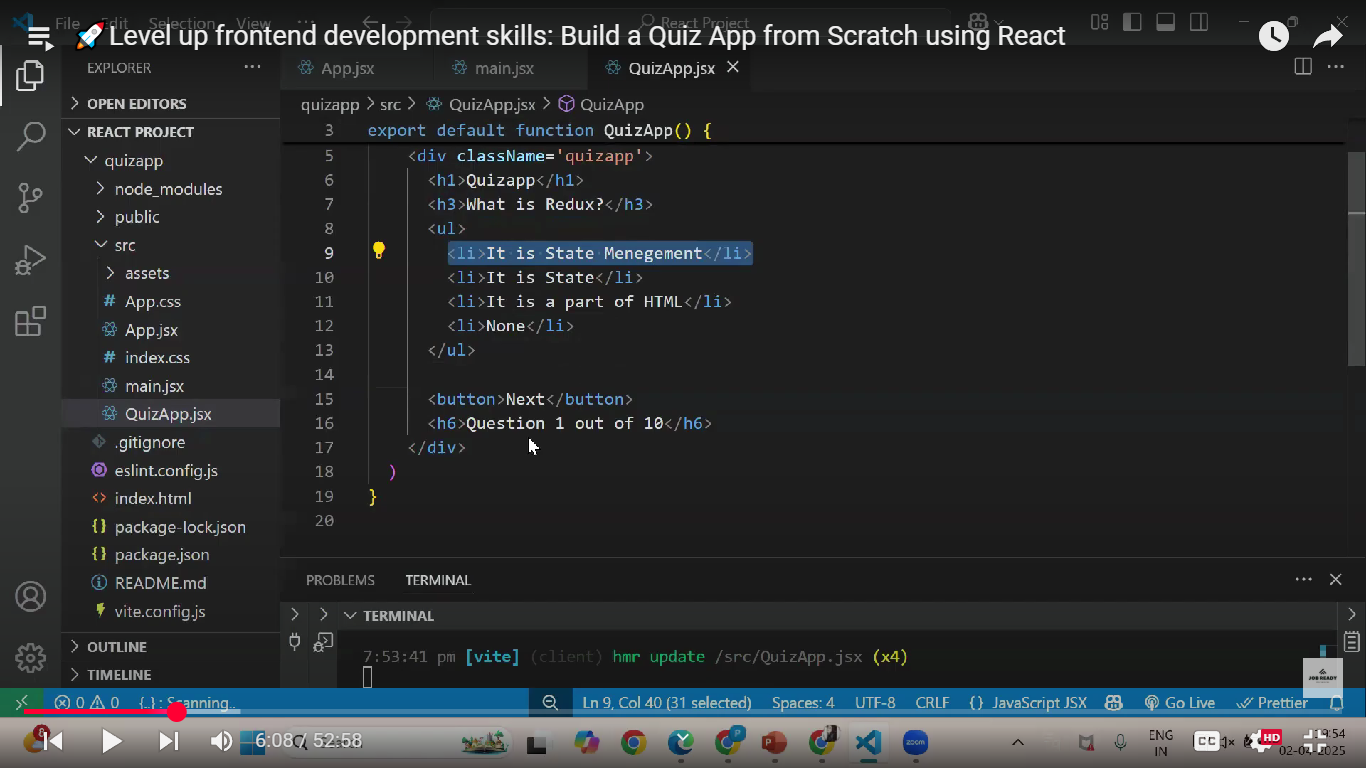
QuizApp

Create a **component** called **QuizApp** which is going to display all the quiz question and option

