**Full EC2 Project Overview: “End-to-End Scalable Web Infrastructure using EC2”**

**WEEK 1 - EC2Quiz Web App**

# **Lab Overview :**

To launch a basic web server using Amazon EC2 (Amazon Linux 2), host a static quiz web page about EC2 concepts, and expose it via a public IP address — using Apache as the web server.

# **Technology Stack** :

1. Cloud Provider - AWS
2. Service - Compute Service - EC2
3. Web Server - Apache HTTP Server
4. Frontend - HTML + CSS + JavaScript (static)
5. Access - SSH via .pem key file

# **Learning Outcomes :**

1. Launch and connect to an EC2 instance
2. Install and configure Apache web server
3. Host static content (HTML/CSS/JS) on EC2
4. Work with security groups and key pairs

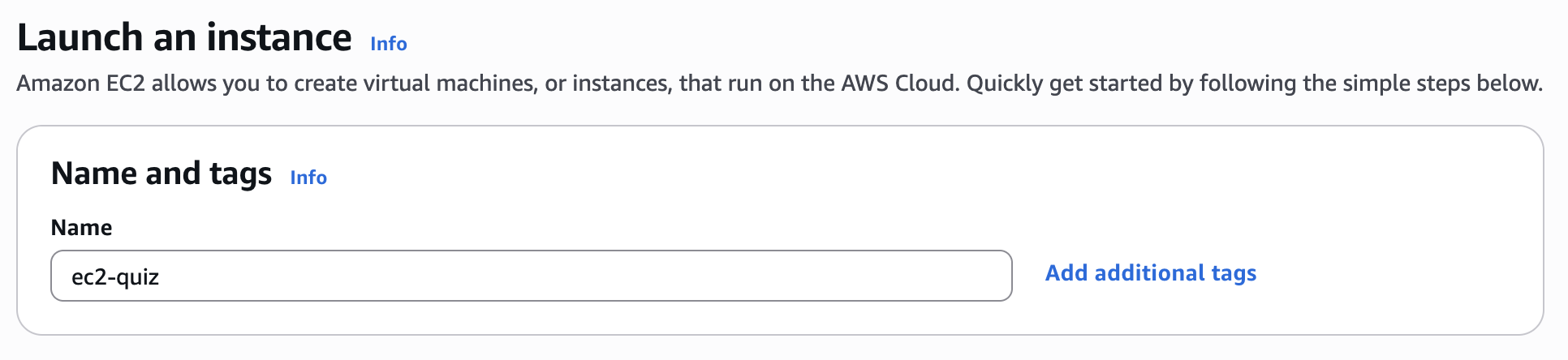
## **App Behavior (EC2Quiz)**

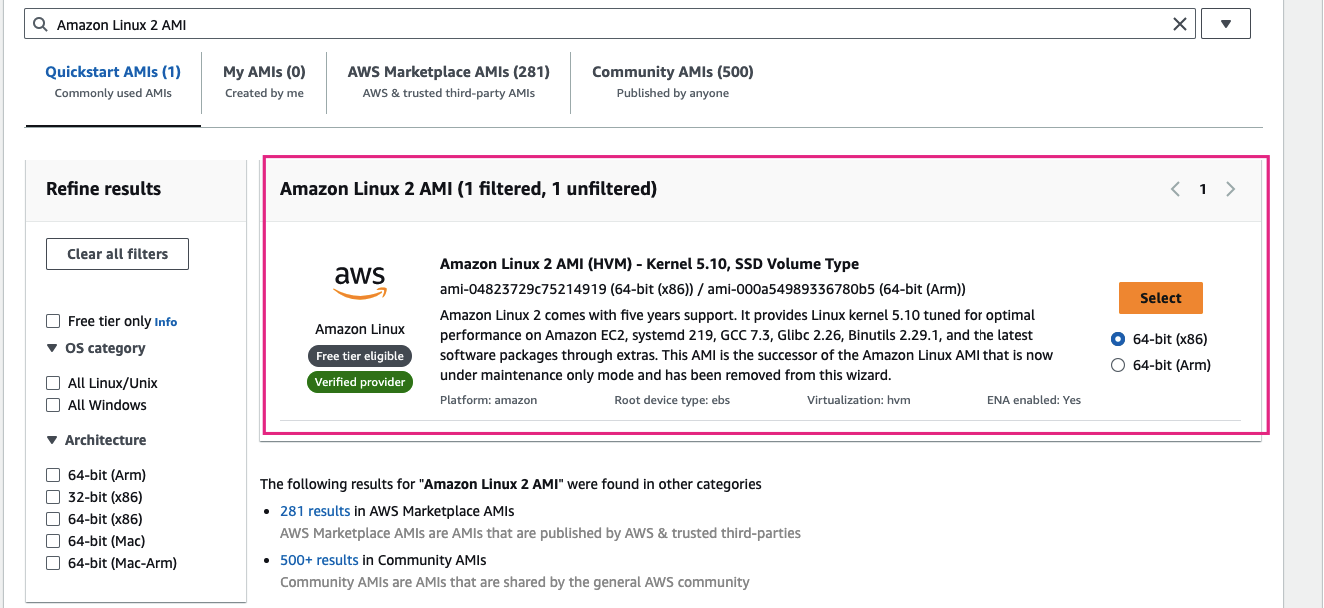
