

TÀI LIỆU HƯỚNG DẪN SỬ DỤNG
USER GUIDE

Claude Code

Công cụ lập trình bằng dòng lệnh của Anthropic
Anthropic's command-line coding tool

Biên soạn theo bản 2.1.268 · macOS · Windows · Linux
Written for version 2.1.268 · macOS · Windows · Linux

Từ cài đặt đến hook, skill, MCP và tự động hoá
From installation to hooks, skills, MCP and automation

Phiên bản 1.0 — Tháng 9 năm 2026
Version 1.0 — September 2026

MỤC LỤC

CONTENTS

Chương 1. Claude Code là gì — <i>Chapter 1. What is Claude Code</i>	5
1.1. Khác gì so với hỏi Claude trên web — <i>How it differs from asking Claude on the web</i>	5
1.2. Những việc Claude Code làm tốt — <i>What Claude Code does well</i>	5
1.3. Các bản dùng được — <i>Available versions</i>	6
1.4. Yêu cầu máy — <i>System requirements</i>	6
Chương 2. Cài đặt và đăng nhập — <i>Chapter 2. Installation and sign-in</i>	8
2.1. Cài đặt trên macOS và Linux — <i>Installing on macOS and Linux</i>	8
2.2. Cài đặt trên Windows — <i>Installing on Windows</i>	8
2.3. Kiểm tra sau khi cài — <i>Checking after installation</i>	8
2.4. Đăng nhập — <i>Signing in</i>	9
2.5. Cập nhật và gỡ cài đặt — <i>Updating and uninstalling</i>	9
Chương 3. Phiên làm việc đầu tiên — <i>Chapter 3. Your first session</i>	10
3.1. Mở đúng chỗ — <i>Opening in the right place</i>	10
3.2. Giao việc — <i>Giving tasks</i>	10
3.3. Kết thúc và quay lại — <i>Ending and returning</i>	10
Chương 4. Bàn phím và thao tác trong phiên — <i>Chapter 4. Keyboard and in-session actions</i>	12
4.1. Phím tắt — <i>Keyboard shortcuts</i>	12
4.2. Ba ký tự mở đầu đáng nhớ — <i>Three leading characters worth remembering</i>	13
4.3. Đưa ảnh vào câu hỏi — <i>Adding images to a prompt</i>	13
4.4. Chế độ vim — <i>Vim mode</i>	13
Chương 5. Danh mục lệnh gạch chéo — <i>Chapter 5. Slash command reference</i>	14
5.1. Phiên làm việc và ngữ cảnh — <i>Sessions and context</i>	14
5.2. Thư mục, mô hình, chế độ — <i>Folders, models, modes</i>	14
5.3. Bộ nhớ, cấu hình, mở rộng — <i>Memory, configuration, extensions</i>	15
5.4. Kỹ năng kèm sẵn — <i>Bundled skills</i>	16
Chương 6. Tham số dòng lệnh — <i>Chapter 6. Command-line options</i>	17
6.1. Các tham số dùng nhiều nhất — <i>The most used options</i>	17
6.2. Chạy không cần ngồi trước màn hình — <i>Running without sitting at the screen</i>	17
6.3. Lệnh con — <i>Subcommands</i>	18
Chương 7. Quyền và an toàn — <i>Chapter 7. Permissions and safety</i>	20
7.1. Các chế độ quyền — <i>Permission modes</i>	20
7.2. Viết quy tắc quyền — <i>Writing permission rules</i>	20
7.3. Phạm vi tệp — <i>File scope</i>	21

7.4. Hộp cát cho lệnh shell — <i>Sandbox for shell commands</i>	21
Chương 8. CLAUDE.md và bộ nhớ — <i>Chapter 8. CLAUDE.md and memory</i>	23
8.1. Các cấp tệp CLAUDE.md — <i>CLAUDE.md file levels</i>	23
8.2. Viết cho hiệu quả — <i>Writing effectively</i>	23
8.3. Chia nhỏ bằng <code>.claude/rules/</code> — <i>Splitting up with .claude/rules/</i>	24
8.4. Bộ nhớ tự động — <i>Auto memory</i>	24
Chương 9. Tệp thiết lập — <i>Chapter 9. Settings files</i>	26
9.1. Bốn tệp và phạm vi — <i>Four files and their scope</i>	26
9.2. Các khoá hay dùng — <i>Commonly used keys</i>	26
9.3. Một tệp thiết lập mẫu — <i>A sample settings file</i>	27
Chương 10. Kỹ năng và lệnh riêng — <i>Chapter 10. Skills and custom commands</i>	29
10.1. Tạo một kỹ năng — <i>Creating a skill</i>	29
10.2. Các trường khai báo — <i>Declaration fields</i>	29
10.3. Tham số — <i>Arguments</i>	30
10.4. Nơi đặt kỹ năng — <i>Where to put skills</i>	30
Chương 11. Trợ lý con và gói mở rộng — <i>Chapter 11. Subagents and plugins</i>	31
11.1. Trợ lý con (subagent) — <i>Subagents</i>	31
11.2. Gói mở rộng (plugin) — <i>Plugins</i>	31
11.3. Dòng trạng thái riêng — <i>Custom status line</i>	31
Chương 12. Hook — tự động hoá theo sự kiện — <i>Chapter 12. Hooks — event-driven automation</i>	32
12.1. Các sự kiện chính — <i>Main events</i>	32
12.2. Cấu trúc khai báo — <i>Declaration structure</i>	32
12.3. Mã thoát quyết định điều gì — <i>What the exit code decides</i>	33
12.4. Ví dụ: chặn lệnh xoá đệ quy — <i>Example: blocking recursive deletion</i>	33
Chương 13. MCP — nối Claude với công cụ ngoài — <i>Chapter 13. MCP — connecting Claude to external tools</i>	34
13.1. Thêm máy chủ — <i>Adding a server</i>	34
13.2. Ba phạm vi — <i>Three scopes</i>	34
13.3. Quản lý — <i>Management</i>	35
Chương 14. Làm việc với Git và GitHub — <i>Chapter 14. Working with Git and GitHub</i>	36
14.1. Những việc giao được bằng một câu — <i>Tasks you can delegate in one sentence</i>	36
14.2. Soát mã — <i>Code review</i>	36
14.3. Dòng ghi công — <i>Attribution line</i>	36
14.4. Nhánh riêng cho mỗi việc — <i>A separate branch for each task</i>	37
Chương 15. Mô hình, ngữ cảnh và chi phí — <i>Chapter 15. Models, context and cost</i>	38
15.1. Chọn mô hình — <i>Choosing a model</i>	38

