<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>TradeMaster Academy - Learn Options & Futures Trading</title>

<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css" rel="stylesheet">

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

<script src="https://unpkg.com/lightweight-charts/dist/lightweight-charts.standalone.production.js"></script>

<style>

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);

color: #333;

line-height: 1.6;

}

.container {

max-width: 1200px;

margin: 0 auto;

padding: 20px;

}

header {

background: rgba(255, 255, 255, 0.95);

backdrop-filter: blur(10px);

padding: 1rem 0;

position: fixed;

width: 100%;

top: 0;

z-index: 1000;

box-shadow: 0 2px 20px rgba(0,0,0,0.1);

}

nav {

display: flex;

justify-content: space-between;

align-items: center;

max-width: 1200px;

margin: 0 auto;

padding: 0 20px;

}

.logo {

font-size: 1.8rem;

font-weight: bold;

color: #667eea;

}

.nav-links {

display: flex;

list-style: none;

gap: 2rem;

}

.nav-links a {

text-decoration: none;

color: #333;

font-weight: 500;

transition: color 0.3s;

}

.nav-links a:hover {

color: #667eea;

}

.user-progress {

display: flex;

align-items: center;

gap: 1rem;

}

.level-badge {

background: #667eea;

color: white;

padding: 0.3rem 0.8rem;

border-radius: 20px;

font-size: 0.8rem;

}

main {

margin-top: 80px;

background: white;

border-radius: 20px;

padding: 2rem;

box-shadow: 0 10px 40px rgba(0,0,0,0.1);

}

.dashboard {

display: grid;

grid-template-columns: 1fr 2fr 1fr;

gap: 2rem;

margin-bottom: 2rem;

}

.progress-panel, .trading-panel, .stats-panel {

background: #f8f9fa;

border-radius: 15px;

padding: 1.5rem;

box-shadow: 0 5px 15px rgba(0,0,0,0.08);

}

.progress-panel h3, .trading-panel h3, .stats-panel h3 {

margin-bottom: 1rem;

color: #667eea;

display: flex;

align-items: center;

gap: 0.5rem;

}

.progress-bar {

background: #e9ecef;

border-radius: 10px;

height: 20px;

margin: 1rem 0;

overflow: hidden;

}

.progress-fill {

background: linear-gradient(90deg, #667eea, #764ba2);

height: 100%;

border-radius: 10px;

transition: width 0.3s ease;

display: flex;

align-items: center;

justify-content: center;

color: white;

font-size: 0.8rem;

font-weight: bold;

}

.milestone {

display: flex;

align-items: center;

gap: 1rem;

margin: 1rem 0;

padding: 0.8rem;

border-radius: 10px;

transition: background 0.3s;

}

.milestone.completed {

background: #d4edda;

border-left: 4px solid #28a745;

}

.milestone.current {

background: #fff3cd;

border-left: 4px solid #ffc107;

}

.milestone.upcoming {

background: #f8f9fa;

border-left: 4px solid #6c757d;

}

.milestone-icon {

width: 30px;

height: 30px;

border-radius: 50%;

display: flex;

align-items: center;

justify-content: center;

color: white;

font-size: 0.8rem;

}

.completed .milestone-icon {

background: #28a745;

}

.current .milestone-icon {

background: #ffc107;

}

.upcoming .milestone-icon {

background: #6c757d;

}

.chart-container {

background: white;

border-radius: 10px;

padding: 1rem;

margin: 1rem 0;

height: 400px;

position: relative;

}

.trading-controls {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));

gap: 1rem;

margin: 1rem 0;

}

.control-group {

display: flex;

flex-direction: column;

gap: 0.5rem;

}

.control-group label {

font-weight: 600;

color: #495057;

}

.control-group input, .control-group select {

padding: 0.8rem;

border: 2px solid #e9ecef;

border-radius: 8px;

font-size: 1rem;

transition: border-color 0.3s;

}

.control-group input:focus, .control-group select:focus {

outline: none;

border-color: #667eea;

