1)profiles.component.html

<form #pform="ngForm" (ngSubmit)=OnSubmit(pform) autocomplete="off">

<div class="Body" [ngStyle]="{background: 'url(./assets/University.jpg)'}">

<div class="container" >

<!---heading---->

<header class="heading"> Registration-Form</header><hr/>

   <!---Form starting---->

   <!--- For Name and Student ID---->

<div class="row">

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Student ID :</label>

</div>

<div class="col-md-8">

<input type="text" name="StdId" id="StdId" required [(ngModel)]="this.service.formData.StdId" placeholder="Enter your Student Id" class="form-control" #StdId="ngModel">

</div>

</div>

</div>

<div class="form-group">

<div class="row">

<input type="hidden" class="form-control" name="StdId" id="StudentName" required [(ngModel)]="this.service.formData.StdId" placeholder="StdId" #StdId="ngModel">

<div class="col-md-4">

<label class="StudentName">Student Name :</label>

</div>

<div class="col-md-8">

     <input type="text" class="form-control" name="StudentName" id="StudentName" required [(ngModel)]="this.service.formData.StudentName" placeholder="Enter your Name" #StudentName="ngModel">

</div>

</div>

</div>

<!--- For Father Name And Mother Name---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Father Name :</label>

</div>

<div class="col-md-8">

     <input type="text" class="form-control" name="FatherName" id="FatherName" required [(ngModel)]="this.service.formData.FatherName" placeholder="Enter your Father Name" #FatherName="ngModel" >

</div>

</div>

</div>

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName"> Mother Name :</label>

</div>

<div class="col-md-8">

<input type="text" class="form-control" name="MotherName" id="MotherName" required [(ngModel)]="this.service.formData.MotherName" placeholder="Enter your Mother Name" #MotherName="ngModel" >

</div>

</div>

</div>

<!--- For College Name And Department Name---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName"> College Name : </label>

</div>

<div class="col-md-8">

<select class="form-control" name="CollegeName" required [(ngModel)]="this.service.formData.CollegeName" #CollegeName="ngModel">

<option class="text" id="CollegeName" selected disabled>Select your College Name</option>

<option>Satyabhama University</option>

<option>JPR Engineering college</option>

<option>New Horizon Engg. college</option>

<option>Ramaiah College</option>

<option>IIT Delhi</option>

<option>IIT Mumbai</option>

<option>IIT Kanpur</option>

<option>IIT Chennai</option>

<option>IIT Kharakpur</option>

<option>NIT Surutkal</option>

<option>SRM University</option>

<option>VIT Vellore</option>

<option>BITS Pilani</option>

<option>Others</option>

</select>

</div>

</div>

</div>

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Department Name :</label>

</div>

<div class="col-md-8">

<select class="form-control" name="DeptName" required [(ngModel)]="this.service.formData.DeptName" #DeptName="ngModel">

<option class="text" id="DeptName"  selected disabled>Select your Department</option>

<option>Mechanical Engineering</option>

<option>Electrical Engineering</option>

<option>Electronics Engineering</option>

<option>Civil Engineering</option>

<option>Computer Sc.Engineering</option>

<option>Information Sc.Engineering</option>

<option>Biotechnogical Engineering</option>

<option>Elect.& Electronics Engg.</option>

<option>Mass And communication</option>

<option>Others</option>

</select>

</div>

</div>

</div>

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Date Of Joining :</label>

</div>

<div class="col-md-8">

<input type="date" name="DOJ" id="DOJ" required [(ngModel)]="this.service.formData.DOJ" placeholder="Enter joining Date" class="form-control" #DOJ="ngModel">

</div>

</div>

</div>

<!-----For Contact Number---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Contact Number :</label>

</div>

<div class="col-md-8">

<input type="number" name="ContactNo" id="ContactNo" required maxlength="10" [(ngModel)]="this.service.formData.ContactNo" placeholder="Enter Contact Number" class="form-control" #ContactNo="ngModel">

</div>

</div>

</div>

<!-----For EmailID---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName" >Email ID :</label>

</div>

<div class="col-md-8">

       <input type="email" name="EmailId" id="EmailId" required [(ngModel)]="this.service.formData.EmailId" placeholder="Enter your Email-Id" class="form-control" #EmailId="ngModel" >

</div>

</div>

</div>

<!-----------For Image-------->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Student Image :</label>

