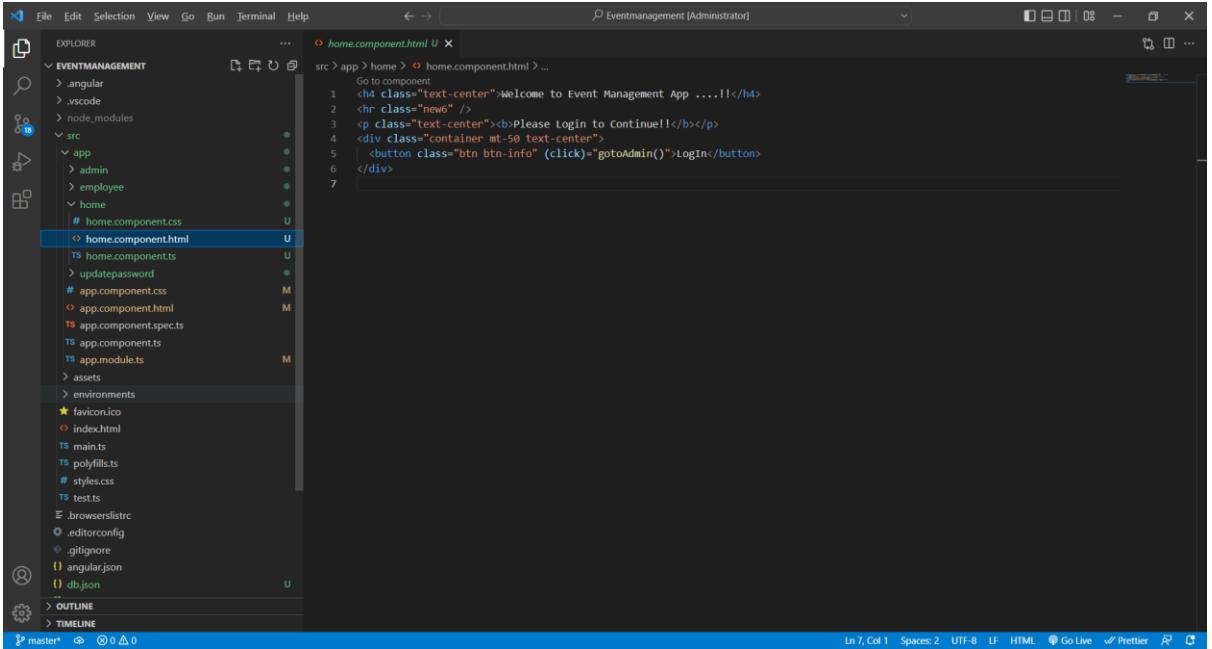


```

src > app > home > TS home.component.ts > HomeComponent > gotoAdmin
1 <import { Component, OnInit } from '@angular/core';
2 <import { Router } from '@angular/router';
3
4 <@Component({
5   selector: 'app-home',
6   templateUrl: './home.component.html',
7   styleUrls: ['./home.component.css']
8 })
9 <export class HomeComponent implements OnInit {
10
11   constructor(private router: Router) {}
12
13   ngOnInit(): void {}
14
15   gotoAdmin() {
16     this.router.navigate(['/admin']);
17   }
18
19 }
20

```

Ln 17, Col 4 | Spaces: 2 | UTF-8 | LF | TypeScript | Go Live | Prettier | R

```

src > app > home > home.component.html > ...
Go to component
1 <h4 class="text-center">Welcome to Event Management App .....!</h4>
2 <hr class="new6" />
3 <p class="text-center"><b>Please login to Continue!!</b></p>
4 <div class="container mt-50 text-center">
5   <button class="btn btn-info" (click)="gotoAdmin()">LogIn</button>
6 </div>
7

```

Ln 7, Col 1 | Spaces: 2 | UTF-8 | LF | HTML | Go Live | Prettier | R

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** On the left, it shows the project structure under "EVENTMANAGEMENT". The "src" folder contains "app", "admin", "employee", and "home". The "home" folder contains "home.component.css", "home.component.html", "home.component.ts", "updatepassword", "app.component.css", "app.component.html", "app.component.spec.ts", "app.component.ts", "app.module.ts", "assets", "environments", "favicon.ico", "index.html", "main.ts", "polyfills.ts", "styles.css", "test.ts", ".browserslistrc", ".editorconfig", ".gitignore", "angular.json", and "db.json".
- Editor:** The main editor area displays the content of "home.component.css".

```
src > app > home > # home.component.css > hr.new6
1 hr.new6 {
2   border: 8px solid #6D214F;
3   border-radius: 5px;
4 }
```
- Bottom Status Bar:** Shows "Line 4, Col 4" and "Spaces: 2, UTF-8, CRLF". It also includes icons for CSS, Go Live, Prettier, and a refresh symbol.

