**Online Test Application.**

Course-end Project 1

DESCRIPTION

**Project Objective:**

As a Full Stack Developer, you have to develop an online test portal for assessment.

**Background of the problem statement:**

As the project is progressing, the management has asked you to create an online test application in order to help the recruitment team take online tests of the candidates coming for interviews.

**You must use the following:**

* Visual Studio Code
* NodeJS
* Angular
* GitHub

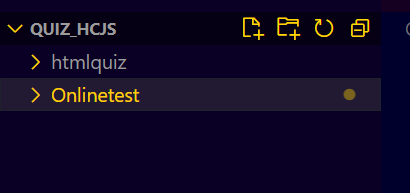
**Following requirements should be met:**

* A few of the source codes should be tracked on GitHub repositories. You need to document the tracked files that are ignored during the final push to the GitHub repository.

# How to start (Steps);

1. Create a Project named as

> ng new Onlinetest



2. Create Component:

Welcome

Questions

Header

Routing.module.ts



To make this router work, we need to set the router outlet in app.component.html

<router-outlet> </router-outlet>

Create a new component: header

Configure Bootstrap

Follow :

[Add Bootstrap 5 to Angular 15 with example & tutorial | Techiediaries](https://www.techiediaries.com/angular-bootstrap/#:~:text=Bootstrap%20may%20be%20integrated%20into%20your%20Angular%20application,arrays%20of%20the%20angular.json%20file%20of%20your%20project)

*. Install bootstrap (*[*https://getbootstrap.com/docs/5.3/getting-started/download/*](https://getbootstrap.com/docs/5.3/getting-started/download/)*) by adding link in html head file*

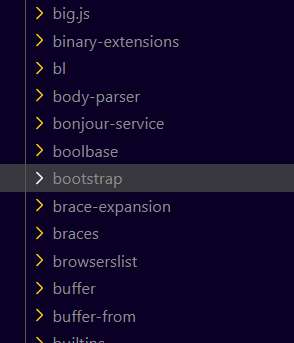
*<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ" crossorigin="anonymous">*

*<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/js/bootstrap.bundle.min.js" integrity="sha384-ENjdO4Dr2bkBIFxQpeoTz1HIcje39Wm4jDKdf19U8gI4ddQ3GYNS7NTKfAdVQSZe" crossorigin="anonymous"></script>*

***OR***

*$npm install* [*bootstrap@5.3.0-alpha3*](mailto:bootstrap@5.3.0-alpha3)

*To test bootstrap:*

**

*Under node\_modules*

*Also in package.json*



How to add bootstrap with your angular project

3 ways :

1 . Thru cdn in index.html

Under <head>

*<!-- <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-GLhlTQ8iRABdZLl6O3oVMWSktQOp6b7In1Zl3/Jr59b6EGGoI1aFkw7cmDA6j6gD" crossorigin="anonymous"> -->*

Above </body>

*<!-- <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js" integrity="sha384-w76AqPfDkMBDXo30jS1Sgez6pr3x5MlQ1ZAGC+nuZB+EYdgRZgiwxhTBTkF7CXvN" crossorigin="anonymous"></script> -->*

2. In style.css (implemented easy way)

@import "~bootstrap-5.3.0-alpha3/dist/css/bootstrap.css";

