

《AI代理與自動化工作流》學生活動手冊

AI代理與自動化工作流 | 6 週 × 3 小時 | 第一版

課程先從Codex入門開始：開啟教師準備的資料夾，讀材料、逐步完成一次情報整理，開啟報告並回顧剛做的步驟。第1週不先定義Skill、不寫範本，也不觸發Skill；交首次報告與操作步驟。第2週再從這些步驟發現可封裝的方法，認識Skill、判斷何時值得建、看使用好處，建立第一份情報Skill後換輸入驗收。

情報研究員 → 營運助理 → 分析與開發助理 | 6週，共18小時

姓名 / 組別：_____（學生作答）

本課用虛構咖啡店練習三種代理工作。營運助理是核心：把信件整理成可查核報名表與草稿，再由人核准。情報提供活動來源；分析與開發提供模擬決策工具。每週閱讀 [lesson-notes.md](#) 對應章節，按六站操作並留下自己的檔案。

課前準備與保存

開啟教師已準備好的「咖啡店三代理」資料夾；老師已放好inputs原始材料、outputs產物位置與records舊版位置。指令中的檔名依實際路徑調整；不要覆寫原始素材。工具不能讀資料夾時上傳或完整貼文；不能存檔時手動另存並重新開啟。

確認你當時的帳號是否可用，免費額度及功能可能不同，本課不保證用量。每週只安排一項必要代理任務；先備齊原始材料與手寫預期，再執行本週任務；看到輸出先核對，再決定要不要重跑。額度不足按手冊人工演練，交規格、手算或規則工具紀錄，代理狀態寫「尚未執行」。網站 lab是規則示範，不連信箱，不是真實AI代理，不替代你自己的執行。

所有信件均為模擬 [example.invalid](#) 地址。本課不真實寄信、不公開報名，不需 n8n、Make或付費API。教師核對範例是預期，不能抄入自己的實際輸出欄。每次記實際材料、Skill版本、操作、輸出位置、核對與狀態。

課前請回答（學生作答）：我想改善的工作_____；我有權使用的材料_____；成果給誰_____；我會如何核對_____。

第 1 週 Codex入門，完成第一次情報整理

交付：[week1-task.md](#)、首次[week1-summary.md](#)、[week1-steps.md](#)及自己的操作紀錄；本週不建立或使用Skill。

六站跟做

1. 看教師示範Codex App，開準備好的資料夾；手寫本次目的、材料與成果。
2. 圈出已提供數字與沒有提供的資訊。A只有三次午間排隊，B為8人便利樣本、5願試、3無需求，C為虛構50 / 55，D沒有成本。
3. 跟老師讀懂一次任務指令：目的、材料位置、四欄摘要、先顯示再保存；不需寫範本或程式。