}

.btn {

padding: 0.8rem 1.5rem;

border: none;

border-radius: 8px;

font-size: 1rem;

font-weight: 600;

cursor: pointer;

transition: all 0.3s;

display: inline-flex;

align-items: center;

gap: 0.5rem;

}

.btn-primary {

background: linear-gradient(135deg, #667eea, #764ba2);

color: white;

}

.btn-primary:hover {

transform: translateY(-2px);

box-shadow: 0 5px 15px rgba(102, 126, 234, 0.4);

}

.btn-success {

background: #28a745;

color: white;

}

.btn-danger {

background: #dc3545;

color: white;

}

.stats-grid {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(150px, 1fr));

gap: 1rem;

margin: 1rem 0;

}

.stat-card {

background: white;

padding: 1rem;

border-radius: 10px;

text-align: center;

box-shadow: 0 2px 10px rgba(0,0,0,0.05);

}

.stat-value {

font-size: 1.8rem;

font-weight: bold;

color: #667eea;

}

.stat-label {

font-size: 0.9rem;

color: #6c757d;

margin-top: 0.5rem;

}

.learning-modules {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));

gap: 2rem;

margin-top: 2rem;

}

.module-card {

background: white;

border-radius: 15px;

padding: 1.5rem;

box-shadow: 0 5px 15px rgba(0,0,0,0.08);

transition: transform 0.3s;

}

.module-card:hover {

transform: translateY(-5px);

}

.module-header {

display: flex;

align-items: center;

gap: 1rem;

margin-bottom: 1rem;

}

.module-icon {

width: 50px;

height: 50px;

border-radius: 10px;

display: flex;

align-items: center;

justify-content: center;

font-size: 1.5rem;

color: white;

}

.beginner { background: #28a745; }

.intermediate { background: #ffc107; }

.advanced { background: #dc3545; }

.professional { background: #6f42c1; }

.module-content {

margin: 1rem 0;

}

.module-topics {

list-style: none;

margin: 1rem 0;

}

.module-topics li {

padding: 0.5rem 0;

border-bottom: 1px solid #e9ecef;

display: flex;

align-items: center;

gap: 0.5rem;

}

.module-topics li:before {

content: "✓";

color: #28a745;

font-weight: bold;

}

.modal {

display: none;

position: fixed;

z-index: 2000;

left: 0;

top: 0;

width: 100%;

height: 100%;

background-color: rgba(0,0,0,0.5);

}

.modal-content {

background-color: white;

margin: 5% auto;

padding: 2rem;

border-radius: 15px;

width: 80%;

max-width: 600px;

box-shadow: 0 10px 40px rgba(0,0,0,0.2);

}

.close {

color: #aaa;

float: right;

font-size: 28px;

font-weight: bold;

cursor: pointer;

}

.close:hover {

color: #000;

}

.options-chain {

background: white;

border-radius: 10px;

overflow: hidden;

margin: 1rem 0;

}

.options-chain table {

width: 100%;

border-collapse: collapse;

}

.options-chain th {

background: #667eea;

color: white;

padding: 1rem;

text-align: center;

}

.options-chain td {

padding: 0.8rem;

text-align: center;

border-bottom: 1px solid #e9ecef;

}

.options-chain tr:hover {

background: #f8f9fa;

}

.call-option {

color: #28a745;

font-weight: bold;

}

.put-option {

color: #dc3545;

font-weight: bold;

}

@media (max-width: 768px) {

.dashboard {

grid-template-columns: 1fr;

}

.nav-links {

display: none;

}

.trading-controls {

grid-template-columns: 1fr;

}

.learning-modules {

grid-template-columns: 1fr;

}

}

.notification {

position: fixed;

top: 100px;

right: 20px;

background: white;

padding: 1rem;

border-radius: 10px;

box-shadow: 0 5px 15px rgba(0,0,0,0.2);

z-index: 1500;

transform: translateX(400px);

transition: transform 0.3s;

}

.notification.show {

transform: translateX(0);