</div>

<div class="col-md-8">

<input type="text" name="StudentImage" id="StudentImage" required [(ngModel)]="this.service.formData.StudentImage" placeholder="Enter your Image Path" class="form-control" #StudentImage="ngModel">

</div>

</div>

</div>

<!-----------For Gender-------->

<div class="form-group">

<div class="row">

<div class="col-md-4 ">

<label class="gender">Gender:</label>

</div>

<div class="col-md-3 gen">

<input type="radio" name="Gender" required [(ngModel)]="this.service.formData.Gender" #Gender="ngModel" id="Gender" value="Male" checked>

<label for="boy">Male</label>

</div>

<div class="col-md-3 gen">

<input type="radio" name="Gender" required [(ngModel)]="this.service.formData.Gender" #Gender="ngModel" id="Gender" value="Female" >

<label for="girl">Female</label>

</div>

</div>

</div>

<!-----------For Submit-------->

<div class="form-group">

<button type="submit" [disabled]="!pform.form.valid" class="btn"> SUBMIT </button>

</div>

     </div>

</div>

</div>

</form>

2)profiles.component.css

.Body{

height: 1350px;

padding-top:50px;

}

/\*-----for border----\*/

.container{

font-family:Roboto,sans-serif;

background-image:url(https://image.freepik.com/free-vector/dark-blue-blurred-background\_1034-589.jpg) ;

border-style: 1px solid grey;

margin: 0 auto;

text-align: center;

opacity: 0.8;

margin-top: 67px;

box-shadow: 2px 5px 5px 0px #eee;

max-width: 550px;

padding-top: 10px;

height: 930px;

margin-top: 140px;

border-radius: 20px;

}

.form-group{

width:100%;

}

/\*--- for label of first and last name---\*/

.StudentName{

margin-left: 2%;

font-family: sans-serif;

font-size: 20px;

color: white;

margin: 10px 0;

}

/\*---for heading-----\*/

.heading{

text-decoration:bold;

text-align : center;

font-size:30px;

color:#F96;

padding-top:10px;

}

.form-control{

margin-left: 30px;

width: 80%;

}

/\*--------------for Gender---------------\*/

/\*--------------label---------\*/

.gender {

color: white;

font-family: sans-serif;

font-size: 20px;

margin-top: 10px;

}

/\*---------- for Input type--------\*/

.gen{

color: white;

font-size: 18px;

padding-top: 13px;

}

.female {

color: white;

font-size: 18px;

margin-top: 14px;

padding-bottom: 16px;

margin-right: 20px;

}

/\*------------For submit button---------\*/

/\* .btn{

color: white;

font-size: 25px;

border: 1px solid white;

background-color: #080808;

width: 35%;

margin-left:30%;

margin-top: 20px;

box-shadow: 0px 2px 2px 0px white;

} \*/

.btn{

box-shadow: 2px 1px 2px 3px #99ccff;

background:#5900a6;

color:#fff;

transition: background-color 1.15s ease-in-out,border-color 1.15s ease-in-out,box-shadow 1.15s ease-in-out;

}

.inputStyle{

border-radius: 5px;

line-height: 1.5em;

margin-right: 10px;

}

@import '~ngx-toastr/toastr.css';

3)profiles.component.ts

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

import {ProfileService} from '../../Shared/profile.service';

import { NgForm, NgFormSelectorWarning } from '@angular/forms';

import { MockNgModuleResolver } from '@angular/compiler/testing';

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

import { ToastrService } from 'ngx-toastr';

import{stdProfile} from '../../Shared/stdProfile.model';

import { from } from 'rxjs';

@Component({

selector: 'app-profiles',

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

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

})

export class ProfilesComponent implements OnInit {

public stdId:number[];

public result:any;

Studentid:number;

std:stdProfile;

constructor(private service:ProfileService,private toastr:ToastrService)

{

this.service.formData={

StdId:null,

StudentName:null,

FatherName:null,

MotherName:null,

DOJ:null,

Gender:null,

DeptName:null,

CollegeName:null,

ContactNo:null,

EmailId:null,

StudentImage:null,

}

this.result=this.service.list;

}

ngOnInit() {

this.resetForm();

this.service.refreshList();

}

OnSubmit(form:NgForm)

{ {debugger}

this.Studentid=form.value.stdId;

this.std=this.service.list.find(e=>e.StdId==this.Studentid);

console.log(this.Studentid);

if(this.std==null)

{

this.InsertRecord(form);

}

else

{

this.UpdateRecord(form);

}

}

InsertRecord(form:NgForm)