- Here are Screen shots of Amin Component:

```

src > app > admin > TS admin.components > AdminComponent
1  import { Component, OnInit } from '@angular/core';
2  import { FormControl, FormGroup, Validators } from '@angular/forms';
3  import { Router } from '@angular/router';
4
5  @Component({
6    selector: 'app-admin',
7    templateUrl: './admin.component.html',
8    styleUrls: ['./admin.component.css'],
9  })
10 export class AdminComponent implements OnInit {
11   constructor(private router: Router) {}
12
13   ngOnInit(): void {
14     this.signupForm = new FormGroup({
15       username: new FormControl(null, Validators.required),
16       password: new FormControl(null, [Validators.required]),
17     });
18   }
19
20   onSubmit() {
21     console.log(this.signupForm.value);
22     this.router.navigate(['/employee']);
23   }
24
25 }
26
27
28
29 }
30

```

```

src > app > admin > admin.component.html
1  <div class="container m-2 border justify-center" style="width: 400px; background-color: #rgb(205, 196, 196)">
2    <div class="text-center">Login Page</div>
3    <form [formGroup]="signupForm" (ngSubmit)="onSubmit()" class="form-center">
4      <div class="row">
5        <div class="form-group">
6          <label for="username">Username:</label>
7          <input type="text" id="username" class="form-control" formControlName="username" />
8          <span class="small" *ngIf="!signupForm.get('username').valid && signupForm.get('username').touched">Please enter a valid username</span>
9        </div>
10       </div>
11       <div class="row">
12         <div class="form-group">
13           <label for="password">Password:</label>
14           <input type="text" id="password" class="form-control" formControlName="password" />
15           <span class="small" *ngIf="!signupForm.get('password').valid && signupForm.get('password').touched">Please enter a valid password</span>
16         </div>
17       </div>
18     </form>
19   </div>
20

```

The screenshot shows two instances of the Visual Studio Code (VS Code) interface side-by-side, both displaying files from a project named "Eventmanagement".

**Top Window (admin.component.html):**

- File Explorer:** Shows the project structure under "EVENTMANAGEMENT".
- Editor:** Displays the content of "admin.component.html".
- Status Bar:** Shows "Ln 15, Col 9" and other standard status bar information.

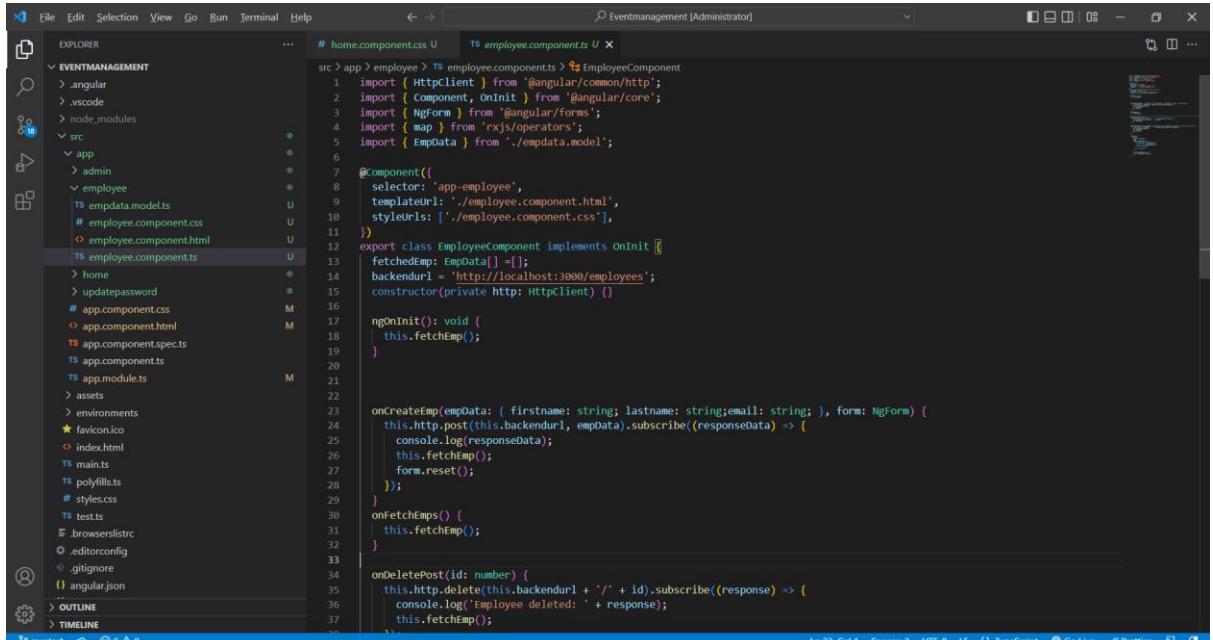
```
src > app > admin > admin.component.html <--> # admin.component.css U
  | formControlName="username"
  |   > span
  |     class="small"
  |       *ngIf="!signupForm.get('username').valid && signupForm.get('username').touched"
  |         > Please enter a valid username</span>
  |       </div>
  |     </div>
  |   <div class="row">
  |     <div class="form-group">
  |       <label for="password">Password</label>
  |       <input type="text" id="password" class="form-control" formControlName="password" />
  |       <span class="small" *ngIf="!signupForm.get('password').valid && signupForm.get('password').touched">Incorrect Password!</span>
  |     </div>
  |   </div>
  |   <button class="btn btn-primary m-2" type="submit">Submit</button>
</div>
```