}

.notification.success {

border-left: 4px solid #28a745;

}

.notification.error {

border-left: 4px solid #dc3545;

}

.notification.info {

border-left: 4px solid #17a2b8;

}

</style>

</head>

<body>

<header>

<nav>

<div class="logo">

<i class="fas fa-chart-line"></i> TradeMaster Academy

</div>

<ul class="nav-links">

<li><a href="#dashboard"><i class="fas fa-tachometer-alt"></i> Dashboard</a></li>

<li><a href="#learning"><i class="fas fa-graduation-cap"></i> Learning</a></li>

<li><a href="#practice"><i class="fas fa-laptop"></i> Practice</a></li>

<li><a href="#portfolio"><i class="fas fa-briefcase"></i> Portfolio</a></li>

<li><a href="#community"><i class="fas fa-users"></i> Community</a></li>

</ul>

<div class="user-progress">

<span class="level-badge" id="userLevel">Beginner</span>

<div style="font-size: 0.9rem;">

<i class="fas fa-coins text-warning"></i>

<span id="userPoints">1,250</span> pts

</div>

</div>

</nav>

</header>

<main class="container">

<div class="dashboard">

<div class="progress-panel">

<h3><i class="fas fa-map-marked-alt"></i> Learning Path</h3>

<div class="progress-bar">

<div class="progress-fill" id="progressBar" style="width: 25%;">25%</div>

</div>

<div class="milestone completed">

<div class="milestone-icon"><i class="fas fa-check"></i></div>

<div>

<strong>Module 1: Basics</strong>

<div style="font-size: 0.8rem; color: #6c757d;">Completed</div>

</div>

</div>

<div class="milestone current">

<div class="milestone-icon"><i class="fas fa-play"></i></div>

<div>

<strong>Module 2: Options</strong>

<div style="font-size: 0.8rem; color: #6c757d;">In Progress</div>

</div>

</div>

<div class="milestone upcoming">

<div class="milestone-icon"><i class="fas fa-lock"></i></div>

<div>

<strong>Module 3: Futures</strong>

<div style="font-size: 0.8rem; color: #6c757d;">Locked</div>

</div>

</div>

<div class="milestone upcoming">

<div class="milestone-icon"><i class="fas fa-lock"></i></div>

<div>

<strong>Module 4: Advanced</strong>

<div style="font-size: 0.8rem; color: #6c757d;">Locked</div>

</div>

</div>

</div>

<div class="trading-panel">

<h3><i class="fas fa-chart-candlestick"></i> Practice Trading</h3>

<div class="chart-container">

<div id="tradingChart" style="height: 100%;"></div>

</div>

<div class="trading-controls">

<div class="control-group">

<label>Asset</label>

<select id="assetSelect">

<option value="AAPL">Apple (AAPL)</option>

<option value="TSLA">Tesla (TSLA)</option>

<option value="SPY">S&P 500 (SPY)</option>

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

</select>

</div>

<div class="control-group">

<label>Strategy</label>

<select id="strategySelect">

<option value="long-call">Long Call</option>

<option value="long-put">Long Put</option>

<option value="covered-call">Covered Call</option>

<option value="protective-put">Protective Put</option>

</select>

</div>

<div class="control-group">

<label>Strike Price</label>

<input type="number" id="strikePrice" value="150" min="1" step="0.01">

</div>

<div class="control-group">

<label>Quantity</label>

<input type="number" id="quantity" value="1" min="1">

</div>

</div>

<div style="display: flex; gap: 1rem; justify-content: center;">

<button class="btn btn-success" onclick="executeTrade('buy')">

<i class="fas fa-arrow-up"></i> Buy

</button>

<button class="btn btn-danger" onclick="executeTrade('sell')">

<i class="fas fa-arrow-down"></i> Sell

</button>

<button class="btn btn-primary" onclick="showOptionsChain()">

<i class="fas fa-list"></i> Options Chain

</button>

</div>

</div>

<div class="stats-panel">

<h3><i class="fas fa-trophy"></i> Your Stats</h3>

<div class="stats-grid">

<div class="stat-card">

<div class="stat-value" id="totalTrades">47</div>

<div class="stat-label">Total Trades</div>

</div>

<div class="stat-card">

<div class="stat-value" id="winRate">68%</div>

<div class="stat-label">Win Rate</div>

</div>

<div class="stat-card">

<div class="stat-value" id="profitLoss">+$1,247</div>

<div class="stat-label">P&L</div>

</div>

<div class="stat-card">

<div class="stat-value" id="streak">5</div>

<div class="stat-label">Win Streak</div>

</div>

</div>

<div style="margin-top: 1.5rem;">

<h4 style="margin-bottom: 1rem; color: #667eea;">Achievements</h4>

<div style="display: flex; gap: 0.5rem; flex-wrap: wrap;">

<div style="background: #ffd700; color: #333; padding: 0.3rem 0.6rem; border-radius: 15px; font-size: 0.8rem;">

<i class="fas fa-medal"></i> First Trade

</div>

<div style="background: #c0c0c0; color: #333; padding: 0.3rem 0.6rem; border-radius: 15px; font-size: 0.8rem;">

<i class="fas fa-chart-line"></i> 10 Trades

</div>

<div style="background: #cd7f32; color: white; padding: 0.3rem 0.6rem; border-radius: 15px; font-size: 0.8rem;">

