json-typings.d.ts

```
declare module "*.json" {
   const value: any;
   export default value;
}
```

db.json

app.module.ts

```
import { NgModule } from '@angular/core';
import { BrowserModule, provideClientHydration } from '@angular/platform-
browser';
```

```
import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { FormsModule, ReactiveFormsModule } from '@angular/forms';
import { RouterModule, Routes } from '@angular/router';
import { HttpClientModule } from '@angular/common/http';
import { HomepageComponent } from './homepage/homepage.component';
import { AdminLoginComponent } from './admin-login/admin-login.component';
import { ChangePasswordComponent } from './change-password/change-
password.component';
import { EmployeesComponent } from './employees/employees.component';
import { AddemployeesComponent } from './addemployees/addemployees.component';
import { ViewEmployeeComponent } from './view-employee/view-employee.component'
const routes:Routes=[
  {path: ",component:AppComponent},
  {path:"home",component:HomepageComponent},
  {path:"login",component:AdminLoginComponent},
  {path:"changepassword",component:ChangePasswordComponent},
  {path:"employees",component:EmployeesComponent},
  {path:"addemployee", component:AddemployeesComponent},
  { path: 'view/:id', component: ViewEmployeeComponent } // Define a route with a
parameter ':id'
@NgModule({
  declarations: [
    AppComponent,
    HomepageComponent,
    AdminLoginComponent,
    ChangePasswordComponent,
    EmployeesComponent,
    AddemployeesComponent,
    ViewEmployeeComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    FormsModule,
    HttpClientModule,
    ReactiveFormsModule,
    RouterModule.forRoot(routes)
```

```
],
providers: [
  provideClientHydration()
],
bootstrap: [AppComponent]
})
export class AppModule { }
```

app.component.html

```
<header class="top-bar">
  <a routerLink="/" routerLinkActive="active-link">Portal</a>&nbsp;&nbsp;&nbsp;
  <a routerLink="/home">Welcome page</a>&nbsp;&nbsp;&nbsp;
  </header>
  <router-outlet></router-outlet></router-outlet>
```

app.component.css

```
.active-link {
    font-weight: bold;
    color: white;
    text-decoration: none; /* Remove underline */
}
.top-bar {
    background-color: black;
    padding: 10px; /* Add padding for spacing */
}

a {
    text-decoration: none;
    margin-right: 10px; /* Adjust spacing between links */
    color: white; /* Set default text color */
}
```

app.component.ts

```
import { Component } from '@angular/core';

@Component({
   selector: 'app-root',
   templateUrl: './app.component.html',
```

```
styleUrl: './app.component.css'
})
export class AppComponent {
  title = 'EventManagement';
}
```

employeeservice.service.ts

```
import { HttpClient } from '@angular/common/http';
import { Injectable } from '@angular/core';
import { Observable } from 'rxjs';
@Injectable({
  providedIn: 'root'
export class EmployeeserviceService {
private url='http://localhost:3000/employees';
  constructor(private http:HttpClient) { }
  private savedPassword: string = 'admin';
  getPassword(): string {
    return this.savedPassword;
  setPassword(newPassword: string): void {
    this.savedPassword = newPassword;
addemployee(employee:any){
  return this.http.post(this.url,employee);
getEmployeeById(id: number): Observable<any> {
  const url = `${this.url}/${id}`;
  return this.http.get(url);
getemployee():Observable<any>{
  return this.http.get(this.url);
```

```
deletEmployee(id: number): Observable<any> {
   const url = `${this.url}/${id}`;
   return this.http.delete(url);
}

updateEmployee(id: number, updatedemployee: any): Observable<any> {
   const url = `${this.url}/${id}`;
   return this.http.put(url, updatedemployee);
}
}
```