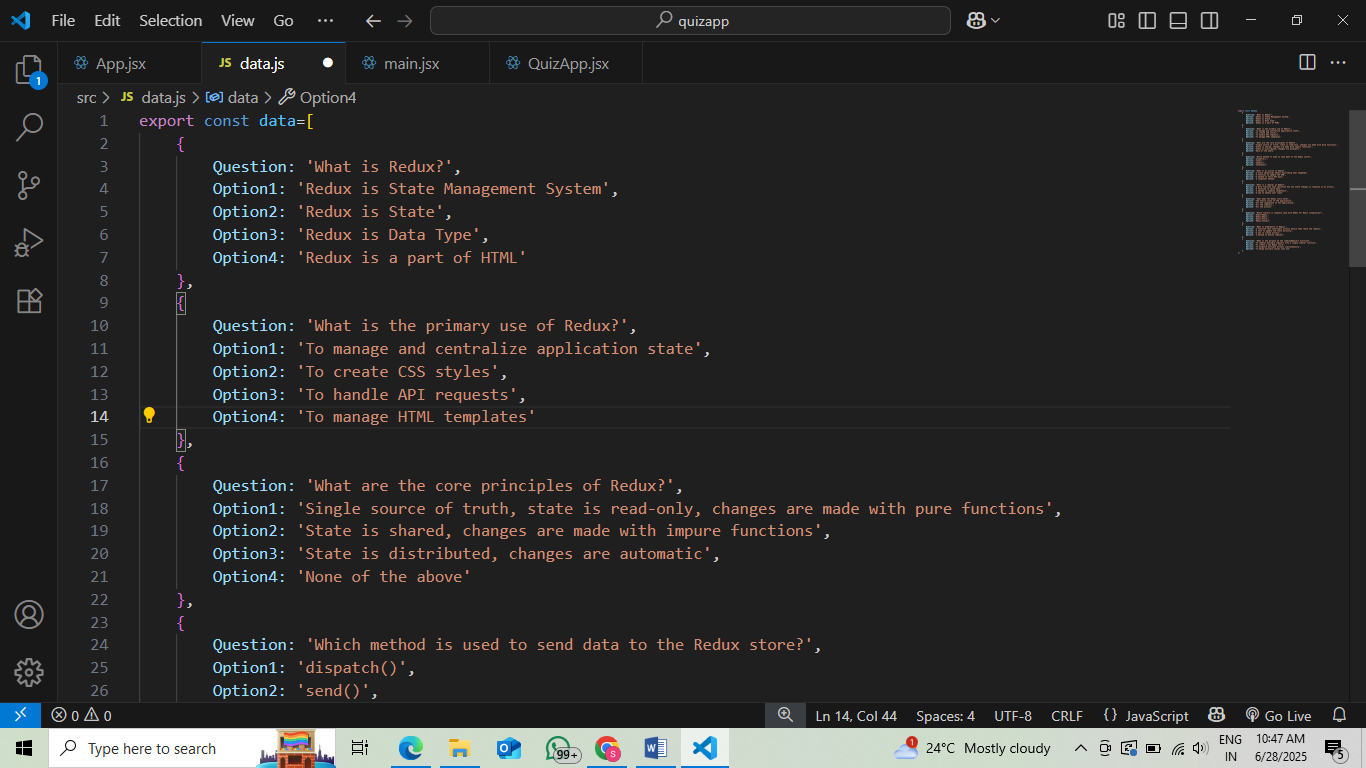
And yes we can create the questions inside these components but what will be diificulty is that the number of line of code will be more as shown below



Now instead of this what we can do is that we can create a **javascript** file where all the questions will be stored in the array as shown below

export const data=[{},{},{},{},{},{},{},{},{},{}];

This is how we need to add all the questions and It need to be in key value pair



Now we will import this to our **QuizApp component** and will make use of **state variable** why you will understand

import React, { useState } from 'react'

import {data} from './data';

 function QuizApp() {

    let[index,setIndex]=useState(0);

    function updateIndex(){

        setIndex(index+1)

