```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Real-Time Crypto Calculator</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
    <style>
        :root {
            --primary-color: #825CFF;
            --background: #f8f9fa;
        }

        body.dark-mode {
            background-color: #121212;
            color: white;
        }

        .calculator-box {
            border-radius: 15px;
            box-shadow: 0 4px 12px rgba(0,0,0,0.1);
            transition: all 0.3s ease;
        }
    </style>
</head>
<body class="p-4">
    <div class="container calculator-box bg-white p-4">
        <button onclick="toggleDarkMode()" class="btn btn-secondary mb-3">Toggle Dark
Mode</button>
        <h2 class="text-center mb-4">💰 Crypto Profit/Loss Calculator</h2>

        <div class="row g-3 mb-4">
            <div class="col-md-6">
                <label class="form-label">Select Cryptocurrency:</label>
                <select id="cryptoSelect" class="form-select">
                    <option value="bitcoin">Bitcoin (BTC)</option>
                    <option value="ethereum">Ethereum (ETH)</option>
                </select>
            </div>
            <div class="col-md-6">
                <label class="form-label">Fiat Currency:</label>
                <select id="fiatSelect" class="form-select">
```

```
                    <option value="usd">USD</option>
                    <option value="inr">INR</option>
                </select>
            </div>
        </div>

        <div class="row g-3 mb-4">
            <div class="col-md-4">
                <input type="number" class="form-control" id="investment" placeholder="Investment
Amount">
            </div>
            <div class="col-md-4">
                <input type="number" class="form-control" id="buyPrice" placeholder="Buying Price">
            </div>
            <div class="col-md-4">
                <input type="number" class="form-control" id="sellPrice" placeholder="Selling Price">
            </div>
        </div>

        <div class="result-box p-3 mb-4 bg-light rounded">
            <h5 class="text-center">Result:</h5>
            <div id="livePrice" class="text-success mb-2"></div>
            <div id="profitLoss" class="fs-4 text-center"></div>
        </div>

        <button onclick="calculateProfit()" class="btn btn-primary w-100">Calculate</button>
        <canvas id="profitChart" class="mt-4"></canvas>
    </div>

<script>
let cryptoData = {};

async function loadCurrencies() {
    try {
        const response = await
fetch('https://api.coingecko.com/api/v3/simple/supported_vs_currencies');
        const currencies = await response.json();
        const fiatSelect = document.getElementById('fiatSelect');

        currencies.forEach(currency => {
            const option = document.createElement('option');
            option.value = currency;
            option.textContent = currency.toUpperCase();
            fiatSelect.appendChild(option);
        });
    } catch (error) {
        console.error("Error loading currencies: ", error);
    }
}

async function updateLivePrice() {
    const crypto = document.getElementById('cryptoSelect').value;
    const fiat = document.getElementById('fiatSelect').value;
    try {
        const response = await
fetch(`https://api.coingecko.com/api/v3/simple/price?ids=${crypto}&vs_currencies=${fiat}`);
        const data = await response.json();
        document.getElementById('livePrice').textContent = `Live Price: ${data[crypto][fiat]}
${fiat.toUpperCase()}`;
    } catch (error) {
        document.getElementById('livePrice').textContent = "Error fetching live price.";
    }
}

function calculateProfit() {
    const investment = parseFloat(document.getElementById('investment').value);
    const buyPrice = parseFloat(document.getElementById('buyPrice').value);
    const sellPrice = parseFloat(document.getElementById('sellPrice').value);
    const fiat = document.getElementById('fiatSelect').value.toUpperCase();

    if (isNaN(investment) || isNaN(buyPrice) || isNaN(sellPrice) || investment <= 0 || buyPrice <= 0
|| sellPrice <= 0) {
        document.getElementById('profitLoss').textContent = "Invalid input. Please enter valid
numbers.";
        return;
    }

    const units = investment / buyPrice;
    const profitLoss = (units * (sellPrice - buyPrice)).toFixed(2);

    document.getElementById('profitLoss').textContent = `Total ${profitLoss >= 0 ? 'Profit' :
'Loss'}: ${Math.abs(profitLoss)} ${fiat}`;
    updateChart(profitLoss);
}

function updateChart(profitLoss) {
    const ctx = document.getElementById('profitChart').getContext('2d');
    new Chart(ctx, {
        type: 'bar',
        data: {
            labels: ['Profit/Loss'],
            datasets: [{
                label: 'Amount',
                data: [profitLoss],
                backgroundColor: profitLoss >= 0 ? 'green' : 'red'
            }]
        }
    });
}

function toggleDarkMode() {
    document.body.classList.toggle('dark-mode');
}

document.getElementById('cryptoSelect').addEventListener('change', updateLivePrice);
document.getElementById('fiatSelect').addEventListener('change', updateLivePrice);

loadCurrencies();
updateLivePrice();
setInterval(updateLivePrice, 60000);
</script>

</body>
</html>
```