Mini Project: Simple JavaScript Quiz App

WEEK 1:

quiz.html:

```
<!DOCTYPE html>
<html lang="en">
 <head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>MCQ Quiz</title>
   <!-- Google Font -->
    k
     href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&disp
lay=swap"
     rel="stylesheet"
    <!-- Stylesheet -->
    <link rel="stylesheet" href="quiz.css" />
 </head>
 <body>
   <div class="image">
    <div class="start-screen">
     <button id="start-button">Start
    </div>
   <div id="display-container">
     <div class="header">
        <div class="number-of-count">
          <span class="number-of-question">1 of 4 questions
       </div>
        <div class="timer-div">
<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16"</pre>
fill="currentColor" class="bi bi-stopwatch-fill" viewBox="0 0 16 16">
    <path d="M6.5 0a.5.5 0 0 0 0 1H7v1.07A7.001 7.001 0 0 0 8 16a7 7 0 0 0</pre>
5.29-11.584.531.531 0 0 0 .013-.0121.354-.354.353.354a.5.5 0 1 0 .707-.7071-
1.414-1.415a.5.5 0 1 0-.707.707l.354.354-.354.354a.717.717 0 0 0-
.012.012A6.973 6.973 0 0 0 9 2.071V1h.5a.5.5 0 0 0 0-1h-3zm2 5.6V9a.5.5 0 0 1-
.5.5H4.5a.5.5 0 0 1 0-1h3V5.6a.5.5 0 1 1 1 0z"/>
 </svg>
          <span class="time-left">20s</span>
        </div>
     </div>
     <div id="container">
        <!-- questions and options will be displayed here -->
     </div>
     <button id="next-button">Next
```

quiz.css:

```
padding: 0;
  margin: 0;
  box-sizing: border-box;
  font-family: "Poppins", sans-serif;
.start-screen,
.score-container {
 position: absolute;
  top: 0;
 width: 100%;
 height: 100%;
 display: flex;
 flex-direction: column;
  align-items: center;
  justify-content: center;
button {
  border: navy;
 outline: grey;
  cursor: pointer;
#start-button,
#restart {
 font-size: 1.3em;
 padding: 0.5em 1.8em;
 border-radius: 0.2em;
#restart {
 margin-top: 0.9em;
#display-container {
```

```
background-color: lighter;
 *{
 padding: 3.1em 1.8em;
 width: 80%;
 max-width: 37.5em;
 margin: 0 auto;
 position: absolute;
 transform: translate(-50%, -50%);
 top: 50%;
 left: 50%;
 border-radius: 0.6em;
.header {
 margin-bottom: 1.8em;
 display: flex;
 justify-content: space-between;
 align-items: center;
 padding-bottom: 0.6em;
 border-bottom: 0.1em solid #949644;
.timer-div {
 background-color: #5da1c0;
 width: 7.5em;
 border-radius: 1.8em;
 display: flex;
 align-items: center;
 justify-content: space-between;
 padding: 0.79em 2.5em;
.question {
 margin-bottom: 1.25em;
 font-weight: 600;
.option-div {
 font-size: 0.9em;
 width: 100%;
 padding: 1em;
 margin: 0.3em 0;
 text-align: left;
 outline: none;
 background: transparent;
 border: 1px solid #615e94;
 border-radius: 0.3em;
.option-div:disabled {
 color: #eeeff0;
 cursor: not-allowed;
```

```
#next-button {
  font-size: 1em;
  margin-top: 1.5em;
  background-color: #3d8f5d;
  color: #ffffff;
  padding: 0.7em 1.8em;
  border-radius: 0.3em;
  float: right;
.hide {
  display: none;
.incorrect {
  background-color: #4c24bb;
  color: #2608cf;
  border-color: #300eee;
.correct {
  background-color: #9cd25b;
  color: #7fda30;
  border-color: #689f38;
#user-score {
  font-size: 1.5em;
  color: #855959;
.image{
  background-color: grey;
  background-size: cover;
  top: 0;
  left: 0;
 width: 100%;
  position: fixed;
  height: 100%;
  color: white;
```

quiz.js:

```
//References
let timeLeft = document.querySelector(".time-left");
let quizContainer = document.getElementById("container");
let nextBtn = document.getElementById("next-button");
let countOfQuestion = document.querySelector(".number-of-question");
let displayContainer = document.getElementById("display-container");
let scoreContainer = document.querySelector(".score-container");
```

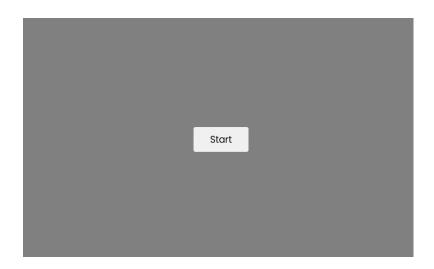
```
let restart = document.getElementById("restart");
let userScore = document.getElementById("user-score");
let startScreen = document.querySelector(".start-screen");
let startButton = document.getElementById("start-button");
let questionCount;
let scoreCount = 0;
let count = 20;
let countdown;
//Questions and Options array
const quizArray = [
 {
    id: "0",
    question: "Which is reserved word in the java programming language?",
    options: ["method","native","reference","array","subclass"],
    correct: "native",
  },
    id: "1",
    question: "Which is a valid key word in java?",
    options: ["interface", "string", "float", "unsigned", "int"],
    correct: "interface",
  },
    id: "2",
    question: "Which keyword is used for Debugging in java?",
    options: ["Abstract","insert","assert","boolean"],
    correct: "assert",
  },
    id: "3",
    question: "Among the following is not a popular IDE for java developmen?",
    options: ["NetBeans","intelliJ IDEA","Visual studio code","Eclpse"],
    correct: "Visual studio code",
 },
];
//Restart Quiz
restart.addEventListener("click", () => {
 initial();
 displayContainer.classList.remove("hide");
 scoreContainer.classList.add("hide");
});
//Next Button
nextBtn.addEventListener(
 "click",
```

