SOURCE CODE- EVENT MANAGEMENT APP

Changepassword.component.html

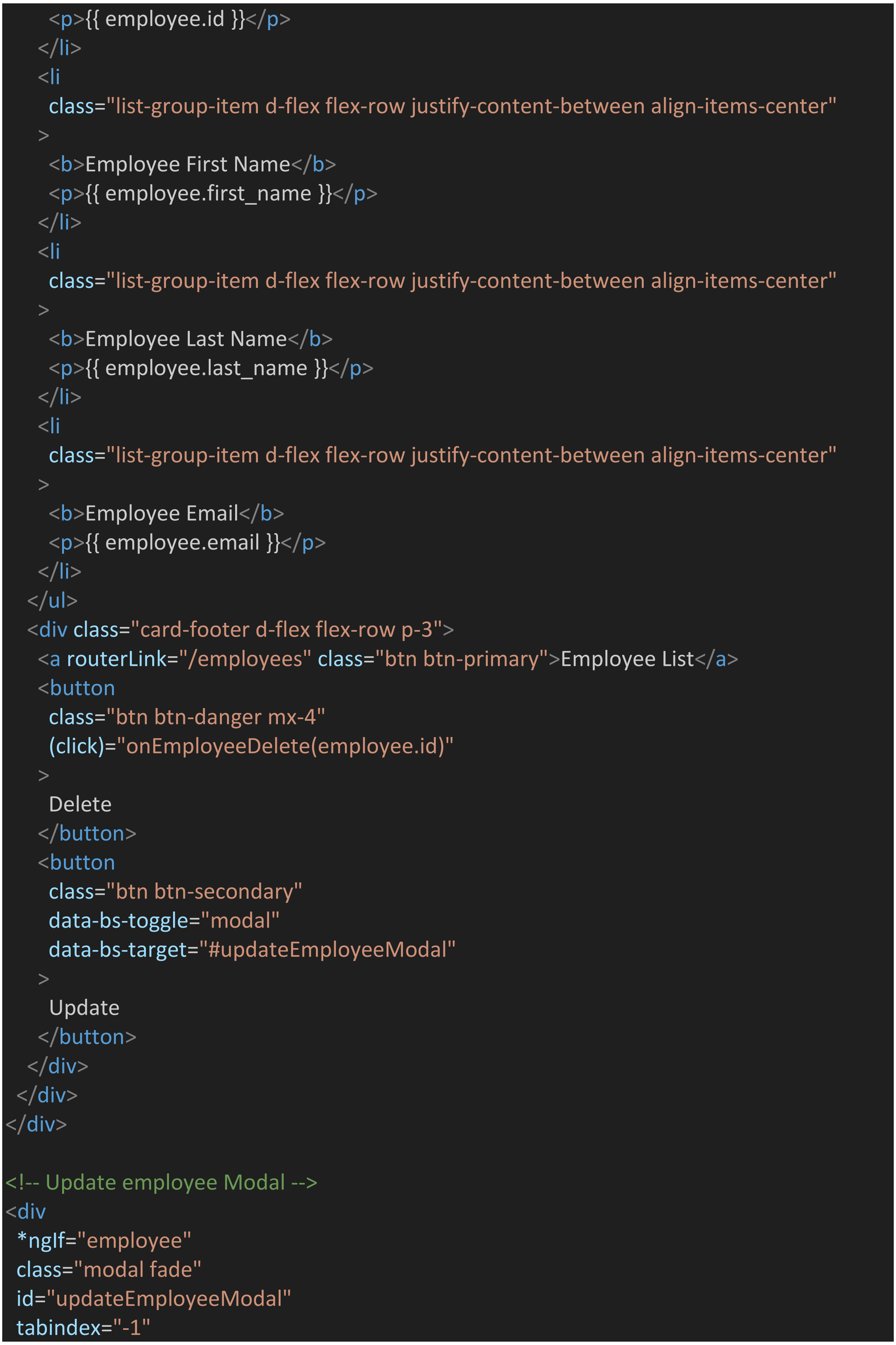
```
<div class="w-100 mt-5 container">
  <form
    #changePasswordForm="ngForm"
    (ngSubmit)="changePasswordForm.form.valid && onPasswordChange()"
    class="w-25 mx-auto"
  >
    <h4>Change Password</h4>
    <div class="my-3">
      <label for="oldPassword" class="form-label">Old Password</label>
      <input
        type="password"
        class="form-control"
        name="oldPassword"
        id="oldPassword"
        placeholder="Old password"
        [(ngModel)]="oldPassword"
        required
      />
    </div>
    <div class="my-3">
      <label for="newPassword" class="form-label">New Password</label>
      <input
        type="password"
        class="form-control"
        name="newPassword"
        id="newPassword"
        placeholder="New password"
        [(ngModel)]="newPassword"
        required
      />
    </div>
    <button type="submit" class="btn btn-primary">
      Submit
    </button>
  </form>
</div>
```

