

Purpose of Quiz

The purpose of my quiz is to test and extend the knowledge of my Year 11 classmates of the TV show: The Rookie. It is also to test my coding skills for this assessment.

Style of question (e.g. multiple choice, short answer)

I chose to use multiple choice questions for my quiz.

Example question and answer

What year did 'The Rookie' first air?

- (a) 2018
- (b) 2019
- (c) 2020
- (d) 2021

Scoring System

1 point per question answered correctly.

When Quiz end

The Quiz ends when the user has completed all the questions.

Boundary conditions I could test

If a user doesn't enter an answer, prompt them to put an input.

Test Case ID	Input Description	Input Provided	Expected Output	Actual Output	Pass /Fail	Changes Made

Q1	Wrong Answer (2019)	Wrong Answer (2019)	It should tell the user that their answer is incorrect and tell the user which one is correct. It should also highlight red.	It tells the user that its answer was incorrect and shows the correct answer, it goes to the next question.	Pass	No changes required
Q1	Correct Answer	'2018'	It should tell the user that their answer is correct.	It comes up with the expected output	Pass	No Changes required
Q1	No input	N/A	The page should prompt the user to select an option.	It pops up with 'Please select an option!'	Pass	No changes required
Q1-7	All Questions correct	All the correct answers	It should come up with a perfect score (7/7). It should say 'You Passed!' The reset button should pop up.	It comes up with the expected output, but 'Correct!' still stays	Pass	No changes required except the reset button looks weird
Q1-7	All Questions incorrect	All the incorrect answers	It should tell the user its score which should be 0/7. Reset button should pop up	It comes up with the expected output.	Pass	No changes required except the reset button looks weird

Reset button	Using the reset button	Pressing the reset button	It should either go back to the start button or go back to the first question.	It went back to the first question of the quiz	Pass	No Changes Required
Next Button		Pressing the next button multiple times	It should stop at the next question	The expected outcome happened because I added a function that gives the next button a cooldown after pressed.	Pass	No changes required
4/7 Correct	4 questions correct	Getting 4 correct answers	It should tell the user that they passed and tell them their score which should be 4/7.	It tells the user their score: 4/7 so the expected outcome happened.	Pass	No changes required
3/7 Correct	3 questions correct	Getting 3 answers correct	It should tell the user that they failed and show their score which should be 3/7.	The expected outcome happens	Pass	No changes required
Repeating the quiz	Pass first time (7/7), fail second time (0/7)	Pass first time, reset and fail second time	They should be different results	The expected outcome happens	Pass	No changes required

Ongoing Improvements:

I need to fix some of my code but I would say that I have completed most of what I have wanted to achieve with my quiz and code. I would like to fix the button placements because it looks weird.

Were Variables used in JavaScript code?

1. currentQuestionIndex

- **What it is:** currentQuestionIndex is a variable that stores the index of the current question being displayed to the user.
- **What it does:** This variable is initialised with a value of 0, meaning the quiz starts with the first question. Each time the user answers a question and proceeds to the next one, the value of currentQuestionIndex is incremented by 1. This allows the program to track which question should be shown next from the questions array.

2. score

- **What it is:** score is a variable that keeps track of the user's correct answers throughout the quiz.
- **What it does:** This variable starts at 0 and increments by 1 every time the user answers a question correctly. By the end of the quiz, score holds the total number of correct answers. This value is then used to determine if the user has passed or failed the quiz, based on a comparison with the pass mark.

