

Claude プロンプト・ベストプラクティス 全コッ一覧 (Anthropic公式)

凡例 ✓=YouTube動画でカバー / △=主に開発者向け (動画では省略) 出典: platform.claude.com/docs/ja/build-with-claude/prompt-engineering/claude-prompting-best-practices

No	大分類	手法 (項目)	概要	動画カバー	対応・メモ
1	モデル固有ガイド	Claude Fable 5 のプロンプト	最新モデルの動作差・スキヤフォルディング変更 (別ページ)	△	開発者向け
2	モデル固有ガイド	Claude Opus 4.8 のプロンプト	応答長・effort・ツールトリガー等の個別調整 (別ページ)	△	開発者向け
3	一般原則	明確かつ直接的に (黄金律)	曖昧に察せず具体的に。文脈の薄い同僚が混乱するならAIも混乱	✓	コッ: 具体的に書く
4	一般原則	コンテキストを追加する	指示の理由・背景 (なぜ) を伝えると的確になる	✓	コッ: 具体的に書く (目的・背景)
5	一般原則	例を効果的に使用する (few-shot)	お手本を見せる。関連性・多様性をもたせ3~5個	✓	コッ: お手本を見せる
6	一般原則	XMLタグで構造化する	指示・資料・例を区切って渡し、取り違えを防ぐ	✓	コッ: 区切りをはっきり
7	一般原則	Claudeに役割を与える	一文でもトーン・視点が目的に焦点化される	✓	コッ: 役割を与える
8	一般原則	長いコンテキストのプロンプト	長文は先頭・質問は末尾 (最大30%向上) / XML構造化 / 引用で根拠づけ	✓	コッ: 区切り・抜き出し
9	一般原則	モデルの自己認識	自分のモデル名を正しく名乗らせる / 既定モデルを指定	△	動画では省略
10	出力とフォーマット	コミュニケーションスタイルと冗長	最新モデルは簡潔。途中報告がほしいなら頼む	△	動画では省略
11	出力とフォーマット	応答のフォーマットを制御する	禁止でなく「こうして」 / XML指示子 / 体裁を合わせる / markdown抑制	△	一部はコッ①に反映
12	出力とフォーマット	LaTeX出力	数式をプレーンテキストにしたいとき指示する	△	動画では省略
13	出力とフォーマット	ドキュメント作成	プレゼン等の作成が得意 / 品質まで具体的に注文	△	動画では省略
14	出力とフォーマット	プリフィル応答からの移行	出力制御 / 前置き排除 / 拒否回避 / 継続 / コンテキスト補充	△	開発者向け
15	ツール使用	ツールの使用	「提案して」でなく「実行して」と明示 / 積極・慎重の切替 / 強い表現を控える	✓	一部: 確認・余計なこと
16	ツール使用	並列ツール呼び出しを最適化する	無関係な処理は同時、依存があれば順番に	△	動画では省略
17	思考と推論	過剰な思考と過度な徹底性	effortで調整 / 「迷ったら使う」等の過剰指示を削る	△	動画では省略
18	思考と推論	思考・インターリーブ思考を活用	一般指示 > 手書き手順 / <thinking>例 / 手動CoT / 自己チェックを依頼	✓	自己チェック = 確認と検証
19	エージェントシステム	長期的な推論と状態追跡	コンテキスト認識 / マルチウィンドウ / 状態管理 (JSON・git・進捗メモ)	✓	コッ: ドキュメントを残す
20	エージェントシステム	自律性と安全性のバランス	消す・上書き等の不可逆操作は実行前に確認させる	✓	コッ: 危険は確認させる
21	エージェントシステム	リサーチと情報収集	成功基準を示す / 複数ソースで裏取り	✓	コッ: 調べてから答えて
22	エージェントシステム	サブエージェントのオーケストレーション	自動で委任 / 過剰使用に注意	△	開発者向け
23	エージェントシステム	複雑なプロンプトを連鎖させる	自己修正: ドラフト→レビュー→改良	△	開発者向け
24	エージェントシステム	ファイル作成を減らす	一時ファイルを最後に片付けさせる	△	開発者向け
25	エージェントシステム	過剰な積極性 (過剰設計を避ける)	頼んだことだけ・シンプルに	✓	一部: 余計なことはさせない
26	エージェントシステム	テスト合格への過度な集中・ハード	一般解を書かせる / 回避策を使わせない	△	開発者向け
27	エージェントシステム	ハルシネーションを最小限に	開いていないコードを推測しない・まず読む	✓	コッ: 調べてから・抜き出し
28	機能別のヒント	向上したビジョン機能	画像・動画の分析 / 関連部分にズームインで精度UP	✓	コッ: 画像で渡す
29	機能別のヒント	フロントエンドデザイン	“AIスロップ”を避け、独自性のあるデザインを指示	△	動画では省略
30	移行に関する考慮事項	旧世代→Claude 4.6への移行	具体化 / 修飾語で品質UP / 機能を明示 / 思考設定をadaptive+effort / プリフィル移行 / 怠惰防止調整	△	開発者向け
31	移行に関する考慮事項	Sonnet 4.5→4.6への移行	effort既定の変更・拡張思考の移行パス	△	開発者向け

作成日: 20260624 ご利用は自己責任でお願いします