homepage.component.html

```
<div class="container">
    <br>
<h1>Event Management App</h1>
By Priyanka M V
<h1>Welcome Admin!!!</h1>
<h3><a routerLink="/login"class="login-link">Click here to
login.</a>&nbsp;&nbsp;&nbsp;</h3>
<br>
<img
src="
VxWM0dKEBfUrr/kx7Rye9zWdDqXHb///8o29RZSKVaSah0WtJgU7wbJz0QGyjUzPAYJjhORZtuVMxrUMs
AH0EWJTarjvTPeiqybSzl3Pymh/N6YdMANlppTsttV8nDuej7mpv/lQsTJDH/mRvqW3ZnVNRiT7TGdygA
OGErMFrLwu0IGB3vXHG9doz1ixrYgl1HPYUAFRAACyozNGUnLVAPJEGaidujlN6JddV7ZNALGSHt5f85N
24+03vbg2fKfHGUXDS8seevouKSgNiGcdQnzcejke2kZi9vTDcAEwgAESH7kpDteX6scJLZW4LngBV8VK
/MW4xTWWHc+Pc10V1NQ5EAAAAVPG8ALUazpuNvS1M1LjgABRtVPkfKf40laGs7MUuAVVwhIiOCXp4dc3X
rk52TabrJgbEho6ETQkaxdrbmj6EekpCogMUAKDCZb3GiWaq5Wp3xuL3/2dkdIknwoJXETWdgLUHyp6vV
oqWkaXjhXHzrgIxtd4XrbHfrkKCMi53XvdGhV578mIlDW2LIZHijo7PmiUuaW7K6P1mxOFNhbnWBX7WaZ
6fYiavBb6QyPEW2RGmuWm7PW4hyZW7TocH/3b3zuqbtizu0NEiVRIB+OEyqkLZfd9FqaM6cPWs8wNNcic
+HT7WFSpVSzt3Arf82Ii2R6eXXcgDyp79JnNVmRmrI8/FxyKemuI66rm+SvZeXRVzYjzqJu8lakndYlbi
GlGQ4l6bPkz/Vf0tTPDRVzSrlAAAYn0lEQVR4nO2di38b1ZXHJdljRZoZSxohIMgaycISsiQSCUm2ZUsB
P0KJbfIwCMckJITiJDhpi0kIFEPSkEAMcbIhYUt2KV2WVxNIt9t9dJf+dXvuY+7cech2QSSjrc6nJZJmN
Jr56pxzf+fc07LL1bG0daxjHetYxzrWsY51rGMd61jH0taxjnVssxaOKff6FJxmSlix5RKeGo5PklfDHW
jElJHxkXBsPD5iAqJMBgKB8RiQGg4sVO/NuTnPsoHsyFB3YDhser0a6O4OVJXYVKA70B29J6fmOAPH6Q7
A/4ZipmCLAazubDyehX+679HJOcuqUVcWWIF1D48z50JpTKl2Y0PIAhNmt/t7tPA4cquhcGwyi3jRYAsP
ZYeik4FuZoHJDiyXK5rFAQh8UIaaoLCiOC6xV8XjARyN0XBY+XsfEpXqeCCLEYGPTcToi8PEqQLgX4qrO
oESfXx4ITvVWlqq2tLD3QULDwXimEF4AXQCoaEMYW9C7oS34ICEKB1uKSzJ08qj3R2bCASQxygoz8erlF
Y1jrxJ86QY9bR4Cz9WlTxtB0tBCT4eBdWwwCV4EKrw8nhM2yuKgzKQbZ00BVQeqWVHu0sGIguGwe6FBfj
vxLAuHYYAlh51MYx00tWiMBSQW7UhLJdrJDaEE1J3XBelCgq8wBBjA/msO9AqVC6Cqi1hgS6FVA6e1V1l
NKYmQLXznhVvnYbXUHk8bTcYIhsJBEagABxnGWmKDoZazlKm0LOWeJaOqk1hKdFJVNzomjNKZdYQpYVkP
oiu1sgsVZWkNoYFuJSwKxrVU9R4APsWlEEK2rRA4ZmbOD/Y2hqWa2g8DnFGDLxscnxoBCst+HdkOEuLxO
```