**Bottom Window (admin.component.css):**

- File Explorer:** Shows the project structure under "EVENTMANAGEMENT".
- Editor:** Displays the content of "admin.component.css".
- Status Bar:** Shows "Ln 4, Col 3" and other standard status bar information.

```
src > app > admin > # admin.component.css ...
  input.ng-invalid.ng-touched {
    border: 1px solid red;
  }
```

- Here are Screen Shots of Admin Component:

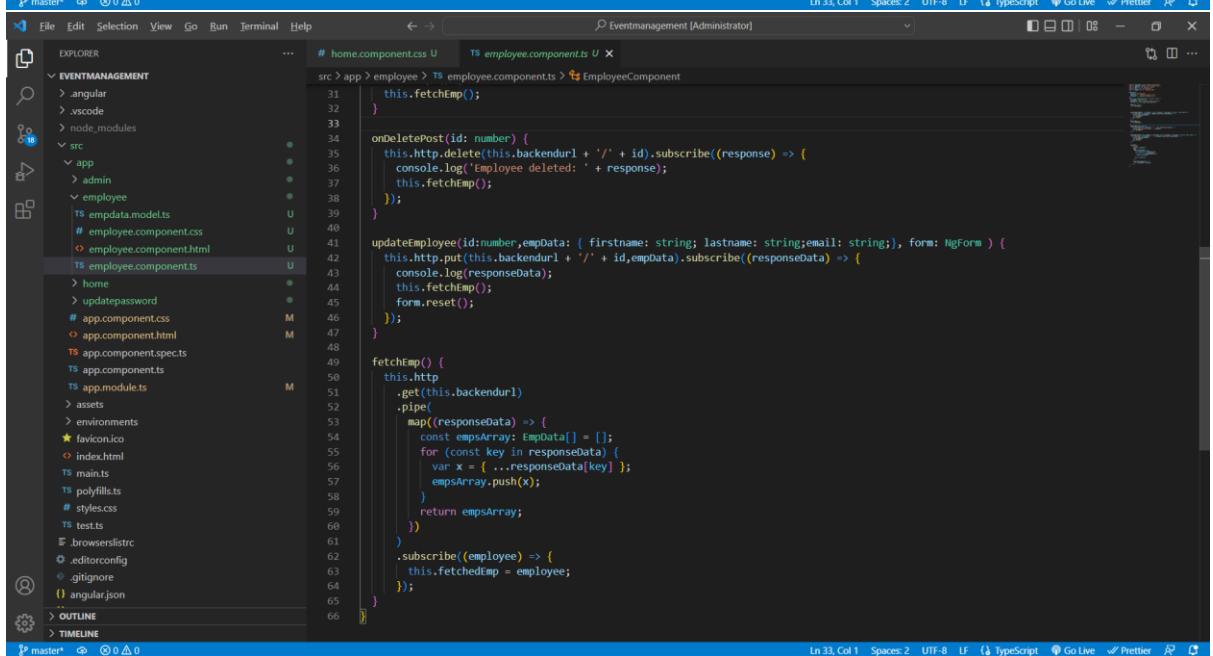


```

# home.component.css U   TS employee.component.ts U X
src > app > employee > TS employee.components > EmployeeComponent
1 import { HttpClient } from '@angular/common/http';
2 import { Component, OnInit } from '@angular/core';
3 import { NgForm } from '@angular/forms';
4 import { map } from 'rxjs/operators';
5 import { EmpData } from './empdata.model';
6
7 @Component({
8   selector: 'app-employee',
9   templateUrl: './employee.component.html',
10  styleUrls: ['./employee.component.css'],
11 })
12 export class EmployeeComponent implements OnInit {
13   fetchedEmp: EmpData[] = [];
14   backendurl = 'http://localhost:3000/employees';
15   constructor(private http: HttpClient) {}
16
17 ngOnInit(): void {
18   this.fetchEmp();
19 }
20
21
22
23 onCreateEmp(empData: { firstname: string; lastname: string; email: string; }, form: NgForm) {
24   this.http.post(this.backendurl, empData).subscribe((responseData) => {
25     console.log(responseData);
26     this.fetchEmp();
27     form.reset();
28   });
29 }
30
31 onFetchEmps() {
32   this.fetchEmp();
33 }
34
35 onDeletePost(id: number) {
36   this.http.delete(this.backendurl + '/' + id).subscribe((response) => {
37     console.log('Employee deleted: ' + response);
38     this.fetchEmp();
39   });
40
41 updateEmployee(id: number, empData: { firstname: string; lastname: string; email: string; }, form: NgForm) {
42   this.http.put(this.backendurl + '/' + id, empData).subscribe((responseData) => {
43     console.log(responseData);
44     this.fetchEmp();
45     form.reset();
46   });
47 }
48
49 fetchEmp() {
50   this.http
51     .get(this.backendurl)
52     .pipe(
53       map((responseData) => {
54         const empsArray: EmpData[] = [];
55         for (const key in responseData) {
56           var x = { ...responseData[key] };
57           empsArray.push(x);
58         }
59         return empsArray;
60       })
61     )
62     .subscribe((employee) => {
63       this.fetchedEmp = employee;
64     });
65   }
66 }