* Homepage **(index.html)** displays a multiple-choice quiz.
* 3 EC2-related questions with selectable options.
* At the end, the user sees their **score** and **correct answers**
* Basic styling for good UI/UX using CSS

## **Task 1: Sign In to the AWS Management Console**

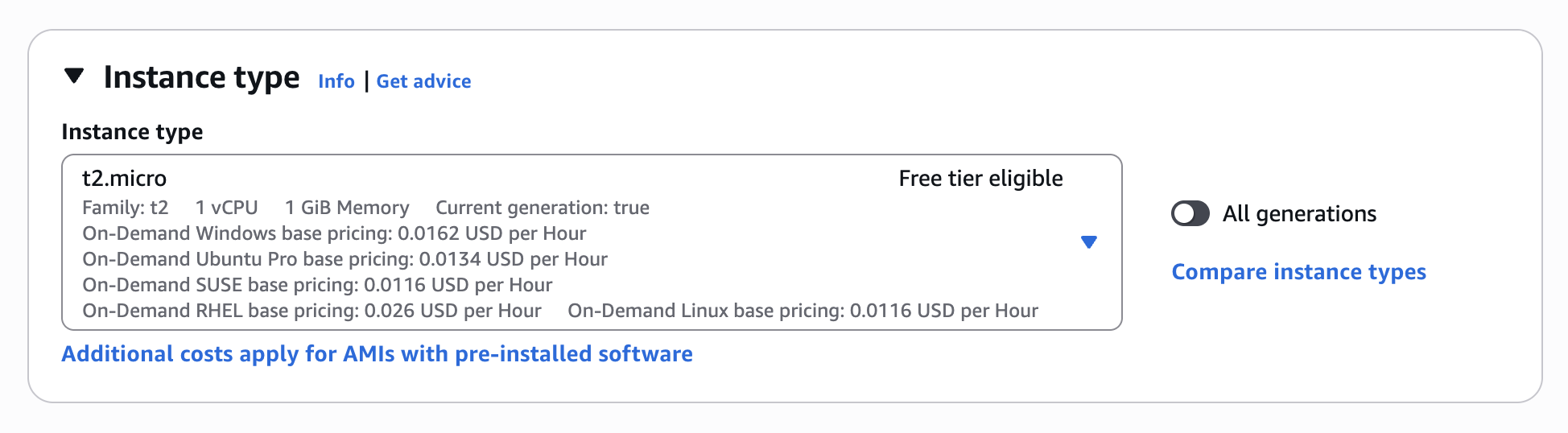
1. Click on the Open Console button, and you will get redirected to AWS Console in a new browser tab.
2. On the AWS sign-in page,
   1. Leave the Account ID as default. Never edit/remove the 12 digit Account ID present in the AWS Console. Otherwise, you cannot proceed with the lab.
   2. Now copy your User Name and Password in the Lab Console to the IAM Username and Password in AWS Console and click on the Sign in button.
3. Once Signed In to the AWS Management Console, Make the default AWS Region as US East (N. Virginia) us-east-1.