4. 在Codex App直接給本次任務與材料，完成本週一項必要任務。先顯示成果等人核對，保存實際輸出；下面是任務指令。

直接讀research-pack.md與research-context.md；全部為虛構，本次不用Skill。

照本次任務要求逐步整理，材料不明先回報，不自行補料。

目的：為課堂同學整理咖啡店提案材料。

只用兩份原文，寫事實、來源、可能解釋、待確認。

對F傳聞十分鐘、G十六席草案、H不同價格條件，引用原句列採用依據，不任選。

第一週G等正式公告，第二週E1出現後才確認12席。

未量測的等待分鐘不補寫；便利樣本不推全校；意向不當交易。

C缺規格日期網址，不能當現價推薦；D沒有成本不能估獲利。

先顯示表格，等待我核對後才存 week1-summary.md。

5. 核對員回到原文查四列。找一個錯誤或缺漏，寫「第幾列 / 來源 / 應改什麼 / 保留什麼限制」，再核對修訂。

實測補充：開啟 [#lab/research](#)，辨認來源id與版本。網站的完成狀態只供演練；你另存自己的摘要，關閉再開。

6. 同學挑A、B各一句回查；在week1-steps.md回顧讀料、整理、核對與保存的實際步驟，下週才由這些步驟認識Skill。

額度不足：手填四列，互查原文，交相同檔案與代理尚未執行標記。

問題與反思

1. 能否由三次排隊算平均等待時間？缺什麼？
2. 願意試用、實際預訂、付款各需要哪種證據？
3. 甲50、乙55可算差5元，為什麼仍不足以推薦甲？

學生作答：我修正_____；依據_____；仍待確認_____。

完成檢查：四項事實都有來源；推論分開；小樣本與虛構限制保留；檔案可開啟。

第 2 週 把步驟做成Skill，換資料重用

回顧第1週的「讀材料→分清事實與推論→附來源→核對→保存報告」後，才介紹以下方法。第2週首次建立情報Skill，不把建立當作第1週的前置要求。

Skill是可按需呼叫、可重複使用的方法包，可含說明、範本、參考資料與選用腳本。相同類型工作反覆發生，且有穩定判斷、品質或格式要求時，才值得建立；一次性簡單任務可直接提示，固定算式用試算表即可。好處是省去每次重講、換輸入仍保留同流程與格式，把經驗留下來測試與改善；不保證永遠正確。第2週先比較「每次重講」的兩份提示範例，再判斷值得建的方法，改短範本，最後換輸入驗收。

Skill是可重複、可帶入不同條件的工作方法。建立階段才改方法範本；使用階段只提供本次任務條件、材料位置與批次，方法母本不改。咖啡店只是練習資料，情報追蹤、報名處理、單品分析三份Skill可在

相同工作方法下換產業、活動與品項：換輸入和政策，不改母本。這不代表能把同一Skill拿去解不同工作方法的問題。cafe-*是本課既有資料夾名稱，名稱不鎖定咖啡商品。

初學者先用教師備妥的完整短範本，不從空白起手。核心方法只留五段：用途、材料、步驟、成果、缺料回報。建立時學生先把方法範本說清楚，再請Codex協助建檔；之後使用只給本次條件、材料位置與批次，能用自己的話說明方法即可；不需懂程式、手打命令或自寫複雜schema。資料夾、Skill安裝、CLI與前端連接由教師課前準備。每份業務Skill用兩週逐步練熟，先求好用再談完整治理。第3站在建立週改範本、使用週只備本次輸入；第4站一次完整呼叫可包含助建檔及業務執行。

學生必做驗收只有兩項：固定同一Skill，換一批新的完整資訊；再試一個缺漏或衝突案例。比較步驟、來源引用、成果格式與品質是否一致，缺料是否誠實回報。SHA、複雜狀態 / 版本、整批故障、防重複事件與完整回歸測試由教師示範或選做，不作初學者必修門檻。

課堂每週一次必要呼叫是為了控制額度；Skill建立後，同一方法長期接續新資料與既有狀態，不每批重建。Skill要讓Codex在「同樣場景、不同資訊」維持一致步驟、引用依據、輸出結構、品質與停止條件。三份必要業務Skill是情報追蹤、報名處理與單品分析；開發 / 回歸驗收Skill選做。第一次做介面是一項單次Codex開發任務，可直接指定已核對規格，不必先建開發Skill。

重複使用核對：先建立一次方法母本，記錄版本；第二次換新資料並讀既有state，產生新報告版本；第三次由教師示範重跑相同批次（學生選做），不能新增重複、清掉待辦或漏掉補件。固定母本後再測新批、缺漏與來源衝突，逐項比一致步驟、引用、輸出欄位與停止條件。來源id、活動容量與數字答案是輸入，不寫死在Skill；例如12席與35元要從該批policy及成本材料讀取。只有必做測試失敗或方法不全才修Skill；學生重看原案與新案，完整回歸由教師示範，不因換批次重建。weekN檔名只為教學辨識；長期保存run-id、輸入 / 報告版本、Skill版本與接續state。

第2週起使用Skill的固定做法

在 Codex App 開啟「咖啡店三代理」資料夾，確認目前工作位置與檔案，再修改教師完整短範本，請Codex協助建檔。三份必要業務文件與一份選做參考為 [research-SKILL.md](#)、[operations-SKILL.md](#)、[analysis-SKILL.md](#)、[build-tool-SKILL.md](#)。三份必要Skill各寫用途、材料、步驟、成果、缺料回報；開發驗收Skill選做。建立週先把方法範本說清楚；使用週只給本次任務條件、材料位置與批次，再在本週必要任務明確指定 Codex讀取並依方法執行；看實際輸出與檔案，而非只看它說已讀Skill。換資料、修規則與重跑都留紀錄。