{

this.service.poststdProfile(form.value).subscribe(res=>{

this.toastr.success("Inserted Successfully","Student Register");

this.resetForm(form)

this.service.refreshList();

});

}

UpdateRecord(form:NgForm)

{

this.service.putstdProfile(form.value).subscribe(res=>{

this.toastr.warning("Updated Successfully","Student Register");

this.resetForm(form)

this.service.refreshList();

});

}

resetForm(form?:NgForm)

{

if(form!=null)

{

form.resetForm();

this.service.formData={

StdId:null,

StudentName:null,

FatherName:null,

MotherName:null,

DOJ:null,

Gender:null,

DeptName:null,

CollegeName:null,

ContactNo:null,

EmailId:null,

StudentImage:null,

}

}

}

}

4)profile-list.component.html

<div class="container-fluid">

<div class="row">

<div class="col-md-3">

<label style="font-size:22px;margin-top:3px;border-radius:5px;box-shadow: 2px 1px 2px 3px #99ccff;opacity:1;margin-left:30px;font-family: Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif" >Enter the Department Name:</label>

</div>

<div class="col-md-3">

<input type="text" class="form-control" [(ngModel)]="department" id=DeptName #DeptName placeholder=" Department ">

</div>

<div class="col-md-3">

<label style="font-size:22px;margin-top:3px;border-radius:5px;box-shadow: 2px 1px 2px 3px #99ccff;opacity:1;margin-left:30px;font-family: Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif">Enter the College Name:</label>

</div>

<div class="col-md-3">

<input type="text" class="form-control" [(ngModel)]="college" id=CollegeName #CollegeName placeholder=" CollegeName ">

</div>

</div>

<div class="row">

<div class="card" \*ngFor="let std of service.list |collegefilterpipe:college|departmentfilterPipe:department" >

<div class="col-md-12">

<img src="{{std.StudentImage}}" (click)=Populateform(std) style="width:250px;height:250px;border-radius:10px ;" >

<h4 (click)=Populateform(std)><b>{{std.StudentName}}</b></h4>

<p (click)=Populateform(std)>{{std.FatherName}}</p>

<p (click)=Populateform(std)>{{std.MotherName}}</p>

<p (click)=Populateform(std)><b>{{std.DeptName}}</b></p>

<p (click)=Populateform(std)><b>{{std.CollegeName}}</b></p>

<p (click)=Populateform(std)>{{std.DOJ|date}}</p>

<p (click)=Populateform(std)>{{std.ContactNo}}</p>

<p (click)=Populateform(std)>{{std.EmailId}}</p>

<p (click)=Populateform(std)>{{std.Gender}}</p>

<p><button (click)="onDelete(std.StdId)" class="btn btn-danger">Delete</button></p>

</div>

</div>

</div>

</div>

5)profile-list.component.css

.row{

background-color:rgb(94, 99, 110) ;

}

.card{

margin-top: 45px;

margin-left: 45px;

}

6)profile-list.component.ts

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

import {ProfileService} from '../../Shared/profile.service';

import{stdProfile} from '../../Shared/stdProfile.model';

import { ToastrService } from 'ngx-toastr';

@Component({

selector: 'app-profile-list',

templateUrl: './profile-list.component.html',

styleUrls: ['./profile-list.component.css']

})

export class ProfileListComponent implements OnInit {

constructor(public service:ProfileService, public toastr:ToastrService) {

}

ngOnInit() {

this.service.refreshList();

}

Populateform(std:stdProfile)

{

this.service.formData=Object.assign({},std);

}

onDelete(id:number)

{

if(confirm("Are you sure to Delete this record?")){

console.log(id);

this.service.deletestdProfile(id).subscribe(res=>

{

this.service.refreshList();

this.toastr.info("Deleted Successfully","Employee Register");

});

}

}

}

7)profile.service.ts

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

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

import { stdProfile } from '../Shared/stdProfile.model';

@Injectable({

providedIn: 'root'

})

export class ProfileService {

formData:stdProfile;

list:stdProfile[];

readonly rootURL="http://localhost:53715/api"

constructor(private http:HttpClient) { }

poststdProfile(formData:stdProfile)

{

return this.http.post(this.rootURL+'/Profile\_Table',formData);

}

refreshList():void

{

this.http.get(this.rootURL+'/Profile\_Table')

.toPromise().then(res=>this.list=res as stdProfile[]);

}

putstdProfile(formData:stdProfile)