y6h/gbDDxLal9YVeRI1HEUKHGUsBKeItqK+lg8i8rt1nwaBqW2LSxFb8Og7tYk0hCabMfMFsLK5EILWYF qUNsVVphUM9FwOIbqQNxFVuLEs/CWGGpxteazVC1dSe2msygBrKNgvBsfHiKeNKygggdy1vBEvLVFoarL OTaDpVQnXGEgNqGFG3GkbtS0mYIsFVdwBx514lv0iSo3DrZZGCow0A0rU1muJ0ocCzlcbAgrq1g8wLoRP 97at84J47ZettvEqjtLglPB7S1lcniqZXOtUvsKrCm+y66pBMtUTgu7yVL7Opaii4NAYHyyGq00xQNEKc Q2fvcPsLYuc8JD1FYgDpGG2qUxOhpOKtFqyyeh1TbuzIDFRrJEM7COQniEdpAhk7WYFmXVtrCoTA9w/QT mbS2sm4lpnZmWHvRuGmqBmqeb4wYB0TKT2t2xaE1jcCGsJzCs8VbCau/+qAvlc9zmM07aKHTaHqRoC8dE xqpdHSs8MpHVihvdcFEIOX9ipIWrjLTk3raOFWWVoI1nxc3rj36U6aza1bGi9sMeqYFa1bzCprNqI8cSk QnaMyWrt/b4vbLdhmmwv80EAT7C/KLUfo4lqIVyRC75tWvBczaEFqezqPIa+kHJXVQlKVSqm2i14eyXmO 7Njfb2ju4qUd9Shrqz2SztXzEFry1hY6vXBJcq6O64/kc0dhWLudGcERbHql0cS5B6ARWyIvvmq9FwbIK MfZNoqZoS1mvrwLD2vsHeUi29KVpqebR3Gr6PQf5FjlXbOJZQy/VSKxouPUtL6eGpqj7rhSMRe5sI7xvN 5TYDSwgVe6efP3Bw0ifpu3PJvW0cy6UyVr05PqkoU5oG5fpZnPpCsKani/7N0JJy0y8kEocOv8h9rKcdH StdpKgOHhwd5JNKzNQmhZhE/qXNEgrSgd7nj7wU2gws5dEjCWQ/Z8c3sGobx2KwDr78crK3rrJrDw+bG/ ALUWUoqxeGyuIiXP/RjaQEOqByDLPidva0o2PBd1ykrI4fhECc1k9ca5Vq5fMQSPewa4htDyMCixuywlU ApRXVvgqpTWGJJTwYHjz+yvT0dO9hLaxojyFL5ge72TojHY5Shctf2ghW6kQK7buGWB3TnNLIqm2iEMUh gXVyevrIoUSC5aAqXk40EgvHoqRralKjgrqaSMwsbbQ4K3UupbiE1C9mZmaOKfTYRlZt1fQT/Dk0IP7yV /0dQDlrTYCkgrQLDiBBxsoDpXa805BcPJ/Iry+q6uMKJY0vq6r7XTs0kri6vwr6qamLVRo7lQrVIrTeXO 3C6wkeFrq3ghSCeIVywDj5LKj5np4zb/QM9OR7ltXmI6JwdGbm7YF8/k3vqV+fyud7bngs9pNd109kUEW Loh/nYJ4ImxiMZxeGjcv71JWenoG33hvoAcuvfNGUlrKEYA30v00de+fUWUC72+RXbeZYmqXOJWbONVEC ivn+nH0I0tx7PdRebRaKwlQicXZg4Ozs2Nhvzs8iR3zXRKt9hkLerly4cOFoalO7CvsgCHtWLhLPQtya0 RJWz87CTrOzW8fef392FnxsJW3vWIIqbbIsv+eWSl3p7+/v6l9LbXzCgrobseo5c/mM5llAy/59yj7g0w OedX7s/OVzwG3A5FqaY8GgXKynhTbApaiPXtrWhezS09UNnEsQVhEduOz3L15HICgtm7wFA196Be/aM/v aWODioVn05IZk51i4Lt8lrzNWOMSU6s3+Lmr9XU/z7VElZYwvQfkCh+AZDOsIIvUBobXbdFCyUHs13wOu BZ610LZ17lcYVk/arukn/Pwl0C+jZcensNQOYLWN4dITl1A9dmKp6tKCQ1CU1X15EoIY1mX08IMPLYHIt JS0jPEArFmA9TGCNZBftRsKUycOHTT01BxqyhryKwarq38v9SYldW5mJqEIa1MpJL6jq8sreZqjPhhAOW sOx+GHlBY9HC86pd0ID87w57deniOwltkc9Nox5repx0EDqEtUcDis1CEjrK6bWH8q4SuXLiSWQExUT1z 9h303VvJ5iiqf370KFPIHrl1D4+GHH32EX33XIs+ldJ6kLHCt21uvXXsbweohUkuCMD2hl5apxcThaaDl 9DBUbiJCeygpgNZ/SRCU1NFLMD4eRXfbTy0mZhknILX8hSt1A677+uXLQGvgg4+qeFhckcyCE6UsDAtca 3Hr3OVFPDLkEShp7+rSYuJVJt8A1pFp5zuWcgU7lgaraxsw+j61hlD1X0HXUoVYfBu51cruG8tQ36Ecpi BVOvDx3Nzcx9cHP1T3YpB7zaykZQ3WwNm335+bu41h9exb3geD5FmoF2Znr9KJXGUpcWh6enRzPf17Z6n tCNYtPcG//o+/PTX+2iefXLq516T6q0uulEKNJnsMC2hdm5u7eP2f9kr/DFT0ZMRgYaQDWF3MAtnbeSI1 kJsO5E8sXZ2ZndVmvcNrJxafP7ymOJqWshe7E4PVv+Ohce8YmHdM6yfYLB4lsHp6Pr08d/HiB19KEgrE3 RbP2s1gDTx6ce43GixcUC6n1PDRq0yeKbGjb/76zd999vuYg39BKfUocqw9Gqz+S/dvRay2jnnXW0CqrJ BrHli5fv36mX+RpC+Rt6TN3ReyD2U19/7tvJb49q1i+akwRwp/9vkWYjs//1fn/nYSTu/bwIDUzUuXxr1 eIDXm9Q6vw0pY1TQEdpwPgNJHNklLWqGwBo4Aq8tjt/OgHFb2vftFLBUz2WdbONvZyoU6rTSVifeubRcS icVT3q3gVd6FdW/JEV51sLCrfIldy9JRkN6FYQHHHaCaOw+wzr322vm4ne3cYrAnHUrrpg4rcSExM3MeU GWj6y9pEOufnlkZ0Cy/kpZwFbjPkuH3vop968wrF+d+NzZ2u8er2ynusfcZI6stXzkTloLaDf3927bt2b Pn1i++9uLkHt0gaaiV0dHeb154410wN775piSRke+GRWh5PBjWe29dP3N7bOw8D+tNHtYfTLC2fOvMHwR K7X366Stre27dugW0bn3y210QsDZYvC2kI24Zyl5ioCXrkgfF4YqVFUluyP0MsF6bnZnhaJkdC+z3zkzy SD61U0Z6+OFte27t0QTJd4N3iIWgG0zWzC37PZIfsKTNrCBraXnNAOtUYubNr7YwWBZUO7d87kzXQpY6/ OAD2B55YMM5ZrBB2W0wGQIRdELeCusGGzVNYXgKpfRnmgQhovVtLEzNcT85dfoxageljTW0mnSbLAiudQ a0gyVprTBYb73zzndnzWnqDnp0h+DRB8Q//huyR5ntdRQtsRZkTlLfEJYYilhg1SVpX94CS9rL5MXAG5X KizwsEnt/gIRFKB1nsP7Ub7Y1B9ES+KuPbNg1EWuyGZa77JGWZmbMYainLIAFb3rPa6VFWe08qcHaaegX 4bLi0c1NpNwVU93c1UdCG7ZJZCusoE9aSiTWzJ61rIvXF93uihHWHS76dp5ksP7Y32Wy/qed41l0cNPis LwBLNVviUIchyf0Ja5aYdHCeeDMfoD1ggEWTlsU0R0d1r/3bzPDuuIcWK6g4boj6zeVVKkRtMKSB9Nrnh PnzLD8n/55BeX4lT9jZ3zWa6WF7RnvcQbrT+Yo7HJQzhJM+Tq4brtS9Uh1axRCHHskT9qSs3yR/fvdL77

