Claude Code 最新コマンドチートシート

目立たないけど効く組み込みコマンド、キー操作、CLI。公式の一覧と v2.1.283 までの変更履歴から ／ 2026-09 時点

Givery

知られていないけど効くコマンド

コマンド	何が起きるか	効く場面
`/insights`	手元の最近のセッションを最大200本分析し、よく触るプロジェクト、使い方、つまずき、試すべき機能を HTML にまとめる	月1回の振り返り。分析にもトークンを使う。出力は `~/.claude/usage-data/`
`/btw 質問`	会話に残さない横道の質問。本筋の文脈を汚さない。引数なしで過去の横道の答えを見返せる	作業中にふと確かめたいこと
`/context`	文脈の使い道を色のグリッドで出し、重いツールや膨らんだ記憶の削り方を提案する	急に遅く、高くなったとき
`/doctor`	重複インストール、PATH、壊れた設定ファイル、使っていないスキルや MCP、遅いフックを洗い出して直す	挙動がおかしいときの最初の一手
`/skill-doctor`	スキルごとの文脈コストと、実際に使われた回数を出す	入れすぎたスキルを切る
`/rewind` （Esc 2回）	会話とコードを前の地点に戻す。選んだ地点から先だけを要約することもできる	方向違いに気づいたとき
`/subtask` `/fork` `/branch`	subtask は会話を引き継いだ裏の作業で結果がここに戻る。fork は別のバックグラウンドセッションに写す。branch は自分が分岐側に移る	横道の作業の任せ方を選ぶ
`/goal 条件`	条件を満たすまで、ターンをまたいで働き続ける。 `/goal clear` で解除	「テストが全部通るまで」
`/cd パス`	会話を保ったまま作業ディレクトリを移す	別リポジトリに続きを持ち込む
`/import codex`	Codex、Gemini CLI、Cursor の指示ファイル、MCP、コマンド、サブエージェント、スキルを取り込む。 `--dry-run` で下見	他のツールからの乗り換え
`/fewer-permission-prompts`	記録から読み取り専用のコマンドを拾い、許可リストを `.claude/settings.json` に足す	確認ダイアログが多すぎるとき
`/team-onboarding`	過去30日の使い方から、新メンバーが最初に貼るガイドを作る	チームに展開する前
`/config key=value`	画面を開かずに設定を変える。例は `/config model=sonnet` 。 `-p` でも使える	スクリプトや素早い切り替え
`/copy 2`	2つ前の応答をコピー。コードブロックを選べ、 `w` でファイルに書き出す	生成物だけを取り出す

会話と文脈

- `/clear 名前` 前の会話に名前を付けて、空から始める。 `/resume` で探せる。費用ゼロ
- `/compact 観点` 残すものを指定して要約する
- `/autocompact 500k` 自動要約が走る大きさを決める。 `auto` で既定に戻す
- `/recap` 今のセッションを1行で要約する
- `/rename` `/resume` 名前を付けて、あとから名前で戻る
- `/export ファイル名` 会話をテキストで書き出す
- `/usage` 費用、プランの上限、統計。 `/cost` と `/stats` は別名
- `/memory` CLAUDE.md の編集と自動メモリの管理
- `/add-dir パス` 別のディレクトリも読み書きできるようにする

並列とバックグラウンド

- `/background` セッションを裏に回して端末を空ける。別名 `/bg`
- `/tasks` 裏で動くサブエージェントや作業の一覧と管理
- `/batch 指示` 大きな変更を5〜30の単位に分け、worktree ごとに並列で実装させる
- `/loop 5m 指示` セッションを開いている間、定期的に実行する
- `/schedule` クラウドで動くルーティンを作る。別名 `/routines`
- `/deep-research 問い` 検索を広げて出典を突き合わせ、出典つきのレポートにする
- `/list-agents` メッセージを送れるサブエージェントや別セッションの一覧

レビューと確認

- `/diff` 編集込みの差分を見る
- `/code-review --fix` 正しさのバグを探して直すまで。 `--comment` で PR に投稿。 `/review` は別名
- `/code-review ultra` クラウドで多数のエージェントが深くレビュー。Pro と Max は3回まで無料
- `/security-review` 既定ブランチとの差分から、インジェクションや認可の穴を探す
- `/simplify` 4つのエージェントで整理だけを提案して直す。バグは見ない
- `/verify` `/run` テストに加えて、アプリを動かして確かめる
- `/autofix-pr` クラウドで PR を見張り、CI の失敗やレビューに修正を push する

設定と拡張

- `/model` ←→ で effort も変える。 `s` でこの回だけ
- `/advisor opus` 要所で別のモデルに助言をもらう
- `/output-style` 応答の書き方を切り替える（例 `concise` ）
- `/update-config` 許可、環境変数、フックの追加を言葉で頼むと settings.json を直す
- `/skills` スキルの一覧。 `t` でトークン順に並べる
- `/reload-skills` 作業中に足したスキルを読み直す
- `/statusline` 欲しい表示を言葉で頼むか、シェルのプロンプトから自動で作る
- `/terminal-setup` Shift+Enter で改行できるようにする（VS Code など）

キー操作と入力

- **Esc / Esc 2回** 応答を止める / 入力を消すか、空なら巻き戻し
- **Shift+Tab** 権限モードを順に切り替える
- **Ctrl+O** やり取りの全記録を開く
- **Ctrl+R** 入力履歴を逆向きに検索
- **Ctrl+G** 外部エディタで長いプロンプトを書く
- **Ctrl+B** 実行中のコマンドや作業を裏に回す
- **Ctrl+S** 書きかけのプロンプトを退避し、もう一度で戻す
- **Ctrl+Enter** 待ち行列のメッセージを今すぐ送る
- **Ctrl+X Ctrl+K** 裏のサブエージェントを全部止める
- **Option+P / T / O** モデル / 思考 / fast mode。Windows は Alt
- **Ctrl+J** どの端末でも改行。画像の貼り付けは Ctrl+V（Windows は Alt+V）
- **! @ ?** 行頭でシェル実行 / ファイル名の補完 / 空欄でショートカット一覧

ターミナルから（CLI）

- **名前で戻る** `claude -n 名前` で始め、 `claude -r 名前` で戻る
- **分岐して再開** `claude -c --fork-session` 。元の会話は残る
- **PR から戻る** `claude --from-pr 123` でその PR の会話を選ぶ
- **並列** `claude -w 名前` で worktree、 `--bg` で裏に。 `claude agents` で一覧
- **切り分け** `claude --safe-mode` で CLAUDE.md、スキル、フック、MCP を全部切って起動
- **診断** `claude doctor` は会話を始めずに導入と設定を点検
- **後片付け** `claude project purge` でプロジェクトの手元の記録を全部消す
- **自動実行** `-p` と `--max-turns` 、 `--max-budget-usd`

消えた・名前が変わった

- **削除** `/pr-comments` （v2.1.91）、 `/vim` （ `/config` の Editor mode へ）、 `/ultraplan`
- **`/review`** `/code-review` の別名になった
- **`/fork`** 別セッションに写す意味に変わった。前の動きは `/subtask`
- **`/extra-usage`** `/usage-credits` に改名。旧名も動く
- **`# でメモリ追記`** 廃止。Claude に頼めば直す
- **Ctrl+L 2回で `/clear`** 廃止。Ctrl+L は再描画だけ

Sources（公式・一次情報 / クリックで開く）

1. Commands
2. Interactive mode
3. CLI reference
4. Checkpointing
5. Manage costs
6. Agent view
7. Goal
8. Ultrareview
9. CHANGELOG