File Name: index.html

Content: Copy the full HTML code provided, including the Binance API integration, the trading forms, and the mock authentication system.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Crypto Exchange</title>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">

<style>

body { font-family: Arial, sans-serif; margin: 0; padding: 0; background-color: #f4f4f4; }

header { background-color: #232323; color: white; padding: 10px; text-align: center; }

header .nav-links { display: flex; justify-content: space-between; }

.content { display: flex; padding: 20px; }

.market { flex: 2; padding-right: 20px; }

.sidebar { flex: 1; background-color: #fff; padding: 20px; }

.crypto-list { display: flex; flex-wrap: wrap; }

.crypto-item { width: 200px; margin: 10px; padding: 20px; background-color: white; box-shadow: 0 2px 5px rgba(0,0,0,0.1); text-align: center; }

.crypto-item h2 { margin-bottom: 10px; font-size: 24px; }

.chart-container { height: 400px; }

</style>

</head>

<body>

<header>

<h1>Crypto Exchange</h1>

<div class="nav-links">

<div>Markets</div>

<div>Login / Register</div>

</div>

</header>

<div class="content">

<div class="market">

<h2>Market Listings</h2>

<div id="crypto-list" class="crypto-list"></div>

<h2>Live Chart</h2>

<div class="chart-container">

<canvas id="priceChart"></canvas>

</div>

</div>

<div class="sidebar">

<h3>Portfolio</h3>

<p>Track your assets here...</p>

<h3>Trade History</h3>

<p>View past trades...</p>

<!-- Add forms for Buy/Sell Orders -->

<h3>Place Buy/Sell Order</h3>

<form id="order-form">

<label for="order-type">Order Type:</label>

<select id="order-type">

<option value="buy">Buy</option>

<option value="sell">Sell</option>

</select>

<label for="crypto-symbol">Cryptocurrency:</label>

<select id="crypto-symbol">

<option value="BTCUSDT">Bitcoin (BTC)</option>

<option value="ETHUSDT">Ethereum (ETH)</option>

</select>

<label for="order-amount">Amount:</label>

<input type="number" id="order-amount" placeholder="Enter amount" required>

<button type="submit">Submit Order</button>

</form>

<div id="order-confirmation" style="display:none; margin-top: 20px;">

<p><strong>Order confirmed:</strong> <span id="order-details"></span></p>

</div>

<!-- User Authentication -->

<h3>User Authentication</h3>

<div id="auth-forms">

<form id="register-form" style="display:none;">

<h4>Register</h4>

<label for="register-username">Username:</label>

<input type="text" id="register-username" required>

<label for="register-password">Password:</label>

<input type="password" id="register-password" required>

<button type="submit">Register</button>

</form>

<form id="login-form">

<h4>Login</h4>

<label for="login-username">Username:</label>

<input type="text" id="login-username" required>

<label for="login-password">Password:</label>

<input type="password" id="login-password" required>

<button type="submit">Login</button>

</form>

<p><a href="#" id="toggle-auth">Don't have an account? Register</a></p>

</div>

</div>

</div>

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<script>

// Binance API for real-time prices

const binanceAPI = 'https://api.binance.com/api/v3/ticker/price';

// Function to fetch crypto prices and display them

async function fetchCryptos() {

try {

const response = await fetch(binanceAPI);

const data = await response.json();

const cryptoList = document.getElementById('crypto-list');

cryptoList.innerHTML = '';

// Display first 6 cryptos

data.slice(0, 6).forEach(crypto => {

const item = document.createElement('div');

item.classList.add('crypto-item');

item.innerHTML = `<h2>${crypto.symbol}</h2><p>Price: $${parseFloat(crypto.price).toFixed(2)}</p>`;

cryptoList.appendChild(item);

});

} catch (error) {

console.error('Error fetching data:', error);

}

}

fetchCryptos();

setInterval(fetchCryptos, 10000); // Refresh every 10 seconds

// Set up Chart.js for live price chart (with placeholder data)

const ctx = document.getElementById('priceChart').getContext('2d');

const priceChart = new Chart(ctx, {

type: 'line',

data: {

labels: ['1h', '2h', '3h', '4h', '5h'],

datasets: [{

label: 'BTC/USD',

data: [32000, 33000, 34000, 35000, 36000],

backgroundColor: 'rgba(75, 192, 192, 0.2)',

borderColor: 'rgba(75, 192, 192, 1)',

borderWidth: 1

}]

},

options: {

scales: {

y: { beginAtZero: false }

}

}

});

// Buy/Sell Order form functionality

document.getElementById('order-form').addEventListener('submit', function(event) {

event.preventDefault(); // Prevent form from submitting

const orderType = document.getElementById('order-type').value;

const cryptoSymbol = document.getElementById('crypto-symbol').value;

const orderAmount = document.getElementById('order-amount').value;

const confirmationText = `${orderType.toUpperCase()} ${orderAmount} ${cryptoSymbol}`;

document.getElementById('order-details').innerText = confirmationText;

document.getElementById('order-confirmation').style.display = 'block';

});

// Mock user authentication (register and login)

let isLoggedIn = false;

let mockUsers = [];

document.getElementById('toggle-auth').addEventListener('click', function() {

const loginForm = document.getElementById('login-form');

const registerForm = document.getElementById('register-form');

loginForm.style.display = loginForm.style.display === 'none' ? 'block' : 'none';

registerForm.style.display = registerForm.style.display === 'none' ? 'block' : 'none';

this.innerText = loginForm.style.display === 'none' ? "Already have an account? Login" : "Don't have an account? Register";

});

// Register form submission

document.getElementById('register-form').addEventListener('submit', function(event) {

event.preventDefault();

const username = document.getElementById('register-username').value;

const password = document.getElementById('register-password').value;

mockUsers.push({ username, password });

alert('User registered successfully!');

document.getElementById('register-username').value = '';

document.getElementById('register-password').value = '';

});

// Login form submission

document.getElementById('login-form').addEventListener('submit', function(event) {

event.preventDefault();

const username = document.getElementById('login-username').value;

const password = document.getElementById('login-password').value;

const user = mockUsers.find(u => u.username === username && u.password === password);

if (user) {

isLoggedIn = true;

alert('Login successful!');

document.getElementById('auth-forms').style.display = 'none';

document.querySelector('header').innerHTML += `<p>Welcome, ${username}</p>`;

} else {

alert('Invalid credentials!');

}