

# Quiz Project

submission By Dinesh Raut

Repository URL: <https://github.com/Dineshraut1903/Online-quiz-application.git>

Date of Submission 19/01/2023

## Technologies used:

Html, CSS, JavaScript, JSON

## Html

### Index.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Java Quiz</title>
  <link rel="stylesheet" href="css/app.css">
  <link href="https://fonts.googleapis.com/css2?family=Lato:wght@300;400&display=swap"
rel="stylesheet">
</head>

<body>
  <div class="container">
    <div id="home" class="flex-center flex-column">
      <h1>Java Quiz</h1>
      <a class="btn" href="html/quiz.html">Start</a>
    </div>
  </div>
</body>
</html>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

quiz.html



```
</body>
</html>
```

### end.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <meta http-equiv="X-UA-Compatible" content="ie=edge" />
  <title>Java Quiz</title>
  <link href="https://fonts.googleapis.com/css2?family=Lato:wght@300;400&display=swap"
rel="stylesheet">
  <link rel="stylesheet" href="../css/app.css" />
</head>

<body>
  <div class="container">
    <div id="end" class="flex-center flex-column">
      <h1 id="finalScore"></h1>

      <a class="btn" href="quiz.html">Start Again</a>
      <a class="btn" href="/">Back to Home</a>
    </div>
  </div>
  <script src="../script/end.js"></script>
</body>
</html>
```

CSS

### app.css

```
:root{ background: rgb(97, 103, 15); background: linear-gradient(90deg,
rgb(232, 180, 10) 100%, rgba(0,212,255,1)
100%, rgb(36, 22, 0) 100%); font-
size: 58%;
} *{ box-sizing: border-box;
font-family: 'Lato', sans-serif;
margin: 0; padding: 0;
color: #333; font-weight: 400;
color: white;
}

.container{ max-
width: 1140px;
margin: 0 auto;
} h1, h2, h3, h4 {
margin-bottom: 1rem;
} h1{ font-size:
6rem; color: rgb(94,
2, 2); margin-
bottom: 5rem;
} h1 > span{
font-size: 2.4rem;
font-weight: 500;
} h2{
font-size: 4.2rem; margin-
bottom: 4rem; font-weight: 700;
} h3{ font-size:
2.8rem; font-
weight: 500;
}

.container{ width:
100vw; height: 90vh;
display: flex; justify-
content: center;
```

```
    align-items: center; max-
width: 80rem;
}
.container >*{
width: 100%;
}

.flex-column {
display: flex;
    flex-direction: column;
}

.flex-center {    justify-
content: center; align-items:
center;
}

.justify-center {
    justify-content: center;
}

.text-center {    text-
align: center;
}

.hidden {
display: none;
}

.btn {    font-size: 2rem;
padding: 1rem 0;    width:
30rem;    text-align: center;
border: 0.1rem solid #bef80e;
margin-bottom: 2.5rem;    text-
decoration: none;    color:
rgb(34, 16, 228);
background-color:#22e622;
border-radius: 15px;    font-
weight: 400;
}

.btn:hover {    cursor: pointer;    box-shadow: 0
0.4rem 1.4rem 0 rgba(86, 185, 235, 0.5);    transform:
translateY(-0.1rem);    transition: transform 150ms;
}

.btn[disabled]:hover {
cursor: not-allowed;    box-
shadow: none;    transform:
none;
}
```

```
    form {      width: 100%;
display: flex;    flex-
direction: column;
align-items: center;
    }      input {      color:#e67e22;      margin-bottom:
2.5rem;      width: 30rem;      padding: 1.2rem;      font-
size: 1.8rem;      border: none;      box-shadow: 0 0.1rem
1.4rem 0 rgba(235, 133, 86, 0.5);
    }
input::placeholder {
color: #aaa;
    }
```

quiz.css

```
*{  font-family: 'Lato',sans-
serif;  word-spacing: 2.5px;
font-weight: 400;  letter-
spacing: 1.5px;
}  h2{      font-
weight:      400;
color:  black;
}

.container{  max-
width: 1140px;
margin: 0 auto;
}

.choice-container {      display: flex;
margin-bottom: 1.5rem;      width: 100%;
font-size: 1.8rem;      border: 0.1rem solid
rgb(86, 165, 235, 0.25);      background-color:
black;      border-radius: 15px;
}

.choice-container:hover {      cursor: pointer;      box-
shadow: 0 0.4rem 1.4rem 0 rgba(104, 50, 210, 0.5);
transform: translateY(-0.1rem);      transition: transform
150ms;
```