```

Ln 33, Col 1 | Spaces: 2 | UTF-8 | LF | TypeScript | Go Live | Prettier |

```

# home.component.css U   TS employee.component.ts U X
src > app > employee > TS employee.components > EmployeeComponent
1 this.fetchEmp();
2
3 onDeletePost(id: number) {
4   this.http.delete(this.backendurl + '/' + id).subscribe((response) => {
5     console.log('Employee deleted: ' + response);
6     this.fetchEmp();
7   });
8 }
9
10 updateEmployee(id: number, empData: { firstname: string; lastname: string; email: string; }, form: NgForm) {
11   this.http.put(this.backendurl + '/' + id, empData).subscribe((responseData) => {
12     console.log(responseData);
13     this.fetchEmp();
14     form.reset();
15   });
16 }
17
18 fetchEmp() {
19   this.http
20     .get(this.backendurl)
21     .pipe(
22       map((responseData) => {
23         const empsArray: EmpData[] = [];
24         for (const key in responseData) {
25           var x = { ...responseData[key] };
26           empsArray.push(x);
27         }
28         return empsArray;
29       })
30     )
31     .subscribe((employee) => {
32       this.fetchedEmp = employee;
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
61
62
63
64
65
66

```

Ln 33, Col 1 | Spaces: 2 | UTF-8 | LF | TypeScript | Go Live | Prettier |

File Edit Selection View Go Run Terminal Help

# home.component.css U employee.component.html 1.U

```
src > app > employee > employee.component.html > html > body > div > nav > div.navbar.navbar-dark.navbar-expand-sm > div.container > div#navbarNav2.collapse.navbar-collapse
```

Go to component

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <title>Event Management App</title>
7     <link rel="stylesheet"
8       href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"
9     />
10    </head>
```

<body>

```
11    <div style="background-color: #438cc4">
12      <nav>
13        <div class="navbar navbar-dark navbar-expand-sm" style="background-color: #a5a7cb">
14          <div class="container">
15            <div class="container2">
16              
17              
18            </div>
19          </div>
20        </div>
21        <div class="collapse navbar-collapse justify-content-end" id="navbarNav2">
22          <ul class="navbar-nav" style="color: #fffff">
23            <li class="nav-item">
24              <a class="nav-link" href="/employee#all" style="text-decoration: none">All Employee</a>
25            </li>
26            <li class="nav-item">
27              <a class="nav-link" href="/employee#new" style="text-decoration: none">New Employee</a>
28            </li>
29            <li class="nav-item">
30              <a class="nav-link" href="/employee#update" style="text-decoration: none">Update Employee</a>
31            </li>
32          </ul>
33        </div>
34      </div>
35      <button class="navbar-toggler" data-toggle="collapse" data-target="#navbarNav2" type="button">
36        <span class="navbar-toggler-icon"></span>
37      </button>
38    </div>
39  </div>
40  <div class="collapse navbar-collapse justify-content-end" id="navbarNav2">
41    <ul class="navbar-nav" style="color: #fffff">
42      <li class="nav-item">
43        <a class="nav-link" href="/employee#all" style="text-decoration: none">All Employee</a>
44      </li>
45      <li class="nav-item">
46        <a class="nav-link" href="/employee#new" style="text-decoration: none">New Employee</a>
47      </li>
48      <li class="nav-item">
49        <a class="nav-link" href="/employee#update" style="text-decoration: none">Update Employee</a>
50      </li>
51    </ul>
52  </div>
53</div>
54</div>
55</div>
56</div>
57</div>
58</div>
59</div>
60</div>
61</div>
62</div>
63</div>
64</div>
65</div>
66</div>
67</div>
68</div>
69</div>
```

Ln 54, Col 22 Spaces: 2 UTF-8 LF HTML Go Live Prettier

File Edit Selection View Go Run Terminal Help

# home.component.css U employee.component.html 1.U