    }

  return (

    <div>

        <h3>{data[index].Question}</h3>

        <ul>

        <li>{data[index].Option1}</li>

        <li>{data[index].Option2}</li>

        <li>{data[index].Option3}</li>

        <li>{data[index].Option3}</li>

        </ul>

        <button onClick={updateIndex}>Next</button>

        <h6>Question {index+1} of {data.length}</h6>

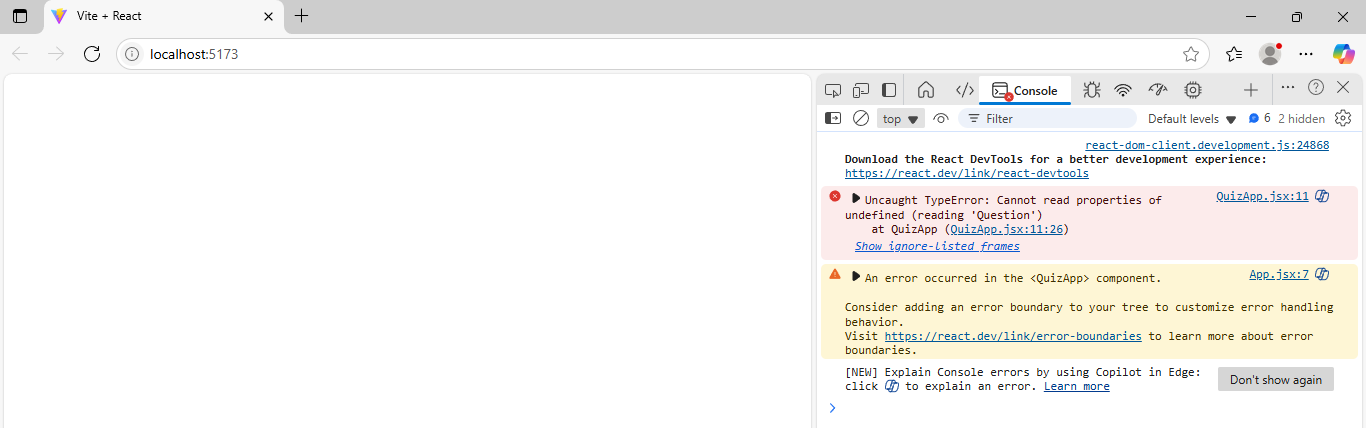
    </div>

  )

}

export default QuizApp;

But over here there is one **problem**  what is that if you ask me upon reaching the last question and when we click the next button since there is no questions is leftout **nothing** will be displayed on the screen but internally the console will throw the **error** as shown below



Now how to handle this bcz the array index went out of the range check the below code where one more **state variable** has been created

import React, { useState } from 'react'

import {data} from './data';

 function QuizApp() {

    let[index,setIndex]=useState(0);

    let[finish,setFinish]=useState(false);

    function updateIndex(){

        if(index<data.length-1){

            setIndex(index+1)}

        else{

            setFinish(true)}

    }

    if(finish){

        return(<div className='scorePage'>

            <h1>Quiz Finished</h1>

        </div>) }

  return (<div className='quiz'>

        <h3>{data[index].Question}</h3>

        <ul>

        <li>{data[index].Option1}</li>

        <li>{data[index].Option2}</li>

        <li>{data[index].Option3}</li>

        <li>{data[index].Option3}</li>

        </ul>

        <button onClick={updateIndex}>Next</button>

        <h6>Question {index+1} of {data.length}</h6>

    </div> )}

export default QuizApp;

For the above I have imported the CSS file

body{

    background-color:#B692C2;

    margin: 0px;padding: 0px;

  }

  .scorePage {

    text-align: center;

    background-color: #E3A5C7;

    border: 2px solid black;

    padding: 40px;

    border-radius: 15px;

    margin-top: 100px;

  }

  .quiz {

      margin: auto;

      margin-top: 50px;

      padding: 20px;

      background-color: #E3A5C7;

      width: 70%;

      border-radius: 10px;

      display: flex;

      align-items: center;

      flex-direction: column;

    }

    .quiz ul li {

      display: flex;

      align-items: center;

      border: 1px solid #694F8E;

      border-radius: 5px;

      padding: 10px;

      margin-bottom: 10px;

      cursor: pointer;

      background-color: #ece7e6;

      width: 500px;

    }

    .quiz ul li.selected {

      background-color: #694F8E;

      color: white;

    }

    .quiz button {

      padding: 10px;

      width: 400px;

      border-radius: 15px;

      background-color: #E49BFF;

      color: black;

      border: 1px solid black;

    }

    .quiz button:hover {

      background-color: aliceblue;

      border: 2px solid plum;

      cursor: pointer;

    }

Where the below line of code will act based on the option user will select

.quiz ul li.selected {

      background-color: #694F8E;

      color: white;

    }

So now what we need to do is that when the user click on the correct option it nee be highlighted let us write the logic for that

Below is the code

let[option,setOption]=useState('');

function selectedOption(opt){

        setOption(opt)

    }

return (<div className='quiz'>

        <h3>{data[index].Question}</h3>

        <ul>

        <li onClick={()=>selectedOption('Option1')}>{data[index].Option1}</li>

        <li onClick={()=>selectedOption('Option2')}>{data[index].Option2}</li>

        <li onClick={()=>selectedOption('Option3')}>{data[index].Option3}</li>

        <li onClick={()=>selectedOption('Option4')}>{data[index].Option4}</li>

        </ul>

        <button onClick={updateIndex}>Next</button>

        <h6>Question {index+1} of {data.length}</h6>

    </div> )}

So what will happen when the user click on the option the **option**  variable will be getting the value as the option user has selected

Now what we need to do based on the option selected when we need to set the **classname** as **selected**  so that the styling which we mentioned will be applied

So for this we will make use of **ternary operator**

className={option=='Option1' ? 'selected':''}

So what will happen is that if the **option** variable is having the value as **Option1** then we need to give the **className** as **selected** else ‘ ‘ (empty string).