Changepassword.component.ts

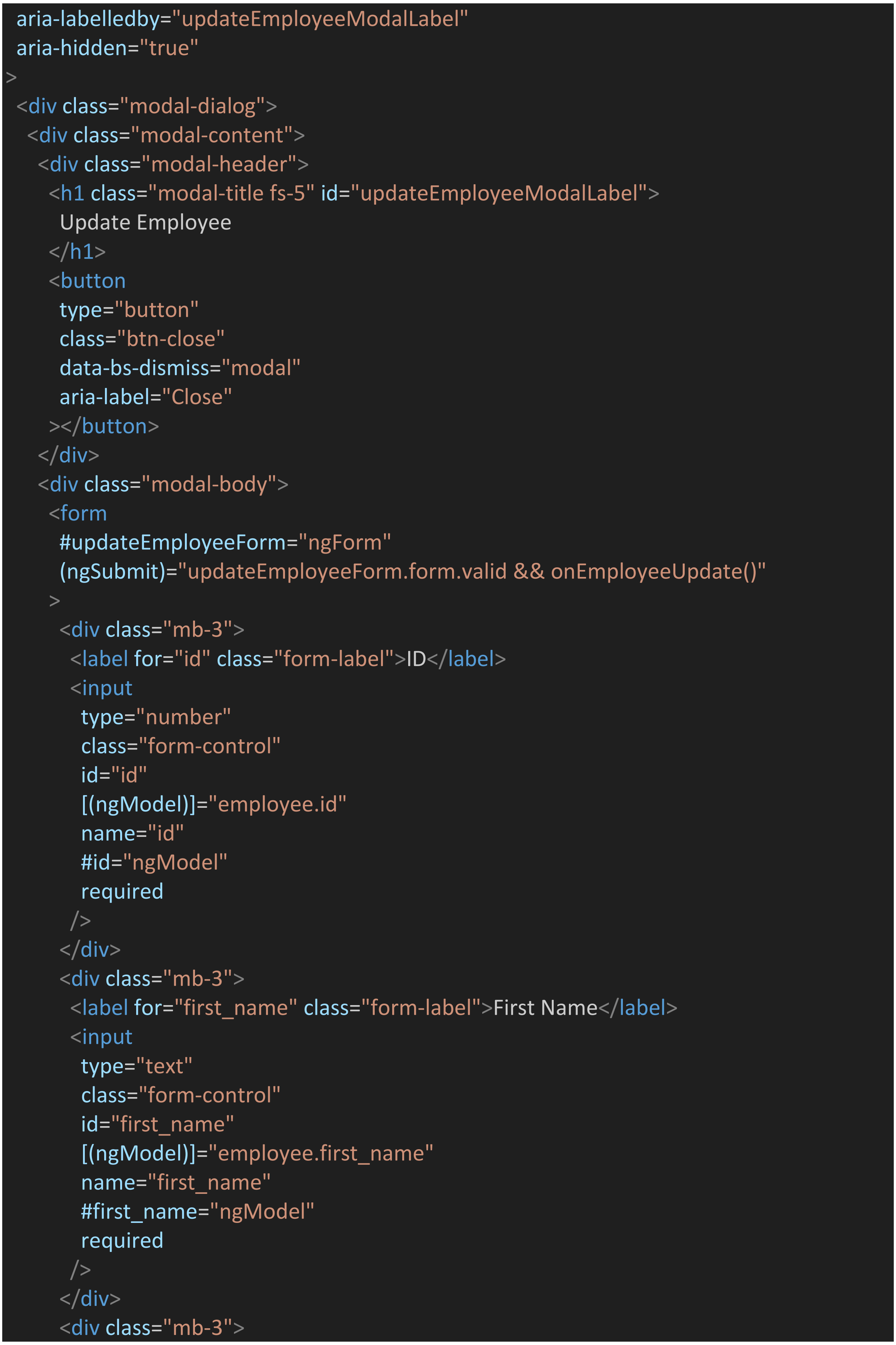
```
import { Component, OnInit } from '@angular/core';
import { EmployeeService } from '../employee.service';

@Component({
  selector: 'app-changepassword',
  templateUrl: './changepassword.component.html',
  styleUrls: ['./changepassword.component.css']
})
export class ChangepasswordComponent implements OnInit {

  public oldPassword:string = "";
  public newPassword:string = "";

  constructor(private employeeService: EmployeeService) {

  }

  ngOnInit(): void {

  }

  onPasswordChange() {
    if(this.oldPassword === this.employeeService.adminPassword) {
      this.employeeService.adminPassword = this.newPassword;
      this.oldPassword = "";
      this.newPassword = "";
      alert("Password Updated");
    }
    else {
      alert("Incorrect old password");
    }
  }

}
```

