CLAUDE.md に書くべきこと チートシート

AIに効くプロジェクト記憶ファイルの構成・書き方・調整 ／ 2026-05 時点 公式仕様準拠

Givery

CLAUDE.mdとは / 階層

毎セッション先頭で全文ロード Claude が会話開始時に読むmarkdownファイル。システムプロンプトではなくユーザーメッセージとして注入されるため、強制ではなく文脈として効く。具体・簡潔ほど追従率が上がる。

4階層・ロード順は広→狭 managed policy → ユーザー~/.claude/CLAUDE.md → プロジェクト./CLAUDE.md または ./.claude/CLAUDE.md → ./CLAUDE.local.md の順に連結。後に読む狭スコープが文脈上で優先されやすい。

```
~/.claude/CLAUDE.md → ./CLAUDE.md → ./CLAUDE.local.md
```

親ディレクトリは起動時・子は遅延 作業ディレクトリから上へ辿った親のCLAUDE.mdは起動時に全文ロード。サブディレクトリ配下はそのフォルダのファイルをClaudeが読んだ時点で初めて読み込む。

managed policyは除外不可 macOS /Library/Application Support/ClaudeCode/CLAUDE.md 等の組織配布版は個人設定で除外できない。会社コーディング規約・セキュリティ方針を全員に強制する用途。

強制したいならhookを併用 CLAUDE.mdは助言であって保証ではない。コミット前lint等の必達処理はPreToolUse/Stop hookやpermissions.denyで技術的に固める。

必ず書く中身

★ プロジェクト概要・構成 アーキテクチャ上の決定とディレクトリ配置。コードを読めば分かる説明ではなく、APIハンドラの置き場所など推測しづらい固定情報。

```
API handlers live in `src/api/handlers/`
```

★ ビルド/テスト/実行コマンド Claudeが推測できないコマンド。標準のnpm test以外に、単体テスト優先や型チェックの作法を明記。

```
Run `npm test` before committing
```

★ コードスタイル規約 言語デフォルトと異なるルールだけ。Claudeが既に知る一般慣習は書かない。

```
Use ES modules (import/export), not CommonJS
```

★ リポジトリ作法 ブランチ命名・PR規約・コミットメッセージ様式など、コードからは読めないチーム合意。

★ 環境の癖・必須env var ローカルRedisが要る、特定の環境変数が必須といった開発環境固有の前提。

★ 既知の落とし穴 非自明な挙動やよくハマる点。コードレビューでClaudeが知っているべきだった指摘は、その場でここへ追記する。

コマンド系

テストランナーと実行範囲 好みのテスト方法を指定。全件実行は遅いので単体優先など、Claudeが選ぶ判断基準を渡す。

```
Prefer running single tests, not the whole suite, for performance
```

完了時の検証手順 一連の変更後に型チェックやビルドを走らせる指示。passかfailの信号を返せる検証を持たせると無人実行が閉じる。

```
Be sure to typecheck when you're done making code changes
```

CLIツールの利用指示 gh/aws/gcloud等を使うよう明記。ghがあればPR作成やissue読みを文脈効率よくこなす。未知ツールは --help から学ばせる。

詳細APIドキュメントは書かない 長いAPI仕様は本文に貼らずリンクで誘導。頻繁に変わる情報・チュートリアルは肥大の原因。

規約・作法

差分のあるスタイルだけ デフォルトと違う規約に絞る。「きれいに書く」のような自明な指示は逆効果。検証可能な具体で書く。

```
Use 2-space indentation
```

ブランチ/PR/コミット規約 命名規則やマージ方針などリポジトリ作法。チーム共有のため./CLAUDE.mdに置きgitへコミット。

矛盾を放置しない 複数のCLAUDE.md・サブディレクトリ・.claude/rules/間で食い違うとClaudeが任意に一方を選ぶ。定期的に棚卸しして衝突を消す。

個人設定はCLAUDE.local.md サンドボックスURLや好みのテストデータなど共有不要の項目は./CLAUDE.local.mdに分け.gitignoreへ。

@import と構成

@path で外部ファイル展開 @相対パス または @絶対パスで他ファイルを取り込む。相対パスはimport元基準で解決。最大4ホップまで再帰可能。

```
git workflow @docs/git-instructions.md
```

importしても文脈は減らない 取り込み先も起動時に全文ロードされ文脈を消費する。分割は整理目的であってトークン削減にはならない。

AGENTS.mdはimportで統合 ClaudeはAGENTS.mdを直接読まない。CLAUDE.md冒頭に@AGENTS.mdと書き、その下にClaude固有指示を足すと両ツールが同一指示を読む。

```
@AGENTS.md\n\n## Claude Code\nUse plan mode for changes under `src/billing/`.
```

