<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>One Piece: Greeting Voyage Quiz</title>

<script src="https://cdn.tailwindcss.com"></script>

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Peralta&family=Inter:wght@400;700&display=swap" rel="stylesheet">

<style>

body {

font-family: 'Inter', sans-serif;

background: #0F2027; /\* fallback for old browsers \*/

background: -webkit-linear-gradient(to right, #2C5364, #203A43, #0F2027); /\* Chrome 10-25, Safari 5.1-6 \*/

background: linear-gradient(to right, #2C5364, #203A43, #0F2027); /\* W3C, IE 10+/ Edge, Firefox 16+, Chrome 26+, Opera 12+, Safari 7+ \*/

background-image: url('https://www.transparenttextures.com/patterns/wood-planks.png');

color: #F0E68C; /\* Khaki \*/

}

.font-one-piece {

font-family: 'Peralta', cursive;

text-shadow: 2px 2px 4px rgba(0,0,0,0.7);

}

.main-container {

background-color: rgba(0, 0, 0, 0.4);

border: 4px solid #D4AF37; /\* Gold \*/

border-radius: 1.5rem;

box-shadow: 0 10px 30px rgba(0,0,0,0.5);

}

.btn-choice {

transition: all 0.2s ease-in-out;

}

.btn-choice:hover {

transform: translateY(-3px);

box-shadow: 0 4px 10px rgba(212, 175, 55, 0.4);

}

.btn-correct {

background-color: #2ECC71 !important;

color: white !important;

border-color: #27AE60 !important;

}

.btn-incorrect {

background-color: #E74C3C !important;

color: white !important;

border-color: #C0392B !important;

}

.btn-disabled {

pointer-events: none;

opacity: 0.7;

}

</style>

</head>

<body class="min-h-screen flex items-center justify-center p-4">

<main id="game-container" class="w-full max-w-2xl mx-auto main-container p-6 md:p-8 text-center">

<!-- Start Screen -->

<div id="start-screen">

<h1 class="text-4xl md:text-6xl font-one-piece text-yellow-400">Greeting Voyage</h1>

<p class="text-lg text-yellow-200 mt-4 mb-8">Help the Straw Hats choose the right greeting!</p>

<button id="start-button" class="font-one-piece bg-yellow-600 hover:bg-yellow-700 text-white font-bold py-3 px-8 rounded-lg shadow-lg transition-transform transform hover:scale-105 text-2xl">

Set Sail!

</button>

</div>

<!-- Quiz Screen -->

<div id="quiz-screen" class="hidden">

<div class="flex justify-between items-center mb-4 text-lg font-bold">

<div>Question: <span id="question-counter" class="text-yellow-400">1/8</span></div>

<div>Score: <span id="score-counter" class="text-yellow-400">0</span></div>

</div>

<img id="char-image" src="" alt="Character" class="mx-auto my-4 w-40 h-40 object-cover rounded-full border-4 border-amber-400 shadow-lg">

<p id="question-prompt" class="text-xl text-yellow-100 mb-6 min-h-[6rem] flex items-center justify-center"></p>

<div id="choices-container" class="grid grid-cols-1 md:grid-cols-2 gap-4 mb-4">

<!-- Choices will be generated by JavaScript -->

</div>

<div id="feedback-container" class="min-h-[2rem] mb-4">

<p id="feedback-text" class="text-2xl font-bold"></p>

</div>

<button id="next-button" class="hidden font-one-piece bg-blue-600 hover:bg-blue-700 text-white font-bold py-2 px-8 rounded-lg shadow-lg transition-transform transform hover:scale-105 text-xl">

Next

</button>

</div>

<!-- End Screen -->

<div id="end-screen" class="hidden">

<h2 class="text-4xl font-one-piece text-yellow-400 mb-4">Voyage Complete!</h2>

<p class="text-2xl text-yellow-100 mb-2">Your Final Score:</p>

<p id="final-score" class="text-6xl font-bold text-white mb-8"></p>

<button id="play-again-button" class="font-one-piece bg-yellow-600 hover:bg-yellow-700 text-white font-bold py-3 px-8 rounded-lg shadow-lg transition-transform transform hover:scale-105 text-2xl">

Play Again

</button>

</div>

</main>

<script>

const questions = [

{

character: 'Nami',

prompt: 'It\'s 7:00 AM. Nami sees her family. What should she say?',

img: 'https://placehold.co/200x200/F39C12/FFFFFF?text=Nami&font=peralta',

options: ['Good morning.', 'Good evening.', 'See you later.', 'How are you?'],

correctAnswer: 'Good morning.'

},

{

character: 'Luffy',

prompt: 'Luffy meets Mr. Porter at 1:00 PM. What is a polite greeting?',

img: 'https://placehold.co/200x200/E74C3C/FFFFFF?text=Luffy&font=peralta',

options: ['Good night.', 'Good afternoon.', 'Goodbye.', 'I\'m sorry.'],

correctAnswer: 'Good afternoon.'

},

{

character: 'Sanji',

prompt: 'Sanji runs into George and asks about his well-being. What does he say?',

img: 'https://placehold.co/200x200/3498DB/FFFFFF?text=Sanji&font=peralta',

options: ['Good evening.', 'Bye. Take care.', 'Not bad.', 'How are you?'],

correctAnswer: 'How are you?'

