玩轉 AI Agent

醫師實戰工作坊｜從 Agent 觀念到 SCI 論文與簡報

3 小時｜先學通用 Agent 觀念，再於 Codex 桌面版（Windows）實作

主線： AI Agent → Git flow ＋ GitHub CLI 版控 → MedSci Skills 寫論文 → 論文轉簡報

教法：環境建置照 SOP 走；學會 Agent 後改為 Agent First，用提示詞讓 Agent 動手

紅線：不貼病患可識別資料（PHI）｜引用必核實｜人負最終責任

講師：江弘竣

【導論】

今日路線圖：四段式，一條主線串到底

0–180 分

① 學習 AI Agent

0–60 分
通用觀念（跨產品）
→ Codex 桌面版 SOP 安裝

② Git flow × gh 版控

60–100 分
GitHub 帳號／gh 安裝／gh auth login
→ Agent 代跑 git 與 gh

③ MedSci Skills 寫 SCI

100–160 分
安裝技能包
→ 文獻、起草、驗證

④ 由論文做簡報

160–180 分
論文 → 簡報（衍生產出）
→ 檢查與收斂

教學節奏： SOP → Agent First

前段（安裝、帳號、授權）一步一步照做；學會 Agent 後，改由你「說目標、給邊界、要回報」，讓 Agent 執行 git／gh 、技能與檔案操作

同一個專案資料夾貫穿全場：論文草稿、版本紀錄、簡報都在裡面

【導論】

學習目標：下課帶走能用的環境與工作流

0–180 分

下課後你能夠

- 用自己的話說明 Agent 的元件與「計畫→執行→驗證」迴圈
- 在 Codex 桌面版開專案、設權限、審查變更
- 用提示詞讓 Agent 跑 git／gh：commit、分支、PR
- 叫用 MedSci Skills 做文獻、起草、驗證引用
- 把論文草稿轉成有講者備註的簡報

你會帶走

- 已登入的 Codex 桌面版＋已授權的 gh
- GitHub 私有 repo：論文草稿＋簡報＋版本歷史
- 已安裝的 MedSci Skills（固定版本）
- 提示詞模板包（git／gh、論文、簡報）
- 安全紅線與投稿前檢查清單

【導論】

投影片標籤與課前環境檢查

開場 5 分

【觀念】 跨產品通用原則與心智模型

【實作】 SOP：一步一步照做（安裝、帳號、授權）

【Agent First】 貼上提示詞，讓 Agent 動手；你審查

【安全】 PHI、引用、權限紅線

【檢查點】 驗收與帶走清單

課前檢查（Windows）

- ☐ Windows 11 建議；Windows 10 需 1809 以上且已更新
- ☐ 可用 winget（Windows 套件管理員）
- ☐ 可安裝軟體的權限（機關電腦可能需資訊室核准）
- ☐ ChatGPT 帳號（Codex 可用方案）
- ☐ 可收信的 Email（申請 GitHub 用）
- ☐ 練習資料：只用公開或合成資料

第一部分

學習 AI Agent：通用觀念

0–25 分 觀念（跨產品）｜25–60 分 收斂到 Codex 桌面版實作

- Agent 是什麼？由哪些元件組成？
- 計畫→執行→驗證迴圈；上下文、權限、工具、技能
- 不同 Agent 產品的共通點：學會觀念，換工具也通用
- 最後才落地：在 Windows 安裝並設定 Codex 桌面版

【觀念】

對話式 AI 會「回答」，Agent 會「動手做」

0–25 分

對話式 AI （Chatbot）

- 一問一答，產出文字建議
- 你複製貼上、自己去執行
- 看不到你的檔案與環境
- 錯了要你自己發現

AI Agent

- 接收目標，自己拆解步驟
- 使用工具：讀寫檔案、跑指令、上網查
- 在你授權的範圍內行動
- 回報結果與變更，供你審查

【觀念】

一個 Agent ＝ 模型＋工具＋上下文＋權限＋技能

0–25 分

模型	工具	上下文	權限	技能
醫療比喻：大腦	醫療比喻：雙手	醫療比喻：病歷	醫療比喻：執業範圍	醫療比喻：標準作業流程
理解目標、推理、產生計畫與文字	讀寫檔案、終端機指令、瀏覽器、外部服務	你給的檔案、說明、對話紀錄；沒給的它不知道	能動哪些資料夾、能否上網、何時要你核准	可重複使用的指令＋腳本＋參考資料包