<i class="fas fa-fire"></i> 5 Streak

</div>

</div>

</div>

</div>

</div>

<section id="learning" class="learning-modules">

<h2 style="text-align: center; margin-bottom: 2rem; color: #667eea;">

<i class="fas fa-graduation-cap"></i> Learning Modules

</h2>

<div class="module-card">

<div class="module-header">

<div class="module-icon beginner">

<i class="fas fa-seedling"></i>

</div>

<div>

<h3>Beginner Level</h3>

<p style="color: #6c757d; font-size: 0.9rem;">Start your trading journey</p>

</div>

</div>

<div class="module-content">

<ul class="module-topics">

<li>What are Options?</li>

<li>Call vs Put Options</li>

<li>Basic Terminology</li>

<li>Risk Management</li>

<li>Your First Trade</li>

</ul>

<button class="btn btn-primary" onclick="startModule('beginner')">

<i class="fas fa-play"></i> Start Learning

</button>

</div>

</div>

<div class="module-card">

<div class="module-header">

<div class="module-icon intermediate">

<i class="fas fa-chart-bar"></i>

</div>

<div>

<h3>Intermediate Level</h3>

<p style="color: #6c757d; font-size: 0.9rem;">Build your skills</p>

</div>

</div>

<div class="module-content">

<ul class="module-topics">

<li>Options Strategies</li>

<li>Greeks Explained</li>

<li>Implied Volatility</li>

<li>Technical Analysis</li>

<li>Position Sizing</li>

</ul>

<button class="btn btn-primary" onclick="startModule('intermediate')">

<i class="fas fa-play"></i> Start Learning

</button>

</div>

</div>

<div class="module-card">

<div class="module-header">

<div class="module-icon advanced">

<i class="fas fa-rocket"></i>

</div>

<div>

<h3>Advanced Level</h3>

<p style="color: #6c757d; font-size: 0.9rem;">Master complex strategies</p>

</div>

</div>

<div class="module-content">

<ul class="module-topics">

<li>Spreads & Combinations</li>

<li>Iron Condors</li>

<li>Butterfly Spreads</li>

<li>Calendar Spreads</li>

<li>Volatility Trading</li>

</ul>

<button class="btn btn-primary" onclick="startModule('advanced')">

<i class="fas fa-play"></i> Start Learning

</button>

</div>

</div>

<div class="module-card">

<div class="module-header">

<div class="module-icon professional">

<i class="fas fa-crown"></i>

</div>

<div>

<h3>Professional Level</h3>

<p style="color: #6c757d; font-size: 0.9rem;">Trade like a pro</p>

</div>

</div>

<div class="module-content">

<ul class="module-topics">

<li>Market Making</li>

<li>Arbitrage Strategies</li>

<li>Risk Management II</li>

<li>Portfolio Hedging</li>

<li>Algorithmic Trading</li>

</ul>

<button class="btn btn-primary" onclick="startModule('professional')">

<i class="fas fa-play"></i> Start Learning

</button>

</div>

</div>

</section>

</main>

<!-- Options Chain Modal -->

<div id="optionsChainModal" class="modal">

<div class="modal-content">

<span class="close" onclick="closeOptionsChain()">&times;</span>

<h2 style="margin-bottom: 1rem; color: #667eea;">

<i class="fas fa-list"></i> Options Chain

</h2>