主評量為三種角色在新批、缺漏與衝突下維持相同品質；未做開發驗收Skill不扣分。最後用教師workflow範本填「先做什麼、交什麼給下一步、哪裡等人」，不需自寫系統規格。網站 lab只幫你理解規則與核對，不能代替這段 Codex實作。工具額度不足時仍改短範本並人工測方法，Codex執行欄保留尚未執行。

下載範本後在專案建立以下真正使用的檔案；範本名稱作教材標籤，修改與交付以專案檔案為唯一母本。metadata name與資料夾由教師預設，學生不需手動設定。

紀錄 01

範本：research-SKILL.md

專案內檔案 / Skill名稱：.agents/skills/cafe-research/SKILL.md；cafe-research

紀錄 02

範本：operations-SKILL.md

專案內檔案 / Skill名稱：.agents/skills/cafe-registration/SKILL.md；cafe-registration

紀錄 03

範本：analysis-SKILL.md

專案內檔案 / Skill名稱：.agents/skills/cafe-analysis/SKILL.md；cafe-analysis

紀錄 04

範本：build-tool-SKILL.md (重複變更 / 回歸驗收選做)

專案內檔案 / Skill名稱：.agents/skills/cafe-build-tool/SKILL.md；cafe-build-tool

你可以明確說「請使用cafe-research」，或指定對應檔案並請Codex照方法執行。讀取後核對實際摘要、表格與停止行為，不能只看工具說已讀。本站不教CLI安裝指令。

路徑與metadata參考OpenAI Build skills。App依老師版本示範，CLI / IDE快捷用法不直接套用到App。

長期情境為同方法的每週情報週報：沿用情報Skill，讀新來源與既有已讀狀態，維持來源核對、衝突處理及同一報告欄位；新增資料可以改結論，不能放寬品質要求。

交付：第一份情報短Skill、第二批摘要、兩項簡短測試紀錄；state與source-log由教師預備，完整交接檔選做。

六站跟做

1. 回看week1-steps.md，比較每次重講與保存方法；先說Skill是什麼、這類反覆整理何時值得建及使用好處，再回查第一批材料。
2. 讀 [research-update.md](#)。A2追加9/24與9/25兩次觀察；B2是另一批6人、2願試、4無需求；E1為CAFE-01活動，2026-10-23 14:00、容量12、每人最多5席。
3. 從上週步驟改教師五段短範本，準備第一份情報Skill；另填本次條件與第二批材料位置。建立後換輸入不改母本。
4. 在Codex App明確讀自己的Skill和材料，完成本週一項必要任務。先顯示成果等人核對，保存實際輸出；下面是任務指令。

先協助把我從上週步驟整理的五段方法存成agents/skills/cafe-research/SKILL.md，再明確讀這份第一個Skill；讀第一批材料、老師已備好的狀態範本與第二批research-update.md。