3. angular.json

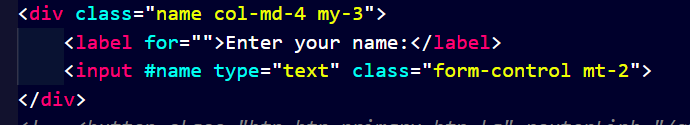
# **Welcome Logic**

As soon as the button <Start Quiz> is activated,

1. It will store the name in local storage and
2. display it throughout out project.

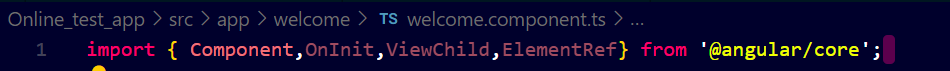
**Steps :**

**1. Welcome.html**

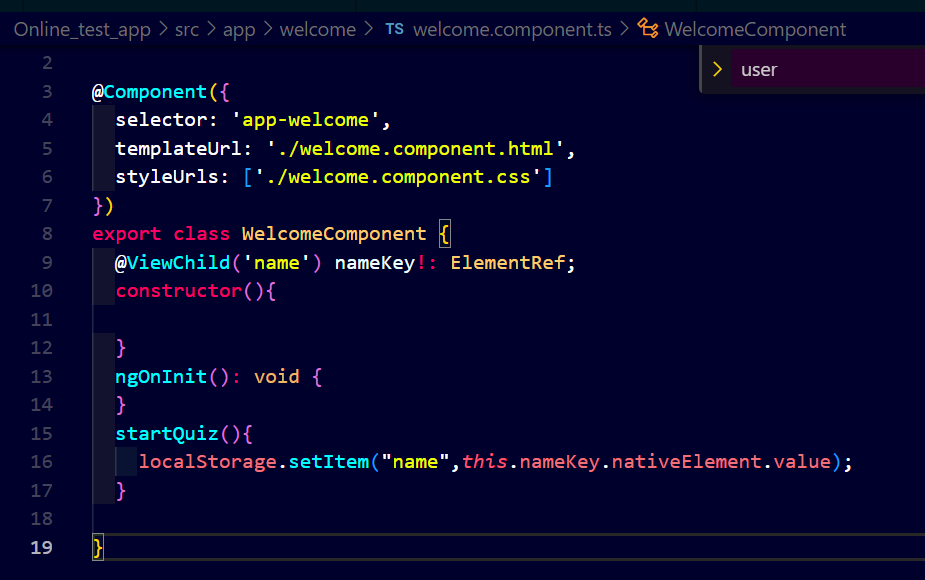




3. Welcome.ts



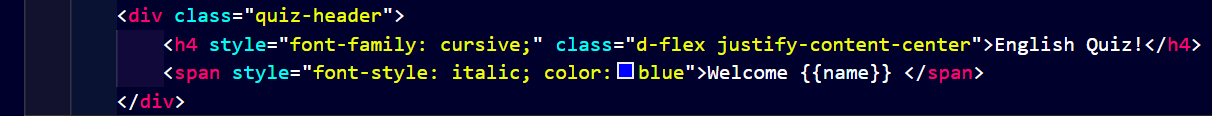
Set name



Get name as the user press button it will display the name



Display the Name from local storage



# For Question

Create a json file : questions.json in asset folder uses Array of Objects.

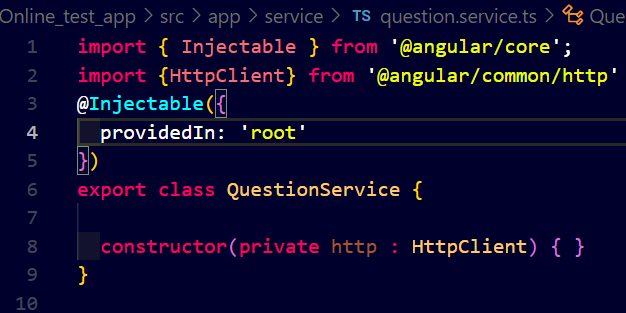
To call this json file , we need a service ,

Inside app foler we created a service folder inside which we create a question service.