15.2. Ngữ cảnh — thứ quyết định chất lượng câu trả lời — <i>Context — what determines response quality</i>	38
15.3. Xem mức dùng — <i>Viewing usage</i>	39
Chương 16. Trình soạn thảo, chạy nền, đám mây — <i>Chapter 16. Editors, background runs, cloud</i>	40
16.1. VS Code và JetBrains — <i>VS Code and JetBrains</i>	40
16.2. Chạy nền — <i>Running in the background</i>	40
16.3. Ứng dụng máy tính và bản chạy trên web — <i>Desktop app and web version</i>	40
Chương 17. Xử lý sự cố — <i>Chapter 17. Troubleshooting</i>	41
17.1. Bảng tra nhanh — <i>Quick reference table</i>	41
17.2. Hai lệnh nên nhớ — <i>Two commands worth remembering</i>	42
Phụ lục A. Bảng tra một trang — <i>Appendix A. One-page quick reference</i>	43
A.1. Mười lệnh dùng hằng ngày — <i>Ten everyday commands</i>	43
A.2. Phím tắt đầy đủ — <i>Full keyboard shortcuts</i>	43
Phụ lục B. Thuật ngữ Anh – Việt — <i>Appendix B. English – Vietnamese glossary</i>	46
Phụ lục C. Các tệp và thư mục — <i>Appendix C. Files and folders</i>	47

Chương 1. Claude Code là gì — Chapter 1. What is Claude Code

Claude Code là công cụ lập trình chạy trong cửa sổ dòng lệnh (Terminal). Thay vì mở trình soạn thảo rồi tự gõ từng dòng mã, bạn mô tả việc cần làm bằng tiếng Việt hoặc tiếng Anh; Claude đọc mã nguồn trong thư mục hiện tại, sửa tệp, chạy lệnh, chạy thử và báo lại kết quả. Công cụ do Anthropic phát hành, dùng chung tài khoản với Claude trên web.

Claude Code is a coding tool that runs in a command-line window (Terminal). Instead of opening an editor and typing each line of code yourself, you describe what needs to be done in Vietnamese or English; Claude reads the source code in the current folder, edits files, runs commands, runs tests and reports back the results. The tool is released by Anthropic and uses the same account as Claude on the web.

1.1. Khác gì so với hỏi Claude trên web — How it differs from asking Claude on the web

Tiêu chí <i>Criterion</i>	Claude trên web (claude.ai) <i>Claude on the web (claude.ai)</i>	Claude Code
Truy cập tệp <i>File access</i>	Phải tự chép dán nội dung vào ô chat <i>You must copy and paste content into the chat box yourself</i>	Tự đọc và sửa tệp trong thư mục làm việc <i>Reads and edits files in the working folder by itself</i>
Chạy lệnh <i>Running commands</i>	Không <i>No</i>	Chạy được lệnh Terminal, cài gói, chạy thử, chạy git <i>Can run Terminal commands, install packages, run tests, run git</i>
Nhớ bối cảnh dự án <i>Remembering project context</i>	Chỉ trong một cuộc trò chuyện <i>Only within one conversation</i>	Đọc tệp <code>CLAUDE.md</code> , nhớ quy ước riêng của dự án <i>Reads the CLAUDE.md file and remembers the project's own conventions</i>
Tự động hoá <i>Automation</i>	Không <i>No</i>	Chạy được trong kịch bản shell, CI, GitHub Actions <i>Can run in shell scripts, CI, GitHub Actions</i>
Nơi chạy <i>Where it runs</i>	Trình duyệt <i>Browser</i>	Terminal, VS Code, JetBrains, ứng dụng máy tính, trình duyệt <i>Terminal, VS Code, JetBrains, desktop app, browser</i>

1.2. Những việc Claude Code làm tốt — What Claude Code does well

- **Sửa lỗi:** dán thông báo lỗi vào, Claude tìm nguyên nhân trong mã nguồn rồi sửa.
 - *Fixing bugs:* paste in the error message, and Claude finds the cause in the source code and fixes it.
- **Viết tính năng mới** theo mô tả bằng lời, có chạy thử trước khi bàn giao.
 - *Writing new features* from a verbal description, with tests run before handover.
- **Giải thích mã nguồn lạ:** mở một dự án chưa từng đọc và hỏi “phần xử lý đăng nhập nằm ở đâu”.
 - *Explaining unfamiliar code:* open a project you have never read and ask “where is the login handling?”.
- **Việc lặp đi lặp lại:** đổi tên biến trên 40 tệp, chuyển đổi định dạng dữ liệu, viết kiểm thử.
 - *Repetitive work:* renaming a variable across 40 files, converting data formats, writing tests.
- **Thao tác Git:** xem khác biệt, tạo commit, mở pull request, xử lý xung đột khi gộp nhánh.
 - *Git operations:* viewing diffs, creating commits, opening pull requests, resolving merge conflicts.