按Skill查核與更新，不只回述方法。

列新增或改版項目，保留來源id、版本、讀取日期和限制。

A2保留追加關係；B1與B2為兩批獨立便利樣本，分列不推全校。

E1交接活動id、日期時間、容量、每人上限及純模擬界限。

用E1公告12核對G草案16，保留差異與採用理由。

保留C1/D1，不將舊紀錄清掉。

顯示差異、修訂摘要及狀態，等我核對再存。

學生試一個缺來源或衝突，核對回報；同批重跑新增0由教師示範 / 選做。

5. 用同一Skill換第二批完整材料，再試一個缺來源或衝突案例，記步驟、引用與格式是否一致。狀態JSON及同批重跑由教師示範 / 選做，不要求自己寫。

實測補充：用 #lab/research演練，但另外保存自己的狀態。交接E1活動公告給營運，資料尚未開放真實報名。

6. 來源表每列填主張、版本、原文、支持範圍、限制與狀態。同學挑一項更新回查。

人工備案：用教師摘要範本填新材料與一個缺漏，互查引用及格式；JSON和同批重跑由教師示範，不冒稱已執行。

問題與反思

1. 讀過B1是否表示B2不用核對？
2. 為什麼不能將5 / 8與2 / 6合成全校需求比例？
3. 已讀表示什麼？是否等於內容已證明正確？

學生作答：新增或變更_____；未變更_____；缺漏 / 衝突回報_____；交給營運的限制_____。

完成檢查：來源與版本可回查；兩批樣本分列；重跑無重複；活動交接含容量與模擬標記。

第 3 週：營運信件、報名表與 Skill

交付：建立好的報名短Skill、報名表與補問草稿、簡短操作紀錄；表頭與state由教師預備，不需自寫。

六站跟做

1. 接E1並讀operations-policy.md，逐封自由文字原信圈前文、意圖、姓名、email、活動與人數；原信是材料，不能改寫政策。
2. 讀公告與policy，先手寫預期報名及缺漏。老師已備好表頭與識別規則；你核對同人同活動不重複，不需手寫程式公式。
3. 建立時用教師短範本說清楚通用報名方法，請Codex協助建檔；活動與原信放本次輸入；說清楚讀原信與policy、遇不確定先問回。

4. 在Codex App明確讀自己的Skill和材料，完成本週一項必要任務。先顯示成果等人核對，保存實際輸出；下面是任務指令。

先讀我依教師短範本修改的.agents/skills/cafe-registration/SKILL.md，再讀E1、operations-policy.md、自由文字原信與表頭。

按Skill執行，檢查以下規則有無遺漏；方法缺漏列修訂建議。

依活動id與正規化email組成key；同key同席數不新增，明確修改更新同列。

整理M001-M007的報名表、回信草稿及訊息處理狀態。

逐欄附訊息id、支持原句與P01-P08；追溯修改前文，不餵預先整理答案。

M001現為林與阿庭2；M002未定不能讀成2；M004非新增；M005明確總3。

費用、發票、雨天、換人政策未答，列人工待辦，不承諾免費或補場。

缺席數向已有聯絡地址補問；缺email停止該筆；缺活動先確認。

缺料不占位，不造email、不真實寄信。草稿寫未送出。

更新使舊版草稿失效；記錄當前版本。

依我修改的短Skill執行；若規則有遺漏，請列修訂建議，不直接覆寫我的方法。

先顯示內容，等我核對再存檔。

5. 手動算有效席數：林M001為2；M004是重複；吳M003為2、M005明確改3；總為5。M002、M006、M007不占位。

實測補充：用 #lab/operations看七封的規則處理。區分「系統判席數範圍」與「代理理解缺漏並寫補問」。Skill放文件夾後不保證自動觸發，本課任務中明確指定讀取。

6. 同學從表格挑一列追查原信，再開草稿檢查沒有「已寄出」。保存首次錯誤與修訂，不覆寫原始材料。

人工備案：用老師表格填姓名、活動、席數與待確認，改短Skill並草擬補問；網站操作另記為示範。

問題與反思

1. M004與M001的訊息id不同，為何仍只算2席？
2. M005為何更新吳的同一列？若修改語意不明，應如何做？
3. M006沒有email，可以由姓名猜地址嗎？為何？

學生作答：我用固定規則處理_____；需要代理理解_____；我的Skill停止條件_____。

完成檢查：有效5席；重複不新增、修改不另建；缺漏不占位；Skill方法可讀；草稿沒有真實送出。

第4週：權限、人工核准與安全重跑

長期情境為同報名方法的每批新信與補件追蹤：沿用營運Skill和既有報名、待辦、核准state；補件有新資訊才更新，未答政策保留待人工，不因下一批到來就清空。

交付：沿用營運短Skill、新批報名與補問草稿、兩項簡短測試紀錄；state、核准log及進階故障測試由教師示範 / 選做。

六站跟做

1. 寫本次權限：讀模擬材料、改本機練習檔、做草稿、經人核准後標模擬送出。本課沒有真實寄信權限。
2. 看教師示範核准指向哪份草稿，學生核對內容再同意，不需自寫id或版本schema。先寫預期：M008陳補3使總8；M009蘇5超容量候補、不占位仍8；M010吳可能三或四位未定，不修改，吳仍3。
3. 沿用五段報名短Skill，只指定本批原信位置、活動公告與policy；不改母本，方法失敗才請Codex協助修訂。複雜核准與故障規則看教師示範。
4. 在Codex App明確讀自己的Skill和材料，完成本週一項必要任務。先顯示成果等人核對，保存實際輸出；下面是任務指令。

