**OnlineTest-Application Developed by-Jyoti Nandini**

**Source Code**

**Components:1)Header Component**

**2)Welcome Component**

**3)Question Component**

**4)service**

**Header.component.html**

**--------------------------------------------**

**<nav class="navbar navbar-dark bg-primary">**

**<div class="container-fluid">**

**<a class="navbar-brand text-bold" href="#">**

**<img src="https://img.icons8.com/color/96/000000/angularjs.png" alt=" ">**

**<span class="text-upercase">ngular Quiz</span>**

**</a>**

**</div>**

**</nav>**

**-------------------------------------------------------------------------------------------------------**

**Header.component.ts**

**----------------------------------------------------**

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

**@Component({**

**selector: 'app-header',**

**templateUrl: './header.component.html',**

**styleUrls: ['./header.component.css']**

**})**

**export class HeaderComponent implements OnInit {**

**constructor() { }**

**ngOnInit(): void {**

**}**

**}**

**--------------------------------------------------------------------------------------------------------------**

**Welcome.component.html**

**---------------------------------------------------------**

**<div class="container bg-light py-5">**

**<h1 class="display-5 fw-bold">Welocme to Quiz App</h1>**

**<p class="col-md-8 fs-4">This quiz will contains total 10 questions. Each Question holds 10 Points</p>**

**<h4>Rules:</h4>**

**<ol>**

**<li>Correct Question gives you 10 points</li>**

**<li>Incorrect question gives to -10 points</li>**

**<li>You will have 60 sec to answer each question</li>**

**<li>Refereshing the page will reset the Quiz</li>**

**</ol>**

**<h1 style="font-family: cursive;text-align:center">All the best!!</h1>**

**<div class="name col-md-4 my-3">**

**<label for="">Enter your name:</label>**

**<input #name type="text" class="form-control">**

**</div>**

**<button class="btn btn-primary btn-lg" routerLink="/question" (click)="startQuiz()">Start the Quiz!!</button>**

**</div>**

**---------------------------------------------------------------------------------------------------------**

**Welcome.component.ts**

**---------------------------------------------------------**

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

**@Component({**

**selector: 'app-welcome',**

**templateUrl: './welcome.component.html',**

**styleUrls: ['./welcome.component.css']**

**})**

**export class WelcomeComponent {**

**@ViewChild('name')namekey!:ElementRef;**

**startQuiz()**

**{**

**localStorage.setItem("name",this.namekey.nativeElement.value);**

**}**

**}**

**------------------------------------------------------------------------------------------------------------**

**Question.component.html**

**-------------------------------------------**

**<div class="container mt-5">**

**<div class="card">**

**<div class="d-flex justify-content-between p-3">**

**<div class="image">**

**<img src="https://img.icons8.com/color/96/000000/angularjs.png" width="90" alt="logo">**

**</div>**

**<div class="quiz-header">**

**<h4 style="font-family: cursive;">Angular & TypeScript Quiz!</h4>**

**<span style="font-style: italic;">Welcome {{name}}</span>**

**</div>**

**</div>**

**<ng-container \*ngIf="!isQuizCompleted">**

**<div class="d-flex justify-content-around py-3">**

**<div class="score">**

**<h5>{{points}}Points</h5>**

**</div>**

**<div class="question-remain">**

**<span style="font-style: italic;">Question {{currentQuestion+1}} of {{questionList.length}}</span>**

**</div>**

**<div class="timer">**

**<h5>{{counter}} sec ⏱</h5>**

**</div>**

**</div>**

**<div class="progress mb-3">**

**<div class="progress-bar progress-bar-striped bg-success" role="progressbar" [ngStyle]="{'width':progress+'%'}" aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>**

**</div>**

**<div class="question">**

**<div class="card">**

**<h3>{{questionList[currentQuestion]?.questionText}}</h3>**

**</div>**

**</div>**

**<div class="options">**

**<ol \*ngFor="let option of questionList[currentQuestion]?.options">**

**<li (click)="answer(currentQuestion+1,option)">**

**<div appChangeBg [isCorrect]="option.correct" class="card">**

**{{option.text}}**

**</div>**

**</li>**

**</ol>**

**</div>**

**<div class="d-flex justify-content-between">**

**<button [disabled]="currentQuestion===0" class="btn" (click)="previousQuestion()"><i class="fa text-primary fa-chevron-left fa-3x" aria-hidden="true"></i></button>**

**<button class="btn" (click)="resetQuiz()"><i class="fa fa-refresh text-primary fa-3x" aria-hidden="true"></i></button>**

**<button [disabled]="currentQuestion===questionList.length-1" class="btn" (click)="nextQuestion()"><i class="fa text-primary fa-chevron-right fa-3x" aria-hidden="true"></i></button>**

**</div>**

**</ng-container>**

**<ng-container \*ngIf="isQuizCompleted">**

**<div class="row d-flex justify-content-between">**

**<img style="width: 50%;" class="img-fluid col-sm-12 mx-auto" src="https://icon-library.com/images/celebration-icon-png/celebration-icon-png-7.jpg" alt="">**