- 換產品時，先問這五個問題：用什麼模型？有哪些工具？我怎麼給上下文？權限怎麼設？能不能裝技能？
- 今天的實作會在 Codex 桌面版逐一對應這五個元件

【觀念】

Agent 的工作迴圈：計畫 → 執行 → 驗證 → 回報

0–25 分



你的三個把關點

計畫後：方向對嗎？（不對就在這裡喊停，最省成本）

執行中：要求核准的動作（刪檔、上網、推送到 GitHub）逐一確認

回報後：看變更內容（diff）與來源，再決定接受或要求重做

【觀念】

上下文決定品質，權限決定風險

0–25 分

上下文（它知道什麼）

- 只知道你給的：檔案、說明、對話
- 給檔案路徑，勝過憑記憶描述
- 專案說明檔：寫下規則，每次自動帶入
  - 例： Codex 讀 AGENTS.md ； Claude Code 讀 CLAUDE.md
- 對話太長會遺忘：大任務拆成多段

權限（它能做什麼）

- 沙盒：限制在專案資料夾內動作
- 核准模式：敏感動作先問你
- 完全存取：可動整台電腦，風險最高
- 最小權限原則：夠用就好
- 版本控管是安全網：改壞可以回復

【觀念】

工具、技能、連接器： Agent 能力的三種擴充

0–25 分

類型	是什麼	醫師比喻	今天會用到
工具	內建能力：讀寫檔、終端機、瀏覽器	聽診器、開單系統	改檔、跑 git ／ gh
技能（ Skills ）	一個資料夾： SKILL.md 說明＋腳本＋參考資料	科內標準作業流程	MedSci Skills
連接器（ MCP 等）	連到外部服務或資料庫的標準介面	院際轉診窗口	（本課不設定）

- Agent Skills 是開放標準：同一個技能包可給多個 Agent 產品使用
- 技能採「漸進載入」：平時只讀名稱與描述，需要時才讀完整內容，節省上下文
- 裝技能＝把資料夾放到該 Agent 會讀取的位置

【觀念】

Agent 產品很多，觀念相通

0–25 分

型態	代表產品（舉例）	適合誰
桌面 App	Codex 桌面版、 Claude Code 桌面版	不想碰程式編輯器的使用者
程式編輯器內建	Cursor 、 GitHub Copilot （ VS Code ）	已在編輯器裡寫東西的人
終端機（ CLI ）	Codex CLI 、 Claude Code	習慣命令列的人

- 以上四個平台都被 MedSci Skills 列為已驗證可讀取 Agent Skills 的主機
- 選工具看三件事：作業系統支援、權限／沙盒設計、你的機構政策
- 功能、方案與價格變動快：以各家官方文件當日內容為準
- 今天實作選 Codex 桌面版： Windows 原生、內建 Git 功能與技能支援

【觀念】

委派心智模型：你是主治，Agent 是住院醫師

0–25 分

好醫囑的五要素

- 目標：要產出什麼（檔案、格式、語言）
- 素材：給哪些檔案；沒給的不得臆造
- 限制：紅線（不捏造、不處理 PHI）
- 步驟：先計畫、等我確認、再動手
- 驗收：怎樣算完成、要回報什麼

常見壞醫囑

- 「幫我寫完一篇論文」
- 沒給素材，卻要求「有引用」
- 沒說紅線，事後才發現亂補數據
- 不看變更就全部接受
- 一次塞十件事，無法中途喊停

【安全】

安全紅線：違反任何一條，寧可不用 AI

0–25 分

- 隱私：不貼、不上傳可識別病患資料（姓名、病歷號、影像、可還原組合）
- 練習：今天只用公開資料或合成資料
- 引用：每一筆都要能回到原文；禁止捏造文獻、 DOI 、 PMID
- 數據：沒有的結果就寫「尚未取得」，不讓 AI 補故事
- 權限：只在自己的專案資料夾工作；帳號密碼與權杖（ token ）不貼進對話
- 責任：作者與講者負最終責任；依期刊與單位政策揭露 AI 使用