**Same thing will apply for all the li tag which has the option**

return (<div className='quiz'>

        <h3>{data[index].Question}</h3>

        <ul>

        <li className={option=='Option1'?'selected':''} onClick={()=>selectedOption('Option1')}>{data[index].Option1}</li>

        <li  className={option=='Option2'?'selected':''} onClick={()=>selectedOption('Option2')}>{data[index].Option2}</li>

        <li  className={option=='Option3'?'selected':''} onClick={()=>selectedOption('Option3')}>{data[index].Option3}</li>

        <li  className={option=='Option4'?'selected':''} onClick={()=>selectedOption('Option4')}>{data[index].Option4}</li>

        </ul>

        <button onClick={updateIndex}>Next</button>

        <h6>Question {index+1} of {data.length}</h6>

    </div> )}

Now one problem will be there what is that if you ask me say suppose in the first question you have selected the 3rd option so when you click next by default in the 2nd question also the 3rd option will selected by default

This is happening bcz option variable has the value of previous option selected

So now what we can do is that in the onclick even function of **next** button there we can set the **option state variable** value as ‘ ’

function QuizApp() {

    let[index,setIndex]=useState(0);

    let[finish,setFinish]=useState(false);

    let[option,setOption]=useState('');

    function updateIndex(){

        if(index<data.length-1){

            setIndex(index+1)

        setOption('');

        }

        else{

            setFinish(true)}

    }

Now we need to calculate the **final score** which will be displayed at the end

So do that what we will be doing we will create a array which is going to store all the **correct answers**

So what we will do in the array is that we will store the correct option of each questions in their particular index

For example if the correct answer of **question1** is **option2** so the question1 is present in the 0th index so the correct option for that question also I will store in the 0th index of the array I have created.

In a same for the question2 is the correct option is 3 then in the 1st index of the array I have created I will store option3

    const correctOptions=['Option2','Option3','Option1','Option1','Option2','Option2','Option2','Option2','Option2','Option2']

After that what we need to do is that as we are aware that the correct option will be stored inside variable called **option** so when the user click on the next button we need to check whether the option selected by the user which is stored inside the **option** variable is t **matching** with the **array** I have created

Also for this let us make use of **score state variable** which is going to hold the score

let[score,setScore]=useState(0)

function updateIndex(){

if(correctOptions[index]==option)

            {

                setScore(score+1);

            }

        if(index<data.length-1){

            setIndex(index+1)

        setOption('');

        }

        else{

            setFinish(true)}

    }

if(finish && score>5){

        return(<div className='scorePage'>

            <h1>Quiz if finished</h1>

            <h1>Youre score is {score} congratualations</h1>

        </div>) }

   if(finish && score<=5)

   {

    return(<div className='scorePage'>

            <h1>Quiz if finished</h1>

            <h1>Youre score is {score} better luck</h1>

        </div>)

   }

**Below is the full code QuizApp component**

import React, { useState } from 'react'

import {data} from './data';

import './Style.css'

 function QuizApp() {

    let[index,setIndex]=useState(0);

    let[finish,setFinish]=useState(false);

    let[option,setOption]=useState('');

    let[score,setScore]=useState(0)

    const correctOptions=['Option2','Option3','Option1','Option1','Option2','Option2','Option2','Option2','Option2','Option2']

    function updateIndex(){

         if(correctOptions[index]==option)

            {

                setScore(score+1);

            }

        if(index<data.length-1){

            setIndex(index+1)

        setOption('');

        }

        else{

            setFinish(true)}

    }

    function selectedOption(opt){

        setOption(opt)

    }

    if(finish && score>5){

        return(<div className='scorePage'>

            <h1>Quiz if finished</h1>

            <h1>Youre score is {score} congratualations</h1>

        </div>) }

   if(finish && score<=5)

   {

    return(<div className='scorePage'>

            <h1>Quiz if finished</h1>

            <h1>Youre score is {score} better luck</h1>

        </div>)

   }

  return (<div className='quiz'>

        <h3>{data[index].Question}</h3>

        <ul>

        <li className={option=='Option1'?'selected':''} onClick={()=>selectedOption('Option1')}>{data[index].Option1}</li>

        <li  className={option=='Option2'?'selected':''} onClick={()=>selectedOption('Option2')}>{data[index].Option2}</li>

        <li  className={option=='Option3'?'selected':''} onClick={()=>selectedOption('Option3')}>{data[index].Option3}</li>

        <li  className={option=='Option4'?'selected':''} onClick={()=>selectedOption('Option4')}>{data[index].Option4}</li>

        </ul>

        <button onClick={updateIndex}>Next</button>

        <h6>Question {index+1} of {data.length}</h6>

    </div> )}

export default QuizApp;