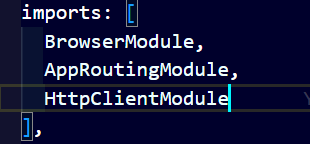
# 

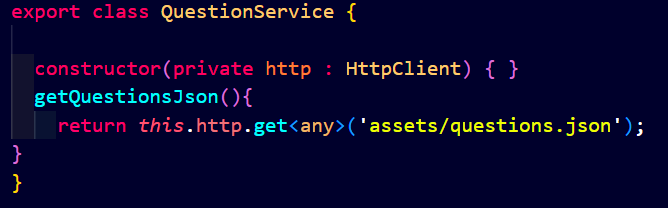
# 

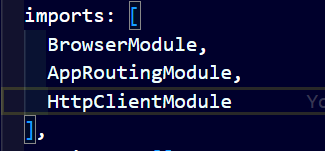
In question.service.ts



In file :app.module.ts

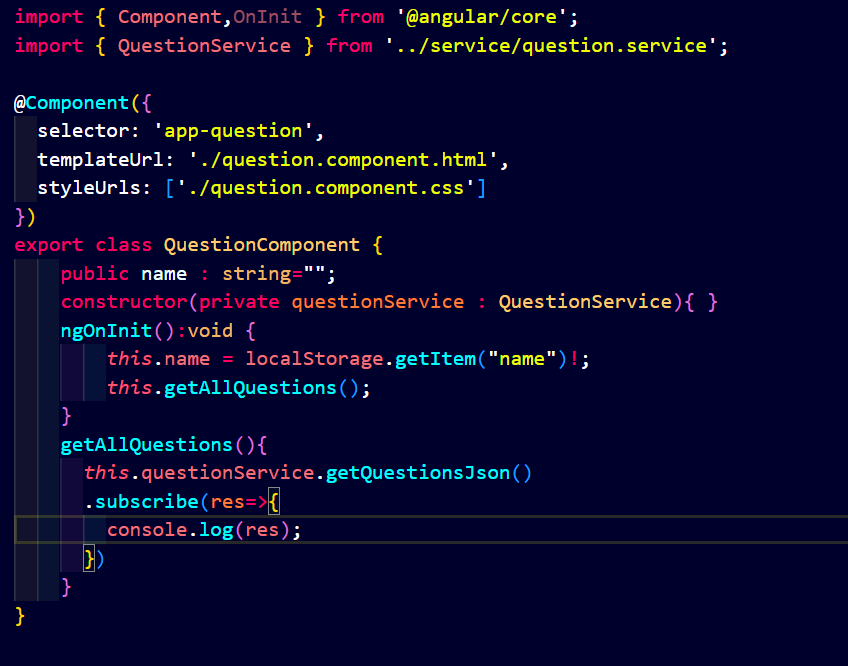




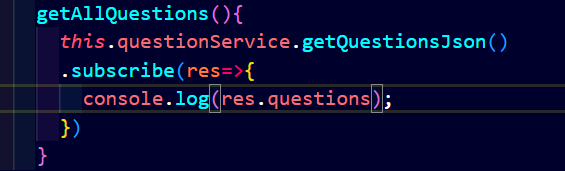


Question.component.ts

getAllQuestions()



Questions.component.ts



# Displaying all questions

# Putting all questions in questionlist

In question.component.ts >

public questionList: any=[];

    public currentQuestion:number =0;

*this*.questionList(res.questions);

export class QuestionComponent {

    public name : string="";

    public questionList: any=[];

    public currentQuestion:number =0;

    constructor(private questionService : QuestionService){ }

    ngOnInit():void {

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

*this*.getAllQuestions();

    }

    getAllQuestions(){

*this*.questionService.getQuestionsJson()

      .subscribe(res=>{

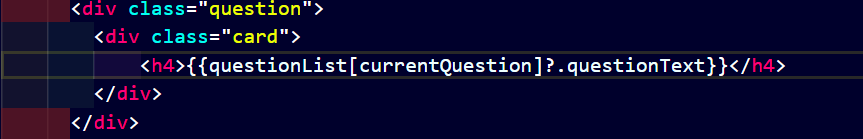
*this*.questionList(res.questions);

      })

    }

}

Question.component.html



# For timer counter and Points

Question.component.ts

Initialize



.html

# 

For next n current question in ts

# 

**Html:**

# 

Displaying Options :

Question.component.html

    <div class="options">

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

            <li>

                <div class="card">

                    {{option.text}}

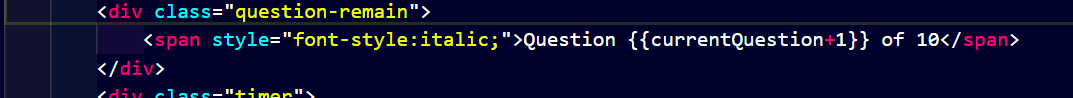
                </div>

            </li>

        </ol>

    </div>

Increment Question counter



Logical Part of Options (On correct ans Increment 10 points)

Question.html

Line 34

   <div class="options">

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

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

                <div class="card">

                    {{option.text}}

                </div>

            </li>

        </ol>

    </div>

Question.ts

 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;

    }

  }

Disabling the Previous button on Question 1

File: Question.html

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

Displaying Timer:

File : Question.ts

import { interval } from 'rxjs';

 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();

  }

Call Start Counter

ngOnInit(): void {

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

*this*.getAllQuestions();

*this*.startCounter();

  }

Reset Quiz on clicking reset btn

File : question.ts

 resetQuiz() {

*this*.resetCounter();

*this*.getAllQuestions();

*this*.points = 0;

*this*.counter = 60;

*this*.currentQuestion = 0;

*this*.progress = "0";

  }

For Progress Bar

File : Question.html

   <div class="progress mb-3">

        <div class="progress-bar progress-bar-striped" role="progressbar" [ngStyle]="{'width':progress+'%'}" aria-valuenow="55" aria-valuemin="0" aria-valuemax="100"></div>

      </div>

File : question.ts

getProgressPercent() {

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

    return *this*.progress;

  }

Changing the background option of correct ans:

At terminal :

ng g d change\_bg

delete spec file

File : change\_bg.directive

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');

    }

  }

}

question.html

        <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>

# Result display

File: question.html

<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>

# References:

<https://www.youtube.com/watch?v=ITPjL7L4sqU&t=1435s>