{

const headers = new HttpHeaders().set('content-type', 'application/json');

const params = new HttpParams().set('ID', formData.StdId.toString());

{

return this.http.put<stdProfile>(this.rootURL+'/Profile\_Table/'+formData.StdId,formData,{headers,params});

}

}

deletestdProfile(id:number)

{

return this.http.delete(this.rootURL+'/Profile\_Table/'+id);

}

}

8)stdprofile.modle.ts

export class stdProfile{

StdId:number;

CollegeName:string;

StudentName:string;

StudentImage:string;

Gender:string;

DOJ:Date;

FatherName:string;

MotherName:string;

DeptName:string;

EmailId:string;

ContactNo:number;

}

9)profile.component.html

<nav class="navbar navbar-inverse">

<div class="container-fluid">

<div class="navbar-header">

<a class="navbar-brand" routerLink="Profiles">REGISTRATION FORM</a>

</div>

<div class="navbar-header">

<a class="navbar-brand" routerLink="List">DETAILS OF THE STUDENT</a>

</div>

<div class="navbar-header">

<a class="navbar-brand" routerLink="admin">ADMIN CONTROL</a>

</div>

</div>

</nav>

<router-outlet></router-outlet>

10)profile.component.css

.navbar{

background-color:black;

}

.navbar-brand{

background-color:black;

border-radius: 5px;

box-shadow: 2px 1px 2px 3px #99ccff;

opacity: 1;

}

11)profile.component.ts

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

@Component({

selector: 'app-profile',

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

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

})

export class ProfileComponent implements OnInit {

constructor() { }

ngOnInit() {

}

}

12)admincreate.component.html

<!--<head>

<meta charset="UTF-8">

<title>Registration Form</title>

 <meta name="viewport" content="width=device-width, initial-scale=1"/>

</head-->

<form #pform="ngForm" (ngSubmit)=OnSubmit(pform) autocomplete="off">

<!-- <div class="Body" [ngStyle]="{background: 'url(./assets/University.jpg)'}"> -->

<div class="container" >

<!---heading---->

<header class="heading"> Registration-Form</header><hr/>

<!---Form starting---->

<!--- For Name And ID---->

<div class="row">

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Student ID :</label>

</div>

<div class="col-md-8">

<input type="text" name="StdId" id="StdId" required [(ngModel)]="this.service.formData.StdId" placeholder="Enter your Student Id" class="form-control" #StdId="ngModel">

</div>

</div>

</div>

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Student Name :</label>

</div>

<div class="col-md-8">

<input type="text" class="form-control" name="StudentName" id="StudentName" required [(ngModel)]="this.service.formData.StudentName" placeholder="Enter your Name" #StudentName="ngModel">

</div>

</div>

</div>

<!--- For Father Name And Mother Name---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Father Name :</label>

</div>

<div class="col-md-8">

<input type="text" class="form-control" name="FatherName" id="FatherName" required [(ngModel)]="this.service.formData.FatherName" placeholder="Enter your Father Name" #FatherName="ngModel" >

</div>

</div>

</div>

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName"> Mother Name :</label>

</div>

<div class="col-md-8">

<input type="text" class="form-control" name="MotherName" id="MotherName" required [(ngModel)]="this.service.formData.MotherName" placeholder="Enter your Mother Name" #MotherName="ngModel" >

</div>

</div>

</div>

<!--- For College Name And Department Name---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName"> College Name : </label>

</div>

<div class="col-md-8">

<input type="text" id="CollegeName" name="CollegeName" required [(ngModel)]="this.service.formData.CollegeName" #CollegeName="ngModel" placeholder="Enter your College Name" class="form-control">

</div>

</div>

</div>

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Department Name :</label>

</div>

<div class="col-md-8">

<input type="text" id="DeptName" name="DeptName" required [(ngModel)]="this.service.formData.DeptName" #DeptName="ngModel" placeholder="Enter Department Name" class="form-control">

</div>

</div>

</div>

<!--- For Date of joining ---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Date Of Joining :</label>

</div>

<div class="col-md-8">

<input type="text" name="DOJ" id="DOJ" required [(ngModel)]="this.service.formData.DOJ" placeholder="Enter joining Date" class="form-control" #DOJ="ngModel">

</div>

</div>

</div>

<!--- For Contact Number---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Contact Number :</label>

</div>

<div class="col-md-8">

<input type="number" name="ContactNo" id="ContactNo" required [(ngModel)]="this.service.formData.ContactNo" placeholder="Enter Contact Number" class="form-control" #ContactNo="ngModel">