<div class="options-chain">

<table>

<thead>

<tr>

<th>Strike</th>

<th>Call Bid</th>

<th>Call Ask</th>

<th>Put Bid</th>

<th>Put Ask</th>

</tr>

</thead>

<tbody id="optionsChainBody">

<!-- Options chain data will be populated here -->

</tbody>

</table>

</div>

</div>

</div>

<!-- Notification System -->

<div id="notification" class="notification">

<span id="notificationText"></span>

</div>

<script>

// Global variables

let chart;

let currentPrice = 150;

let portfolio = {

cash: 10000,

positions: [],

totalValue: 10000

};

let userLevel = 'beginner';

let userPoints = 1250;

let progress = 25;

// Initialize the application

document.addEventListener('DOMContentLoaded', function() {

initializeChart();

updateDashboard();

generateOptionsChain();

});

// Initialize TradingView-style chart

function initializeChart() {

const chartContainer = document.getElementById('tradingChart');

// Create a simple line chart for demo purposes

const ctx = document.createElement('canvas');

ctx.id = 'priceChart';

chartContainer.appendChild(ctx);

const data = generatePriceData();

chart = new Chart(ctx, {

type: 'line',

data: {

labels: data.labels,

datasets: [{

label: 'Price',

data: data.prices,

borderColor: '#667eea',

backgroundColor: 'rgba(102, 126, 234, 0.1)',

borderWidth: 2,

fill: true,

tension: 0.4

}]

},

options: {

responsive: true,

maintainAspectRatio: false,

plugins: {

legend: {

display: false

}

},

scales: {

y: {

beginAtZero: false,

grid: {

color: 'rgba(0,0,0,0.1)'

}

},

x: {

grid: {

display: false

}

}

}

}

});

// Update price every 2 seconds

setInterval(updatePrice, 2000);

}

// Generate realistic price data

function generatePriceData() {

const labels = [];

const prices = [];

let price = currentPrice;

for (let i = 0; i < 24; i++) {

labels.push(`${i}:00`);

price += (Math.random() - 0.5) \* 2;

prices.push(Math.max(0, price));

}

return { labels, prices };

}

// Update price

function updatePrice() {

const change = (Math.random() - 0.5) \* 2;

currentPrice = Math.max(0, currentPrice + change);

const data = generatePriceData();

chart.data.labels = data.labels;

chart.data.datasets[0].data = data.prices;

chart.update();

}

// Execute trade

function executeTrade(type) {

const asset = document.getElementById('assetSelect').value;

const strategy = document.getElementById('strategySelect').value;

const strike = parseFloat(document.getElementById('strikePrice').value);

const quantity = parseInt(document.getElementById('quantity').value);

const trade = {

id: Date.now(),

type: type,

asset: asset,

strategy: strategy,

strike: strike,

quantity: quantity,

price: currentPrice,

timestamp: new Date(),

value: currentPrice \* quantity \* 100 // Options multiplier

};

portfolio.positions.push(trade);

updatePortfolio();

showNotification(`${type.toUpperCase()} order executed for ${quantity} contracts of ${asset}`, 'success');

// Update stats

updateStats();