ohv/awfIyVt47Gqzj/2GBta3qmC6XWDJevDy43qkBAbPKIv7ot9Y6kmByQgusOxorr/f4cQLrpBXWJcc4 ljG9Y1rrSC2EQNvf8LZgw1JFq6ZxU3Zf22q2wFc7vyKP/vM4zfTWwXC7YwZDwW90QeuMh6hi9jM+BsJ1y YzKZZL6EXnwId7uz2a5Z09SWP9lze+POwaWaMnXcq0ZLPzbOiFtf00qQxUPzwrvr5Y5WHJQfvZ+zh7Ker 1fc0//Gyet43+x5nfnKAc+V0ijZpPoZMKPwQ0aYqwsmd0KYPHfQ0SWKyM8ra+9Xv4pca3jNoOhg5TDoAV WxH6FGf2pCn0w70VdKyiZUbkEiS8i0d6VZ5FFsd3/5JP3P1mNMhP/B+f3/7XA6nKOcuDzO3WEiO2tANr6 BX3wnOZhJT2W37IQDPJ11E27Om68hwKwnuTax4q6s8lguM0x09SCxIUKdRU7pSWwFQq6J44a0jfRDhJ3Z QYFJzM/JKMtgcUtYf2WKId+C6ybjsnv/LevXY+d0mKs0mWdLR+HkZBkQJUCC92n220PsYcetE3EsKop/B jdG7qz2WDoIOUQssJyBy2exW62kdJ62BpgBQEWl+vU7Tt27Pj+AVtDm3bseB0bPHj88be+q5fKL+/ccvz kyb9YYR1yDCxeOfRqGJLmDM/d+e3XYy/yHA+rYXzT42DbH7S17Y9T2/7gz5546q8HKhWQFe6K+yTAshkM HaQcuJYDy0ERnzEOuZu4oNzTPdEAyyR1UXgVKnoYPsceVRpoWzhUOo2eVPRDnH715HHrYOgg5cDDek6DF WwYLtxwI2VI31/OcTW1GZYL3fjJNgcLFf1xXUTd1qBNW+z0y9ussBxURtfZxOef4v9HN+PNgVzYyoPc6G ACjIyDFZG4znVNFENJu2ochJilMuzqd0zK4mDJdQZLLnEXbmDlkThPlOuqPjTa6A2R07ucB8PR1WATVk8 9bFakD1IOAIslqnS0nTTXLTWyMnSzgj40nU1FKTKUQDKt7zlo7M3ysJ6wwHKQctBDRXbrVaLsZttNrHhY sjstpfVng1ZYDElSEvSiGvbUHdJo9z1gjULnKAeXS6teIE81WAaSNR1gvuHNI9U4PBJf/Fh792x+Ealcr sdYVnFuJ0NlhbP7Dtjkd+coB5dL+5KTaTHNYAW1Pryl/cnDQkscmKalJR9nAjteBJK/HnpyWUKwKn/9Gb InsD1FbE+XxRykHPQ6uuwS9Jo6SH+CwKZVrCftIC5w9LxknuhgxQHmqNdVchl/Q5CfjNYP/wc4/WgqTJt e7ervckwZrYcKGsyEkkmV2iwh4jvwqHTWU7ylV8F8iSgRVXdJFQ3BlaceeIS377d///12bPBkO7VHtm9m 7cVdMhYqaAJMjxQimmxYcbBIuw9qRe09Zu3AshTxOZH5raz6kzhnzc/fV2fCvpYKZfwpWn8rKWLzTloZr 4WKXIbrEVgpgy/cjhWX0WkjmSUxgzhDptGh4yQ32KoCzvByJuOus1fLL18fEbaqrClcMTPvIFiaM+FiRW BCAAWOLSs+R5H5HImlcVM3WpC0xiuJaZEVSrJKpXAmI9fSbMhUPX2kcoCTkims+fm7TWQd00JFxhcqaFo IFKY9K4+HwdJmCtkSHNNUtuanIODIcw6LSLYBLNBc9DOTaYnCAswa4P15B+V3GipUfrNIkWtmMWqBJbvp T23rac9YSmt0G21oDLXKGcKQDLwAq8wWMAUbrj7qR2ItQmJarGccdMc5HQyTxC1Yc8s0sWULq6b9YS8Ns Wy8WVAgQ4Guv1jCl+n0awVguZl4gdDPUD8C/kHcexYbfc6BpYmfMvnyWaQgcb4BrGAB78IrqLLh2BSCPk iyyXyAhWO0nOlzJ5kHwgAxn6FgxcEIfpsQ6nPOvU/0/CNae0WIbASLSQeSsvD3rumzJD/vT6NT1hcDCP4 kg4VR9mUq7iSEMYX1FuqaHwm+JB4uBF+fzzGwaGQEtQti+X5DWHIZT33hd2muZUha1GH41pgWcAgW+qQM