- **Việc không phải lập trình:** đọc một thư mục tài liệu rồi tóm tắt, đổi tên hàng loạt tệp, xử lý bảng tính, soạn tài liệu như chính tệp bạn đang đọc.
 - ***Non-programming work:** reading a folder of documents and summarising it, batch-renaming files, processing spreadsheets, writing documents such as the very file you are reading.*

GHI CHÚ — Claude Code chỉ nhìn thấy thư mục bạn khởi động nó (thư mục làm việc) và các thư mục thêm bằng `--add-dir`. Nó không tự lục các nơi khác trong máy.

NOTE — *Claude Code only sees the folder you start it in (the working folder) and folders added with `--add-dir`. It does not search other places on the computer by itself.*

1.3. Các bản dùng được — Available versions

Nơi chạy <i>Where it runs</i>	Cách mở <i>How to open</i>	Dùng khi <i>Use when</i>
Terminal	Gõ <code>claude</code> trong thư mục dự án <i>Type <code>claude</code> in the project folder</i>	Cách dùng chính, đầy đủ tính năng nhất <i>The main way to use it, with the most complete features</i>
VS Code / Cursor	Cài tiện ích mở rộng Claude Code <i>Install the Claude Code extension</i>	Muốn thấy khác biệt tệp ngay trong trình soạn thảo <i>You want to see file diffs right in the editor</i>
JetBrains	Cài plugin từ Marketplace <i>Install the plugin from the Marketplace</i>	Dùng IntelliJ, PyCharm, WebStorm... <i>You use IntelliJ, PyCharm, WebStorm...</i>
Ứng dụng máy tính <i>Desktop app</i>	Tải Claude Code cho macOS / Windows <i>Download Claude Code for macOS / Windows</i>	Muốn giao diện cửa sổ, chạy nhiều phiên song song <i>You want a windowed interface and several sessions running in parallel</i>
Trình duyệt <i>Browser</i>	<code>claude.ai/code</code>	Giao việc từ máy khác, chạy trên máy chủ của Anthropic <i>Assigning work from another computer, running on Anthropic's servers</i>

1.4. Yêu cầu máy — System requirements

- macOS 13 trở lên, Windows 10 trở lên (có hoặc không có WSL), hoặc Linux (Ubuntu 20.04+, Debian 10+).
 - *macOS 13 or later, Windows 10 or later (with or without WSL), or Linux (Ubuntu 20.04+, Debian 10+).*
- RAM tối thiểu 4 GB.
 - *At least 4 GB of RAM.*
- Kết nối Internet — toàn bộ phần suy luận chạy trên máy chủ Anthropic, máy của bạn chỉ là nơi thao tác tệp.
 - *An Internet connection — all reasoning runs on Anthropic's servers; your computer is only where files are handled.*
- Tài khoản Claude gói Pro/Max, hoặc khoá API Anthropic tính tiền theo lượng dùng, hoặc tài khoản Amazon Bedrock / Google Vertex AI của doanh nghiệp.
 - *A Claude Pro/Max account, or an Anthropic API key billed by usage, or a business Amazon Bedrock / Google Vertex AI account.*

- Node.js chỉ cần khi cài bằng npm. Bản cài đặt gốc (native) đã kèm sẵn, không cần Node.
 - *Node.js is only needed when installing with npm. The native installer includes everything and does not need Node.*

Chương 2. Cài đặt và đăng nhập — Chapter 2. Installation and sign-in

2.1. Cài đặt trên macOS và Linux — Installing on macOS and Linux

Cách được khuyến nghị là bản cài đặt gốc — một tệp chạy sẵn, tự cập nhật, không phụ thuộc Node.js. Mở Terminal và chạy:

The recommended method is the native installer — a ready-to-run file that updates itself and does not depend on Node.js. Open Terminal and run:

```
curl -fsSL https://claude.ai/install.sh | bash
```

Lệnh này đặt chương trình vào `~/.local/share/claude/versions/` và tạo liên kết `~/.local/bin/claude`. Nếu gõ `claude` mà máy báo không tìm thấy lệnh, thư mục `~/.local/bin` chưa nằm trong biến `PATH`. Thêm bằng:

This command places the program in `~/.local/share/claude/versions/` and creates the link `~/.local/bin/claude`. If you type `claude` and the computer reports that the command is not found, the `~/.local/bin` folder is not yet in the `PATH` variable. Add it with:

```
# với zsh — vỏ lệnh mặc định của macOS
# for zsh — the default macOS shell
echo 'export PATH="$HOME/.local/bin:$PATH"' >> ~/.zshrc
source ~/.zshrc
```

Cài bằng npm — Installing with npm

Nếu máy đã có Node.js 18 trở lên và bạn quen dùng npm:

If the computer already has Node.js 18 or later and you are used to npm:

```
npm install -g @anthropic-ai/claude-code
```

CẢNH BÁO — Không chạy lệnh cài npm kèm `sudo`. Cài bằng quyền quản trị làm tệp cấu hình thuộc về người dùng root, sau đó Claude Code không ghi được thiết lập và báo lỗi quyền truy cập. Nếu lỡ làm rồi, gỡ bằng `sudo npm uninstall -g @anthropic-ai/claude-code` rồi cài lại theo cách gốc ở trên.