Employee.component.html

```
<div class="w-100 mt-4 container">
  <div class="card" *ngIf="employee">
    <div class="card-header p-3">
      Details of <b>{{ employee.first_name }} {{ employee.last_name }}</b>
    </div>
    <ul class="list-group list-group-flush">
      <li
        class="list-group-item d-flex flex-row justify-content-between align-items-center"
      >
        <b>Employee Id</b>
```



```
        <p>{{ employee.id }}</p>
      </li>
      <li
        class="list-group-item d-flex flex-row justify-content-between align-items-center"
      >
        <b>Employee First Name</b>
        <p>{{ employee.first_name }}</p>
      </li>
      <li
        class="list-group-item d-flex flex-row justify-content-between align-items-center"
      >
        <b>Employee Last Name</b>
        <p>{{ employee.last_name }}</p>
      </li>
      <li
        class="list-group-item d-flex flex-row justify-content-between align-items-center"
      >
        <b>Employee Email</b>
        <p>{{ employee.email }}</p>
      </li>
    </ul>
    <div class="card-footer d-flex flex-row p-3">
      <a routerLink="/employees" class="btn btn-primary">Employee List</a>
      <button
        class="btn btn-danger mx-4"
        (click)="onEmployeeDelete(employee.id)"
      >
        Delete
      </button>
      <button
        class="btn btn-secondary"
        data-bs-toggle="modal"
        data-bs-target="#updateEmployeeModal"
      >
        Update
      </button>
    </div>
  </div>
</div>

<!-- Update employee Modal -->
<div
  *ngIf="employee"
  class="modal fade"
  id="updateEmployeeModal"
  tabindex="-1"
```



```
  aria-labelledby="updateEmployeeModalLabel"
  aria-hidden="true"
>
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h1 class="modal-title fs-5" id="updateEmployeeModalLabel">
          Update Employee
        </h1>
        <button
          type="button"
          class="btn-close"
          data-bs-dismiss="modal"
          aria-label="Close"
        ></button>
      </div>
      <div class="modal-body">
        <form
          #updateEmployeeForm="ngForm"
          (ngSubmit)="updateEmployeeForm.form.valid && onEmployeeUpdate()"
        >
          <div class="mb-3">
            <label for="id" class="form-label">ID</label>
            <input
              type="number"
              class="form-control"
              id="id"
              [(ngModel)]="employee.id"
              name="id"
              #id="ngModel"
              required
            />
          </div>
          <div class="mb-3">
            <label for="first_name" class="form-label">First Name</label>
            <input
              type="text"
              class="form-control"
              id="first_name"
              [(ngModel)]="employee.first_name"
              name="first_name"
              #first_name="ngModel"
              required
            />
          </div>
          <div class="mb-3">
```

```
            <label for="last_name" class="form-label">Last Name</label>
            <input
              type="text"
              class="form-control"
              id="last_name"
              [(ngModel)]="employee.last_name"
              name="last_name"
              #last_name="ngModel"
              required
            />
          </div>
          <div class="mb-3">
            <label for="email" class="form-label">Email</label>
            <input
              type="email"
              class="form-control"
              id="email"
              [(ngModel)]="employee.email"
              name="email"
              #email="ngModel"
              required
            />
          </div>
          <button type="submit" class="btn btn-primary" data-bs-dismiss="modal">
            Submit
          </button>
          <button
            type="button"
            class="mx-4 btn btn-secondary"
            data-bs-dismiss="modal"
          >
            Close
          </button>
        </form>
      </div>
    </div>
  </div>
</div>
```