【觀念】

觀念落地：今天的實作環境統一用 Codex 桌面版

25–60 分

為什麼選它（依官方文件）

• Windows 原生執行（PowerShell + Windows 沙盒）

- 內建 Git 功能，可審查、復原變更
- 支援技能（Skills）與外掛
- 核准模式：敏感動作先問你

五元件對應

- 模型：由 OpenAI 帳號方案提供
- 工具：檔案編輯、整合終端機
- 上下文：開啟的專案資料夾＋AGENTS.md
- 權限：「需核准」模式（Ask for approval）
- 技能：~/.agents/skills 下的技能包

【實作】

CP1 ｜安裝 Codex 桌面版（Windows）

約 10 分

1. 開啟 Microsoft Store，搜尋並安裝官方 App
   官方文件目前名稱為 ChatGPT 桌面版（內含 Codex），發行者須為 OpenAI
2. 或在 PowerShell 用指令安裝
   winget install --id 9PLM9XGG6VKS -s msstore
3. 開啟 App，用你的 ChatGPT 帳號登入
   機關電腦若被擋：改用助教備援機，勿自行關閉防護
4. 在 App 內切換到 Codex，看到「新對話」入口
   介面文字以當日版本為準

✓ 驗收：已登入、可開新的 Codex 對話；尚未開啟任何含真實病患資料的資料夾

講師待確認｜App 名稱、Store 頁面與 winget ID 請於課前以官方 Windows 頁面再確認一次

【實作】

CP2 ｜建立專案資料夾並設定權限

約 8 分

1. 在「文件」下建立資料夾： ai-agent-workshop
   完整路徑： C:\Users\ 你的帳號 \Documents\ai-agent-workshop
2. 把講義包 take-home 解壓縮到此資料夾
   內含 SAFETY.md 、提示詞模板、論文與簡報範本
3. 在 Codex 選擇「開啟資料夾／專案」並選它
   之後所有工作都在這個資料夾
4. 輸入框下方權限設為「需核准」（ Ask for approval ）
   不要使用完全存取模式