</div>

</div>

</div>

<!-----For EmailId---->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName" >Email :</label>

</div>

<div class="col-md-8">

<input type="email" name="EmailId" id="EmailId" required [(ngModel)]="this.service.formData.EmailId" placeholder="Enter your email" class="form-control" #EmailId="ngModel" >

</div>

</div>

</div>

<!-----------For Image-------->

<div class="form-group">

<div class="row">

<div class="col-md-4">

<label class="StudentName">Student Image :</label>

</div>

<div class="col-md-8">

<input type="text" name="StudentImage" id="StudentImage" required [(ngModel)]="this.service.formData.StudentImage" placeholder="Enter your Image Path" class="form-control" #StudentImage="ngModel">

</div>

</div>

</div>

<!-----------For Gender-------->

<div class="form-group">

<div class="row">

<div class="col-md-4 ">

<label class="gender">Gender:</label>

</div>

<div class="col-md-3 gen">

<input type="radio" name="Gender" required [(ngModel)]="this.service.formData.Gender" #Gender="ngModel" id="Gender" value="Male" checked>

<label for="boy">Male</label>

</div>

<div class="col-md-3 gen">

<input type="radio" name="Gender" required [(ngModel)]="this.service.formData.Gender" #Gender="ngModel" id="Gender" value="Female" >

<label for="girl">Female</label>

</div>

</div>

</div>

<!-----------For Submit-------->

<div class="form-group">

<button type="submit" [disabled]="!pform.form.valid" class="btn"> SUBMIT </button>

</div>

</div>

</div>

</form>

<!-- {{form.value|json}} -->

13)admincreate.component.css

.Body{

height: 1350px;

padding-top:50px;

}

/\*-----for border----\*/

.container{

font-family:Roboto,sans-serif;

background-image:url(https://image.freepik.com/free-vector/dark-blue-blurred-background\_1034-589.jpg) ;

border-style: 1px solid grey;

margin: 0 auto;

text-align: center;

opacity: 0.8;

margin-top: 67px;

box-shadow: 2px 5px 5px 0px #eee;

max-width: 550px;

padding-top: 10px;

height: 930px;

margin-top: 0px;

border-radius: 20px;

}

.form-group{

width:100%;

}

/\*--- for All the label and columns---\*/

.StudentName{

margin-left: 2%;

font-family: sans-serif;

font-size: 20px;

color: white;

margin: 10px 0;

}

/\*---for heading-----\*/

.heading{

text-decoration:bold;

text-align : center;

font-size:30px;

color:#F96;

padding-top:10px;

}

.form-control{

margin-left: 30px;

width: 80%;

}

/\*--------------for Gender---------------\*/

/\*--------------label---------\*/

.gender {

color: white;

font-family: sans-serif;

font-size: 20px;

margin-top: 10px;

}

/\*---------- for Input type--------\*/

.gen{

color: white;

font-size: 18px;

padding-top: 13px;

}

.female {

color: white;

font-size: 18px;

margin-top: 14px;

padding-bottom: 16px;

margin-right: 20px;

}

/\*------------For submit button---------\*/

.btn{

box-shadow: 2px 1px 2px 3px #99ccff;

background:#5900a6;

color:#fff;

transition: background-color 1.15s ease-in-out,border-color 1.15s ease-in-out,box-shadow 1.15s ease-in-out;

}

.inputStyle{

border-radius: 5px;

line-height: 1.5em;

margin-right: 10px;

}

@import '~ngx-toastr/toastr.css';

14)admincreate.component.ts

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

import { ProfileService } from 'src/app/Shared/profile.service';

import { ToastrService } from 'ngx-toastr';

import { stdProfile } from 'src/app/Shared/stdProfile.model';

import { NgForm } from '@angular/forms';

@Component({

selector: 'app-admincreate',

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

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

})

export class AdmincreateComponent implements OnInit {

public stdId:number[];

public result:any;

Studentid:number;

std:stdProfile;

constructor(private service:ProfileService,private toastr:ToastrService) {

this.service.formData={

StdId:null,

StudentName:null,

FatherName:null,

MotherName:null,

DOJ:null,

Gender:null,

DeptName:null,

CollegeName:null,

ContactNo:null,

EmailId:null,

StudentImage:null,

}

this.result=this.service.list;

}

ngOnInit() {

this.resetForm();

this.service.refreshList();