}

// Update portfolio

function updatePortfolio() {

const totalValue = portfolio.cash + portfolio.positions.reduce((sum, pos) => sum + pos.value, 0);

portfolio.totalValue = totalValue;

document.getElementById('totalTrades').textContent = portfolio.positions.length;

document.getElementById('profitLoss').textContent = `+$${(totalValue - 10000).toLocaleString()}`;

}

// Update statistics

function updateStats() {

const trades = portfolio.positions.length;

const wins = Math.floor(trades \* 0.68); // Simulated win rate

const winRate = trades > 0 ? Math.round((wins / trades) \* 100) : 0;

document.getElementById('winRate').textContent = `${winRate}%`;

// Update streak (simplified)

const streak = Math.floor(Math.random() \* 10) + 1;

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

}

// Show options chain

function showOptionsChain() {

document.getElementById('optionsChainModal').style.display = 'block';

}

// Close options chain

function closeOptionsChain() {

document.getElementById('optionsChainModal').style.display = 'none';

}

// Generate options chain data

function generateOptionsChain() {

const tbody = document.getElementById('optionsChainBody');

const strikes = [];

// Generate strikes around current price

for (let i = -5; i <= 5; i++) {

strikes.push(currentPrice + (i \* 5));

}

strikes.forEach(strike => {

const row = document.createElement('tr');

const callBid = (Math.random() \* 5 + 0.5).toFixed(2);

const callAsk = (parseFloat(callBid) + 0.1).toFixed(2);

const putBid = (Math.random() \* 5 + 0.5).toFixed(2);

const putAsk = (parseFloat(putBid) + 0.1).toFixed(2);

row.innerHTML = `

<td><strong>$${strike.toFixed(2)}</strong></td>

<td class="call-option">$${callBid}</td>

<td class="call-option">$${callAsk}</td>

<td class="put-option">$${putBid}</td>

<td class="put-option">$${putAsk}</td>

`;

tbody.appendChild(row);

});

}

// Start learning module

function startModule(level) {

showNotification(`Starting ${level} module...`, 'info');

// Simulate module progress

let progress = 0;

const interval = setInterval(() => {

progress += 10;

updateProgress(progress);

if (progress >= 100) {

clearInterval(interval);

completeModule(level);

}

}, 500);

}

// Update progress

function updateProgress(newProgress) {

progress = newProgress;

document.getElementById('progressBar').style.width = `${progress}%`;

document.getElementById('progressBar').textContent = `${progress}%`;

}

// Complete module

function completeModule(level) {

userPoints += 250;

document.getElementById('userPoints').textContent = userPoints.toLocaleString();

showNotification(`Congratulations! You completed the ${level} module and earned 250 points!`, 'success');

// Update user level based on points

if (userPoints >= 5000) {

userLevel = 'professional';

} else if (userPoints >= 3000) {

userLevel = 'advanced';

} else if (userPoints >= 1500) {

userLevel = 'intermediate';

}

document.getElementById('userLevel').textContent = userLevel.charAt(0).toUpperCase() + userLevel.slice(1);

}

// Update dashboard

function updateDashboard() {

// Update all dashboard elements

updatePortfolio();

updateStats();

}

// Show notification

function showNotification(message, type = 'info') {

const notification = document.getElementById('notification');

const notificationText = document.getElementById('notificationText');

notification.className = `notification ${type}`;

notificationText.textContent = message;

notification.classList.add('show');

setTimeout(() => {

notification.classList.remove('show');

}, 3000);

}

// Close modal when clicking outside

window.onclick = function(event) {

const modal = document.getElementById('optionsChainModal');

if (event.target === modal) {

closeOptionsChain();

}

}

// Smooth scrolling for navigation

document.querySelectorAll('a[href^="#"]').forEach(anchor => {

anchor.addEventListener('click', function (e) {

e.preventDefault();

const target = document.querySelector(this.getAttribute('href'));

if (target) {

target.scrollIntoView({

behavior: 'smooth',

block: 'start'

});

}

});

});

</script>

</body>

</html>