**Code of the Project**

[**https://github.com/rishav1729/Assessment4/tree/master**](https://github.com/rishav1729/Assessment4/tree/master)

**For vs code like format:**

[**https://github1s.com/rishav1729/Assessment4/tree/master**](https://github1s.com/rishav1729/Assessment4/tree/master)

**D:\EventManagement\src\app\employee-dashboard\employee-dashboard.component.html**

<nav class="navbar navbar-light bg-primary">

    <div class="container-fluid">

      <h1 style="color: white">Event Management Employee Admin Panel</h1>

      <div class="d-fles">

        <button

        (click)="clickAddEmployee()"

          type="button"

          data-toggle="modal"

          data-target="#exampleModal"

          class="btn btn-success"

        >

          Add Employee

        </button>

      </div>

    </div>

  </nav>

  <table class="table mt-3">

    <thead>

      <tr>

        <th scope="col">Employee Id</th>

        <th scope="col">First Name</th>

        <th scope="col">Last Name</th>

        <th scope="col">Email Id</th>

        <th scope="col">Mobile No.</th>

        <th scope="col">Salary</th>

        <th scope="col">Action</th>

      </tr>

    </thead>

    <tbody>

      <tr \*ngFor="let temp of employeeData">

        <td>{{ temp.id }}</td>

        <td>{{ temp.firstName }}</td>

        <td>{{ temp.lastName }}</td>

        <td>{{ temp.email }}</td>

        <td>{{ temp.mobile }}</td>

        <td>{{ temp.salary }}</td>

        <td>

          <button

            type="button"

            data-toggle="modal"

            data-target="#exampleModal"

            class="btn btn-info"

            (click)="onEdit(temp)"

          >

            Edit

          </button>

          <button (click)="deleteEmployee(temp)" class="btn btn-danger mx-3">

            Delete

          </button>

        </td>

      </tr>

    </tbody>

  </table>

  <!-- Button trigger modal -->

  <!-- Modal -->

  <div

    class="modal fade"

    id="exampleModal"

    tabindex="-1"

    role="dialog"

    aria-labelledby="exampleModalLabel"

    aria-hidden="true"

  >

    <div class="modal-dialog" role="document">

      <div class="modal-content">

        <div class="modal-header">

          <h5 class="modal-title" id="exampleModalLabel">Employee Details</h5>

          <button

            type="button"

            class="close"

            data-dismiss="modal"

            aria-label="Close"

          >

            <span aria-hidden="true">&times;</span>

          </button>

        </div>

        <div class="modal-body">

          <form [formGroup]="formValue">

            <div class="form-group">

              <label for="exampleInputEmail1">First Name</label>

              <input

                type="text"

                class="form-control"

                id="exampleInputEmail1"

                aria-describedby="emailHelp"

                formControlName="firstName"

              />

            </div>

            <div class="form-group">

              <label for="exampleInputPassword1">Last Name</label>

              <input

                type="text"

                class="form-control"

                id="exampleInputPassword1"

                formControlName="lastName"

              />

            </div>

            <div class="form-group">

              <label for="exampleInputPassword1">Email</label>

              <input

                type="email"

                class="form-control"

                id="exampleInputPassword1"

                formControlName="email"

              />

            </div>

            <div class="form-group">

              <label for="exampleInputPassword1">Mobile No.</label>

              <input

                type="text"

                class="form-control"

                id="exampleInputPassword1"

                formControlName="mobile"

              />

            </div>

            <div class="form-group">

              <label for="exampleInputPassword1">Salary</label>

              <input

                type="text"

                class="form-control"

                id="exampleInputPassword1"

                formControlName="salary"

              />

            </div>

          </form>

        </div>

        <div class="modal-footer">

          <button

            type="button"

            id="cancel"

            class="btn btn-secondary"

            data-dismiss="modal"

          >

            Close

          </button>

          <button

            type="button"

            (click)="postEmployeeDetails()"

            class="btn btn-primary"

            \*ngIf="showAdd"

          >

            Add

          </button>

          <button

            type="button"

            (click)="updateEmployee()"

            class="btn btn-primary"

            \*ngIf="showUpdate"