大規模は.claude/rules/へ 話題別にrulesファイルへ分割。paths frontmatterで該当ファイルを開いた時だけロードする条件付きルールにでき、文脈を節約。

```
---\npaths:\n - "src/api/**/*.ts"\n---
```

HTMLコメントは保守者メモ用 <!-- -->ブロックコメントは文脈注入前に除去される。トークンを使わず人間向け注記を残せる。コードブロック内のコメントは保持。

書き方のコツ(調整)

200行未満を目安 1ファイル200行を超えると文脈消費が増え追従が落ちる。各行に『これを消すとClaudeがミスするか』を問い、ノーなら削る。

```
if not needed → cut it
```

強調語で追従率を上げる 守らせたい一行にIMPORTANTやYOU MUSTを付けて重みを足す。プロンプトと同じ感覚でチューニングする。

見出しと箇条書きで構造化 markdownヘッダーとbulletで関連指示をグルーピング。密な段落よりスキャンしやすく追従しやすい。

コード同様に運用 挙動が崩れたらレビューし定期的に剪定。変更後はClaudeの挙動が実際に変わったかで効果を検証する。

曖昧さを疑う CLAUDE.mdに書いたのに質問が来る・ルールが無視されるなら、文言が曖昧か全体が長すぎてルールが埋もれている。

アンチパターン

肥大化で指示が埋もれる 長すぎるCLAUDE.mdは半分無視される。ルールが効かない時はまず長さを疑い、容赦なく剪定するかhookへ変換。

READMEの丸写し コードを読めば分かること・ファイル単位の説明は書かない。概要が要るなら@README.mdで参照させる。

頻繁に変わる情報の放置 詳細API仕様や移ろう手順を本文に書くと古い情報が残り誤動作の元。リンク誘導かskillに逃がす。

都度しか要らない知識を常駐 たまにしか使わないドメイン知識やワークフローはskillへ。全会話を膨らませず必要時だけロードされる。

自明な精神論 『clean codeを書け』のような中身のない指示はノイズ。具体的・検証可能な記述に置き換える。

運用コマンド

操作	コマンド/記法	効果
雛形自動生成	/init	コードベースを解析しビルド/テスト/規約入りCLAUDE.mdを生成。既存があれば改善提案
メモリ編集	/memory	ロード中のCLAUDE.md・CLAUDE.local.md・rulesを一覧、選んでエディタで開く。auto memoryのトグルも
会話から追記	# テキスト	プロンプト冒頭の#で、その場の内容をメモリへ保存。『CLAUDE.mdに追加して』と頼んでもよい
別ファイル取込	@path/to/file	@相対/絶対パスで起動時に全文展開。最大4ホップ再帰
AGENTS.md統合	@AGENTS.md	CLAUDE.md冒頭に書くと既存AGENTS.mdを読み込み、下にClaude固有指示を追加できる
話題別ルール	.claude/rules/*.md	話題ごとに分割。paths frontmatterで該当ファイル時のみロードする条件付きにできる
不要な親を除外	claudeMdExcludes	settings.local.jsonでglob指定し、無関係な祖先CLAUDE.mdを読み込まない。managed版は除外不可

Tips・補足

守らせたい一行にはIMPORTANT/YOU MUSTを付けて追従率を底上げする

サブディレクトリのCLAUDE.mdはそのフォルダのファイルを読んだ時に遅延ロード。monorepoの局所規約に有効

ClaudeはAGENTS.mdを直接読まない。@AGENTS.md importかsymlinkで両ツールの指示を一本化する

auto memory(MEMORY.md)はClaudeが自動で書く学習メモ。先頭200行/25KBのみ常駐、CLAUDE.mdとは別系統

強制が必要な処理はCLAUDE.mdではなくhookやpermissions.denyで技術的に固める

Sources（公式・一次情報 / クリックで開く）

1. **公式メモリ仕様** — code.claude.com/docs/en/memory
2. **公式ベストプラクティス** — code.claude.com/docs/en/best-practices
3. **Skills** — code.claude.com/docs/en/skills
4. **Hooks ガイド** — code.claude.com/docs/en/hooks-guide
5. **Settings** — code.claude.com/docs/en/settings
6. **context window** — code.claude.com/docs/en/context-window