}

OnSubmit(form:NgForm)

{ {debugger}

this.Studentid=form.value.StdId;

this.std=this.service.list.find(e=>e.StdId==this.Studentid);

console.log(this.Studentid);

if(this.std==null)

// if(form.value.eid=this.Studentid)

{

this.InsertRecord(form);

}

else

{

this.UpdateRecord(form);

}

}

InsertRecord(form:NgForm)

{

this.service.poststdProfile(form.value).subscribe(res=>{

this.toastr.success("Inserted Successfully","Student Register");

this.resetForm(form)

this.service.refreshList();

});

}

UpdateRecord(form:NgForm)

{

this.service.putstdProfile(form.value).subscribe(res=>{

this.toastr.warning("Updated Successfully","Student Register");

this.resetForm(form)

this.service.refreshList();

});

}

resetForm(form?:NgForm)

{

if(form!=null)

{

form.resetForm();

this.service.formData={

StdId:null,

StudentName:null,

FatherName:null,

MotherName:null,

DOJ:null,

Gender:null,

DeptName:null,

CollegeName:null,

ContactNo:null,

EmailId:null,

StudentImage:null,

}

}

}

}

15)amdindetails.component.html

<div class="container">

<div class="row">

<div class="card" \*ngFor="let std of service.list" >

<div class="col-md-12">

<img src="{{std.StudentImage}}" (click)=Populateform(std) style="width:250px;height:250px;border-radius:10px ;" >

<h4 (click)=Populateform(std)><b>{{std.StudentName}}</b></h4>

<p (click)=Populateform(std)>{{std.FatherName}}</p>

<p (click)=Populateform(std)>{{std.MotherName}}</p>

<p (click)=Populateform(std)><b>{{std.DeptName}}</b></p>

<p (click)=Populateform(std)><b>{{std.CollegeName}}</b></p>

<p (click)=Populateform(std)>{{std.DOJ|date}}</p>

<p (click)=Populateform(std)>{{std.ContactNo}}</p>

<p (click)=Populateform(std)>{{std.EmailId}}</p>

<p (click)=Populateform(std)>{{std.Gender}}</p>

<p><button (click)="onDelete(std.StdId)" class="btn btn-danger">Delete</button></p>

</div>

</div>

</div>

</div>

16)admindetails.component.ts

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

import { ProfileService } from 'src/app/Shared/profile.service';

import { ToastrService } from 'ngx-toastr';

import { stdProfile } from 'src/app/Shared/stdProfile.model';

@Component({

selector: 'app-admindetails',

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

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

})

export class AdmindetailsComponent implements OnInit {

constructor(public service:ProfileService, public toastr:ToastrService) { }

ngOnInit() {

this.service.refreshList();

}

Populateform(std:stdProfile)

{

this.service.formData=Object.assign({},std);

}

onDelete(id:number)

{

if(confirm("Are you sure to Delete this record?")){

console.log(id);

this.service.deletestdProfile(id).subscribe(res=>

{

this.service.refreshList();

this.toastr.info("Deleted Successfully","Employee Register");

});

}

}

}

17)admin.component.html

<div class="jumbotron" [ngStyle]="{background: 'url(./assets/backimage.jpg)'}">

<div class="row">

<div class="col-md-5">

<app-admincreate></app-admincreate>

</div>

<div class="col-md-7">

<app-admindetails></app-admindetails>

</div>

</div>

</div>

18)admin.component.css

.col-md-7{

opacity: 0.9;

}

19)admin.component.ts

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

@Component({

selector: 'app-admin',

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

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

})

export class AdminComponent implements OnInit {

constructor() { }

ngOnInit() {

}

}

20)filterPipe.ts

import {Pipe,PipeTransform} from '@angular/core';

import {stdProfile} from '../../Shared/stdProfile.model';

@Pipe({

name:'collegefilterpipe'

})

export class collegefilterpipe implements PipeTransform{

transform(reg:stdProfile[],college:string)

{

if(!reg||!college)

{

return reg;

}

return reg.filter((x=>x.CollegeName.toLowerCase().indexOf(college.toLowerCase())!==-1));

}

}

@Pipe({

name:'departmentfilterPipe'

})

export class departmentfilterPipe implements PipeTransform{

transform(reg:stdProfile[],department:string)

{

if(!reg|| !department)

{

return reg;

}

return reg.filter((x=>x.DeptName.toLowerCase().indexOf(department.toLowerCase())!==-1));

}

}