          >

            Update

          </button>

        </div>

      </div>

    </div>

  </div>

**D:\EventManagement\src\app\employee-dashboard\employee-dashboard.component.ts**

import { Component, OnInit } from '@angular/core';

import { FormBuilder, FormGroup } from '@angular/forms';

import { EmployeeModel } from './employee-dashboard.model';

import { ApiService } from '../shared/api.service';

@Component({

  selector: 'app-employee-dashboard',

  templateUrl: './employee-dashboard.component.html',

  styleUrls: ['./employee-dashboard.component.css']

})

export class EmployeeDashboardComponent implements OnInit {

  formValue !: FormGroup;

  employeeModelObj : EmployeeModel = new EmployeeModel();

  employeeData : any;

  showAdd !: boolean;

  showUpdate !: boolean;

  constructor(private formbuiler: FormBuilder,

    private api : ApiService) { }

  ngOnInit(): void {

    this.formValue = this.formbuiler.group({

      firstName : [''],

      lastName : [''],

      email : [''],

      mobile : [''],

      salary : ['']

    })

    this.getAllEmployee();

  }

  clickAddEmployee()

  {

    this.formValue.reset();

    this.showAdd= true;

    this.showUpdate= false;

  }

  postEmployeeDetails()

  {

    this.employeeModelObj.firstName = this.formValue.value.firstName;

    this.employeeModelObj.lastName = this.formValue.value.lastName;

    this.employeeModelObj.email = this.formValue.value.email;

    this.employeeModelObj.mobile = this.formValue.value.mobile;

    this.employeeModelObj.salary = this.formValue.value.salary;

    this.api.postEmployee(this.employeeModelObj)

    .subscribe(res=>{

      console.log(res);

      alert("employee Added")

      let ref = document.getElementById('cancel')

      ref?.click();

      this.formValue.reset();

      this.getAllEmployee();

    },

    err=>{

      alert("Something went wrong")

    })

  }

  getAllEmployee(){

    this.api.getEmployee()

    .subscribe(res=>{

      this.employeeData = res;

    })

  }

  deleteEmployee(temp : any){

    this.api.deleteEmployee(temp.id)

    .subscribe(res=>{

      alert("employee Deleted");

      this.getAllEmployee();

    })

  }

  onEdit(temp : any)

  {

    this.showAdd= false;

    this.showUpdate= true;

    this.employeeModelObj.id = temp.id;

    this.formValue.controls['firstName'].setValue(temp.firstName);

    this.formValue.controls['lastName'].setValue(temp.lastName);

    this.formValue.controls['email'].setValue(temp.email);

    this.formValue.controls['mobile'].setValue(temp.mobile);

    this.formValue.controls['salary'].setValue(temp.salary);

  }

  updateEmployee()

  {

    this.employeeModelObj.firstName = this.formValue.value.firstName;

    this.employeeModelObj.lastName = this.formValue.value.lastName;

    this.employeeModelObj.email = this.formValue.value.email;

    this.employeeModelObj.mobile = this.formValue.value.mobile;

    this.employeeModelObj.salary = this.formValue.value.salary;

    this.api.updateEmployee(this.employeeModelObj, this.employeeModelObj.id)

    .subscribe(res=>{

      alert("Updated Successfully")

      let ref = document.getElementById('cancel')

      ref?.click();

      this.formValue.reset();

      this.getAllEmployee();

    })

  }

}

**D:\EventManagement\src\app\employee-dashboard\employee-dashboard.model.ts**

export class EmployeeModel{

    id : number = 0;

    firstName : string = '';

    lastName : string = '';

    email : string = '';

    mobile : string = '';

    salary : string = '';

}

**D:\EventManagement\src\app\login\login.component.css**

.card{

    border: none;

    width: 500px;

    padding: 40px;

    position: absolute;

    top: 50%;

    left: 50%;

    transform: translate(-50%,50%);

    background: hsla(205, 46%, 30%, 1);

    background: linear-gradient(90deg, hsla(205, 46%, 30%, 1) 0% ,hsla(205, 29%, 36%, 1)100%);

}

.card label h1,

h6{

    color:  #fff;

}

label{

    color: #fff;