WARNING — Do not run the `npm install` command with `sudo`. Installing with administrator rights makes the configuration files owned by the root user, after which Claude Code cannot write its settings and reports permission errors. If you have already done so, remove it with `sudo npm uninstall -g @anthropic-ai/claude-code` and then reinstall using the native method above.

2.2. Cài đặt trên Windows — Installing on Windows

Trên PowerShell, chạy:

In PowerShell, run:

```
irm https://claude.ai/install.ps1 | iex
```

Bản chạy thẳng trên Windows cần Git for Windows (Claude Code dùng Git Bash để chạy các lệnh shell). Cách còn lại là cài trong WSL — khi đó mọi thứ giống hệt Linux.

The version that runs directly on Windows needs Git for Windows (Claude Code uses Git Bash to run shell commands). The alternative is to install inside WSL — everything then works exactly as on Linux.

2.3. Kiểm tra sau khi cài — Checking after installation

```
# Xem phiên bản đang dùng
# Show the version in use
claude --version
```

```
# Kiểm tra sức khỏe bản cài: đường dẫn, quyền, cập nhật, cấu hình
# Check install health: paths, rights, updates, config
claude doctor
```

Trong một phiên đang chạy, gõ `/doctor` cho bản kiểm tra đầy đủ hơn — lệnh này còn sửa được một số lỗi cấu hình.

In a running session, type `/doctor` for a more thorough check — this command can also fix some configuration errors.

2.4. Đăng nhập — *Signing in*

Lần đầu gõ `claude`, chương trình hỏi cách xác thực:

The first time you type `claude`, the program asks how to authenticate:

- 1. Tài khoản Claude (Pro/Max)** — mở trình duyệt để đăng nhập rồi quay lại Terminal. Dùng chung hạn mức với claude.ai, không phải trả thêm theo lượt.
 - 1. Claude account (Pro/Max)** — opens the browser to sign in, then returns to Terminal. Shares the usage limit with claude.ai, with no extra per-use charge.
- 2. Anthropic Console (khoá API)** — tính tiền theo số token dùng, cần thẻ thanh toán.
 - 2. Anthropic Console (API key)** — billed by the number of tokens used; requires a payment card.

Thông tin đăng nhập lưu trong Keychain của macOS. Các lệnh liên quan:

Sign-in credentials are stored in the macOS Keychain. Related commands:

```
claude auth          # xem trạng thái đăng nhập
                    # show sign-in status

# tạo khoá dùng lâu dài cho máy chủ, CI (cần gói trả phí)
# create a long-lived key for servers, CI (paid plan needed)
claude setup-token
```

Trong phiên: `/login` để đổi tài khoản, `/logout` để đăng xuất, `/status` để xem đang dùng tài khoản nào.

Within a session: `/login` to switch accounts, `/logout` to sign out, `/status` to see which account is in use.

2.5. Cập nhật và gỡ cài đặt — *Updating and uninstalling*

```
claude update        # kiểm tra và cài bản mới
                    # check for and install a new version
claude install stable # cài lại bản ổn định
                    # reinstall the stable version
claude install latest # cài bản mới nhất
                    # install the latest version
claude migrate-installer # chuyển từ bản npm sang bản gốc
                    # switch from the npm version to native
```

GHI CHÚ — Bản gốc tự kiểm tra cập nhật ở nền. Muốn tắt, đặt biến môi trường `DISABLE_AUTOUPDATER=1` trong tệp `~/.zshrc`.

NOTE — The native version checks for updates in the background. To turn this off, set the environment variable `DISABLE_AUTOUPDATER=1` in the `~/.zshrc` file.

Chương 3. Phiên làm việc đầu tiên — Chapter 3. Your first session

3.1. Mở đúng chỗ — Opening in the right place

Claude Code lấy thư mục hiện hành làm phạm vi làm việc. Hãy chuyển vào thư mục dự án rồi mới gọi: *Claude Code uses the current folder as its working scope. Move into the project folder before calling it:*

```
cd ~/du-an/trang-web
claude
```

Lần đầu mở một thư mục, chương trình hỏi bạn có tin thư mục này không. Câu hỏi đó tồn tại vì Claude sẽ đọc các tệp cấu hình nằm trong thư mục — chỉ trả lời có với thư mục do bạn tạo hoặc từ nguồn tin cậy. *The first time you open a folder, the program asks whether you trust it. The question exists because Claude will read the configuration files in the folder — answer yes only for folders you created or that come from a trusted source.*

Muốn cho Claude thấy thêm thư mục khác ngoài thư mục hiện tại: *To let Claude see another folder besides the current one:*

```
claude --add-dir ~/du-an/thu-vien-chung
```

3.2. Giao việc — Giving tasks

Gõ yêu cầu bằng tiếng Việt rồi nhấn Enter. Vài ví dụ theo mức độ: *Type your request in Vietnamese and press Enter. A few examples by level:*

```
> dự án này làm gì, cấu trúc thư mục ra sao
> what does this project do, how is the folder structure organised
> thêm nút đăng xuất vào trang chủ
> add a sign-out button to the home page
> chạy thử toàn bộ test rồi sửa những chỗ hỏng
> run the whole test suite, then fix whatever is broken
> đổi tên hàm tinhTien thành tinhThanhTien ở tất cả các tệp
> rename the function tinhTien to tinhThanhTien in all files
```

Claude làm việc theo vòng: đọc tệp liên quan → đề xuất thay đổi → xin phép trước khi sửa tệp hoặc chạy lệnh → báo kết quả. Mỗi lần xin phép, bạn chọn cho phép một lần, cho phép luôn với loại thao tác đó, hoặc từ chối kèm lời giải thích.

