14. Query Bitcoin Blockchain Data Using BlockCypher API in Node.js

Why: Local nodes are resource-intensive; APIs provide quick access to balances, transactions. BlockCypher's RESTful endpoints speed up analytics or integrations, key for 2026's on-chain data apps like portfolio trackers amid rising adoption.

How to Implement: Use axios for HTTP requests to fetch data. Handle rate limits with async/await. Integrate in bots or dashboards for real-time monitoring.

```
javascript
const axios = require('axios');

// Hack: Get wallet balance
async function getBitcoinBalance(address) {
    try {
        const response = await
axios.get(`https://api.blockcypher.com/v1/btc/main/addrs/${address}/balance`);
        const balance = response.data.balance / 100000000;  // Satoshi to BTC
        return balance;
    } catch (error) {
        console.error(error);
        return null;
    }
}

// Example
getBitcoinBalance('1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa').then(console.log);  // Satoshi
Nakamoto's genesis address
```

Analysis: API calls are sub-second, vs. hours for full node syncs. BlockCypher supports testnets; in 2026, with BTC at new highs, this enables scalable apps, though paid tiers handle high volumes, reducing costs over self-hosted explorers.