}

**D:\EventManagement\src\app\login\login.component.html**

<div class="container">

  <div class="row">

       <div class="col-md-6">

          <div class="card">

              <div class="text-center">

                  <h1>Login</h1>

                  <h6>Please enter email & password !!</h6>

              </div>

              <form>

                  <form [formGroup]="loginForm" (ngSubmit)="login()">

                      <div class="mb-3">

                        <label for="exampleInputEmail1" class="form-label">Email address</label>

                        <input  formControlName="email" type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">

                        <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>

                      </div>

                      <div class="mb-3">

                        <label for="exampleInputPassword1" class="form-label">Password</label>

                        <input formControlName="password" type="password" class="form-control" id="exampleInputPassword1">

                      </div>

                      <button type="submit" class="btn btn-primary">Login</button>

                    </form>

              </form>

              <a style="color: #fff; margin: top 10px;" routerLink="/signup">New user? Click to signup!!</a>

          </div>

       </div>

  </div>

</div>

**D:\EventManagement\src\app\login\login.component.ts**

import { Component, OnInit } from '@angular/core';

import { FormBuilder , FormGroup, Validators} from '@angular/forms';

import { HttpClient } from '@angular/common/http';

import { Router } from '@angular/router';

@Component({

  selector: 'app-login',

  templateUrl: './login.component.html',

  styleUrls: ['./login.component.css']

})

export class LoginComponent implements OnInit {

  routerLinkActive="active";

  public loginform!:FormData;

  constructor(private formBuilder: FormBuilder,private http:HttpClient,private route:Router) { }

  email = '';

  password = '';

  submitted = false;

 loginForm: FormGroup=new FormGroup({});

  ngOnInit() {

    this.loginForm = this.formBuilder.group({

        email: ['', [Validators.required, Validators.email]],

        password: ['', [Validators.required, Validators.minLength(6)]]

    });

  }

  get f() { return this.loginForm.controls; }

  baseUrl = 'http://localhost:3000/signupUsers';

  login() {

    this.http.get<any>(this.baseUrl)

    .subscribe(res=>{

      const u=res.find((a:any)=>{

        return a.email===this.loginForm.value.email && a.password===this.loginForm.value.password

      })

      if(u){

        alert("Login success");

        this.loginForm.reset();

        this.route.navigate(['/dashboard'])

      }else{

        alert("Invalid Credentials")

      }

    })

  }

}

**D:\EventManagement\src\app\shared\api.service.ts**

import { Injectable } from '@angular/core';

import {HttpClient} from  '@angular/common/http';

import {map } from 'rxjs/operators'

@Injectable({

  providedIn: 'root'

})

export class ApiService {

  constructor(private http: HttpClient) { }

  postEmployee(data : any){

    return this.http.post<any>("http://localhost:3000/posts", data)

    .pipe(map((res:any)=>{

      return res;

    }))

  }

  getEmployee(){

    return this.http.get<any>("http://localhost:3000/posts")

    .pipe(map((res:any)=>{

      return res;

    }))

  }

  updateEmployee(data : any, id: number){

    return this.http.put<any>("http://localhost:3000/posts/"+id, data)

    .pipe(map((res:any)=>{

      return res;

    }))

  }

  deleteEmployee(id : number){

    return this.http.delete<any>("http://localhost:3000/posts/"+id)

    .pipe(map((res:any)=>{

      return res;

    }))

  }

}

**D:\EventManagement\src\app\signup\signup.component.css**

.card{

    border: none;

    width: 500px;

    padding: 40px;

    position: absolute;

    top: 50%;

    left: 50%;

    transform: translate(-50%,50%);

    background: hsla(205, 46%, 30%, 1);

    background: linear-gradient(90deg, hsla(205, 46%, 30%, 1) 0% ,hsla(205, 29%, 36%, 1)100%);

}

.card label h1,

h6{

    color:  #fff;

}

label{

    color: #fff;

}

**D:\EventManagement\src\app\signup\signup.component.html**

<div class="container">

    <div class="row">

         <div class="col-md-6">

            <div class="card">

                <div class="text-center">

                    <h1>Sign Up</h1>

                    <h6>Register Yourself !!</h6>

                </div>

                <form>

                    <form [formGroup]="signupForm" (ngSubmit)="signUp()">

                        <div class="mb-3">

                            <label for="exampleInputPassword1" class="form-label">Full Name</label>

                            <input formControlName="fullname" type="text" class="form-control" id="exampleInputPassword1">