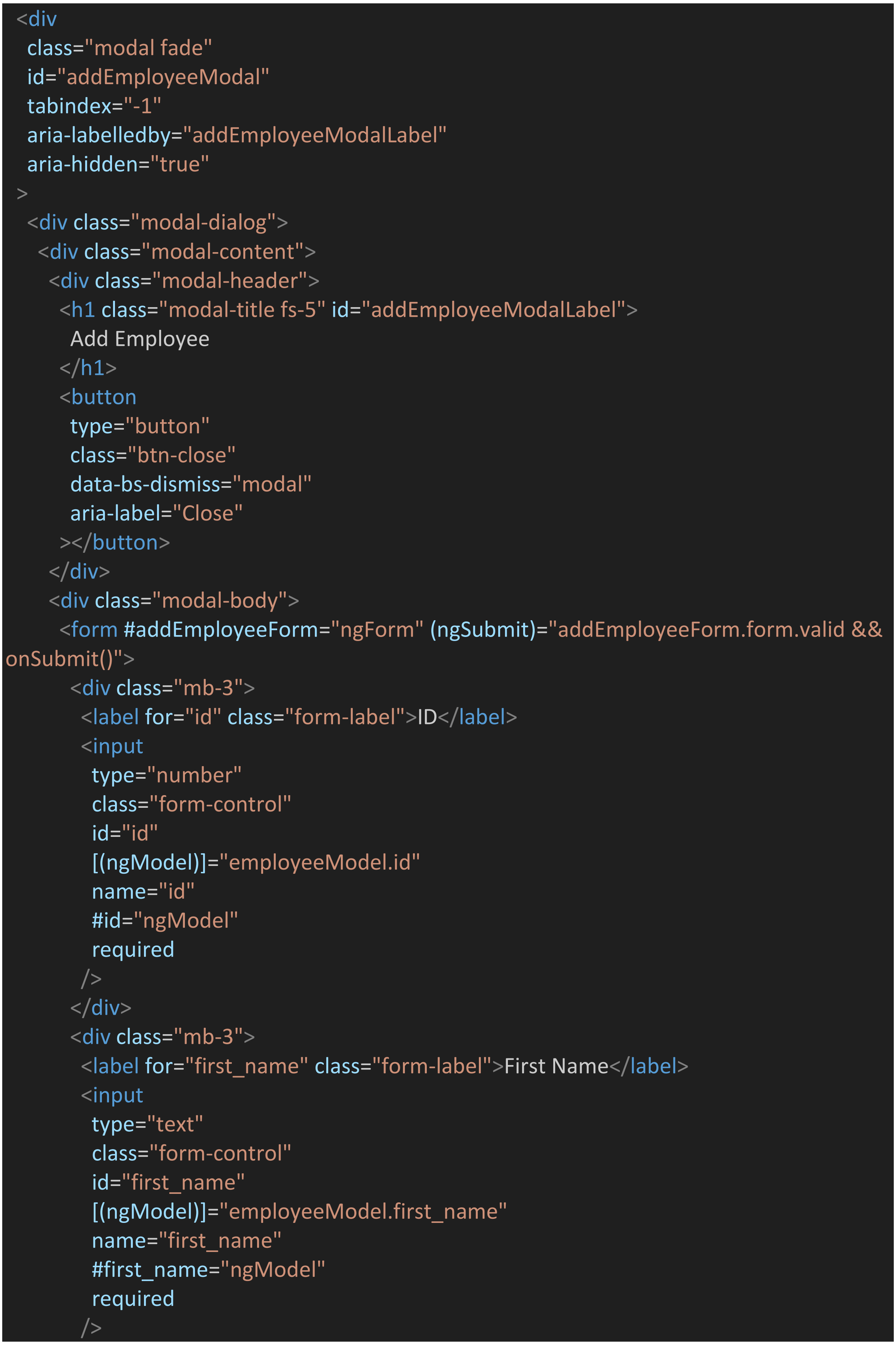
Employee.component.ts

```
import { Component, OnInit } from '@angular/core';
import Employee from '../models/Employee';
import { ActivatedRoute, Router } from '@angular/router';
import { EmployeeService } from '../employee.service';
```

```
@Component({
  selector: 'app-employee',
  templateUrl: './employee.component.html',
  styleUrls: ['./employee.component.css']
})
export class EmployeeComponent implements OnInit {
  public employee!: Employee;

  constructor(private employeeService: EmployeeService, private route: ActivatedRoute,
private router: Router) {

  }

  ngOnInit() {
    this.route.params.subscribe(params => {
      this.employeeService.getEmployee(+params["id"]).subscribe((data:Employee)=> {
        this.employee = data;
      })
    });

  }

  onEmployeeDelete(id:number) {
    this.employeeService.deleteEmployee(id).subscribe();
    this.router.navigate([`employees`]);
  }

  onEmployeeUpdate() {
    this.employeeService.updateEmployee(this.employee).subscribe((data:Employee)=> {
      console.log(`Employee updated: ${data}`);
    });
  }
}
```