```

}

.choice-prefix {    padding:
1.5rem 2.5rem;      background-
color: #e67e22;      color: rgb(49,
21, 232);          border-top-left-
radius: 15px;      border-bottom-
left-radius: 15px;
}

.choice-text {
padding: 1.5rem;
width: 100%;
}

.correct {          background-
color: #4e7e8a;
}

.incorrect {        background-
color: #801336;
}

#hud {              display: flex;
justify-content: space-between;
text-transform: uppercase;
}

.hud-prefix {        text-
align: center;      font-
size: 2rem;
color: rgb(49, 21, 232);
}

.hud-main-text {      text-
align: center;
color:black;
}

```

Javascript:

quiz.js

```

const question = document.getElementById("question"); const choices =
Array.from(document.getElementsByClassName("choice-text")); const
questionCounterText = document.getElementById("questionCounter"); const
scoreText = document.getElementById("score");
let currentQuestion = {};
let acceptingAnswers = false;
let score = 0; let
questionCounter = 0; let
availableQuestions = [];
let questions = [];
fetch("../json/questions.json")
  .then(res => {
return res.json();
  })
  .then(loadedQuestions => {
console.log(loadedQuestions);    questions
= loadedQuestions;    startQuiz();
  });

const CORRECT_BONUS = 10; const
MAX_QUESTIONS = 5;
startQuiz = () => {
questionCounter = 0;    score = 0;
availableQuestions = [...questions];
getNewQuestion();
};
getNewQuestion = () => {    if (availableQuestions.length === 0 ||
questionCounter >= MAX_QUESTIONS) {
localStorage.setItem('mostRecentScore', score);    return
window.location.assign("../html/end.html");
  }
  questionCounter++;
  questionCounterText.innerText = `${questionCounter}/${MAX_QUESTIONS}`;
  const questionIndex = Math.floor(Math.random() *
availableQuestions.length);    currentQuestion =
availableQuestions[questionIndex];    question.innerText =
currentQuestion.question;
  choices.forEach(choice => {    const number =
choice.dataset["number"];    choice.innerText =
currentQuestion["choice" + number];    });
  availableQuestions.splice(questionIndex, 1);
  acceptingAnswers = true;
};

```

```

    choices.forEach(choice => {
choice.addListener("click", e => {
if (!acceptingAnswers) return;
    acceptingAnswers = false;    const selectedChoice =
e.target;    const selectedAnswer =
selectedChoice.dataset["number"];
    const classToApply =    selectedAnswer == currentQuestion.answer
? "correct" : "incorrect";
    if (classToApply === "correct")
{
incrementScore(CORRECT_BONUS);
    }
selectedChoice.parentElement.classList.add(classToApply);
    setTimeout(() => {
selectedChoice.parentElement.classList.remove(classToApply);
getNewQuestion();
    }, 400);
    });
}); incrementScore = num =>
{
    score += num;
scoreText.innerText = score;
};

```

end.js

```

const username = document.getElementById('username'); const
saveScoreBtn = document.getElementById('saveScoreBtn'); const
finalScore = document.getElementById('finalScore'); const
mostRecentScore = localStorage.getItem('mostRecentScore');
finalScore.innerText = "Total Score is: "+mostRecentScore+ "
points!";

username.addListener('keyup', () => {
saveScoreBtn.disabled = !username.value;
}); saveHighScore =
(e) => {
    e.preventDefault();
};

```

## JSON

questions.json

```
[
{
  "question": "Which component is used to compile, debug and execute the java programs?",
  "choice1": "JRE",
  "choice2": "JIT",
  "choice3": "JDK",
  "choice4": "JVM",
  "answer": 3
},
{
  "question": "Which one of the following is not a Java feature?",
  "choice1": "Object-oriented",
  "choice2": "Use of pointers",
  "choice3": "Portable",
  "choice4": "Dynamic and Extensible",
  "answer": 2
},
{
  "question": "Which of these cannot be used for a variable name in Java",
  "choice1": "identifier & keyword",
  "choice2": "identifier",
  "choice3": "keyword",
  "choice4": "none of the mentioned",
  "answer": 3
},
{
  "question": "Which of the following is not an OOPS concept in Java?",
  "choice1": "Polymorphism",
  "choice2": "Inheritance",
  "choice3": "Compilation",
  "choice4": "Encapsulation",
  "answer": 3
},
{
  "question": "What is not the use of “this” keyword in Java?",
  "choice1": "Referring to the instance variable when a local variable has the same name",
  "choice2": "Passing itself to the method of the same class",
  "choice3": "Passing itself to another method",
  "choice4": "Calling another constructor in constructor chaining",
  "answer": 2
}
]
```