```
src > app > employee > employee.component.html > html > body > div > nav > div.navbar.navbar-dark.navbar-expand-sm > div.container > div#navbarNav2.collapse.navbar-collapse
```

Go to component

```
33    <li class="nav-item">
34      <a class="nav-link" href="/employee#all" style="text-decoration: none">All Employee</a>
35    </li>
36    <button class="navbar-toggler" data-toggle="collapse" data-target="#navbarNav2" type="button">
37      <span class="navbar-toggler-icon"></span>
38    </button>
39  </div>
40  <div class="collapse navbar-collapse justify-content-end" id="navbarNav2">
41    <ul class="navbar-nav" style="color: #fffff">
42      <li class="nav-item">
43        <a class="nav-link" href="/employee#all" style="text-decoration: none">All Employee</a>
44      </li>
45      <li class="nav-item">
46        <a class="nav-link" href="/employee#new" style="text-decoration: none">New Employee</a>
47      </li>
48      <li class="nav-item">
49        <a class="nav-link" href="/employee#update" style="text-decoration: none">Update Employee</a>
50      </li>
51    </ul>
52  </div>
53</div>
54</div>
55</div>
56</div>
57</div>
58</div>
59</div>
60</div>
61</div>
62</div>
63</div>
64</div>
65</div>
66</div>
67</div>
68</div>
69</div>
```

Ln 54, Col 22 Spaces: 2 UTF-8 LF HTML Go Live Prettier

The screenshot shows the VS Code interface with the following details:

- File Explorer:** On the left, it shows the project structure under the "EVENTMANAGEMENT" folder. The "employee.component.html" file is currently selected.
- Code Editor:** The main area displays the HTML template for the Employee component. The code includes a navigation bar, a section for employee details, and a list group for employees.
- Status Bar:** At the bottom, it shows "Line 54, Col 22" and other status indicators like "Spaces: 2", "UTF-8", "LF", "HTML", "Go Live", and "Prettier".

```
<div class="row m-3" id="all">
<div class="col">
<h1 style="color: #ff00ff">All Employee Details</h1>
<p *ngIf="fetchedEmp.length < 1">No Employees Available !!</p>
<ul class="list-group" *ngIf="fetchedEmp.length > 1">
<li class="list-group-item border border-circle">
<ngFor let emp of fetchedEmp</ngFor>
<div style="width: 20%; background-color: #d2d2e2">
<p>{{ emp.firstname }}</p>
<p>{{ emp.lastname }}</p>
<p>{{ emp.email }}</p>
</li>
</ul>
</div>
<br class="new1" />
<div class="container m-2 justify-content-center" style="text-align: center">
```

```
# home.component.css U employee.component.html 1 U
src\app\employee> employee.component.html > html > body > div > nav > div.navbar.navbar-dark.navbar-expand-sm > div.container > div.navbarNav2.collapse.navbar-collapse > div > h1 style="color: #ffffff">Add New Employee</h1>
<form #ngform>
  <div class="row m-3">
    <div class="form-group align-center" style="color: #ffffff">
      <label for="firstname">First Name</label>
      <input style="background-color: #d2dae2" type="text" id="firstname" class="form-control" name="firstname" ngModel required>
    </div>
    <div class="row m-3">
      <div class="form-group" style="color: #ffffff">
        <label for="lastname">Last Name</label>
        <input style="background-color: #d2dae2" type="text" id="lastname" class="form-control" name="lastname" ngModel required>
      </div>
    </div>
  </div>
</form>
```

File Edit Selection View Go Run Terminal Help

# home.component.css U employee.component.html L U

```
src > app > employee > employee.component.html > html > body > div > nav > div.navbar.navbar-dark.navbar-expand-sm > div.container > div#navbarNav2.collapse.navbar > div
```

EXPLORER

- <\_ angular
- <\_ vscode
- <\_ node\_modules
- <\_ src
  - <\_ app
    - <\_ admin
    - <\_ employee
      - TS empdata.model.ts
      - # employee.component.css
      - employee.component.html 1, U
      - TS employee.component.ts
      - > home
      - > updatepassword
  - # app.component.css
  - o app.component.html M M
  - TS app.component.spec.ts
  - TS app.component.ts
  - TS app.module.ts M
  - > assets
  - > environments
  - ★ favicon.ico
  - o index.html
  - TS main.ts
  - TS polyfills.ts
  - # styles.css
  - TS test.ts
  - ! browserslistrc
  - o editorconfig
  - o .gitignore
  - ( angular.json
- <\_ OUTLINE
- <\_ TIMELINE

Ln 54, Col 22 Spaces: 2 UTF-8 LF HTML Go Live Prettier

File Edit Selection View Go Run Terminal Help

# home.component.css U employee.component.html L U