Claude works in a loop: read the relevant files → propose changes → ask permission before editing files or running commands → report the results. Each time it asks, you choose to allow once, always allow that type of action, or decline with an explanation.

MẸO — Yêu cầu càng cụ thể, kết quả càng sát. “Sửa lỗi đăng nhập” mơ hồ hơn nhiều so với “khi mật khẩu sai, trang báo lỗi trắng thay vì hiện thông báo — hãy sửa và thêm test cho trường hợp đó”.

TIP — *The more specific the request, the closer the result. “Fix the login bug” is far vaguer than “when the password is wrong, the page shows a blank error instead of a message — fix it and add a test for that case”.*

3.3. Kết thúc và quay lại — Ending and returning

Thao tác Action	Cách làm How to do it
Thoát phiên Exit the session	Gõ <code>/exit</code> , hoặc nhấn Ctrl + D <i>Type <code>/exit</code>, or press Ctrl + D</i>
Dừng việc Claude đang làm	Nhấn Esc một lần

<i>Stop what Claude is doing</i>	<i>Press Esc once</i>
Quay lại cuộc trò chuyện gần nhất <i>Return to the most recent conversation</i>	<code>claude -c</code> (hoặc <code>claude --continue</code>) <code>claude -c</code> (or <code>claude --continue</code>)
Chọn phiên cũ trong danh sách <i>Choose an old session from a list</i>	<code>claude --resume</code> , hoặc gõ <code>/resume</code> khi đang trong phiên <code>claude --resume</code> , or type <code>/resume</code> while in a session
Xoá nội dung trò chuyện, giữ phiên <i>Clear the conversation, keep the session</i>	Gõ <code>/clear</code> Type <code>/clear</code>
Quay ngược về trạng thái trước <i>Go back to an earlier state</i>	Gõ <code>/rewind</code> , hoặc nhấn Esc hai lần Type <code>/rewind</code> , or press Esc twice

Lịch sử trò chuyện lưu trong `~/.claude/projects/`, tách theo từng thư mục dự án. Nhờ vậy `claude -c` mở đúng cuộc trò chuyện thuộc thư mục bạn đang đứng.

Conversation history is stored in `~/.claude/projects/`, separated by project folder. That is why `claude -c` opens the conversation belonging to the folder you are in.

Chương 4. Bàn phím và thao tác trong phiên — Chapter 4. Keyboard and in-session actions

4.1. Phím tắt — Keyboard shortcuts

Phím Key	Tác dụng Effect
Esc	Dừng ngay việc Claude đang làm, giữ nguyên cuộc trò chuyện <i>Immediately stop what Claude is doing, keeping the conversation</i>
Esc Esc	Mở bảng quay ngược (rewind) để lùi về một bước trước đó <i>Open the rewind panel to go back one step</i>
Ctrl + C	Xoá dòng đang gõ; nhấn hai lần liên tiếp để thoát <i>Clear the line being typed; press twice in a row to exit</i>
Ctrl + D	Thoát phiên <i>Exit the session</i>
Ctrl + L	Xoá màn hình, giữ nguyên ngữ cảnh <i>Clear the screen, keeping the context</i>
Ctrl + R	Tìm trong lịch sử câu lệnh đã gõ <i>Search the history of typed prompts</i>
Mũi tên ↑ / ↓ <i>Arrow keys ↑ / ↓</i>	Duyệt lại các câu đã gõ trước đó <i>Browse previously typed prompts</i>
Shift + Tab	Đổi chế độ quyền: hỏi từng bước → tự nhận sửa tệp → chế độ lập kế hoạch <i>Switch permission mode: ask at each step → auto-accept file edits → plan mode</i>
Tab	Gợi ý hoàn tất tên tệp, tên lệnh <i>Autocomplete file names and command names</i>
Shift + Enter	Xuống dòng trong cùng một câu (chạy <code>/terminal-setup</code> một lần để bật) <i>New line within the same prompt (run /terminal-setup once to enable)</i>
Option + Enter	Xuống dòng — cách dùng được trên hầu hết Terminal của macOS <i>New line — the method that works in most macOS Terminals</i>
Ctrl + B	Đưa việc đang chạy xuống nền, lấy lại ô nhập <i>Send the running task to the background and get the input box back</i>
Ctrl + V	Dán ảnh từ clipboard vào câu hỏi <i>Paste an image from the clipboard into the prompt</i>

GHI CHÚ — Phím Shift + Enter chỉ hoạt động sau khi chạy `/terminal-setup` — lệnh này ghi thêm một dòng cấu hình cho Terminal hoặc iTerm2. Nếu không muốn cấu hình, dùng Option + Enter, hoặc gõ dấu \ rồi Enter.

NOTE — Shift + Enter only works after running `/terminal-setup` — this command adds a configuration line for Terminal or iTerm2. If you do not want to configure it, use Option + Enter, or type \ and then Enter.

