Claude Code — Document Creation: 20 Essential Guidelines

Setup & Context

1. **CLAUDE.md is your constitution** Place it in project root. Define role, frameworks, output standards. Claude Code reads it automatically at every session start.
2. **One project folder = one context** Keep all related files (sources, drafts, outputs) in the same directory tree. Use `@folder` to reference entire directories when needed.
3. **Start sessions with explicit goal** First prompt: state deliverable, format, audience. Example: "Create a 2000-word policy brief on X for EU decision-makers."

Research & Input

4. **Use @file for local sources** `@document.pdf @notes.md` injects content without copy-paste. More efficient than uploading repeatedly.
5. **Web search is a tool, not a crutch** Ask Claude Code to search, then critically evaluate. Request source URLs, check dates, cross-reference.
6. **Save research as you go** Ask: "Save key findings to `research_notes.md`". Creates persistent reference, reduces context re-consumption.
7. **Separate facts from analysis** Keep a `sources.md` for raw data/quotes, a separate file for your synthesis. Easier to audit, revise, cite.

Brainstorming & Structure

8. **Brainstorm in a scratch file** "Create `brainstorm.md` with 10 angles on topic X". Iterate there before committing to structure.
9. **Outline before drafting** Request: "Create `outline.md` with H2 structure, 2-3 bullet points per section". Approve structure, then expand.
10. **Challenge your own outline** Ask: "What's weak or missing in this outline?" Claude Code should critique, not just execute.

Drafting & Writing

11. **Draft in sections, not monoliths** "Write section 2 based on outline, save to `draft.md`". Incremental builds = cheaper revisions.
12. **Specify tone per output** Same content, different targets: "Rewrite executive summary for technical audience" vs "for ministerial briefing".
13. **Keep master draft sacred** Work on `draft_v1.md`, `draft_v2.md` or use `_working.md` for experiments. Merge only validated changes.

Revision & Editing

14. **Revisions = str_replace, not rewrite** Ask: "In `draft.md`, revise paragraph 3 to strengthen the

causal link". Claude Code edits surgically.

15. **Batch small edits** Collect 3-5 minor fixes, request in one prompt: "Apply these edits to `draft.md`: [list]". Saves turns.
16. **Request diff summary** After edits: "Show me what changed". Keeps you in control without re-reading entire document.

Outputs & Assets

17. **Image prompts are a deliverable** Request: "Write 3 image generation prompts for hero image, style: [X], aspect: 16:9". Save to `image_prompts.md`.
18. **Export-ready formats** For Hugo: "Format as markdown with YAML frontmatter: title, date, summary, tags". Ready to drop in `content/posts/`.
19. **Version your outputs** Use dates or versions: `brief_2025-01-22.md`. Never overwrite without explicit intent.

Workflow Discipline

20. **End sessions with state summary** Ask: "Summarize current state: what's done, what's pending, what files exist". Copy to `STATUS.md` for next session continuity.

Quick Reference Commands

Task	Prompt pattern
Inject file	`@filename.md`
Create file	"Create `filename.md` with..."
Edit file	"In `filename.md`, replace X with Y"
Web research	"Search for..., save findings to `research.md`"
Outline	"Create outline for [topic] in `outline.md`"
Section draft	"Write section N, append to `draft.md`"
Critique	"What's weak in this analysis?"
Image prompt	"Write image prompt for [concept], style [X]"
Status	"Summarize session state"