```
src > app > employee > employee.component.html > html > body > div > nav > div.navbar.navbar-dark.navbar-expand-sm > div.container > div#navbarNav2.collapse.navbar > div
```

EXPLORER

- <\_ angular
- <\_ vscode
- <\_ node\_modules
- <\_ src
  - <\_ app
    - <\_ admin
    - <\_ employee
      - TS empdata.model.ts
      - # employee.component.css
      - employee.component.html 1, U
      - TS employee.component.ts
      - > home
      - > updatepassword
  - # app.component.css
  - o app.component.html M M
  - TS app.component.spec.ts
  - TS app.component.ts
  - TS app.module.ts M
  - > assets
  - > environments
  - ★ favicon.ico
  - o index.html
  - TS main.ts
  - TS polyfills.ts
  - # styles.css
  - TS test.ts
  - ! browserslistrc
  - o editorconfig
  - o .gitignore
  - ( angular.json
- <\_ OUTLINE
- <\_ TIMELINE

Ln 54, Col 22 Spaces: 2 UTF-8 LF HTML Go Live Prettier

The screenshot shows the Visual Studio Code interface with the file `employee.component.html` open in the editor. The code is an Angular component template for deleting employees. It includes a modal dialog with buttons for 'Delete' and 'Update', a list group for employees, and a confirmation message at the bottom.

```
<button type="button" data-bs-toggle="modal" data-bs-target="#exampleModal" class="btn btn-info">Update</button>
</div>
</div>
</form>
</div>

<hr class="new1" />

<div class="row m-3" id="delete" style="width: 100%">
<div class="col">
<h1 style="color: #ffff00">Delete Employee</h1>
<p style="color: #ffff00" class="text-start" *ngIf="fetchedEmp.length < 1">
    No Employees Available !!
</p>
<ul class="list-group" *ngIf="fetchedEmp.length >= 1">
<li style="width: ; background-color: #d2dae2;" class="list-group-item" *ngFor="let emp of fetchedEmp">
        <p>{{ emp.firstname }}</p>
        <p>{{ emp.lastname }}</p>
        <p>{{ emp.email }}</p>
        <button class="btn btn-sm btn-danger float-right" (click)="onDeletePost(emp.id)">
            Delete
        </button>
    </li>
</ul>
</div>
<hr class="new1" />
</div>
</div>
```

The screenshot shows the Visual Studio Code interface with the file `employee.component.html` open in the editor. The code is identical to the one in the first screenshot, but the entire file content is highlighted in blue, indicating it has been selected or is being edited.

```
<button type="button" data-bs-toggle="modal" data-bs-target="#exampleModal" class="btn btn-info">Update</button>
</div>
</div>
</form>
</div>

<hr class="new1" />

<div class="row m-3" id="delete" style="width: 100%">
<div class="col">
<h1 style="color: #ffff00">Delete Employee</h1>
<p style="color: #ffff00" class="text-start" *ngIf="fetchedEmp.length < 1">
    No Employees Available !!
</p>
<ul class="list-group" *ngIf="fetchedEmp.length >= 1">
<li style="width: ; background-color: #d2dae2;" class="list-group-item" *ngFor="let emp of fetchedEmp">
        <p>{{ emp.firstname }}</p>
        <p>{{ emp.lastname }}</p>
        <p>{{ emp.email }}</p>
        <button class="btn btn-sm btn-danger float-right" (click)="onDeletePost(emp.id)">
            Delete
        </button>
    </li>
</ul>
</div>
<hr class="new1" />
</div>
</div>
```

The screenshot shows two instances of the Visual Studio Code (VS Code) interface. Both instances have the title bar "Eventmanagement [Administrator]" and the status bar "Ln 7, Col 17 | Spaces: 2 | UTF-8 | CRLF".

**Top Instance:**

- Explorer:** Shows the project structure under "EVENTMANAGEMENT".
- File:** Employee component CSS file: "# employee.component.css U".

```
src > app > employee > # employee.component.css > container2
1 hr.new {
2   border-top: 1px dashed black;
3 }
4 .container2{
5   align-items: center;
6
7   height: 25%;
8 }
```

**Bottom Instance:**

- Explorer:** Shows the project structure under "EVENTMANAGEMENT".
- File:** TypeScript model file: "empdata.model.ts U".