Employees.component.html

```
<div class="mt-4 container">
  <button
    class="btn btn-secondary"
    data-bs-toggle="modal"
    data-bs-target="#addEmployeeModal"
  >
    Add New Employee
  </button>

  <!-- Modal -->
```



```
 <div
  class="modal fade"
  id="addEmployeeModal"
  tabindex="-1"
  aria-labelledby="addEmployeeModalLabel"
  aria-hidden="true"
 >
  <div class="modal-dialog">
   <div class="modal-content">
    <div class="modal-header">
     <h1 class="modal-title fs-5" id="addEmployeeModalLabel">
      Add Employee
     </h1>
     <button
      type="button"
      class="btn-close"
      data-bs-dismiss="modal"
      aria-label="Close"
     ></button>
    </div>
    <div class="modal-body">
     <form #addEmployeeForm="ngForm" (ngSubmit)="addEmployeeForm.form.valid &&
onSubmit()">
      <div class="mb-3">
       <label for="id" class="form-label">ID</label>
       <input
        type="number"
        class="form-control"
        id="id"
        [(ngModel)]="employeeModel.id"
        name="id"
        #id="ngModel"
        required
       />
      </div>
      <div class="mb-3">
       <label for="first_name" class="form-label">First Name</label>
       <input
        type="text"
        class="form-control"
        id="first_name"
        [(ngModel)]="employeeModel.first_name"
        name="first_name"
        #first_name="ngModel"
        required
       />
      </div>
      <div class="mb-3">
       <label for="last_name" class="form-label">Last Name</label>
       <input
        type="text"
        class="form-control"
        id="last_name"
        [(ngModel)]="employeeModel.last_name"
        name="last_name"
        #last_name="ngModel"
        required
       />
      </div>
      <div class="mb-3">
       <label for="email" class="form-label">Email</label>
       <input
        type="email"
        class="form-control"
        id="email"
        [(ngModel)]="employeeModel.email"
        name="email"
        #email="ngModel"
        required
       />
      </div>
      <button type="submit" class="btn btn-primary" data-bs-
dismiss="modal">Submit</button>
      <button
       type="button"
       class="mx-4 btn btn-secondary"
       data-bs-dismiss="modal"
      >
       Close
      </button>
     </form>
    </div>
   </div>
  </div>
 </div>

 <!-- Employee details table -->
 <table class="mt-4 table table-light">
  <thead>
   <tr>
    <th scope="col">#</th>
    <th scope="col">First Name</th>
    <th scope="col">Last Name</th>
    <th scope="col">Email</th>
    <th scope="col">Action</th>
   </tr>
  </thead>

  <tbody>
   <tr *ngFor="let employee of employees">
    <th scope="row">{{ employee.id }}</th>
    <td>{{ employee.first_name }}</td>
    <td>{{ employee.last_name }}</td>
    <td>{{ employee.email }}</td>
    <td>
     <button class="btn btn-secondary" (click)="onDetailsShow(employee.id)">View
Details</button>
    </td>
   </tr>
  </tbody>
 </table>
</div>
```

