







The screenshot shows a code editor with the file `Stock_Price_Generator.js` open. The code generates random stock price data for a 3000-year period, starting from year 3000. It uses a random factor to calculate price changes based on a volatility of 2%.

```
1  const fs = require('fs');
2
3  const rawData : any[] = [];
4  let currentPrice : number = 100; // 시작 가격
5  const volatility : number = 0.02; // 하루 변동성 (2%)
6
7
8  rawData.push('("2999-12-31", 100.00)');
9
10 // 3000년 1월 2일부터 3000년 12월 31일까지 데이터 생성
11 let currentDate : Date = new Date( year: 3000, monthIndex: 0, date: 2); //
12
13 for (let i : number = 1; i <= 365; i++) {
14     const dateString : string = currentDate.toISOString().split( separator: 'T')[0]; //
15     const randomFactor : number = (Math.random() - 0.5) * 2 * volatility; // -2% ~ 2% 범위로 랜덤 변동
16     currentPrice *= (1 + randomFactor); // 변동성 반영
17     rawData.push(`("${dateString}", ${currentPrice.toFixed( fractionDigits: 2)})`); // 변동 후 가격 기록
18     currentDate.setDate(currentDate.getDate() + 1);
19 }
20
21 const fileContent : string = rawData.join(',\n');
22 const now : Date = new Date();
23 const fileName : string = `stock_data_${now.toISOString().split( separator: 'T')[0]}_${now.getHours()}-${now.getMinutes()}-${now.getSeconds()}.txt`;
24
25 fs.writeFileSync(fileName, fileContent);
26
27 console.log(`마지막 줄: ${rawData[rawData.length - 1]}`);
28 console.log(`${fileName} 파일에 데이터가 저장되었습니다.`);
29
```

The execution output at the bottom shows the final price and the file name:

```
마지막 줄: ("3000-12-31", 119.43)
stock_data_2025-04-16_15-35-55.txt 파일에 데이터가 저장되었습니다.

종료 코드 0(으)로 완료된 프로세스
```

Stock_Price_Generator.js

1년간 랜덤 주식 가격 생성