gpUWtA9NqnXmR6KcRJSEdF/DMbBceCjSFZIWKZuAVeL++BJtXBm6NNTd+CFS4IZe1NdBZiFdoQ2bkXSoT 6sChEYSqzOpzzGqlEaAXqhoSWtjWDhlaclXCzB+VohM3Rv4UXUl4wiD7RgWfLZAs2Ao1MfaraqMg1rtc8 wt5mTY5vsFpHmyHizqHUEf/8dTqWtxbkRDM8nnZ+pCBBa8BO2GSKtTfw7WPTossZ7EmDKOUaVk0Dd03Ac 3gkV0Ak5Z/ER90jjy0ZR1LIE0TU9gNYIEFvIfAnEQ/Ej38WAQ/escCU9a6AblTSNlQ1glw5QqHRm4bilR u2ZRT1Ik+b1SNQjKgXxTND6DoErZ/mIticYLpibutWnrHLhvn850rQOrLmspi78Kmv10nYAX0Jhra+LJF JZQQrBwUqMyI+Lh/AicdRDSwb3PIbBIyuJbKxJd17cxrJDp758R12LigVTNEdPPmJIeh1yi5QIoB+qMZJ wM+ub7uCOWwS/FQp/HGdqBTFYk9a+OdT6blzu02Seb/1YcGUflGi/YLAtMBCmij5AACycw9IR+baE6V92 IIUjxUO9s6ndj74KhvCKX2MlILCetAwuPXOB55ujAIcYGVqw/zY6FtKbMYAl1BEsmzRicPeW6wY9AUaii Lr3urRFn0GU3twqSVs12sHB8QcqyHA17DW2tYOFgNzmGiGqj73xG12Z48JRLId6PQJiGRF+fdab7XhjuM egii66C9GHPqjeFhbVSJGRzODSkURD4cdJaA+OP1JIkhUX/Tl8S6RGjH6nBsph2iCrFeYWJLLZaDcdmU1 geLDaDNkOUgAStNj0Lcs1u0TN2Zq1s76uj2KOTQmSgbBj8SKyPpqU+hzSW0YIDbQJLX62GERaaepbfmOc 4w6GE3QlrEvt7gAAora6kvgZ6Az0BjDEYyvB+BIqipjqjCy8cPfRiRcsr3Gq1QqRynxxqCitdlu1WzOCD gGPJSJEj4RCs2bESSoAa4xHSfSHsZ7QHI5SDctBvRAO+L81nnABLWUok3volSe/8ajX/4UOJo02jUELdv yY3jokNmdyBhzRJ0PamFuR9xJcESOZocNSGBFEqlX1Q3RjqI3+ywUuve2fK1UTimB8/NK7AQj+E24wVbj tYZ+qpoQUfOJ255aB9XkbDpMw6xiClSJPPJUrliuTy++YzpgOW553QWBY8Lx1eJIvqTCuwAKLlNnpuc0m 2zBCyY/qSyOsEtSTbDQH4szTUYj2jqv60T8UKvjA6OlosFsumUhDEaibjgMay2ChOV/CZmVdgfXf4SNOU heqdSLJpQ06oJ0s+yF1ld73px0YKaSza5+dDcq7uK8BgoQ7uyuWK6C+ozGdM1Y07k3FAY1ks9PYWUcayr MCqTE9HzL/uxDZKhXq93uweO0EabAzuqtdybjln732CVG7UdpVUwSVmMrnR31wx10sL/nKt0PClpXpxMG NU/VAbZhwg4QVfoZJzuwSXhYdbHh0t29xribb5y8VcRR5tGmPgH7273IMAq2jrfmKhuKuYK476RTXTtys XqZzuLRbwn3sSwMR0w1zdqH2Zn1zCCxub6pfcu2qCZLFao5zL9RY81i1SYzQHm0aLPtHuiGIplxv0qyGp

```
XHMVIsW07S7lUKhQL5VAZtXTkgqD4Om6fjDRD6rU+IZ6pm77Ycx+NCvV59+M+erlUshijXqoXhqsNaxbQ
rXBWiHkC43WbA9X2CUX0L+NYgEyd6lut08DnVkaPdivbefPtbG/ZH7D/sz61/CjB0vVtzlDH2b3IjmN5p
vstvl8IR/ZgLc32Ym92PQDNnOKvKV/tGttIgz/v9iPZtWxjnWsYx3rWMc61rGOdaxjHetYxzrWsY51rGM
d61jHfpD9H867Du/VuRQMAAAAAElFTkSuQmCC"
alt="Image Description" style="width: 750px; height: auto;"
class="center-image">
</div>
```