4.2. Ba ký tự mở đầu đáng nhớ — Three leading characters worth remembering

Gõ Type	Nghĩa là Meaning	Ví dụ Example
@	Chèn đường dẫn tệp hoặc thư mục vào câu hỏi <i>Insert a file or folder path into the prompt</i>	<code>giải thích @src/app.py</code> <code>explain @src/app.py</code>
!	Chạy thẳng một lệnh shell, kết quả đưa vào ngữ cảnh <i>Run a shell command directly; the output goes into the context</i>	<code>!git status</code>
#	Ghi điều vừa nói vào bộ nhớ dự án (tệp CLAUDE.md) <i>Save what was just said to the project memory (CLAUDE.md file)</i>	<code># dự án này dùng 4 dấu cách để</code> <code>thụt lề</code> <code># this project uses 4 spaces for</code> <code>indentation</code>
/	Gọi lệnh dựng sẵn hoặc kỹ năng <i>Call a built-in command or a skill</i>	<code>/review</code>

Dấu @ có gợi ý hoàn tất: gõ `@src/` rồi nhấn Tab để chọn tệp. Với máy chủ MCP, @ còn dùng để gọi tài nguyên bên ngoài, ví dụ `@github:issue://123`.

The @ sign has autocompletion: type @src/ and press Tab to choose a file. With MCP servers, @ is also used to call external resources, for example @github:issue://123.

4.3. Đưa ảnh vào câu hỏi — Adding images to a prompt

- Kéo tệp ảnh thả vào cửa sổ Terminal.
 - *Drag an image file and drop it into the Terminal window.*
- Chụp màn hình (Cmd + Ctrl + Shift + 4 để chép vào clipboard) rồi nhấn Ctrl + V.
 - *Take a screenshot (Cmd + Ctrl + Shift + 4 to copy it to the clipboard), then press Ctrl + V.*
- Gõ đường dẫn ảnh sau dấu @.
 - *Type the image path after the @ sign.*

Cách này hợp khi cần Claude dựng giao diện theo bản thiết kế, hoặc đọc thông báo lỗi trong ảnh chụp màn hình.

This is useful when you need Claude to build an interface from a design, or to read an error message in a screenshot.

4.4. Chế độ vim — Vim mode

Gõ `/vim` để bật chế độ soạn thảo kiểu vim cho ô nhập: phím Esc chuyển sang chế độ lệnh, i để gõ, dd xóa dòng, w/b nhảy theo từ. Bật một lần, chương trình nhớ cho các phiên sau.

Type /vim to turn on vim-style editing for the input box: Esc switches to command mode, i to type, dd deletes a line, w/b jump by word. Turn it on once and the program remembers it for later sessions.

Chương 5. Danh mục lệnh gạch chéo — Chapter 5. Slash command reference

Gõ dấu `/` ở đầu dòng để mở danh sách lệnh; gõ tiếp vài chữ cái để lọc. Bảng dưới liệt kê các lệnh dựng sẵn theo nhóm công việc.

Type `/` at the start of the line to open the command list; type a few more letters to filter. The tables below list the built-in commands by task group.

5.1. Phiên làm việc và ngữ cảnh — Sessions and context

Lệnh Command	Tác dụng Effect
<code>/clear</code> [tên] <code>/clear</code> [name]	Bắt đầu cuộc trò chuyện mới, ngữ cảnh rỗng <i>Start a new conversation with an empty context</i>
<code>/compact</code> [chỉ dẫn] <code>/compact</code> [instructions]	Tóm tắt cuộc trò chuyện để giải phóng ngữ cảnh <i>Summarise the conversation to free up context</i>
<code>/context</code> [all]	Vẽ lưới màu cho thấy ngữ cảnh đang bị chiếm bởi thứ gì <i>Draw a coloured grid showing what is taking up the context</i>
<code>/autocompact</code> [auto số token] <code>/autocompact</code> [auto token count]	Đặt ngưỡng tự động nén <i>Set the automatic compaction threshold</i>
<code>/resume</code> [tên số] <code>/resume</code> [name number]	Quay lại một cuộc trò chuyện trước <i>Return to a previous conversation</i>
<code>/branch</code> [tên] <code>/branch</code> [name]	Rẽ nhánh cuộc trò chuyện để thử hướng khác, giữ nguyên bản gốc <i>Branch the conversation to try another direction, keeping the original</i>
<code>/rewind</code> [mốc N] <code>/rewind</code> [checkpoint N]	Lùi cả mã nguồn lẫn hội thoại về một mốc trước <i>Roll both the code and the conversation back to an earlier checkpoint</i>
<code>/export</code> [tên tệp] <code>/export</code> [file name]	Xuất cuộc trò chuyện ra tệp văn bản <i>Export the conversation to a text file</i>
<code>/copy</code> [N]	Chép câu trả lời gần nhất vào clipboard <i>Copy the most recent answer to the clipboard</i>
<code>/exit</code>	Thoát chương trình <i>Exit the program</i>

5.2. Thư mục, mô hình, chế độ — Folders, models, modes

Lệnh Command	Tác dụng Effect
<code>/cd</code> <đường dẫn> <code>/cd</code> <path>	Chuyển phiên sang thư mục làm việc khác <i>Move the session to another working folder</i>
<code>/add-dir</code> <đường dẫn> <code>/add-dir</code> <path>	Cho phép truy cập thêm một thư mục trong phiên này <i>Allow access to one more folder in this session</i>