**<div class="result text-center col-md-6 col-sm-12">**

**<h3>Congratulations!! <br>You have completed the quiz. <br>Below is your result:</h3>**

**<p>Total Question Attempted : {{questionList.length}} </p>**

**<p>Total Correct Answered : {{correctAnswer}} </p>**

**<p>Total Wrong Answered : {{inCorrectAnswer}} </p>**

**<p>Your Score : {{points}} Points </p>**

**</div>**

**</div>**

**</ng-container>**

**</div>**

**</div>**

**--------------------------------------------------------------------------------------**

**Question.component.ts**

**----------------------------------------------------------------------------**

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

**import { interval } from 'rxjs';**

**import { QuestionService } from '../service/question.service';**

**@Component({**

**selector: 'app-question',**

**templateUrl: './question.component.html',**

**styleUrls: ['./question.component.css']**

**})**

**export class QuestionComponent implements OnInit {**

**public name: string = "";**

**public questionList: any = [];**

**public currentQuestion: number = 0;**

**public points: number = 0;**

**counter = 60;**

**correctAnswer: number = 0;**

**inCorrectAnswer: number = 0;**

**interval: any;**

**progress: string = "0";**

**isQuizCompleted : boolean = false;**

**constructor(private questionService: QuestionService) { }**

**ngOnInit(): void {**

**this.name = localStorage.getItem("name")!;**

**this.getAllQuestions();**

**this.startCounter();**

**}**

**getAllQuestions() {**

**this.questionService.getQuestionJson()**

**.subscribe(res => {**

**console.log(res.questions);**

**this.questionList = res.questions;**

**})**

**}**

**nextQuestion() {**

**this.currentQuestion++;**

**}**

**previousQuestion() {**

**this.currentQuestion--;**

**}**

**answer(currentQno: number, option: any) {**

**if(currentQno === this.questionList.length){**

**this.isQuizCompleted = true;**

**this.stopCounter();**

**}**

**if (option.correct) {**

**this.points += 10;**

**this.correctAnswer++;**

**setTimeout(() => {**

**this.currentQuestion++;**

**this.resetCounter();**

**this.getProgressPercent();**

**}, 1000);**

**} else {**

**setTimeout(() => {**

**this.currentQuestion++;**

**this.inCorrectAnswer++;**

**this.resetCounter();**

**this.getProgressPercent();**

**}, 1000);**

**this.points -= 10;**

**}**

**}**

**startCounter() {**

**this.interval = interval(1000)**

**.subscribe(val => {**

**this.counter--;**

**if (this.counter === 0) {**

**this.currentQuestion++;**

**this.counter = 60;**

**this.points -= 10;**

**}**

**});**

**setTimeout(() => {**

**this.interval.unsubscribe();**

**}, 600000);**

**}**

**stopCounter() {**

**this.interval.unsubscribe();**

**this.counter = 0;**

**}**

**resetCounter() {**

**this.stopCounter();**

**this.counter = 60;**

**this.startCounter();**

**}**

**resetQuiz() {**

**this.resetCounter();**

**this.getAllQuestions();**

**this.points = 0;**

**this.counter = 60;**

**this.currentQuestion = 0;**

**this.progress = "0";**

**}**

**getProgressPercent() {**

**this.progress = ((this.currentQuestion / this.questionList.length) \* 100).toString();**

**return this.progress;**

**}**

**}**

**------------------------------------------------------------------------------------------------------------------------**

**question.service.ts**

**---------------------------------------**

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

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

**@Injectable({**

**providedIn: 'root'**

**})**

**export class QuestionService {**

**constructor(private http:HttpClient) { }**

**getQuestionJson()**

**{**

**return this.http.get<any>("assets/questions.json");**

**}**

**}**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**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 { WelcomeComponent } from './welcome/welcome.component';**