homepage.component.css

```
.container {
   text-align: center; /* Center-align content */
   background-image: url('https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcTj7emkxfyB-
DOexTHJdfe5PFDW0JDmPiqHxw&usqp=CAU');
   background-size: cover; /* Adjusts the background image to cover the entire
   background-position: center; /* Center the background image */
   font-family: Arial, sans-serif; /* Example font family */
   width: auto;
   height: 800px;
  .center-image {
   display: block; /* Set the image to display as a block element */
   margin: 0 auto; /* Set left and right margins to auto for horizontal
centering */
  .login-link{
       color: black;
   text-decoration: none; /* Remove underline */
```

homepage.component.ts

```
import { Component } from '@angular/core';

@Component({
   selector: 'app-homepage',
   templateUrl: './homepage.component.html',
```

```
styleUrl: './homepage.component.css'
})
export class HomepageComponent {
}
```

admin-login.component.html

```
<title>Admin Login</title>
  <link rel="stylesheet" href="admin-login.component.css">
  <!-- Bootstrap CSS -->
  <link rel="stylesheet"</pre>
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <div class="container">
    <h2 class="mt-4 mb-4">Admin Login</h2>
    <form (ngSubmit)="login()" class="login-form">
      <div class="form-group">
        <label for="username">Username:</label>
        <input type="text" id="username" name="username" [(ngModel)]="username"</pre>
class="form-control">
      </div>
      <div class="form-group">
        <label for="password">Password:</label>
        <input type="password" id="password" name="password"</pre>
[(ngModel)]="password" class="form-control">
      <button type="submit" class="btn btn-primary">Login</button>
    </form>
  </div>
  <!-- Bootstrap JS and jQuery -->
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"><</pre>
/script>
 <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></scrip</pre>
```

admin-login.component.css

```
.container {
 text-align: center;
 background-image: url('https://wallpaperaccess.com/full/2886484.jpg');
 background-size: cover;
 background-position: center center;
 font-family: 'Times New Roman', Times, serif;
 margin: 0;
 padding: 0;
 display: flex;
 justify-content: center;
 align-items: center;
 width: 100vw;
 height: 100vh;
.login-form {
 background-color: rgba(255, 255, 255, 0.8);
 padding: 20px;
 border-radius: 8px;
.login-form div {
 margin-bottom: 10px;
```

admin-login.component.ts

```
import { Component } from '@angular/core';
import { Router } from '@angular/router';

@Component({
    selector: 'app-admin-login',
    templateUrl: './admin-login.component.html',
    styleUrls: ['./admin-login.component.css']
})
export class AdminLoginComponent {
    username: string = '';
    password: string = '';
    constructor(private router: Router) {}
    login(): void {
```

```
// authentication logic
const isValid = this.authenticate();

if (isValid) {
    alert('Successfully Logged in');
    this.router.navigate(['/changepassword']); // Navigate to change password
page on successful login
    } else {
        alert('Invalid credentials. Please try again.'); // Show error message for
invalid credentials
    }
}

authenticate(): boolean {
    return this.username === 'admin' && this.password === 'admin';
}
```

change-password.component.html

```
<div class="container">
 <div class="center-content">
   <h2>Change Password</h2>
   <form (ngSubmit)="changePassword()" class="password-form">
        <label for="oldPassword">0ld Password:</label>
        <input type="password" id="oldPassword" name="oldPassword"</pre>
[(ngModel)]="oldPassword">
     </div>
     <div>
       <label for="newPassword">New Password:</label>
        <input type="password" id="newPassword" name="newPassword"</pre>
[(ngModel)]="newPassword">
     </div>
     <br>
     <button type="submit">Change Password</button>
   </form><br>
   <h1>0r</h1><br>
   <h2>View Employees List</h2>
    <a routerLink="/employees" class="link">To view the list of employee details,
click here.</a>
 </div>
```

```
.container {
 text-align: center;
 background-image: url('https://www.itl.cat/pngfile/big/135-1358691_tablet-blue-
purple-black.jpg');
  background-size: cover;
 background-position: center;
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  display: flex;
  justify-content: center;
  align-items: center;
 height: 100vh;
password-form button {
  background-color: #da38d2;
  color: white; /* White text color */
  padding: 10px 20px; /* Padding */
  border: none; /* No border */
  border-radius: 4px; /* Rounded corners */
  cursor: pointer; /* Cursor style */
  transition: background-color 0.3s ease; /* Smooth background color transition
.password-form button:hover {
  background-color: #5a1a35;
.center-content {
  background-color: rgba(255, 255, 255, 0.8);
  padding: 20px;
  border-radius: 8px;
.password-form div {
 margin-bottom: 10px;
.link {
  display: block;
 margin-top: 20px;
```

```
color: black;
text-decoration: none; /* Remove underline */
}
```