## **Task 2 : Launch an EC2 Instance with desired specifications**

1. Ensure you are in the US East (N. Virginia) us-east-1 Region to begin launching an EC2 instance in the Amazon cloud.
2. Navigate to EC2 by clicking on the Services menu in the top, then click on EC2 in the Compute section.
3. Click on the Instances option on the left panel, and then click on the Launch Instances button.
4. Name : Enter **ec2-quiz**  
   
5. Search for **Amazon Linux 2 AMI** in the search box and select it by clicking on the Select button.

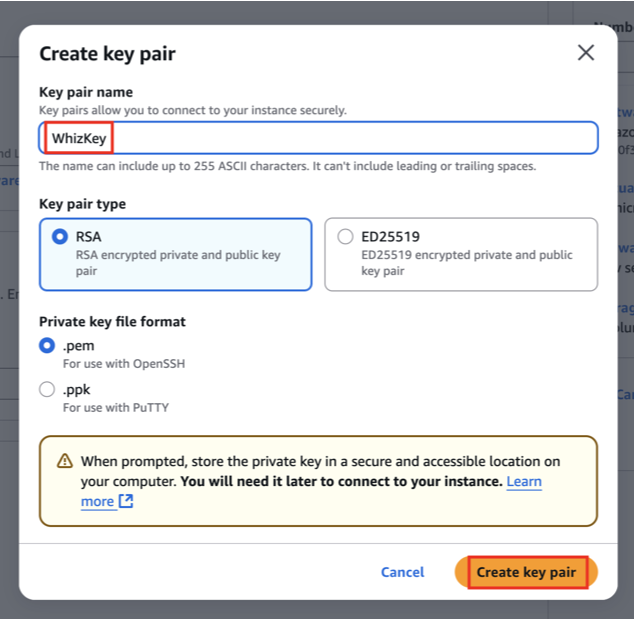


1. An instance type : Select **t2.micro**



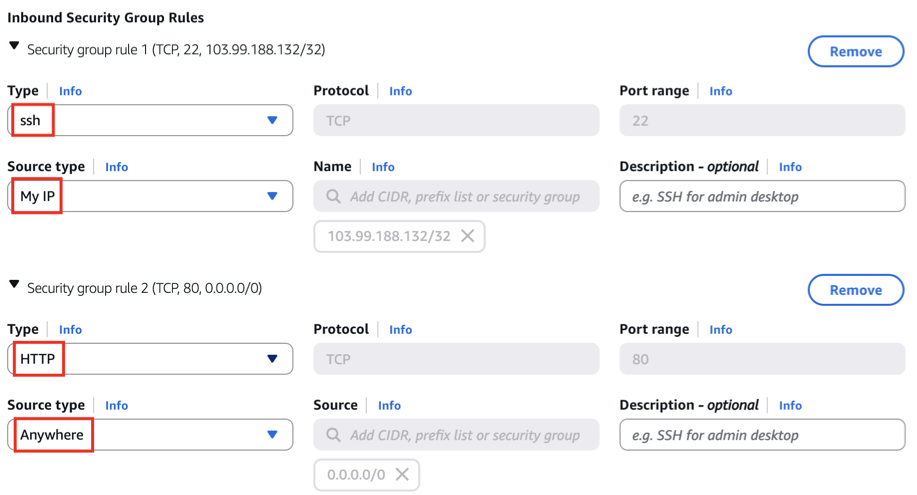
1. For Key pair(login): Select Create a new key pair Button