✓ 驗收： Codex 顯示專案資料夾；權限為需核准模式

【Agent First】

CP3 ｜讓 Agent 寫專案規則（AGENTS.md）

約 10 分

你只要做

- 貼上右側提示詞
- 先看它的計畫，回覆「確認」
- 看變更內容（diff）：綠色新增、紅色刪除
- 符合才接受；不符就說哪裡要改

可複製提示詞（貼到 Codex）

請閱讀本資料夾的 SAFETY.md 與 README.md。

任務：建立 AGENTS.md，作為本專案給 Agent 的固定規則。

內容：繁體中文；不處理可識別病患資料；不捏造引用或數據，不確定標【待核實】；任何刪檔、上網、推送到 GitHub 前先問我。

步驟：先列計畫等我確認，再建檔，最後用三行說明你做了什麼。

✓ 驗收：資料夾出現 AGENTS.md；你親自看過變更並接受

【檢查點】

從這裡開始：說目標，讓 Agent 動手

55–60 分

說目標 → 給邊界 → 要計畫 → 審變更 → 要回報

之後你不必親手學的

- git 指令細節
- gh 參數組合
  - 技能資料夾放哪裡
  - 檔案格式轉換

第一部分檢查點

- ☐ 能說出 Agent 五元件
- ☐ Codex 已登入、已開專案資料夾
- ☐ 權限為需核准模式
- ☐ AGENTS.md 已建立並審查過

第二部分

Git flow × GitHub CLI：讓 Agent 幫你版控

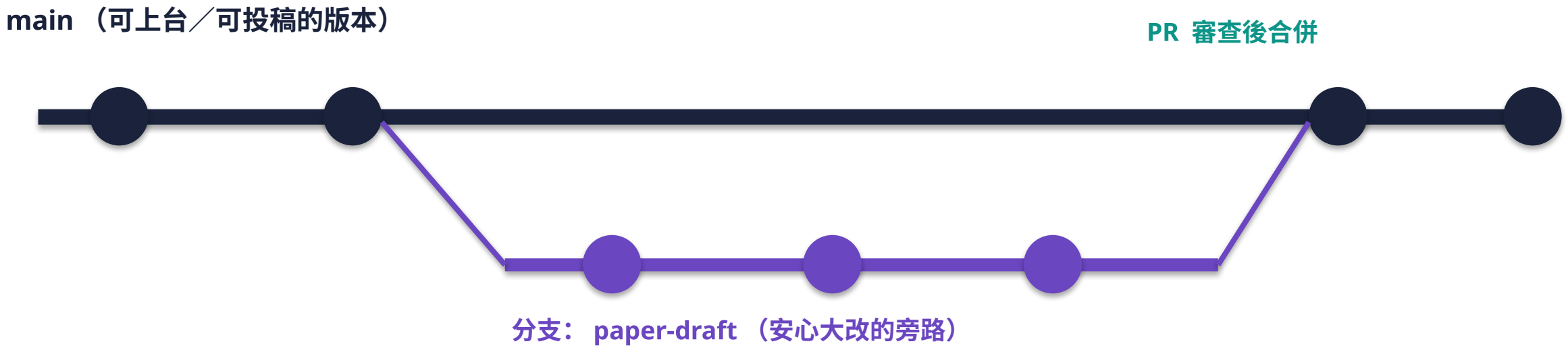
60–100 分｜先 SOP 申請與授權，再 Agent First

- Git flow 淺白版： main 、分支、 commit 、 PR
- SOP： 申請 GitHub 帳號 → 安裝 Git 與 gh → gh auth login
- Agent First： 讓 Agent 建私有 repo 、 commit 、開 PR
- 取代舊版的 GitHub Desktop 路線

【觀念】

Git flow 淺白版：像病歷一樣留下每次修改

60–65 分



詞	白話	醫療比喻
repo	這個專案的家（含所有版本）	一本病歷
commit	一次有意義的存檔＋一句說明	一筆病程紀錄
branch	從主線分出的修改空間	會診前的草稿
PR	請人看過再合回主線	主治簽核

【實作】

CP4 ｜申請 GitHub 帳號（全班一起）

約 8 分

1. 瀏覽器開啟 github.com，點「Sign up」
   使用可長期收信的 Email（建議個人或學術信箱）
2. 設定使用者名稱與強密碼
   名稱會公開：用專業名稱，不含病患或醫院內部資訊
3. 完成 Email 驗證碼
   收信匣找不到請查垃圾郵件
4. 到 Settings → Password and authentication 啟用兩步驟驗證
   建議用驗證器 App；備援碼離線保存

✓ 驗收：能登入 GitHub，右上角看到自己的頭像

【實作】

CP5 ｜安裝 Git 與 GitHub CLI （gh）

約 7 分

1. 在 Codex 整合終端機或 PowerShell 貼上：安裝 Git
   winget install --id Git.Git -e --source winget
2. 安裝 GitHub CLI
   winget install --id GitHub.cli --source winget
3. 關閉後重新開啟終端機視窗（新分頁不夠）
   安裝程式會修改 PATH，需新視窗才生效
4. 確認版本： git --version 與 gh --version
   gh 需 2.90 以上才有安裝技能用的 gh skill 指令

✓ 驗收：兩個指令都印出版本號； gh 版本 ≥ 2.90

【實作】

CP6 ｜授權本機 gh： gh auth login

約 8 分

1. 在終端機輸入 gh auth login
   以下依序選擇，方向鍵移動、 Enter 確認
2. 要登入哪裡？選 GitHub.com；通訊協定選 HTTPS
   被問是否用 GitHub 認證 Git： 選 Yes
3. 驗證方式選「用瀏覽器登入」，複製畫面上的一次性代碼
   瀏覽器開啟後貼上代碼並按授權
4. 回終端機輸入 gh auth status 確認
   看到已登入的帳號即完成； Agent 之後可代你使用 gh