```
src > app > employee > empdata.models > EmpData
1 export interface EmpData {
2   id: number;
3   firstname: string;
4   lastname: string;
5   email: string;
6 }
```

- Here are screen Shots of Update Password;

The image shows two side-by-side screenshots of the Microsoft Edge browser running in a Windows environment. Both windows are displaying the same Angular application code within the Visual Studio Code code editor.

The top window displays the file `src/app/updatepassword/updatepassword.component.ts`. The code defines a component with a constructor that takes a `Router` dependency. It contains a `signupForm` `FormGroup` and methods `ngOnInit` and `onSubmit`. The `onSubmit` method logs the form value and navigates to the '/admin' route.

```
# home.component.css U TS updatepassword.component.ts U
src > app > updatepassword > TS updatepassword.component.ts > UpdatepasswordComponent > onSubmit
11     constructor(private router: Router) {}
12
13
14     signupForm: FormGroup;
15
16
17
18     ngOnInit(): void {
19         this.signupForm = new FormGroup({
20             username: new FormControl(null, Validators.required),
21             password: new FormControl(null, [Validators.required]),
22         });
23     }
24
25
26     onSubmit() {
27         console.log(this.signupForm.value);
28         this.router.navigate(['/admin']);
29     }
30
31
32 }
```

The bottom window displays the file `src/app/updatepassword/updatepassword.component.html`. It contains the template for the update password component, featuring two `form-group` sections for 'username' and 'password' with validation messages.

```
# home.component.css U TS updatepassword.component.html U
src > app > updatepassword > TS updatepassword.component.html > ...
Go to component
1 <div
2     class="container m-2 border justify-center"
3     style="width: 400px; background-color: #darkgrey"
4     >
5         <h4 class="text-center">Update Password</h4>
6         <form [formGroup]="signupForm" (ngSubmit)="onSubmit()">
7             <div class="row">
8                 <div class="form-group">
9                     <label for="username">Username</label>
10                    <input
11                        type="text"
12                        id="username"
13                        class="form-control"
14                        formControlName="username"
15                    />
16                    <span
17                        class="small"
18                        *ngIf="!
19                            signupForm.get('username').valid &&
20                            signupForm.get('username').touched
21                        "
22                        >Please enter a valid username:</span>
23                >
24            </div>
25        </div>
26        <div class="row">
27            <div class="form-group">
28                <label for="password">NewPassword</label>
29                <input
30                    type="text"
31                    id="password"
32                    class="form-control"
33                    formControlName="password"
34                />
35                <span
36                    class="small"
37                    *ngIf="!
38                        signupForm.get('password').valid &&
39                        signupForm.get('password').touched
40                "
41                >Please enter a valid password:</span>
42            </div>
43        </div>
44    </form>
45
```

File Edit Selection View Go Run Terminal Help

# home.component.css U    # updatepassword.component.html U

```

src > app > updatepassword > # updatepassword.component.html ...
1  <div>
2    <form>
3     <div>
4      <input type="text" id="username" formControlName="username"/>
5      <span class="small" *ngIf="!signupForm.get('username').valid && signupForm.get('username').touched">
6        Please enter a valid username</span>
7     </div>
8    </div>
9    <div class="row">
10    <div class="form-group">
11     <label for="password">New Password</label>
12     <input type="text" id="password" formControlName="password"/>
13     <span class="small" *ngIf="!signupForm.get('password').valid && signupForm.get('password').touched">
14      Incorrect Password!</span>
15    </div>
16    </div>
17    <button class="btn btn-primary m-2" type="submit">Update</button>
18  </div>
19 
```

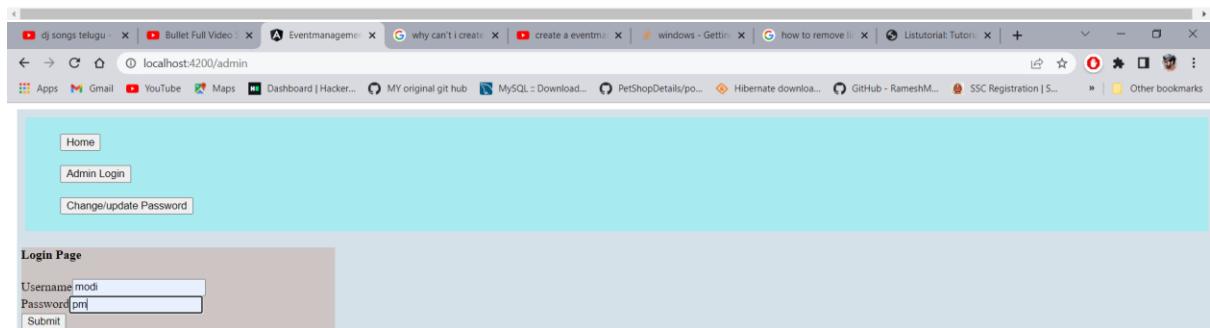
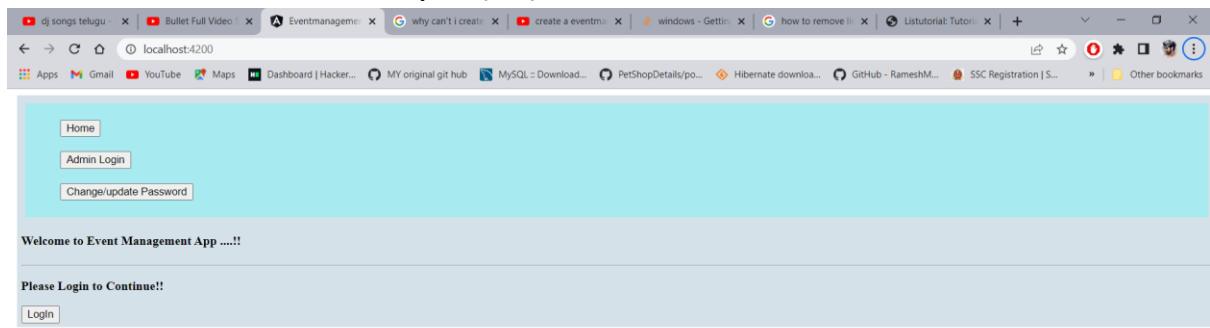
File Edit Selection View Go Run Terminal Help

# home.component.css U    # updatepassword.component.css U