},

{

character: 'Zoro',

prompt: 'Zoro meets Mr. Lang at 7:00 PM. What greeting is appropriate?',

img: 'https://placehold.co/200x200/2ECC71/FFFFFF?text=Zoro&font=peralta',

options: ['Good morning.', 'Good evening.', 'Good afternoon.', 'Hello.'],

correctAnswer: 'Good evening.'

},

{

character: 'Usopp',

prompt: 'Usopp is leaving his friend Adnan. What should he say?',

img: 'https://placehold.co/200x200/F1C40F/000000?text=Usopp&font=peralta',

options: ["I'm fine, thanks.", 'See you later.', 'Hello.', 'Good morning.'],

correctAnswer: 'See you later.'

},

{

character: 'Chopper',

prompt: 'Chopper is saying goodbye to a friend. What is a simple way to do this?',

img: 'https://placehold.co/200x200/E91E63/FFFFFF?text=Chopper&font=peralta',

options: ['How are you?', 'Goodbye.', 'Not bad.', 'Good afternoon.'],

correctAnswer: 'Goodbye.'

},

{

character: 'Robin',

prompt: 'It\'s 8:00 PM and Robin is telling Fatima it\'s time for bed. What does she say?',

img: 'https://placehold.co/200x200/9B59B6/FFFFFF?text=Robin&font=peralta',

options: ['Good morning.', 'Good night.', 'Hello.', 'See you later.'],

correctAnswer: 'Good night.'

},

{

character: 'Franky',

prompt: 'Someone asks Franky, "How\'s it going?". He feels okay. What\'s his reply?',

img: 'https://placehold.co/200x200/3498DB/FFFFFF?text=Franky&font=peralta',

options: ["I'm sorry.", 'Not bad.', 'Bye. Take care.', 'Good morning.'],

correctAnswer: 'Not bad.'

}

];

// DOM Elements

const startScreen = document.getElementById('start-screen');

const quizScreen = document.getElementById('quiz-screen');

const endScreen = document.getElementById('end-screen');

const startButton = document.getElementById('start-button');

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

const playAgainButton = document.getElementById('play-again-button');

const questionCounter = document.getElementById('question-counter');

const scoreCounter = document.getElementById('score-counter');

const charImage = document.getElementById('char-image');

const questionPrompt = document.getElementById('question-prompt');

const choicesContainer = document.getElementById('choices-container');

const feedbackText = document.getElementById('feedback-text');

const finalScore = document.getElementById('final-score');

let currentQuestionIndex = 0;

let score = 0;

let shuffledQuestions = [];

function startGame() {

startScreen.classList.add('hidden');

endScreen.classList.add('hidden');

quizScreen.classList.remove('hidden');

currentQuestionIndex = 0;

score = 0;

shuffledQuestions = [...questions].sort(() => Math.random() - 0.5);

scoreCounter.textContent = score;

nextButton.classList.add('hidden');

showQuestion();

}

function showQuestion() {

resetState();

const currentQuestion = shuffledQuestions[currentQuestionIndex];

questionCounter.textContent = `${currentQuestionIndex + 1}/${shuffledQuestions.length}`;

charImage.src = currentQuestion.img;

charImage.alt = currentQuestion.character;

questionPrompt.textContent = currentQuestion.prompt;

currentQuestion.options.forEach(option => {

const button = document.createElement('button');

button.textContent = option;

button.classList.add('btn-choice', 'bg-amber-100', 'hover:bg-amber-200', 'text-stone-800', 'font-bold', 'py-3', 'px-4', 'rounded-lg', 'border-2', 'border-amber-300');

button.addEventListener('click', selectAnswer);

choicesContainer.appendChild(button);

});

}

function resetState() {

nextButton.classList.add('hidden');

feedbackText.textContent = '';

while (choicesContainer.firstChild) {

choicesContainer.removeChild(choicesContainer.firstChild);

}

}

function selectAnswer(e) {

const selectedButton = e.target;

const correctAnswer = shuffledQuestions[currentQuestionIndex].correctAnswer;

// Disable all buttons

Array.from(choicesContainer.children).forEach(button => {

button.classList.add('btn-disabled');

// Highlight the correct answer

if(button.textContent === correctAnswer) {

button.classList.add('btn-correct');

}

});

if (selectedButton.textContent === correctAnswer) {

score++;

scoreCounter.textContent = score;

feedbackText.textContent = 'Correct!';

feedbackText.classList.add('text-green-400');

feedbackText.classList.remove('text-red-400');

} else {

selectedButton.classList.add('btn-incorrect');

feedbackText.textContent = 'Wrong!';

feedbackText.classList.add('text-red-400');

feedbackText.classList.remove('text-green-400');

}

nextButton.classList.remove('hidden');

}

function handleNextButton() {

currentQuestionIndex++;

if (currentQuestionIndex < shuffledQuestions.length) {

showQuestion();

} else {

showEndScreen();

}

}

function showEndScreen() {

quizScreen.classList.add('hidden');

endScreen.classList.remove('hidden');

finalScore.textContent = `${score} / ${shuffledQuestions.length}`;

}

// Event Listeners

startButton.addEventListener('click', startGame);

nextButton.addEventListener('click', handleNextButton);

playAgainButton.addEventListener('click', startGame);

</script>

</body>

</html>