Employees.component.ts

```
import { Component, OnInit } from '@angular/core';
import Employee from '../models/Employee';
import { EmployeeService } from '../employee.service';
import { ActivatedRoute, Router } from '@angular/router';

@Component({
 selector: 'app-employees',
 templateUrl: './employees.component.html',
 styleUrls: ['./employees.component.css']
})
export class EmployeesComponent implements OnInit {

 public employees:Employee[] = [];
 public employeeModel = new Employee(1,"","","");

 constructor(private employeeService:EmployeeService, private route: ActivatedRoute,
private router: Router) {

 }

 ngOnInit(): void {
  this.employeeService.loadAllEmployees().subscribe((data:Employee[])=> {
   this.employees = data;
```

```
    })
  }

  onSubmit() {
    this.employeeService.addEmployee(this.employeeModel).subscribe((data:Employee)=> {
      this.employees.push(data);
    });
    this.employeeModel = new Employee(1,"","","");
  }

  onDetailsShow(id:number) {
    this.router.navigate([`employees`, id]);
  }

}
```

Login.component.html

```
<div class="w-100 mt-5 container">
  <form
    #loginForm="ngForm"
    (ngSubmit)="loginForm?.form?.valid && onLogin()"
    class="w-25 mx-auto"
  >
    <h4>Login</h4>
    <div class="my-3">
      <label for="email" class="form-label">Email</label>
      <input
        type="email"
        class="form-control"
        name="email"
        id="email"
        placeholder="Email"
        [(ngModel)]="email"
        required
      />
      <div
        *ngIf="
          loginForm.controls['email']?.invalid &&
          loginForm.controls['email']?.touched
        "
      >
        <span *ngIf="loginForm.controls['email'].errors?.['required']" class="mt-4 text-sm
text-danger"
          >Email is required.</span
        >
      </div>
    </div>
    <div class="my-3">
      <label for="password" class="form-label">Password</label>
      <input
        type="password"
        class="form-control"
        name="password"
        id="password"
        placeholder="Password"
        [(ngModel)]="password"
        required
      />
      <div
        *ngIf="
          loginForm.controls['password']?.invalid &&
          loginForm.controls['password']?.touched
        "
      >
        <span *ngIf="loginForm.controls['password'].errors?.['required']" class="text-sm text--
danger"
          >Password is required.</span
        >
      </div>
    </div>
    <button type="submit" class="btn btn-primary"
[disabled]="loginForm?.form?.invalid">Submit</button>
  </form>
</div>
```

Login.component.ts

```
import { Component, OnInit } from '@angular/core';
import Employee from '../models/Employee';
import { EmployeeService } from '../employee.service';
import { NgForm } from '@angular/forms';
import { Router } from '@angular/router';

@Component({
  selector: 'app-login',
  templateUrl: './login.component.html',
  styleUrls: ['./login.component.css']
})
export class LoginComponent implements OnInit {

  public email:string = "";
  public password:string = "";

  constructor(private employeeService: EmployeeService, private router: Router) {}

  ngOnInit(): void {

  }

  onLogin() {
    if(this.email === this.employeeService.adminEmail && this.password ===
this.employeeService.adminPassword) {
      localStorage.setItem("admin", this.email);
      this.email = "";
      this.password = "";
      this.router.navigate(['employees']);
      this.employeeService.setIsLoggedIn(true);
    }
    else {
      this.employeeService.setIsLoggedIn(false);
      alert("Incorrect credentials");
    }
  }

}
```

Navbar.component.html

```
<nav class="navbar navbar-dark navbar-expand-lg bg-dark">
  <div class="container d-flex justify-content-center">
    <a class="navbar-brand fw-bold" routerLink="/">Portal</a>
    <button
      class="navbar-toggler"
      type="button"
      data-bs-toggle="collapse"
      data-bs-target="#navbarNav"
      aria-controls="navbarNav"
      aria-expanded="false"
      aria-label="Toggle navigation"
    >
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNav">
      <ul class="navbar-nav d-flex align-items-center">
        <li class="nav-item" *ngIf="!isUser">
          <a class="nav-link" routerLink="/login" routerLinkActive="active"
            >Login</a
```

```
     >
    </li>
    <li class="nav-item" *ngIf="isUser">
     <a class="nav-link" routerLink="/employees" routerLinkActive="active"
      >Employees</a
     >
    </li>
    <li class="nav-item" *ngIf="isUser">
     <a
      class="nav-link"
      routerLink="/change-password"
      routerLinkActive="active"
      >Change Password</a
     >
    </li>
    <li class="nav-item" *ngIf="isUser">
     <a routerLink="#" class="nav-link" style="cursor: pointer" (click)="logoutUser()">
      Logout
     </a>
    </li>
   </ul>
  </div>
 </div>
</nav>
```