```

src > app > updatepassword > # updatepassword.component.css ...
1  input.ng-invalid.ng-touched {
2    border: 1px solid red;
3 } 
```

## ❖ Here are output(UI) Screen Shots:



A screenshot of a web browser window showing a password update form. The URL in the address bar is `localhost:4200/updatepassword`. The page has a light blue header with three buttons: "Home", "Admin Login", and "Change/update Password". Below the header is a grey section titled "Update Password" containing three input fields: "Username modi", "NewPassword ornew", and a "Update" button.

A screenshot of a web browser window showing an employee management application. The URL in the address bar is `localhost:4200/employee/all`. The page has a purple header with two user icons (male and female) and the text "Employee Management App". Below the header is a sidebar with a list of navigation items: "All Employee", "New Employee", "Update Employee", and "Delete Employee". The main content area is titled "All Employee Details" and lists five employees in a table:

Employee	Email
Steve Palmer	steve@codingthesmartway.com
Ann Smith	ann@codingthesmartway.com
Harry Kumar	hkumar01@example.com
Max Jhonson	max11@example.com
modi	

Screenshot of a web browser showing three employee management pages:

- Add New Employee:** A form with fields for FirstName (amit), LastName (shah), Email (amit@shah.com), and an ADD button.
- Update Employee:** A form with fields for ID (redacted) and New Email (redacted), with an Update button.
- Delete Employee:** A list of employees with delete buttons:
  - Steve Palmer (steve@codingthesmartway.com)
  - Ann Smith (ann@codingthesmartway.com)
  - Harry Kumar (hkumar01@example.com, id: 4)
  - Max Johnson (max1@example.com, id: 5)

Below these pages is a screenshot of the Postman API testing tool:

**localhost:3000/employees**

**GET** localhost:3000/employees

KEY	VALUE	DESCRIPTION	Bulk Edit
Body	Pretty	Status: 200 OK Time: 40 ms Size: 936 B	Save Response
	Raw		
	Preview		
	Visualize		
	JSON		

```

1 {
2   "id": 2,
3   "firstname": "Steve",
4   "lastname": "Palmer",
5   "email": "steve@codingthesmartway.com"
6 },
7 {
8   "id": 3,
9   "firstname": "Ann",
10  "lastname": "Smith",
11  "email": "ann@codingthesmartway.com"
12 },
13 {
14   "firstname": "Harry",
15   "lastname": "Kumar",
16   "email": "hkumar01@example.com",
17   "id": 4
18 },
19 {
20   "firstname": "Max",
21   "lastname": "Johnson",
22   "email": "max1@example.com",
23 }

```

The screenshot shows a web browser window with a title bar containing multiple tabs. The active tab is titled "Delete Employee". The main content area displays a list of employees with their names, last names, and email addresses. Each entry includes a "Delete" button.

Name	Last Name	Email	Action
Steve	Palmer	steve@codingthesmartway.com	Delete
Ann	Smith	ann@codingthesmartway.com	Delete
Harry	Kumar	hkumar01@example.com	Delete
Max	Jhonson	max11@example.com	Delete
modi	n		

I added the code in separate pdf.