<code>/model [tên]</code> <code>/model [name]</code>	Đổi mô hình và lưu làm mặc định <i>Change the model and save it as the default</i>
<code>/effort [mức]</code> <code>/effort [level]</code>	Đặt mức suy luận: low, medium, high, xhigh, max <i>Set the reasoning level: low, medium, high, xhigh, max</i>
<code>/fast [on off]</code>	Bật/tắt chế độ trả lời nhanh <i>Turn fast response mode on/off</i>
<code>/permissions</code>	Quản lý các quy tắc cho phép / hỏi / chặn <i>Manage allow / ask / deny rules</i>
<code>/status</code>	Xem mô hình, tài khoản, trạng thái phiên <i>View the model, account and session status</i>
<code>/usage</code>	Xem lượng token đã dùng trong phiên (<code>/cost</code> là tên gọi khác) <i>View the tokens used in the session (/cost is another name for it)</i>
<code>/config [khóa=giá trị]</code> <code>/config [key=value]</code>	Mở bảng thiết lập hoặc đặt thẳng một tùy chọn <i>Open the settings panel or set an option directly</i>
<code>/theme, /color</code>	Đổi bảng màu, đổi màu thanh nhập <i>Change the colour theme and the input bar colour</i>

5.3. Bộ nhớ, cấu hình, mở rộng — *Memory, configuration, extensions*

Lệnh <i>Command</i>	Tác dụng <i>Effect</i>
<code>/init</code>	Quét mã nguồn và tạo tệp CLAUDE.md cho dự án <i>Scan the source code and create a CLAUDE.md file for the project</i>
<code>/memory</code>	Mở các tệp CLAUDE.md, bật/tắt bộ nhớ tự động <i>Open the CLAUDE.md files, turn auto memory on/off</i>
<code>/hooks</code>	Xem các móc lệnh (hook) đang cấu hình <i>View the configured hooks</i>
<code>/plugin</code>	Quản lý gói mở rộng <i>Manage plugins</i>
<code>/mcp</code> <code>[reconnect enable disable]</code>	Quản lý kết nối máy chủ MCP <i>Manage MCP server connections</i>
<code>/keybindings</code>	Mở tệp phím tắt để sửa <i>Open the keybindings file for editing</i>
<code>/ide</code>	Quản lý kết nối với trình soạn thảo <i>Manage connections to editors</i>
<code>/agents, /list-agents</code>	Xem và tạo trợ lý con <i>View and create subagents</i>
<code>/tasks</code>	Xem việc đang chạy nền <i>View tasks running in the background</i>
<code>/doctor</code>	Kiểm tra và sửa lỗi cấu hình <i>Check and fix configuration errors</i>

5.4. Kỹ năng kèm sẵn — Bundled skills

Ngoài lệnh dựng sẵn, bản cài kèm sẵn một số kỹ năng (skill) cũng gọi bằng dấu gạch chéo:

Besides built-in commands, the installation includes some skills that are also called with a slash:

Lệnh Command	Tác dụng Effect
<code>/code-review [mức] [--fix]</code> <code>/code-review [level] [--fix]</code>	Soi phần mã vừa thay đổi hoặc một pull request tìm lỗi <i>Review recently changed code or a pull request for bugs</i>
<code>/security-review</code>	Soi phần thay đổi tìm lỗ hổng bảo mật <i>Review changes for security vulnerabilities</i>
<code>/simplify [--fix]</code>	Đề xuất rút gọn mã <i>Suggest code simplifications</i>
<code>/plan [mô tả]</code> <code>/plan [description]</code>	Vào chế độ lập kế hoạch cho thay đổi lớn <i>Enter plan mode for large changes</i>
<code>/verify</code>	Chạy các bước kiểm tra lại việc vừa làm <i>Run steps to re-check the work just done</i>
<code>/debug [mô tả]</code> <code>/debug [description]</code>	Bật ghi nhật ký gỡ lỗi và truy sự cố <i>Turn on debug logging and troubleshoot</i>
<code>/loop [chu kỳ] [câu lệnh]</code> <code>/loop [interval] [prompt]</code>	Chạy lặp một câu lệnh trong khi phiên còn mở <i>Run a prompt repeatedly while the session is open</i>
<code>/subtask [chỉ dẫn]</code> <code>/subtask [instructions]</code>	Giao một việc phụ cho trợ lý con <i>Hand a side task to a subagent</i>
<code>/insights</code>	Dựng báo cáo HTML phân tích các phiên gần đây <i>Build an HTML report analysing recent sessions</i>
<code>/install-github-app</code>	Cài ứng dụng GitHub cho kho mã <i>Install the GitHub app for the repository</i>

GHI CHÚ — Danh sách lệnh thay đổi theo phiên bản. Gõ `/help` trong phiên để xem danh sách đúng với bản đang cài, và `/release-notes` để xem thay đổi mới nhất.

NOTE — The command list changes between versions. Type `/help` in a session to see the list for the installed version, and `/release-notes` to see the latest changes.