JavaScript Code:

```
// Quiz questions stored in an array of objects

const questions = [

  {

    question: "Which year was 'The Rookie' first aired?",
```

```
options: ["2018", "2019", "2020"],

answer: "2018",

feedback: "The show first aired in 2018."

},

{

question: "Who plays the main character, John Nolan?",

options: ["Nathan Fillion", "Eric Winter", "Alyssa Diaz"],

answer: "Nathan Fillion",

feedback: "Nathan Fillion stars as John Nolan."

},

{

question: "What is John Nolan's occupation before joining the LAPD?",

options: ["Lawyer", "Construction Worker", "Graphic Designer"],

answer: "Construction Worker",

feedback: "John Nolan was a construction worker before joining the LAPD."

},
```

```
{  
  
  question: "Which city does 'The Rookie' take place in?",  
  
  options: ["Los Angeles", "New York", "Chicago"],  
  
  answer: "Los Angeles",  
  
  feedback: "The show is set in Los Angeles."  
  
},  
  
{  
  
  question: "Which network airs 'The Rookie'?",  
  
  options: ["ABC", "CBS", "NBC"],  
  
  answer: "ABC",  
  
  feedback: "The Rookie airs on ABC."  
  
},  
  
{  
  
  question: "Which country was 'The Rookie' filmed in?",  
  
  options: ["Australia", "New Zealand", "Canada"],  
  
  answer: "New Zealand",
```

```
        feedback: "The show was filmed in New Zealand."

    },

    {

        question: "What is John Nolan's badge number?",

        options: ["001", "007", "180"],

        answer: "180",

        feedback: "John Nolan's badge number is 180."

    },

];

// Variables to keep track of current question index and score

let currentQuestionIndex = 0;

let score = 0;

const passMark = Math.ceil(questions.length / 2);

// Function to update the progress bar
```

```
function updateProgressBar() {

    const progressBar = document.getElementById('progress-bar');

    const progress = ((currentQuestionIndex + 1) / questions.length) * 100;

    progressBar.style.width = progress + '%';

}

// Function to display the current question

function showQuestion() {

    const questionContainer = document.getElementById('question-container');

    const currentQuestion = questions[currentQuestionIndex];

    questionContainer.innerHTML = `<h2>${currentQuestion.question}</h2>`;

    questionContainer.innerHTML += createOptionsHtml(currentQuestion.options);

    updateProgressBar();

}
```



```
// Function to create HTML for options

function createOptionsHtml(options) {

    return options.map(option => `

        <div>

            <input type="radio" name="option" value="${option}">

            <label>${option}</label>

        </div>

    `).join('');

}

// Function to handle feedback display

function displayFeedback(isCorrect, feedback) {

    const feedbackDiv = document.getElementById('feedback');

    feedbackDiv.textContent = isCorrect ? "Correct! " + feedback : `Wrong! ${feedback}`;

    feedbackDiv.className = isCorrect ? 'feedback correct' : 'feedback incorrect';

}
```

```
// Function to move to the next question

function nextQuestion() {

    const nextButton = document.getElementById('next-button');

    nextButton.disabled = true; // Disable the button


    const selectedOption = document.querySelector('input[name="option"]:checked');

    const feedbackDiv = document.getElementById('feedback');


    // Ensure the user has selected an option

    if (!selectedOption) {

        alert('Please select an option!');

        nextButton.disabled = false; // Re-enable the button if no option selected

        return;

    }

}
```

```
const selectedAnswer = selectedOption.value;

const currentQuestion = questions[currentQuestionIndex];

const isCorrect = selectedAnswer === currentQuestion.answer;

if (isCorrect) {

    score++;

}

// Display feedback and move to the next question or show results if it's the last question
displayFeedback(isCorrect, currentQuestion.feedback || "No additional feedback available.");

currentQuestionIndex++;

if (currentQuestionIndex < questions.length) {

    setTimeout(() => {

        feedbackDiv.textContent = ''; // Clear feedback

        showQuestion();

    });

}
```

```
        nextButton.disabled = false; // Re-enable the button after question is shown

    }, 1000);

} else {

    setTimeout(() => {

        showResults(); // Show results after feedback

        nextButton.disabled = false; // Re-enable the button for reset

    }, 1000);

}

}
```

```
// Function to display the final results
```

```
function showResults() {

    const scoreDiv = document.getElementById('score');

    const feedbackDiv = document.getElementById('feedback');

    const passed = score >= passMark;
```

```
// Hide the question container and next button, show the reset button

document.getElementById('question-container').style.display = 'none';

document.getElementById('next-button').style.display = 'none';

document.getElementById('reset-button').style.display = 'block';


// Clear the feedback

feedbackDiv.textContent = '';


// Display the user's score and whether they passed

scoreDiv.textContent = `You scored ${score} out of ${questions.length}. You ${passed ? 'passed!' : 'did not pass. Try again!'}`;

scoreDiv.className = passed ? 'result passed' : 'result failed';

}


// Function to reset the quiz
```

```
function resetQuiz() {

    currentQuestionIndex = 0;

    score = 0;

    document.getElementById('question-container').style.display = 'block';

    document.getElementById('next-button').style.display = 'block';

    document.getElementById('reset-button').style.display = 'none';

    document.getElementById('feedback').textContent = '';

    document.getElementById('score').textContent = '';

    showQuestion(); // Start the quiz again
}

// Function to start the quiz, triggered by the start button

function startQuiz() {

    document.getElementById('start-container').style.display = 'none';

    document.getElementById('progress-container').style.display = 'block';

    document.getElementById('question-container').style.display = 'block';
```

```
document.getElementById('next-button').style.display = 'block';

showQuestion(); // Display the first question
}

// Event listener to start the quiz when the page loads
window.addEventListener('load', () => {

    document.getElementById('start-button').focus();

});
```