

第48回　「MCP: Advanced Topics」
Anthropic 公式 Claude Academy

Claude Code

標準入出力でつなぐ

公式の教材を、順番に

原石工房
隠れた才能に気づき、花開く

この資料について

原石工房

Anthropic とは関係のない、非公式の資料です

公式の学習サイト Claude Academy の日本語訳です

原文（英語）MCP: Advanced Topics ／ The STDIO transport

改訂で、この回に出てくる最初の挨拶は無くなりました

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

全51回の、今日はここ

原石工房

Claude Code 101	1 2 3 4 5 6 7 8 9 10 11 12	第1回〜第12回
Claude Code in Action	13 14 15 16 17 18 19 20 21	第13回〜第21回
スキル	22 23 24 25 26 27	第22回〜第27回
手分け（subagents）	28 29 30 31	第28回〜第31回
MCP 入門	32 33 34 35 36 37 38 39 40	第32回〜第40回
MCP 応用	41 42 43 44 45 46 47 **48** 49 50 51	第41回〜第51回

今日はここ　第48回

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

今日のレッスン

原石工房

MCP: Advanced Topics の第8回

原題　The STDIO transport

訳　標準入出力でつなぐ

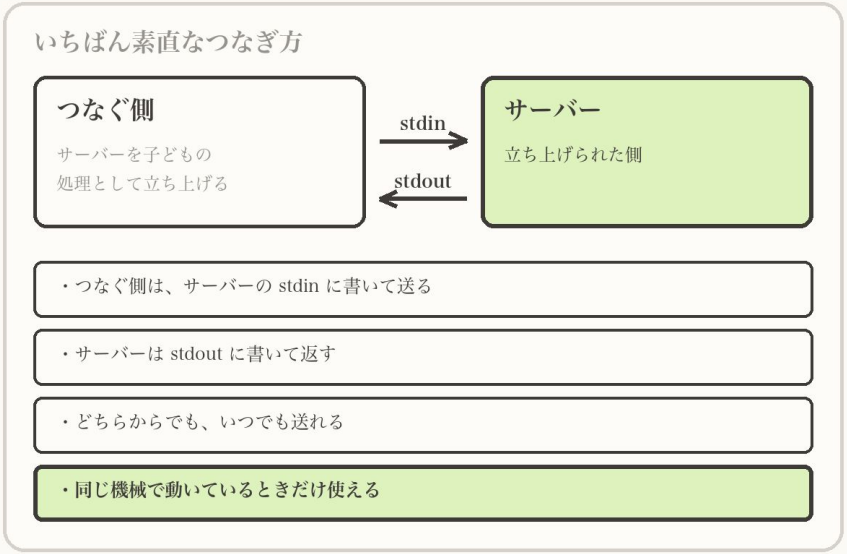
文を運ぶ道筋のうち、いちばん素直なものを見ます

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

① 子どもの処理として立ち上げる

原石工房

原題 The Stdio Transport



管の両端は、同じ機械の中にしかない

つなぐ側が、サーバーを子どもの処理として立ち上げます

stdin で送り、stdout で返す。どちらからでも、いつでも

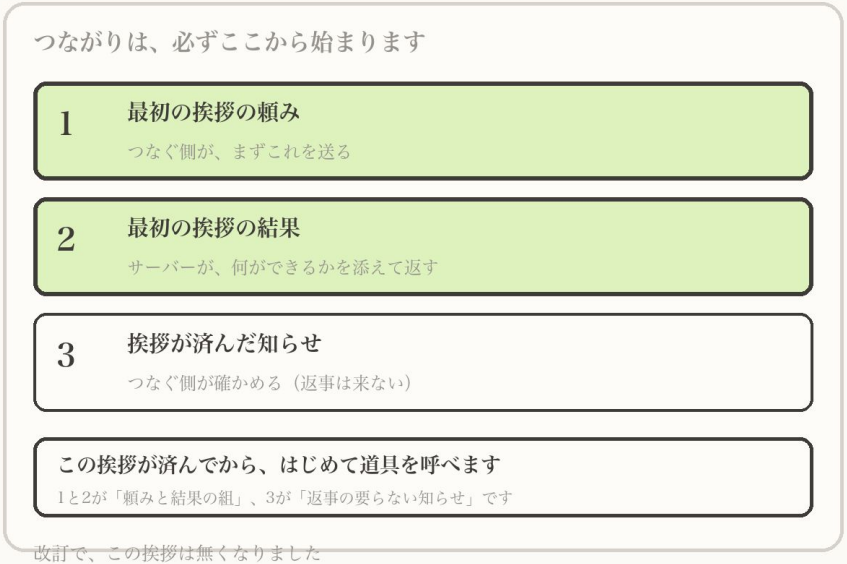
同じ機械で動いているときだけ使えます

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

② 必ず、3つの挨拶から

原石工房

原題 MCP Connection Sequence



前回の2種類が、さっそく両方出てくる

最初の挨拶の頼み、その結果、挨拶が済んだ知らせ

済んでから、はじめて道具を呼べます

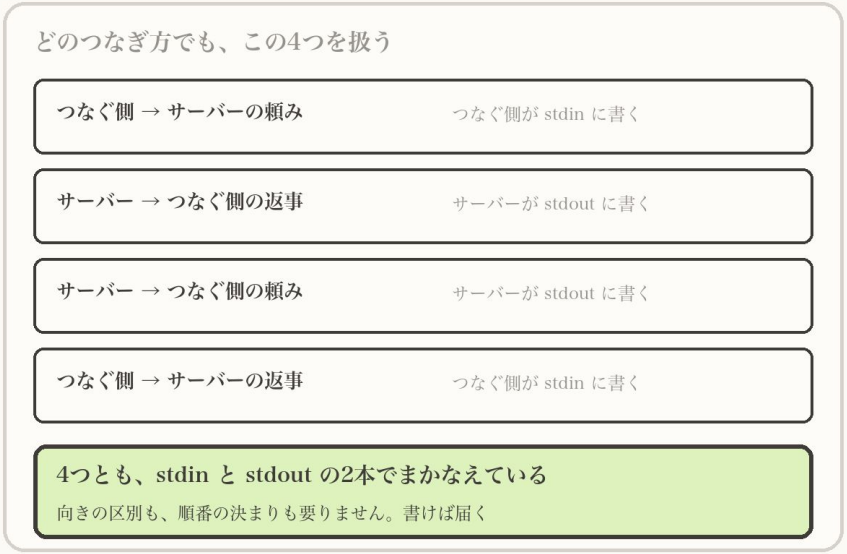