✓ 驗收： gh auth status 顯示已登入你的帳號

講師待確認｜ gh 提示文字可能隨版本調整；課前用當日版本走一次並截圖

【Agent First】

讓 Agent 建私有 repo 並完成第一次 commit

約 8 分

你只要做

- 貼上提示詞
- 每個 git ／ gh 指令出現核准時，看懂再按允許
- 最後到 GitHub 網頁確認 repo 是私有

可複製提示詞（貼到 Codex）

請在本專案資料夾用 git 與 gh 完成版控設定：

1. 確認 gh auth status 已登入；若 git 尚未設定姓名與 Email，先問我。
2. git init，建立 .gitignore （排除 data/raw 、 *.xlsx 原始資料與暫存檔）。
3. 第一次 commit，訊息用繁中：「初始化：專案規則與講義」。
4. 用 gh 建立「私有」 repo： ai-agent-workshop，並推送。

每一步先說明要執行的指令，等我核准。完成後給我 repo 網址。

✓ 驗收： GitHub 上有私有 repo，內含 1 筆 commit

【Agent First】

讓 Agent 跑 Git flow：分支→ PR→ 合併

約 8 分

你只要做

- 貼上提示詞
- 到 GitHub 網頁看 PR 的變更（Files changed）
- 你自己按下合併，或請 Agent 合併

可複製提示詞（貼到 Codex）

請幫我練習一次 Git flow：

1. 從 main 建立分支 paper-draft。
2. 在 paper/ 建立 manuscript.md，只放 IMRAD 四個章節標題（繁中說明）。
3. commit，訊息：「新增論文骨架」，推送分支。
4. 用 gh pr create 開 PR 到 main，標題與說明用繁中，列出變更檔案。
5. 回報 PR 網址，等我審查後再合併。

✓ 驗收：PR 已建立並合併；main 有「新增論文骨架」這筆紀錄

【安全】

Git 安全守則與第二部分檢查點

95–100 分

【安全】版控守則

- repo 一律先設私有；公開前再三檢查
  - 原始資料、病患資料不進 repo（.gitignore）
  - 一次性代碼、 token 、密碼不貼進對話
  - 推送、合併、刪分支： Agent 須先問你
- commit 訊息寫人話：未來的你看得懂

【檢查點】你應該有

- ☐ GitHub 帳號＋兩步驟驗證
- ☐ git 、 gh 已安裝； gh auth status 正常
- ☐ 私有 repo： ai-agent-workshop
- ☐ 一個已合併的 PR （論文骨架）
- ☐ 會用提示詞讓 Agent 跑 git ／ gh

第三部分

MedSci Skills × Agent： 協作 SCI 期刊論文

100–160 分｜認識技能包 → 安裝 → 文獻 → 起草 → 驗證

- MedSci Skills 是什麼、有哪些技能
- Agent First： 讓 Codex 安裝並確認技能
- 以技能鏈協作：文獻檢索、論文起草、報告規範與引用驗證
- 每一步都 commit： 論文的版本歷史就是你的研究紀錄

【觀念】

MedSci Skills： 給醫學研究用的開源技能包

100–105 分

已查證的事實

• GitHub： Aperivue/medsci-skills （MIT 授權）

• 作者： Yoojin Nam, MD （首爾峨山醫學中心放射科）

• 目前 54 個技能；最新版 v6.1.0

• 可用於 Claude Code 、 Codex 、 Cursor 、 GitHub Copilot

• 涵蓋文獻、研究設計、統計、圖表、寫作、報告規範、投稿

它自己說明的限制

• 產出需由合格研究者逐一審查

• 不是診斷工具，也不是自動作者

• 檢查器「沒找到問題」≠ 論文正確

• 不要提供可識別病患資料

• 部分功能在非 Claude 平台會退化為手動步驟

【觀念】

技能地圖：依研究階段分組（節選）

105–110 分

研究階段	代表技能（節選）	做什麼
專案入口	orchestrate	描述需求，分派到合適技能
文獻與引用	search-lit 、 verify-refs	API 查證的文獻檢索；稽核引用真偽
資料與設計	design-study 、 deidentify	研究設計檢查；本機去識別化
分析與圖表	analyze-stats 、 make-figures	統計分析碼與出版級圖表
寫作	write-paper 、 polish-language	IMRAD 八階段起草；語言潤飾
審查與規範	check-reporting 、 self-review	報告規範逐項稽核；投稿前自評
投稿與簡報	find-journal 、 present-paper	選期刊；由論文製作簡報

