

Claude CLI Cheatsheet

The Claude CLI is a powerful command-line tool for interacting with Claude directly from your terminal. Here is a reference guide for common commands and flags to help you get started and work more efficiently.

Installation

To get started, you'll need Node.js and npm.

```
# Install the Claude Code CLI globally
npm install -g @anthropic-ai/claude-code
```

Starting and Continuing Sessions

These are the fundamental commands for starting new conversations or resuming old ones.

Command	Description	Example
claude	Starts a new interactive chat session.	claude
claude "query"	Starts an interactive session with an initial prompt.	claude "Explain this project's architecture."
claude -c or claude --continue	Continues the most recent conversation in the current directory.	claude -c
claude -r "<session-id>" or claude --resume "<session-id>"	Resumes a specific session using its unique ID.	claude -r "abc123"

Non-Interactive and Scripting Commands

For quick, one-off queries or for use in scripts.

Command	Description	Example
claude -p "query" or claude --print "query"	Executes a prompt and prints the response, then exits.	claude -p "Write a shell script to list all .js files."

<code>cat file.txt claude -p "query"</code>	Pipes the content of a file to Claude for processing.	<code>cat README.md claude -p "summarize this file"</code>
<code>claude --output-format json</code>	Specifies the output format for a non-interactive query. Useful for scripting.	<code>claude -p "List my project files" --output-format json</code>

In-Session Slash Commands

These commands are used *within* an active interactive session.

Command	Description	Example
<code>/help</code>	Displays a list of all available slash commands.	<code>/help</code>
<code>/init</code>	Creates a CLAUDE.md file in your project root for providing project-wide context.	<code>/init</code>
<code>/clear</code>	Clears the current conversation history and context, but stays in the same session.	<code>/clear</code>
<code>/compact</code>	Summarizes the current conversation to reduce the token count and prevent context window limits.	<code>/compact</code>
<code>/review</code>	Asks Claude to review a file, code block, or pull request.	<code>/review my_new_feature.js</code>
<code>/model</code>	Allows you to switch to a different Claude model (e.g., Opus, Sonnet) for the current session.	<code>/model sonnet</code>
<code>> summarize this project</code>	A natural language prompt to get an overview of the codebase.	<code>> summarize this project</code>

Important Flags and Options

These flags can be added to your claude command to customize its behavior.

Flag	Description	Example
--model <name>	Specifies the model to use for the session.	claude --model claude-3-5-sonnet-20240620
--verbose	Enables verbose logging to see the full turn-by-turn output for debugging.	claude --verbose
--add-dir <path>	Adds an additional directory to Claude's context.	claude --add-dir ../docs
--dangerously-skip-permissions	Bypasses permission prompts. Use with caution.	claude --dangerously-skip-permissions
--allowedTools "<tools>"	A space-separated list of tools Claude can use without prompting.	claude --allowedTools "Bash(git status:*)"