* Key pair name: **WhizKey**
* Key pair type: RSA
* Private key file format: .pem

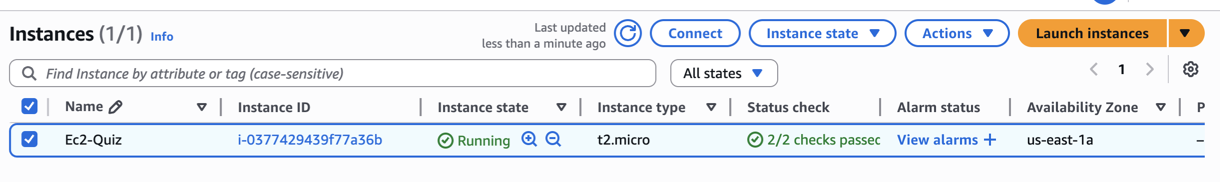


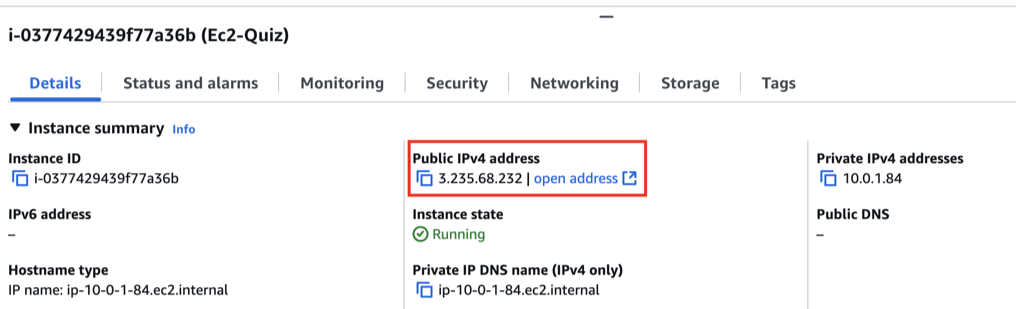
1. In Network Settings Click on Edit Button:

* Allow **SSH (port 22)** from **My IP**
* Allow **HTTP (port 80)** from **Anywhere**