Chương 6. Tham số dòng lệnh — Chapter 6. Command-line options

6.1. Các tham số dùng nhiều nhất — The most used options

Tham số <i>Option</i>	Tác dụng <i>Effect</i>
<code>-c, --continue</code>	Tiếp tục cuộc trò chuyện gần nhất trong thư mục hiện tại <i>Continue the most recent conversation in the current folder</i>
<code>-r, --resume [giá trị]</code> <code>-r, --resume [value]</code>	Mở lại phiên theo mã phiên, hoặc mở bảng chọn <i>Reopen a session by session ID, or open the picker</i>
<code>-p, --print</code>	Trả lời một lượt rồi thoát — dùng cho kịch bản, đường ống <i>Answer once and exit — for scripts and pipelines</i>
<code>--model <tên></code> <code>--model <name></code>	Chọn mô hình cho phiên (<code>opus</code> , <code>sonnet</code> , <code>haiku</code> , <code>fable</code> ...) <i>Choose the model for the session (opus, sonnet, haiku, fable...)</i>
<code>--effort <mức></code> <code>--effort <level></code>	Mức suy luận: <code>low</code> , <code>medium</code> , <code>high</code> , <code>xhigh</code> , <code>max</code> <i>Reasoning level: low, medium, high, xhigh, max</i>
<code>--permission-mode <chế độ></code> <code>--permission-mode <mode></code>	Chế độ quyền khi khởi động <i>Permission mode at startup</i>
<code>--add-dir <thư mục...></code> <code>--add-dir <folders...></code>	Cho phép truy cập thêm thư mục <i>Allow access to additional folders</i>
<code>-n, --name <tên></code> <code>-n, --name <name></code>	Đặt tên hiển thị cho phiên <i>Set a display name for the session</i>
<code>--agent <tên></code> <code>--agent <name></code>	Chạy phiên dưới dạng một trợ lý con đã định nghĩa <i>Run the session as a defined subagent</i>
<code>--settings <tệp JSON></code> <code>--settings <file JSON></code>	Nạp thêm thiết lập từ tệp hoặc chuỗi JSON <i>Load additional settings from a file or JSON string</i>
<code>-w, --worktree [tên]</code> <code>-w, --worktree [name]</code>	Tạo worktree Git riêng cho phiên <i>Create a separate Git worktree for the session</i>
<code>--safe-mode</code>	Tắt mọi tùy biến (<code>CLAUDE.md</code> , <code>skill</code> , <code>hook</code> , <code>MCP</code> ...) để dò lỗi cấu hình <i>Turn off all customisations (CLAUDE.md, skills, hooks, MCP...) to track down configuration errors</i>
<code>-d, --debug [bộ lọc]</code> <code>-d, --debug [filter]</code>	Bật nhật ký gỡ lỗi, lọc theo nhóm ví dụ <code>api</code> , <code>hooks</code> <i>Turn on debug logging, filtered by category, e.g. api, hooks</i>
<code>-v, --version</code>	In số phiên bản <i>Print the version number</i>

6.2. Chạy không cần ngồi trước màn hình — Running without sitting at the screen

Chế độ `-p` biến Claude Code thành một lệnh shell bình thường: nhận đầu vào, in kết quả, thoát. Đây là cách đưa Claude vào kịch bản tự động.

The `-p` mode turns Claude Code into an ordinary shell command: it takes input, prints the result and exits. This is how Claude is brought into automated scripts.

```
# Hỏi một câu, lấy câu trả lời dạng văn bản
# Ask one question, get the answer as text
claude -p "liệt kê các hàm chưa có kiểm thử trong thư mục src"
claude -p "list the functions without tests in the src folder"

# Đưa dữ liệu qua đường ống
# Pass data through a pipe
git diff | claude -p "viết mô tả commit cho phần thay đổi này"
git diff | claude -p "write a commit message for these changes"

# Lấy kết quả dạng JSON để xử lý tiếp
# Get JSON output for further processing
claude -p --output-format json "đếm số tệp Python" | jq -r .result
claude -p --output-format json "count the Python files" | jq -r .result

# Giới hạn tiền chi cho một lượt chạy
# Limit the money spent on one run
claude -p --max-budget-usd 2 "rà soát toàn bộ kho mã"
claude -p --max-budget-usd 2 "review the entire repository"
```

CẢNH BÁO — Ở chế độ `-p`, hộp thoại hỏi có tin thư mục hay không bị bỏ qua. Chỉ chạy trong thư mục bạn biết rõ nguồn gốc.

WARNING — In `-p` mode, the dialog asking whether you trust the folder is skipped. Only run it in folders whose origin you know well.

6.3. Lệnh con — Subcommands

Lệnh Command	Tác dụng Effect
<code>claude doctor</code>	Kiểm tra sức khỏe bản cài <i>Check the health of the installation</i>
<code>claude update</code>	Kiểm tra và cài bản mới <i>Check for and install a new version</i>
<code>claude install [bản]</code> <code>claude install [version]</code>	Cài bản gốc — <code>stable</code> , <code>latest</code> hoặc số phiên bản cụ thể <i>Install the native version — stable, latest or a specific version number</i>
<code>claude migrate-installer</code>	Chuyển từ bản npm sang bản gốc <i>Switch from the npm version to the native version</i>
<code>claude setup-token</code>	Tạo khoá dùng lâu dài cho máy chủ, CI <i>Create a long-lived key for servers, CI</i>
<code>claude mcp ...</code>	Thêm, xoá, liệt kê máy chủ MCP <i>Add, remove, list MCP servers</i>
<code>claude plugin ...</code>	Quản lý gói mở rộng <i>Manage plugins</i>
<code>claude agents</code>	Liệt kê các phiên đang chạy nền <i>List sessions running in the background</i>
<code>claude attach <mã></code> <code>claude attach <id></code>	Mở lại một phiên nền trong cửa sổ hiện tại <i>Reopen a background session in the current window</i>
<code>claude logs <mã></code> <code>claude logs <id></code>	In phần đầu ra gần đây của một phiên nền <i>Print the recent output of a background session</i>

<code>claude stop <mã></code> <code>claude stop <id></code>	Dừng một phiên nền, giữ nguyên nội dung trò chuyện <i>Stop a background session, keeping the conversation</i>
<code>claude import [nguồn]</code> <code>claude import [source]</code>	Nhập cấu hình từ công cụ lập trình AI khác <i>Import configuration from other AI coding tools</i>