完整 54 個技能與分組見 GitHub README；不確定用哪個，先叫用 orchestrate

【Agent First】

CP7 ｜讓 Agent 安裝 MedSci Skills

110–120 分

你只要做

- 貼上提示詞，核准安裝指令
- 安裝完：完全關閉並重開 Codex
- 開新對話，輸入驗證句

可複製提示詞（貼到 Codex）

請幫我安裝 MedSci Skills 給 Codex 使用（使用者層級）：

1. 確認 gh --version ≥ 2.90。
2. 執行： gh skill install Aperivue/medsci-skills --all --pin v6.1.0 --agent codex --scope user
3. 若 gh skill 失敗，改用（需 Node 18+、Python 3.9+）： npx medsci-skills@6.1.0 install
4. 告訴我技能裝在哪個資料夾。不要改動本專案以外的其他檔案。

重開後驗證：「請確認 MedSci Skills 已安裝，列出 5 個適合今天實作的技能。」

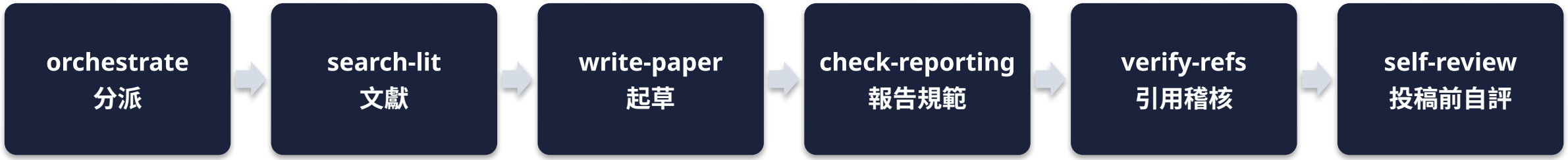
✓ 驗收： Codex 能列出 MedSci 技能（如 search-lit、write-paper）

講師待確認｜課堂統一用哪個安裝管道與版本請事先決定；無終端機權限者可改用官方 Windows 教室安裝包

【觀念】

SCI 論文技能鏈： Agent 起草，技能把關，你決定

120–123 分



在 Codex 叫用技能

用「$ 技能名稱」明確叫用（例： $search-lit ），或直接描述需求讓 Agent 自行選用

- 有自己去識別化的資料時，可在起草前加入 analyze-stats → make-figures
- 今天時間有限：完成「研究問題 → 文獻 → 一段草稿 → 驗證」的最小迴圈
- 每完成一步就請 Agent commit ，保留可追溯的研究紀錄

【Agent First】

研究問題 → 文獻檢索（search-lit）

123–135 分

你只要做

- 把練習研究問題寫進 paper/research-question.md
- 貼上提示詞
- 抽查 3 篇：到 PubMed 打開原文確認

可複製提示詞（貼到 Codex）

請先在 paper-draft 分支工作。

閱讀 paper/research-question.md，使用 $search-lit 技能：

1. 先列出檢索策略（關鍵字、同義詞、資料庫）給我確認。
2. 確認後檢索，只收錄經 API 查證的文獻，輸出 paper/refs.bib 與 paper/lit-table.md（作者、年份、設計、主要發現、PMID／DOI）。
3. 不得憑記憶補文獻；查不到就說查不到。
4. commit：「文獻檢索初版」。

✓ 驗收：lit-table.md 與 refs.bib 存在；你抽查的 3 篇都真實存在

講師待確認｜課堂練習研究問題與資料集請事先指定（建議用公開資料，例如 MedSci 範例所用的公開資料集）

【Agent First】

論文起草（write-paper）：先大綱，再一段

135–148 分

你只要做

- 貼上提示詞
- 先審大綱：每個主張都有來源嗎?
- 只核准寫一段（建議方法或前言）

可複製提示詞（貼到 Codex）

請使用 $write-paper 技能，以 paper/lit-table.md、refs.bib 與 research-question.md 為唯一素材：

1. 先產出 IMRAD 大綱到 paper/manuscript.md，每條主張標註來源；無來源者標【待核實】。
2. 等我確認大綱後，只撰寫「前言」一段（或我指定的段落）。
3. 不得產生任何結果數字；沒有資料就寫「資料尚未取得」。
4. commit：「論文大綱與前言初稿」。