Navbar.component.ts

```
import { Component, OnInit } from '@angular/core';
import { EmployeeService } from '../employee.service';
import { Router } from '@angular/router';

@Component({
 selector: 'app-navbar',
 templateUrl: './navbar.component.html',
 styleUrls: ['./navbar.component.css']
})
export class NavbarComponent implements OnInit {

 public isUser:boolean = false;

 constructor(private employeeService:EmployeeService, private router:Router) {

 }

 ngOnInit(): void {
  this.employeeService.getIsLoggedIn().subscribe((value:boolean) => {
      this.isUser = value;
    });
  }

  logoutUser() {
    localStorage.removeItem("admin");
    this.employeeService.setIsLoggedIn(false);
    this.router.navigate(['login']);
  }

}
```

App-routing.modules.ts

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { EmployeesComponent } from './employees/employees.component';
import { LoginComponent } from './login/login.component';
import { EmployeeComponent } from './employee/employee.component';
import { ChangepasswordComponent } from
'./changepassword/changepassword.component';

const routes: Routes = [
  {
    path: "",
    redirectTo: "/login",
    pathMatch: "full"
  },
  {
    path: "employees",
    component: EmployeesComponent,
  },
  {
    path: "employees/:id",
    component: EmployeeComponent,
  },
  {
    path: "login",
    component: LoginComponent
  },
  {
    path: "change-password",
    component: ChangepasswordComponent
  },
];
```

```
@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```

employee.service.ts

```
import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';
import { BehaviorSubject, Observable, tap } from 'rxjs';
import Employee from './models/Employee';

@Injectable({
  providedIn: 'root',
})
export class EmployeeService {
  public adminEmail: string = 'mounika@gmail.com';
  public adminPassword: string = '12345';
  public employees: Employee[] = [];
  public isLoggedIn: BehaviorSubject<boolean> = new
BehaviorSubject<any>(localStorage.getItem("admin") ? true : false);

  constructor(private http: HttpClient) {}

  public loadAllEmployees = (): Observable<Employee[]> => {
    return this.http
      .get<Employee[]>('http://localhost:3000/employees');
  };

  public addEmployee = (employee: Employee): Observable<Employee> => {
    return this.http
      .post<Employee>('http://localhost:3000/employees', employee);
  };

  public getEmployee = (id:number):Observable<Employee>=> {
    return this.http
      .get<Employee>(`http://localhost:3000/employees/${id}`);
  }

  public deleteEmployee = (id:number):Observable<Employee>=> {
    return this.http
      .delete<Employee>(`http://localhost:3000/employees/${id}`);
  }

  public updateEmployee = (employee:Employee):Observable<Employee>=> {
    return this.http
      .put<Employee>(`http://localhost:3000/employees/${employee.id}`, employee);
  }

  public setIsLoggedIn(value: boolean): void {
    this.isLoggedIn.next(value);
  }

  public getIsLoggedIn(): Observable<boolean> {
    return this.isLoggedIn.asObservable();
  }
}
```

Data.json

```
{
  "employees": [
    {
      "id": 1,
      "first_name": "Sebastian",
      "last_name": "Eschweiler",
      "email": "sebastian@codingthesmartway.com"
    },
    {
      "id": 2,
      "first_name": "Steve",
      "last_name": "Palmer",
      "email": "steve@codingthesmartway.com"
    },
    {
      "id": 3,
      "first_name": "Ann",
      "last_name": "Smith",
      "email": "ann@codingthesmartway.com"
    }
    {
      "id": 4,
      "first_name": "mounika",
      "last_name": "mouni",
      "email": "mounika@gmail.com"
    }
    }
 }
```

Index.html

```
<!doctype html>
```

```html
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>EventManagement</title>
  <base href="/">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="icon" type="image/x-icon" href="favicon.ico">
</head>
<body>
  <app-root></app-root>
</body>
</html>
```

Note:

Commands to use:

```
cd <filename>
npm install
npm install -g json-server
npm run json-server
ng serve
```