change-password.component.ts

```
import { Component } from '@angular/core';
import { EmployeeserviceService } from '../employeeservice.service';
@Component({
  selector: 'app-change-password',
  templateUrl: './change-password.component.html',
  styleUrls: ['./change-password.component.css']
export class ChangePasswordComponent {
  oldPassword: string = '';
  newPassword: string = '';
  constructor(private EmployeeserviceService: EmployeeserviceService) {}
  changePassword(): void {
    const savedPassword = this.EmployeeserviceService.getPassword();
    if (this.oldPassword === savedPassword) {
      this.EmployeeserviceService.setPassword(this.newPassword);
      alert('Password changed successfully!');
    } else {
      alert('Failed to change password. Please try again.');
```

employees.component.html

```
<br><br><div class="container"><br><br><br><br><bri><interpretation of the container o
```

```
ID
     First Name
     Last Name
     Email
     Actions
   </thead>
  {{ employee.id }}
     {{ employee.first_name }}
     {{ employee.last_name }}
     {{ employee.email }}
     <!-- View button with routerLink -->
      <button [routerLink]="'/view/' + employee.id" class="table-</pre>
button">view</button>
    </div>
```

employees.component.css

```
.container {
    background-image: url('https://cdn.wallpapersafari.com/9/55/kGvVJa.jpg');
    background-size: cover;
    background-position: center;
    display: flex;
    flex-direction: column;
    align-items: center;
    height: 100vh;
    margin: 0;
    padding: 0;
    position: relative; /* Ensure it starts from the top */
    top: -35px; /* Position from the top */

}
.table-button{
    background-color: #da38d2;
    color: white; /* White text color */
```

```
padding: 10px 20px; /* Padding */
   border: none; /* No border */
   border-radius: 4px; /* Rounded corners */
   cursor: pointer; /* Cursor style */
   transition: background-color 0.3s ease; /* Smooth background color transition
.table-button:hover{
   background-color: #5a1a35;
 .mat-button{
   background-color: #da38d2;
 color: white; /* White text color */
 padding: 10px 20px; /* Padding */
 border: none; /* No border */
 border-radius: 4px; /* Rounded corners */
 cursor: pointer; /* Cursor style */
 transition: background-color 0.3s ease; /* Smooth background color transition
 .mat-button:hover{
   background-color: #5a1a35;
 /* Style for the table */
 table {
   width: 80%; /* Adjust the width as needed */
   border-collapse: collapse;
   margin-bottom: 20px;
 th, td {
   border: 1px solid #ddd;
   padding: 8px;
   text-align: left;
   background-color: #f2f2f2;
```

```
/* Style for the button */
button {
   margin-bottom: 20px;
}

.table-b{
   background-color: rgba(255, 255, 255, 0.8);
   padding: 20px;
   border-radius: 8px;
   margin-bottom: 10px;
}
```

employees.component.ts

```
import { Component, OnInit } from '@angular/core';
import { EmployeeserviceService } from '../employeeservice.service';
@Component({
  selector: 'app-employees',
 templateUrl: './employees.component.html',
  styleUrl: './employees.component.css'
export class EmployeesComponent implements OnInit {
  employees: any[]=[];
  id=""
  first_name=""
  last name=""
  email=""
    constructor(private employeeservice:EmployeeserviceService) { }
    ngOnInit(): void {
     this.getemployees();
    getemployees():void{
    this.employeeservice.getemployee().subscribe(data=>{this.employees=data});
  deletEmployee(id: number): void {
   this.employeeservice.deletEmployee(id).subscribe(() => {
```

```
this.employees = this.employees.filter((employee) => employee.id !== id);
});
}

cancelUpdate(employee: any): void {
  employee.editing = false;
}

updateEmployee(employee: any): void {
  this.employeeservice.updateEmployee(employee.id, {
    first_name: employee.first_name,
    last_name: employee.last_name,
    email: employee.email

}).subscribe(() => {
  employee.editing = false;
  });
}

}
```

addemployees.component.html

addemployees.component.css

```
.container {
```

```
text-align: center;
    background-image: url('https://img.freepik.com/premium-photo/night-sky-is-
dark-blue-color-with-clouds-stars_662214-55728.jpg?w=360');
    background-size: contain;
    background-position: center;
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    position: relative; /* Ensure it starts from the top */
 top: -15px; /* Position from the top */
.add-form{
    display: flex;
    flex-direction: column;
    align-items: center;
    background-color: rgba(255, 255, 255, 0.8);
    padding: 50px;
    border-radius: 100px;
    margin-bottom: 10px;
.button-click{
   background-color: #da38d2;
    color: white; /* White text color */
    padding: 10px 20px; /* Padding */
    border: none; /* No border */
    border-radius: 4px; /* Rounded corners */
    cursor: pointer; /* Cursor style */
    transition: background-color 0.3s ease; /* Smooth background color transition
.button-click:hover{
    background-color: #5a1a35;
```