****

1. Simply Click Launch Instance without any changes.





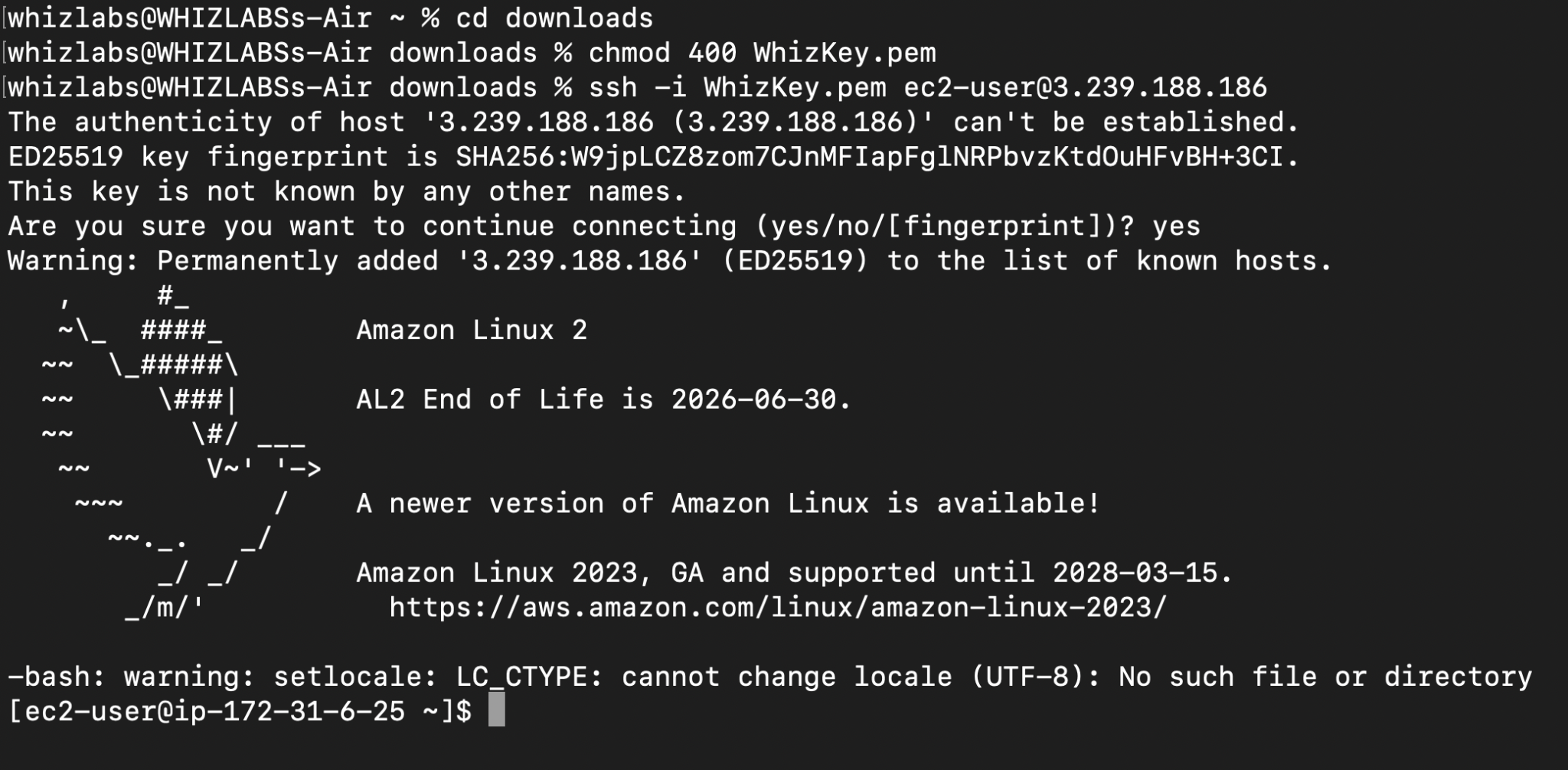
1. After it's running, **copy the Public IPv4 address** and save it.

## **Task 3 : Connect to the EC2 Instance**

1. This step uses your terminal, not the EC2 Console.
2. Open terminal where your keypair.pem file is located
3. Run the following:

**chmod 400 keypair.pem**

**ssh -i keypair.pem ec2-user@<IP-Address>**



## **Task 4: Install Apache and Enable CGI**

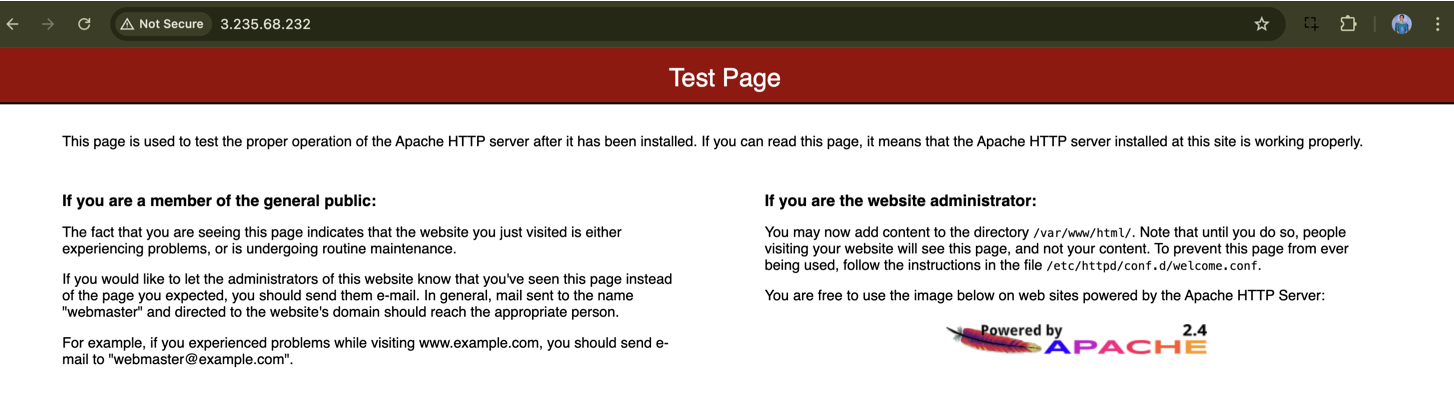
1. Apache will serve your web page, and CGI will allow us to run dynamic scripts from the browser.
2. Update the instance: **sudo yum update -y**
3. Install Apache HTTP Server: **sudo yum install httpd -y**
4. Start & enable Apache:

**sudo systemctl start httpd**

**sudo systemctl enable httpd**

1. Test the application, Go to your web browser and open,

**http://<your-ec2-public-ip>**

****

## **Task 5 : Create and Deploy the EC2 Quiz Web Page**

In this step, we'll set up a static website hosted by Apache on the EC2 instance. The site will present a multiple-choice EC2 quiz built with **HTML, CSS, and JavaScript.** The quiz will randomly load 3 questions every time the page is refreshed.

* Replace Apache’s default page with our custom quiz app.
* Create three files: **index.html, style.css, and script.js.**
* Place these files in Apache’s default web root **/var/www/html/**.

#### 

1. Create **index.html** – structure and layout of the quiz

* Open the HTML file using nano:

**sudo nano /var/www/html/index.html**

* Paste the following code:

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8">**

**<title>EC2 Quiz</title>**

**<link rel="stylesheet" href="style.css">**

**</head>**

**<body>**

**<div class="quiz-container">**

**<h1>EC2 Quiz</h1>**

**<div id="question"></div>**

**<ul id="options"></ul>**

**<button onclick="nextQuestion()">Next</button>**

**<div id="result"></div>**

**</div>**

**<script src="script.js"></script>**

**</body>**

**</html>**

* Save and exit:

Press Ctrl + X, then Y, then Enter

1. Create **style.css** – quiz page styling

* Open the CSS file using nano:

**sudo nano /var/www/html/style.css**

* Paste the following code:

**body {**

**font-family: Arial, sans-serif;**

**background-color: #f4f4f4;**

**padding: 20px;**

**text-align: center;**

**}**

**.quiz-container {**

**background: #fff;**

**padding: 30px;**

**border-radius: 10px;**

**width: 400px;**

**margin: auto;**

**box-shadow: 0 0 10px rgba(0,0,0,0.1);**

**}**

**ul {**

**list-style: none;**

**padding: 0;**

**}**

**li {**

**margin: 10px 0;**

**cursor: pointer;**

**background: #eee;**

**padding: 10px;**

**border-radius: 5px;**

**}**

**li:hover {**

**background: #ddd;**

**}**

**button {**

**margin-top: 20px;**

**padding: 10px 20px;**

**}**

**#result {**

**margin-top: 20px;**

**font-weight: bold;**

**}**

* Save and exit:

Press Ctrl + X, then Y, then Enter

1. Create **script.js** – quiz logic and randomization

* Open the JavaScript file using nano :

sudo nano /var/www/html/script.js

* Paste the following code:

**const allQuestions = [**

**{**

**question: "What does EC2 stand for?",**

**options: ["Elastic Compute Cloud", "Elastic Cluster Core", "Extended Cloud Control", "Elastic Core Cloud"],**

**answer: 0**

**},**

**{**

**question: "Which port is used for SSH?",**

**options: ["22", "80", "443", "3306"],**

**answer: 0**

**},**

**{**

**question: "Which instance type is free tier eligible?",**

**options: ["t2.micro", "m5.large", "c5.xlarge", "p3.2xlarge"],**

**answer: 0**

**},**

**{**

**question: "Which pricing model is best for short-term, interruption-tolerant workloads?",**

**options: ["Spot", "Reserved", "On-Demand", "Dedicated"],**

**answer: 0**

**},**

**{**

**question: "Which of the following is NOT a valid EC2 instance family?",**

**options: ["t", "m", "z", "d"],**

**answer: 3**

**},**

**{**

**question: "How do you connect to EC2 from local terminal?",**

**options: ["RDP", "FTP", "SSH", "SCP"],**

**answer: 2**

**},**

**{**

**question: "What does an IAM role provide for EC2?",**

**options: ["User login", "Access to AWS services", "Billing info", "Domain registration"],**

**answer: 1**

**},**

**{**

**question: "Where are web files served from in Apache?",**

**options: ["/etc/httpd/", "/var/www/html/", "/home/ec2-user/", "/usr/share/"],**

**answer: 1**

**},**

**{**

**question: "Which tool can monitor EC2 performance?",**

**options: ["CloudTrail", "CloudWatch", "S3", "Route 53"],**

**answer: 1**

**},**

**{**

**question: "How can you restrict