前回の2種類が、ここで両方そろって出てきます

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

③ 4つの場面を、2本でまかなう

原石工房

原題 Four Communication Scenarios



これが、両方向の理想の形

つなぐ側の頼みも返事も、stdin に書く

サーバーの頼みも返事も、stdout に書く

向きの区別も、順番の決まりも要りません

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

素直さが、あとで物差しになる

原石工房

stdin と stdout の、たった2本

- つなぐ側 → サーバーの頼み　stdin
- サーバー → つなぐ側の返事　stdout
- サーバー → つなぐ側の頼み　stdout
- つなぐ側 → サーバーの返事　stdin

4つの場面が、向きの区別なしでまかなえています

書けば届く。それだけです

同じ機械の中だから、余計な決まりごとが一切要りません

HTTP では、このうち2つがそのままではできなくなります

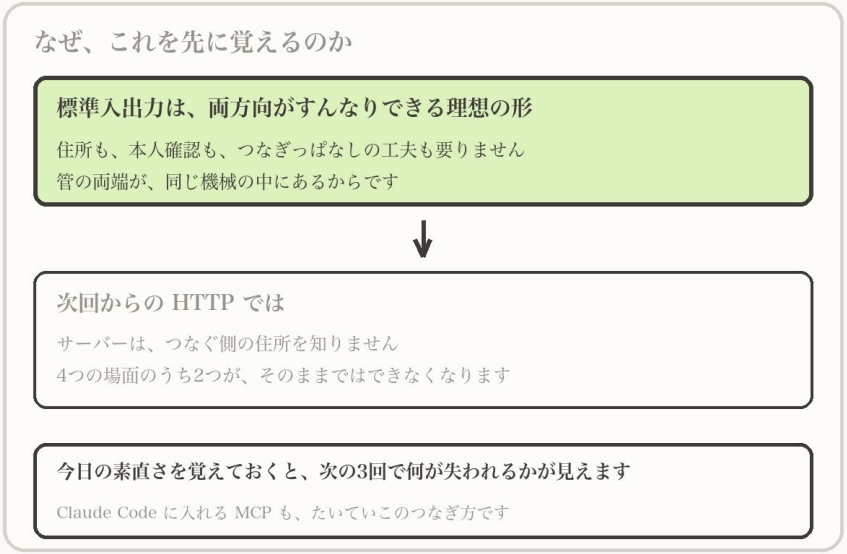
素直さが、あとで物差しになる

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

※ 補足　なぜ「同じ機械だけ」なのか

原石工房

ここから先は原文にありません



窮屈さに取りかかる前の、物差し

管の両端は、同じ機械の中にしかないからです

裏を返すと、余計な決まりごとが一切要りません

Claude Code に入れる MCP も、たいていこのつなぎ方です

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

※ 補足　次回から、窮屈になります

原石工房

HTTP では、サーバーはつなぐ側の住所を知りません

4つの場面のうち2つが、そのままではできなくなります

今日の素直さを覚えておくと、何が失われるかが見えます

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

今回のまとめ

原石工房

サーバーを子どもの処理として立ち上げ、stdin と stdout で話す

画面に JSON を貼れば、その場で試せる

4つの場面が2本でまかなえる。これが理想の形です

出どころ＝ Anthropic 公式 Claude Academy「MCP: Advanced Topics」／
https://academy.claude.com/courses/model-context-protocol-advanced-topics/the-stdio-transport ／ 非公式の日本語訳

その仕組みは、どこまで素直でいられますか



原石工房

隠れた才能に気づき、花開く