

Claude Code Agent Teams 設定與使用指南

一、什麼是 Agent Teams ?

Agent Teams 讓多個 Claude Code 實例一起工作：

- **Team Lead (主控)**：你對話的那一個，負責建立團隊、分配任務、彙整結果。
- **Teammates (隊友)**：各自獨立的 Claude Code 實例，有各自的 context、可彼此傳訊、從共享任務清單領任務。
- **Subagents** 是在「同一個 session」裡派出去的小幫手，只回報結果；**Agent Teams** 是「多個完整 session」並行協作，隊友之間可以互相溝通。

適合：研究/審查、多模組並行開發、多假設除錯、前後端+測試分頭改。不適合：單純線性任務或同一檔案多人改（容易衝突）。

二、啟用 Agent Teams (設定)

此功能為實驗功能，預設關閉，需手動啟用。

方法 1：環境變數

在啟動 Claude Code 的 shell 裡設定 (PowerShell 範例)：

```
$env:CLAUDE_CODE_EXPERIMENTAL_AGENT_TEAMS = "1"  
claude
```

方法 2：settings.json (建議)

在使用者或專案的 `settings.json` 裡加上 `env`：

- **使用者**：`%USERPROFILE%\.claude\settings.json` (Windows)
- **專案**：專案目錄下的 `.claude/settings.json`

內容範例：

```
{  
  "env": {  
    "CLAUDE_CODE_EXPERIMENTAL_AGENT_TEAMS": "1"  
  }  
}
```

若檔案已存在，只要在頂層加 `"env": { "CLAUDE_CODE_EXPERIMENTAL_AGENT_TEAMS": "1" }` 即可，不必刪掉其他設定。

三、顯示模式 (teammateMode) — Windows 注意

Agent Teams 有兩種顯示方式：

模式	說明	Windows 支援
in-process	所有隊友都在同一個終端裡，用 Shift+↑/↓ 切換隊友、輸入即傳給該隊友	✓ 支援，且是 Windows 上實際可用的方式
split panes (分窗)	每個隊友一個窗格，需 tmux 或 iTerm2	✗ 官方寫明不支援 VS Code 內建終端、Windows Terminal、Ghostty

因此在你目前的環境 (Windows) 下，會以 **in-process** 為主。可明確設成 in-process，避免自動選到不支援的 split：

在 `settings.json` 中：

```
{
  "env": {
    "CLAUDE_CODE_EXPERIMENTAL_AGENT_TEAMS": "1"
  },
  "teammateMode": "in-process"
}
```

或單次執行：

```
claude --teammate-mode in-process
```

四、第一次使用：建立團隊

啟用後，用自然語言請 Claude 建立 agent team 並說明任務與角色即可。例如：

我正在設計一個幫開發者追蹤程式碼裡 TODO 註解的 CLI 工具。

請建立一個 agent team，從不同角度探索：

一個隊友負責 UX、一個負責技術架構、一個扮演唱反調的人。

或：

建立一個 4 人團隊，並行重構這幾個模組，每個隊友都用 Sonnet。

Claude (作為 lead) 會：

1. 建立團隊與共享任務清單
2. 為每個角色/人數 spawn 隊友
3. 分配任務、彙整結果
4. 結束時可請 lead 做 cleanup

五、操作與控制

1. 和隊友互動 (in-process 模式)

- Shift+↑ / Shift+↓ : 切換目前選中的隊友 (要傳話給誰) 。
- 選中後直接輸入 : 訊息會送給該隊友。
- Enter : 進入該隊友的 session 檢視 ; Escape : 中斷該隊友目前這輪。
- Ctrl+T : 切換任務清單顯示。

2. 用自然語言指揮 Lead

例如：

- 「請 researcher 那個隊友關閉」 → lead 會送關閉請求給該隊友
- 「等所有隊友完成任務後再繼續」
- 「清理團隊」 → 移除團隊資源 (需先讓隊友都關閉)

3. Delegate 模式 (只協調、不寫碼)

若希望 lead 只做分配與協調、不要自己動手寫程式：

- 先建立好團隊後，按 Shift+Tab 切換到 delegate mode 。
- 此模式下 lead 只能用協調相關工具 (spawn、傳訊、關閉隊友、管理任務)，不會直接改程式碼。

4. 任務與計畫審核

- 共享任務清單：lead 建立任務，隊友可「認領」或由 lead 指定；任務可有依賴 (pending / in progress / completed) 。
- 計畫審核：可要求某隊友「先提出計畫、等 lead 核准再實作」：
 - 例：「spawn 一個架構師隊友來重構 auth 模組，在改任何東西前要先經過計畫審核。」
 - 隊友會先以唯讀方式規劃，送審後由 lead 批准或退回並給回饋。

六、關閉與清理

1. 關閉單一隊友：對 lead 說「請某某隊友關閉」，lead 會送關閉請求，隊友可同意或說明理由拒絕。
2. 清理團隊：對 lead 說「清理團隊」；若還有隊友在跑，cleanup 會失敗，需先關閉所有隊友再清理。
3. 重要：只用 lead 做 cleanup，不要讓隊友執行 cleanup，以免團隊狀態不一致。

七、Hooks 與品質控管 (選用)

可用 Hooks 在「隊友閒置」或「任務完成」時做檢查：

- TeammateIdle：隊友即將閒置時執行；若 exit code 為 2，可回傳意見並讓隊友繼續工作。
- TaskCompleted：任務要被標成完成時執行；exit code 2 可阻止完成並回傳意見。

設定方式見官方 [Hooks](#) 文件。

八、目前限制與注意事項

- 實驗功能：行為與介面可能變動。
- Session 恢復：/resume 、 /rewind 不會還原 in-process 隊友；恢復後若 lead 還對舊隊友傳訊會失敗，需請 lead 重新 spawn 。
- 任務狀態：有時隊友沒正確把任務標成完成，會卡住依賴；可手動更新狀態或請 lead 提醒該隊友。

- **一 session 一團隊**：同一個 lead 一次只能帶一個團隊，要開新團隊前先 cleanup 現有團隊。
 - **不能巢狀**：隊友不能再開團隊或 spawn 子隊友，只有 lead 能管理團隊。
 - **Token 用量**：每個隊友都是獨立 session，整體 token 會明顯高於單一 session，適合「值得並行」的任務。
-

九、快速檢查清單 (Windows)

1. 在 `%USERPROFILE%\.claude\settings.json` 或專案 `.claude/settings.json` 加上：
 - `"env": { "CLAUDE_CODE_EXPERIMENTAL_AGENT_TEAMS": "1" }`
 - (建議) `"teammateMode": "in-process"`
 2. 重新啟動 Claude Code (若用環境變數則在設好變數的 shell 裡啟動)。
 3. 在 REPL 裡用一句話描述任務與角色，請 Claude 建立 agent team。
 4. 用 **Shift+↑/↓** 切換隊友、**Ctrl+T** 看任務清單，並用自然語言對 lead 下指令。
-

參考連結

- 官方文件：[Orchestrate teams of Claude Code sessions](#)
- 設定參考：[Claude Code settings](#)
- Subagents 比較：[Create custom subagents](#)