

⚡ SESSION MANAGEMENT

<code>/clear</code>	Wipe conversation history (<i>/reset /new</i>)
<code>/compact [focus]</code>	Summarize context, keep key info
<code>/resume [id]</code>	Resume a past session (<i>/continue</i>)
<code>/rename [name]</code>	Name the current session
<code>/branch [name]</code>	Fork conversation at this point (<i>/fork</i>)
<code>/rewind</code>	Rewind to an earlier point (<i>/checkpoint</i>)
<code>/export [file]</code>	Export conversation as text
<code>/copy [N]</code>	Copy last (or Nth) response to clipboard
<code>/exit</code>	Exit Claude Code (<i>/quit</i>)

📊 CONTEXT & INFORMATION

<code>/context</code>	Visualize context usage as colored grid
<code>/usage</code>	Show plan usage limits
<code>/cost</code>	Show token usage & cost stats
<code>/diff</code>	Interactive diff viewer for changes
<code>/doctor</code>	Diagnose installation & settings
<code>/insights</code>	Analyze session patterns & friction
<code>/help</code>	Show all commands
<code>/release-notes</code>	View full changelog
<code>/pr-comments [PR]</code>	Fetch GitHub PR comments

💡 **Pro tip:** Use `/compact` at ~80% context. Use `/clear` when switching tasks entirely.

💡 **Cron / at scheduling:**
`echo 'cd /project && claude -p "$(cat prompt.md)" > output.md 2>&1' | at 21:30`

🧠 MODEL & MODE CONTROL

<code>/model [name]</code>	Switch model (← → to adjust effort)
<code>/effort [level]</code>	Set low / medium / high / max / auto. max applies to the current session only and requires Opus 4.6
<code>/fast [on/off]</code>	Toggle speed-optimized mode
<code>/plan [desc]</code>	Enter plan mode (read-only reasoning)
<code>/sandbox</code>	Toggle sandboxed execution

⚙️ CONFIG & PERMISSIONS

<code>/config</code>	Open settings UI (<i>/settings</i>)
<code>/permissions</code>	Manage allow/ask/deny rules (<i>/allowed-tools</i>)
<code>/memory</code>	Edit CLAUDE.md & auto-memory
<code>/init</code>	Bootstrap project with CLAUDE.md
<code>/hooks</code>	View lifecycle hook configs
<code>/keybindings</code>	Open keybinding config file
<code>/login</code>	Sign in to Anthropic account
<code>/logout</code>	Sign out
<code>/privacy-settings</code>	Update privacy prefs (Pro/Max)
<code>/extra-usage</code>	Configure pay-as-you-go overflow
<code>/color [color]</code>	Set prompt bar color for session

💡 **Effort levels:** low for quick tasks, high for complex reasoning, max for Opus 4.6 deep think (session only).

🔌 EXTENSIONS & INTEGRATIONS

<code>/mcp</code>	Manage MCP server connections
<code>/plugin</code>	Manage plugins (install/remove)
<code>/reload-plugins</code>	Hot-reload all active plugins
<code>/agents</code>	Manage sub-agent configurations
<code>/chrome</code>	Configure browser automation
<code>/ide</code>	Manage IDE integrations
<code>/install-github-app</code>	Set up Claude GitHub Actions
<code>/install-slack-app</code>	Install Claude for Slack
<code>/add-dir <path></code>	Add a working directory this session
<code>/desktop</code>	Open session in Desktop app (<i>/app</i>)
<code>/remote-control</code>	Enable remote from claude.ai (<i>/rc</i>)

🎯 BUNDLED SKILLS

<code>/simplify</code>	3-agent parallel code review
<code>/batch <task></code>	Run task across multiple files
<code>/debug</code>	Structured debugging workflow
<code>/loop [interval] <prompt></code>	Schedule recurring prompt in-session
<code>/schedule [desc]</code>	Create cloud scheduled task
<code>/security-review</code>	Scan branch diff for vulnerabilities
<code>/btw <question></code>	Quick side question (no context cost)

💡 **/loop syntax:** `/loop 5m check deploy status` — supports `s m h d` units. Session-scoped; dies on exit.

🔧 CLI FLAGS (TERMINAL)

<code>claude -p "prompt"</code>	Non-interactive single prompt
<code>--output-format json</code>	JSON / stream-json / text output
<code>--model <name></code>	Choose model at launch
<code>--dangerously-skip-permissions</code>	No approval prompts (CI/cron)
<code>--allowedTools "Bash,Read"</code>	Pre-approve specific tools
<code>--permission-mode acceptEdits</code>	Auto-accept file writes
<code>--append-system-prompt "..."</code>	Add to system prompt
<code>--system-prompt-file f.md</code>	Replace system prompt from file
<code>--continue</code>	Resume most recent conversation
<code>--resume <id></code>	Resume specific session
<code>--bare</code>	Skip hooks/skills/MCP discovery
<code>--json-schema '{...}'</code>	Enforce structured output schema

🗝️ KEYBOARD SHORTCUTS

<code>Esc</code>	Cancel current generation / exit menu
<code>Shift + Tab</code>	Cycle: normal → auto → plan mode
<code>Ctrl + 0</code>	Toggle verbose / thinking output
<code>Ctrl + R</code>	Reverse search command history
<code>Ctrl + F</code> <code>Ctrl + F</code>	Kill all background agents (press twice)
<code>@ file</code>	Attach file/dir to context
<code>! command</code>	Run shell with conversation context
<code>/vim</code>	Enable vim keybindings in prompt