addemployees.component.ts

```
import { Component } from '@angular/core';
import { EmployeeserviceService } from '../employeeservice.service';
import { Router } from '@angular/router';
@Component({
  selector: 'app-addemployees',
  templateUrl: './addemployees.component.html',
  styleUrls: ['./addemployees.component.css']
export class AddemployeesComponent {
  employees: any[] = [];
  first_name = "";
  last_name = "";
  email = "";
  constructor(
    private employeeservice: EmployeeserviceService,
    private router: Router
  ) {}
  addemployee(): void {
      console.log('First Name:', this.first_name);
      console.log('Last Name:', this.last_name);
      console.log('Email:', this.email);
    if (this.first_name.trim() && this.last_name.trim() && this.email.trim()) {
      const newEmployee = {
       first_name: this.first_name.trim(),
       last_name: this.last_name.trim(),
       email: this.email.trim()
      };
      this.employeeservice.addemployee(newEmployee).subscribe(_ => {
        alert('Employee added successfully!');
       this.router.navigate(['/employees']); // Navigate to employees list after
addition
      });
    } else {
      alert('Please fill in all the fields');
```

```
<div class="container">
 <div *ngIf="employee" class="view-contain">
   <h2>Employee Details</h2>
   <form *ngIf="!isEditing" (ngSubmit)="toggleEditMode()">
     ID: {{ employee.id }}
     First Name: {{ employee.first_name }}
     Last Name: {{ employee.last_name }}
     Email: {{ employee.email }}
     <!-- Toggle Edit Mode Button -->
     <button mat-raised-button color="primary" routerLink="/employees">Employee
List</button>&nbsp;
     <button type="button" (click)="toggleEditMode()">Edit</button>&nbsp;
     <button (click)="deleteEmployee()">Delete</button>
   </form>
   <form *ngIf="isEditing" (ngSubmit)="updateEmployee()">
     <label for="firstName">First Name:</label>
     <input type="text" id="firstName" [(ngModel)]="employee.first_name"</pre>
name="firstName">
     <label for="lastName">Last Name:</label>
     <input type="text" id="lastName" [(ngModel)]="employee.last_name"</pre>
name="lastName">
     <label for="email">Email:</label>
     <input type="email" id="email" [(ngModel)]="employee.email" name="email">
     <!-- Update Button -->
     <button type="submit">Update</button>&nbsp;&nbsp;
     <!-- Cancel Editing Button -->
     <button type="button" (click)="cancelEditMode()">Cancel/button>
</div>
```

view-employee.component.css

```
.container {
    background-image: url('https://img.freepik.com/premium-photo/sky-
night 646510-7882.jpg?size=626&ext=jpg&ga=GA1.1.1880011253.1700265600&semt=ais');
```

```
background-size: cover;
   background-position: center;
   display: flex;
   flex-direction: column;
   align-items: center;
   height: 100vh;
   margin: 0;
   padding: 0;
   position: relative; /* Ensure it starts from the top */
   top: -35px; /* Position from the top */
   display: flex;
   justify-content: center;
   align-items: center;
   height: 100vh;
 .view-contain {
   width: 70%;
   padding: 20px;
   background-color: #e0d5d5;
   border-radius: 8px;
   text-align: center;
 /* Existing styles for .container and .view-contain */
.view-contain button {
   margin: 5px;
   padding: 10px 20px;
   border: none;
   border-radius: 5px;
   cursor: pointer;
   transition: background-color 0.3s ease;
 .view-contain button[type="submit"] {
   background-color: #4caf50; /* Green */
   color: white;
 .view-contain button[type="button"],
 .view-contain button:not([type]) {
   background-color: #f44336; /* Red */
   color: white;
```

```
.view-contain button:hover {
  opacity: 0.8;
}
.view-contain form input[type="email"] {
  width: 200px;
}
```

view-employee.component.ts

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
import { EmployeeserviceService } from '../employeeservice.service';
@Component({
  selector: 'app-view-employee',
  templateUrl: './view-employee.component.html',
  styleUrls: ['./view-employee.component.css']
export class ViewEmployeeComponent implements OnInit {
  employee: any;
  isEditing = false;
  originalEmployee: any;
  constructor(
   private route: ActivatedRoute,
   private router: Router,
    private employeeService: EmployeeserviceService
  ) {}
  ngOnInit(): void {
    this.route.params.subscribe(params => {
      const employeeId = params['id'];
     this.employeeService.getEmployeeById(employeeId).subscribe(data => {
        this.employee = data;
       this.originalEmployee = { ...data }; // Create a copy for editing
      });
    });
  toggleEditMode(): void {
    this.isEditing = !this.isEditing;
```

```
cancelEditMode(): void {
   this.employee = { ...this.originalEmployee }; // Reset to original values
   this.isEditing = false;
 updateEmployee(): void {
   if (this.employee) {
     this.employeeService.updateEmployee(this.employee.id,
this.employee).subscribe(updatedEmployee => {
       this.employee = updatedEmployee;
       this.originalEmployee = { ...updatedEmployee }; // Update original values
       this.isEditing = false;
     });
 deleteEmployee(): void {
   if (this.employee) {
     this.employeeService.deletEmployee(this.employee.id).subscribe(() => {
       // Redirect to the employee list or another page after deletion
       this.router.navigate(['/employees']);
     });
```