**import { QuestionComponent } from './question/question.component';**

**import { HeaderComponent } from './header/header.component';**

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

**import { ChangeBgDirective } from './change-bg.directive';**

**@NgModule({**

**declarations: [**

**AppComponent,**

**WelcomeComponent,**

**QuestionComponent,**

**HeaderComponent,**

**ChangeBgDirective**

**],**

**imports: [**

**BrowserModule,**

**AppRoutingModule,HttpClientModule**

**],**

**providers: [],**

**bootstrap: [AppComponent]**

**})**

**export class AppModule { }**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**app-routing.module.ts**

**--------------------------------------------------------**

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

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

**import { WelcomeComponent } from './welcome/welcome.component';**

**import { QuestionComponent } from './question/question.component';**

**const routes: Routes = [**

**{path:"",redirectTo:'welcome',pathMatch:"full"},**

**{path:"welcome",**

**component:WelcomeComponent},**

**{path:"question",**

**component:QuestionComponent}**

**];**

**@NgModule({**

**imports: [RouterModule.forRoot(routes)],**

**exports: [RouterModule]**

**})**

**export class AppRoutingModule { }**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**app.component.html**

**-----------------------------------**

**<app-header></app-header>**

**<router-outlet></router-outlet>**

**------------------------------------------------------**

**chang-bg.directive.ts**

**-----------------------------------------------------**

**import { Directive, ElementRef, HostListener, Input, Renderer2 } from '@angular/core';**

**@Directive({**

**selector: '[appChangeBg]'**

**})**

**export class ChangeBgDirective {**

**@Input() isCorrect : Boolean = false;**

**constructor(private el : ElementRef, private render : Renderer2) { }**

**@HostListener('click') answer(){**

**if(this.isCorrect){**

**this.render.setStyle(this.el.nativeElement,'background','green');**

**this.render.setStyle(this.el.nativeElement,'color','#fff');**

**this.render.setStyle(this.el.nativeElement,'border','2px solid grey');**

**}else{**

**this.render.setStyle(this.el.nativeElement,'background','red');**

**this.render.setStyle(this.el.nativeElement,'color','#fff');**

**this.render.setStyle(this.el.nativeElement,'border','2px solid grey');**

**}**

**}**

**}**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Questions.json**

**-----------------------------------**

**{**

**"questions": [{**

**"questionText": "Which of the following does TypeScript use to specify types?",**

**"options": [{**

**"text": ":",**

**"correct": true**

**},**

**{**

**"text": ";"**

**},**

**{**

**"text": "!"**

**},**

**{**

**"text": "&"**

**}**

**],**

**"explanation": "TS uses a colon (:) to separate the property name from the property type"**

**},**

**{**

**"questionText": "Which of the following is NOT a type used in TypeScript?",**

**"options": [{**

**"text": "number"**

**},**

**{**

**"text": "string"**

**},**

**{**

**"text": "boolean"**

**},**

**{**

**"text": "enum",**

**"correct": true**

**}**

**],**

**"explanation": "enum is not used as a type in TypeScript"**

**},**

**{**

**"questionText": "How can we specify properties and methods for an object in TypeScript?",**

**"options": [{**

**"text": "Use classes."**

**},**

**{**

**"text": "Use interfaces.",**

**"correct": true**

**},**

**{**

**"text": "Use enums."**

**},**

**{**

**"text": "Use async/await."**

**}**

**],**

**"explanation": "interfaces are typically used to list the properties and methods for an object"**

**},**

**{**

**"questionText": "How else can Array<number> be written in TypeScript?",**

**"options": [{**

**"text": "@number"**

**},**

**{**

**"text": "number[]",**

**"correct": true**

**},**

**{**

**"text": "number!"**

**},**

**{**

**"text": "number?"**

**}**

**],**

**"explanation": "number[] is another way of writing Array<number> in TypeScript"**

**},**

**{**

**"questionText": "In which of these does a class take parameters?",**

**"options": [{**

**"text": "constructor",**

**"correct": true**

**},**

**{**

**"text": "destructor"**

**},**

**{**

**"text": "import"**

**},**

**{**

**"text": "subscribe"**

**}**

**],**

**"explanation": "a constructor is used by a class to take in parameters"**

**},**

**{**

**"questionText": "Which is NOT an access modifier?",**

**"options": [{**

**"text": "private"**

**},**

**{**

**"text": "protected"**

**},**

**{**

**"text": "public"**

**},**

**{**

**"text": "async",**

**"correct": true**

**}**

**],**

**"explanation": "async is not used as an access modifier type in TypeScript"**

**},**

**{**

**"questionText": "Which keyword allows us to share information between files in TypeScript?",**

**"options": [{**

**"text": "import"**

**},**

**{**

**"text": "export",**

**"correct": true**

**},**

**{**

**"text": "async"**

**},**

**{**

**"text": "constructor"**

**}**

**],**

**"explanation": "the export keyword allows for the information to be transmitted between files"**

**},**

**{**

**"questionText": "Which is an array method to generate a new array based on a condition?",**

**"options": [{**

**"text": "filter",**

**"correct": true**

**},**

**{**

**"text": "map"**

**},**

**{**

**"text": "async"**

**},**

**{**

**"text": "enum"**

**}**

**],**

**"explanation": "filter is a method used to conditionally create a new array"**

**},**

**{**

**"questionText": "How is a property accessible within a class?",**

**"options": [{**

**"text": "Using this.propertyName",**

**"correct": true**

**},**

**{**

**"text": "Accessors"**

**},**

**{**

**"text": "Destructuring"**

**},**

**{**

**"text": "Arrow function"**

**}**

**],**

**"explanation": "this.propertyName is the way to access a specific property within a class"**

**},**

**{**

**"questionText": "Which keyword allows us to share information between files in TypeScript?",**

**"options": [{**

**"text": "import"**

**},**

**{**

**"text": "export",**

**"correct": true**

**},**

**{**

**"text": "async"**

**},**

**{**

**"text": "constructor"**

**}**

**],**

**"explanation": "the export keyword allows for the information to be transmitted between files"**

**}**

**]**

**}**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**