                          </div>

                          <div class="mb-3">

                            <label for="exampleInputPassword1" class="form-label">Mobile Number</label>

                            <input  formControlName= "mobile" type="number" class="form-control" id="exampleInputPassword1">

                          </div>

                        <div class="mb-3">

                          <label for="exampleInputEmail1" class="form-label">Email Number</label>

                          <input  formControlName ="email" type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">

                          <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>

                        </div>

                        <div class="mb-3">

                          <label for="exampleInputPassword1" class="form-label">Password</label>

                          <input formControlName = "password"type="password" class="form-control" id="exampleInputPassword1">

                        </div>

                        <button type="submit" class="btn btn-primary">Sign Up</button>

                      </form>

                      <a  style="color: #fff; margin: top 10px;" routerLink="/login">Already registered? Click to Login!!</a>

                </form>

            </div>

         </div>

    </div>

</div>

**D:\EventManagement\src\app\signup\signup.component.ts**

import { HttpClient } from '@angular/common/http';

import { Component, OnInit } from '@angular/core';

import {FormGroup,FormBuilder} from '@angular/forms';

import { Router } from '@angular/router';

@Component({

  selector: 'app-signup',

  templateUrl: './signup.component.html',

  styleUrls: ['./signup.component.css']

})

export class SignupComponent implements OnInit {

  public signupForm!: FormGroup;

  constructor( private formBuilder : FormBuilder , private http: HttpClient , private router:Router) { }

  ngOnInit(): void {

    this.signupForm = this.formBuilder.group({

      fullname:[''],

      email:[''],

      password:[''],

      mobile:[''],

    })

  }

 signUp(){

      this.http.post<any>("http://localhost:3000/signupUsers",this.signupForm.value)

      .subscribe(res=>{

        alert("signup Successfull");

        this.signupForm.reset();

        this.router.navigate(['login']);

      },err=>{

        alert("something went wrong")

      })

 }

}

**D:\EventManagement\src\app\app-routing.module.ts**

import { NgModule } from '@angular/core';

import { RouterModule, Routes } from '@angular/router';

import { EmployeeDashboardComponent } from './employee-dashboard/employee-dashboard.component';

import { LoginComponent } from './login/login.component';

import { SignupComponent } from './signup/signup.component';

const routes: Routes = [

  {path:'', redirectTo:'login', pathMatch:'full'},

  {path:'login', component:LoginComponent},

  {path:'signup',component:SignupComponent },

  {path:'dashboard', component:EmployeeDashboardComponent}

];

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

**D:\EventManagement\src\app\app.module.ts**

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { AppRoutingModule } from './app-routing.module';

import { AppComponent } from './app.component';

import { FormsModule,FormBuilder, ReactiveFormsModule } from '@angular/forms';

import { HttpClient, HttpClientModule } from '@angular/common/http';

import { EmployeeDashboardComponent } from './employee-dashboard/employee-dashboard.component';

import { LoginComponent } from './login/login.component';

import { SignupComponent } from './signup/signup.component';

import { RouterModule } from '@angular/router';

@NgModule({

  declarations: [

    AppComponent,

    EmployeeDashboardComponent,

    LoginComponent,

    SignupComponent,

  ],

  imports: [

    BrowserModule,

    AppRoutingModule,

    ReactiveFormsModule,

    RouterModule,

    HttpClientModule,

  ],

  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule { }

**D:\EventManagement\src\app\app.component.html**

<router-outlet></router-outlet>

**D:\EventManagement\db.json**

{

  "posts": [

    {

      "id": 3,

      "firstName": "Khushi",

      "lastName": "kumari",

      "email": "khusi@gmail.com",

      "mobile": "8457962130",

      "salary": "$ 3500"

    }

  ],

  "signupUsers": [

    {

      "fullname": "rishav",

      "email": "r@gmail.com",

      "password": "12345",

      "mobile": 9304201452,

      "id": 4

    },

    {

      "fullname": "Raghav",

      "email": "raghav@gmail.com",

      "password": "123",

      "mobile": 7412589630,

      "id": 5

    }

  ],

  "profile": {

    "name": "typicode"

  }

}