✓ 驗收：manuscript.md 有大綱＋一段草稿；無來源句已標【待核實】

【Agent First】

驗證：報告規範＋引用稽核＋開 PR

148–158 分

你只要做

- 貼上提示詞
- 讀稽核報告： MISMATCH ／ UNVERIFIED ／ FABRICATED 優先處理
- 在 PR 中審查後合併

可複製提示詞（貼到 Codex）

請對 paper/manuscript.md 執行驗證：

1. $verify-refs： 稽核所有引用，輸出 qc/reference_audit.json ，並用繁中摘要有問題的條目。
2. $check-reporting： 依研究設計選擇合適的報告規範（先告訴我你選哪一個），列出 MISSING ／ PARTIAL 項目。
3. 不要自動修改引用；把待辦整理成 paper/todo.md 。
4. commit 並用 gh 開 PR 到 main： 「論文初稿＋驗證報告」。

✓ 驗收：有引用稽核報告與待辦清單； PR 已開啟

【安全】

論文誠信：技能是第二道防線，你是第一道

158–160 分

- 引用：稽核「 OK 」只代表文獻存在且資料吻合，不代表支持你的論點——仍要讀原文
- 數據：結果只來自你真實擁有、已去識別的資料； Agent 不得「補齊」
- 病患資料：需處理真實資料時，先用本機去識別化（ deidentify 技能在本機執行、不連網）並經 IRB 規範
- 揭露：依目標期刊與單位政策，於方法或致謝說明 AI 與工具使用（可註明 MedSci Skills 版本）
- 署名： AI 不是作者；所有作者對全文負責

第四部分

由論文做簡報：讓 Agent 產出衍生成果

160–180 分｜論文 → 簡報 → 檢查 → 收斂

- 簡報是論文的衍生產出：內容只能來自論文與已核實來源
- 用 present-paper 技能產生有講者備註的簡報
- 在分支上產出、用 PR 合併：論文與簡報版本一致

【Agent First】

讓 Agent 從論文產生簡報（present-paper）

160–172 分

你只要做

- 先寫下：聽眾、時長、三個核心訊息
- 貼上提示詞
- 審每頁講稿：有無論文裡沒有的主張?

可複製提示詞（貼到 Codex）

請建立分支 slides，使用 $present-paper 技能，以 paper/manuscript.md 為唯一來源：

聽眾：本科住院醫師；時長： 10 分鐘；核心訊息：（填三點）。

1. 先給我逐頁腳本（每頁一句重點＋講者備註）等我確認。
2. 確認後產出 slides/talk.pptx，繁體中文，字型 Microsoft JhengHei。
3. 論文中【待核實】的內容不得寫成肯定句；不新增引用。
4. commit 並開 PR：「由論文產生簡報初版」。

✓ 驗收： slides/talk.pptx 已產生；每頁主張都能對回論文

【檢查點】

簡報檢查與今日帶走清單

172–176 分

【檢查點】簡報上台前

- 每頁主張都能對回論文段落與來源
- 沒有論文裡不存在的數字或結論
- 在 PowerPoint 開啟檢查字型與排版
- 分享給他人時用「去除講者備註」版本
- PR 合併後， main 同時有論文與簡報

【檢查點】今日帶走

☐ Codex 桌面版＋ AGENTS.md 專案規則

☐ GitHub 私有 repo ＋多筆 commit 與 PR

☐ MedSci Skills v6.1.0 已安裝

☐ 論文大綱＋一段草稿＋稽核報告

☐ 由論文產生的簡報＋提示詞模板包

總結：一條可重複的工作流



- 明天的第一步：選一個真實小題目，用提示詞模板讓 Agent 開分支、寫一段、開 PR
- 原則不變：說目標、給邊界、要計畫、審變更、要回報
- 紅線不變：不貼 PHI 、引用必核實、人負最終責任

Q&A ｜建議提問方向： PHI 與去識別、引用驗證、 AI 揭露、機關電腦安裝限制