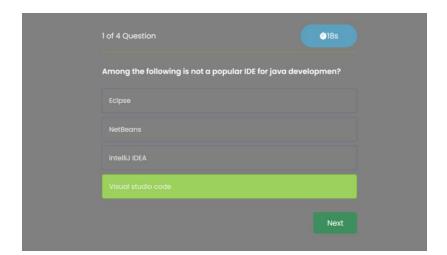
```
(displayNext = () => {
    //increment questionCount
    questionCount += 1;
    //if last question
    if (questionCount == quizArray.length) {
      //hide question container and display score
      displayContainer.classList.add("hide");
      scoreContainer.classList.remove("hide");
      //user score
      userScore.innerHTML =
        "Your score is " + scoreCount + " out of " + questionCount;
    } else {
      //display questionCount
      countOfQuestion.innerHTML =
        questionCount + 1 + " of " + quizArray.length + " Question";
      //display quiz
      quizDisplay(questionCount);
      count = 20;
      clearInterval(countdown);
      timerDisplay();
 })
);
//Timer
const timerDisplay = () => {
 countdown = setInterval(() => {
    count--;
    timeLeft.innerHTML = `${count}s`;
    if (count == 0) {
      clearInterval(countdown);
      displayNext();
  }, 1000);
};
//Display quiz
const quizDisplay = (questionCount) => {
  let quizCards = document.querySelectorAll(".container-mid");
 //Hide other cards
  quizCards.forEach((card) => {
   card.classList.add("hide");
  });
  //display current question card
 quizCards[questionCount].classList.remove("hide");
};
//Quiz Creation
```

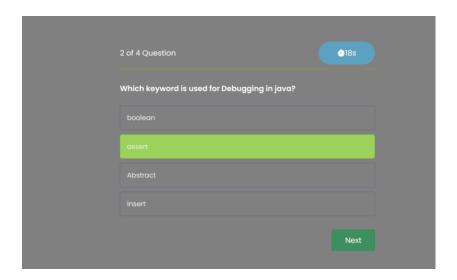
```
function quizCreator() {
  //randomly sort questions
  quizArray.sort(() => Math.random() - 0.5);
  //generate quiz
  for (let i of quizArray) {
    //randomly sort options
    i.options.sort(() => Math.random() - 0.5);
    //quiz card creation
    let div = document.createElement("div");
    div.classList.add("container-mid", "hide");
    //question number
    countOfQuestion.innerHTML = 1 + " of " + quizArray.length + " Question";
    //question
    let question_DIV = document.createElement("p");
    question DIV.classList.add("question");
    question DIV.innerHTML = i.question;
    div.appendChild(question_DIV);
    //options
    div.innerHTML += `
    <button class="option-div"</pre>
onclick="checker(this)">${i.options[0]}</button>
     <button class="option-div"</pre>
onclick="checker(this)">${i.options[1]}</button>
      <button class="option-div"</pre>
onclick="checker(this)">${i.options[2]}</button>
       <button class="option-div"</pre>
onclick="checker(this)">${i.options[3]}</button>
   quizContainer.appendChild(div);
//Checker Function to check if option is correct or not
function checker(userOption) {
  let userSolution = userOption.innerText;
 let question =
    document.getElementsByClassName("container-mid")[questionCount];
  let options = question.querySelectorAll(".option-div");
  //if user clicked answer == correct option stored in object
 if (userSolution === quizArray[questionCount].correct) {
    userOption.classList.add("correct");
   scoreCount++;
  } else {
    userOption.classList.add("incorrect");
    //For marking the correct option
   options.forEach((element) => {
```

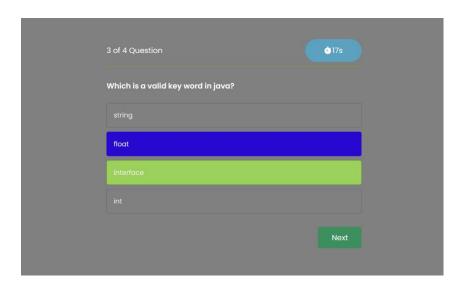
```
if (element.innerText == quizArray[questionCount].correct) {
        element.classList.add("correct");
   });
  //clear interval(stop timer)
  clearInterval(countdown);
  //disable all options
  options.forEach((element) => {
    element.disabled = true;
  });
//initial setup
function initial() {
  quizContainer.innerHTML = "";
  questionCount = 0;
  scoreCount = 0;
  count = 20;
  clearInterval(countdown);
  timerDisplay();
 quizCreator();
  quizDisplay(questionCount);
//when user click on start button
startButton.addEventListener("click", () => {
  startScreen.classList.add("hide");
  displayContainer.classList.remove("hide");
 initial();
});
//hide quiz and display start screen
window.onload = () => {
  startScreen.classList.remove("hide");
  displayContainer.classList.add("hide");
```

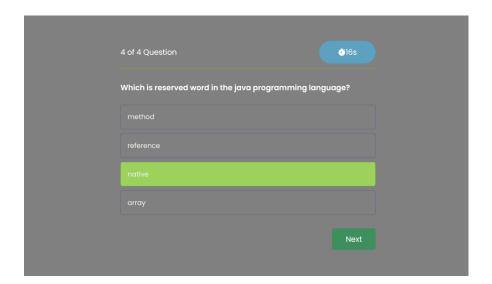
Output:













GITHUB URL:

https://github.com/Asha-Asha/quiz-application-in-js/tree/2609c73786a7f1905e2ad12fb8a2924e07058041/week1%20project