

A clear PowerShell path into T3N

A proposed companion to the developer quickstart.

01 / The observed gap

The quickstart and file-writing guide show Bash commands. PowerShell uses different environment-variable and file-writing syntax. This is a shell-portability gap in the inspected instructions; official Windows support remains to be confirmed.

02 / A small, concrete improvement

Add a PowerShell 7 tab. Use a masked prompt for the API key, and a single-quoted here-string when writing TypeScript so its template literals remain unchanged.

```
$env:T3N_API_KEY = Read-Host -Prompt 'API key' -MaskInput

@'
const sampleId = 'demo';
console.log(`Ready: ${sampleId}`);
'@ | Set-Content -LiteralPath './new-sample.ts' -Encoding utf8
```

Illustration only. Use a new filename: Set-Content replaces an existing file. This snippet demonstrates shell syntax, not an SDK connection.

03 / Evidence and remaining validation

The companion setup checker passed five local fixture scenarios. The literal file-writing example was also checked. SDK installation, API authentication, and the full client walkthrough have not been run; these belong in the agreed verification scope.

A focused next step

Up to two documentation pages and one agreed walkthrough. **600 USDC**: 300 to start, 300 after acceptance. Delivery within 48 hours after scope, test access, and deposit are confirmed.

SOURCE NOTES

[T3N developer quickstart](#)

[T3N guide to writing code samples](#)