EC2 access to only your IP?",**

**options: ["IAM policy", "NACL", "Security group rule", "Subnet ACL"],**

**answer: 2**

**}**

**];**

**// Shuffle and pick 3 random questions**

**const quizData = allQuestions.sort(() => 0.5 - Math.random()).slice(0, 3);**

**let current = 0;**

**let score = 0;**

**function loadQuestion() {**

**const q = quizData[current];**

**document.getElementById("question").innerText = q.question;**

**const options = document.getElementById("options");**

**options.innerHTML = "";**

**q.options.forEach((opt, i) => {**

**const li = document.createElement("li");**

**li.innerText = opt;**

**li.onclick = () => {**

**if (i === q.answer) score++;**

**nextQuestion();**

**};**

**options.appendChild(li);**

**});**

**}**

**function nextQuestion() {**

**current++;**

**if (current < quizData.length) {**

**loadQuestion();**

**} else {**

**document.getElementById("question").innerText = "Quiz Completed!";**

**document.getElementById("options").innerHTML = "";**

**document.getElementById("result").innerText = `Your score: ${score}/${quizData.length}`;**

**document.querySelector("button").style.display = "none";**

**}**

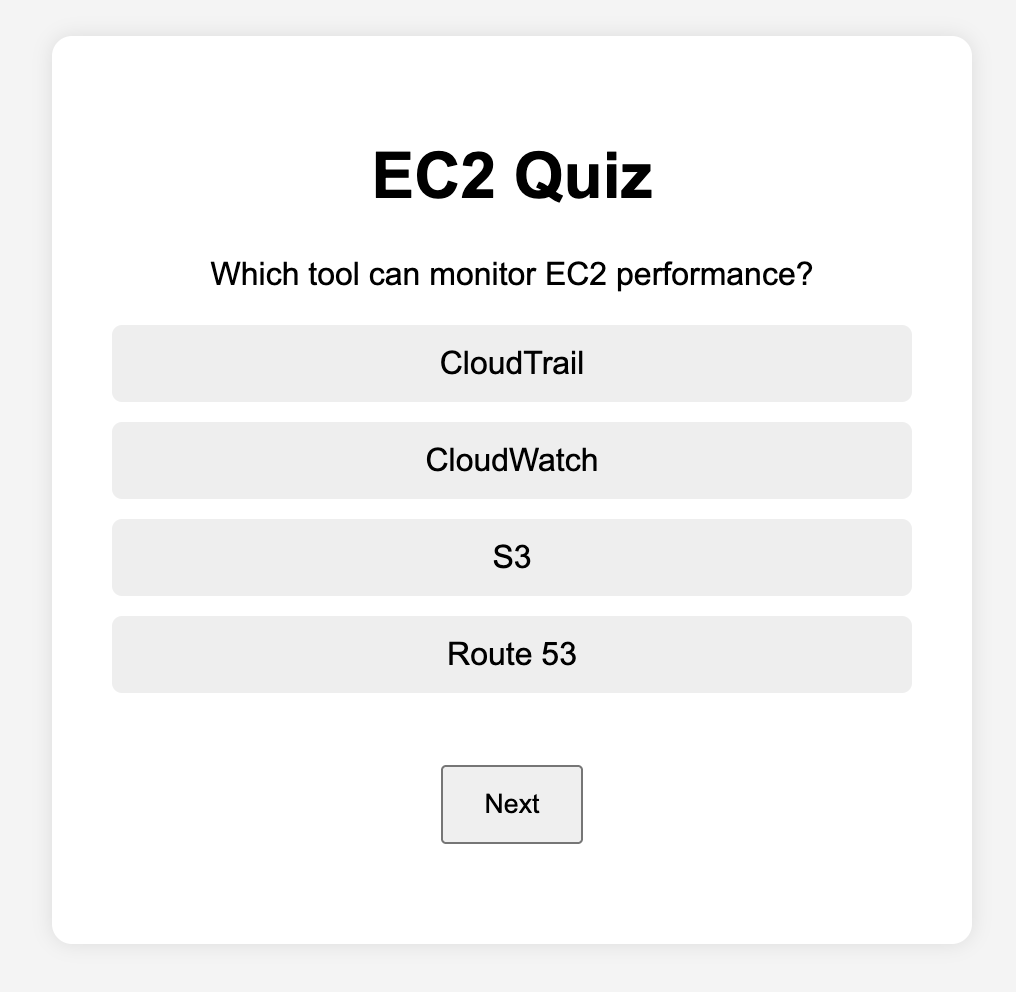
**}**

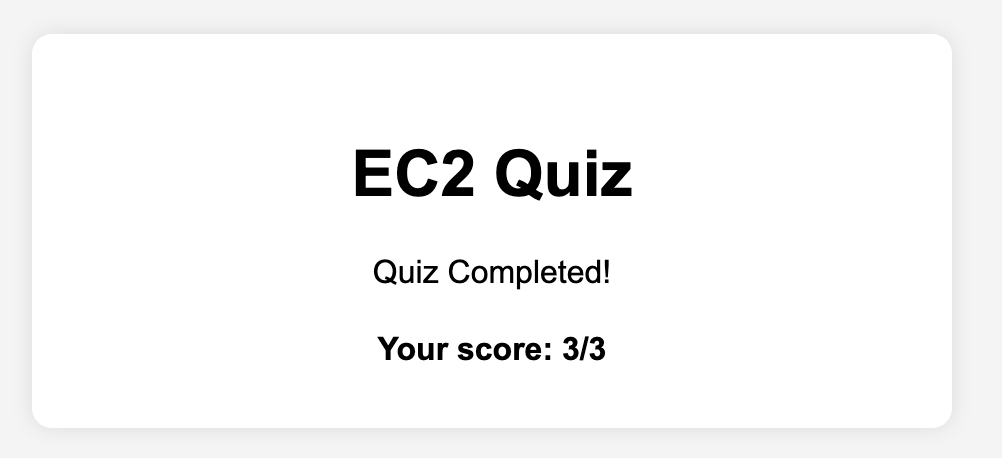
**window.onload = loadQuestion;**

1. Test the Quiz Application:

* Open a browser and visit:

**http://<your-ec2-public-ip>**

****



* You should see the quiz load with **3 random EC2 questions**. After answering, your score is displayed.