讀上週表、狀態、.agents/skills/cafe-registration/SKILL.md與operations-test-cases.md。
 處理M008-M010：補件可加入；整團超容量候補不占位；M010可能3或4未定，保留原3與待確認。
 依P04修改先扣舊席數再驗容量；P05未答仍待人工；P08讀寫失敗停整批，不假成功。
 已有訊息id不再處理；不同id同key同席數不新增；修改更新同列並使舊草稿 / 核准失效。
 輸出有效席數、候補、停止原因、當前版本與待核准草稿。
 同批重跑防重複由教師另示範 / 選做，這次必做只核對新批與一個缺漏 / 衝突。
 缺欄停止整批的副本測試由教師另示範 / 選做，不是本次必做。
 只顯示紀錄，不真實寄信。

5. 回查實際輸出：林2、吳3、陳3為8；候補不加入合計；三或四位未定沒有覆寫吳。自己選一個缺料或衝突案例核對回報；缺欄整批停止由教師示範。

教師示範 / 選做：在營運lab模擬：核對草稿→核准特定版本→標模擬送出→再點。第二次不新增送出事件。修改內容後舊核准失效。這些都是演練，不能說郵件已寄出。

6. 用教師兩欄紀錄填「新完整資料結果」與「一個缺漏 / 衝突回報」，同學看方法與格式是否一致；完整版本與故障測試選做。

人工備案：按十封信順序演算狀態，在表上標每次當前席數與版本；核准 / 送出以模擬log呈現，代理尚未執行另標。

問題與反思

1. 同一人舊草稿已核准，修改席數後是否可直接模擬送出新草稿？
2. 缺email只停該筆、缺必需CSV欄位停整批，兩者差在哪裡？
3. 教師示範重跑後，如何看出沒有多一份報名或清掉待辦？（延伸選做）

學生作答：未通過_____；規則修訂_____；換料結果_____；核准綁定版本_____。

完成檢查：總8且候補另列；舊核准不沿用；重讀不新增；缺欄不覆寫；送出僅模擬且不重複事件。

第 5 週：分析問題、財務清理與手算

長期情境為每月新報價與CSV：沿用分析Skill，把新的成本說明、店長brief與CSV跨文件核對，保持單位、共享固定、公式、引用與停止條件一致；新案沒有細項就保留缺漏，不編原因。

交付：建立好的單品分析短Skill、有來源的CSV與比較摘要、原案 / 新案手算及一個缺料測試；規格和清理表範本由教師預備，完整分析測試選做。

六站跟做

1. 先讀decision-brief店長問題及成本原文，再對CSV表頭與單位。美式 $16 + 5 + 4 = 25$ 、拿鐵 $20 + 15 + 5 = 40$ 、店租 $5000 + 折舊1000 = 6000$ 共享。
2. 寫公式與預期：每杯毛利=售價-變動成本；毛利率=毛利 / 售價；單品月損平=向上取整固定 / 毛利。美式35元58.3333%172杯；拿鐵40元50%150杯。
3. 依完整短範本改分析Skill，請Codex協助建檔，指定決策brief與來源整理成CSV和摘要，列單位、公式、共享固定、單品假設、非法值與驗收。
4. 在Codex App明確讀自己的Skill和材料，完成本週一項必要任務。先顯示成果等人核對，保存實際輸出；下面是任務指令。

先讀我依教師短範本修改的agents/skills/cafe-analysis/SKILL.md，再讀decision-brief、analysis-spec與三份CSV。

引用成本原句整理finance-normalized.csv與decision-summary.md，再對提供CSV。

比較單杯毛利與毛利率，說能回答什麼、仍缺什麼；更新無細項，不編原因。

依Skill清理與計算，保留輸入版本與實際結果。

先檢查表頭與單位，逐列標缺漏、非數字與非法範圍；不補0。

售價 >0 ，變動成本 ≥ 0 ，固定 ≥ 0 ；毛利 >0 才算損平，固定0允許0。

固定成本共享不逐列相加；只做單品假設，不將活動報名當銷售量。

列原案與更新案的毛利、毛利率與向上取整損平，逐項對手算預期。

選一筆缺料或衝突說明原因與回報，不改原檔、不估真實獲利；其餘完整測試由教師示範。

5. 打開 #lab/analysis。輸入60、25、6000，查35與172；再讀更新65、29、7200，查36與200。把網站規則結果與手算、代理輸出三者分開記。

教師示範 / 選做：測四筆錯誤、固定0及毛利0；學生必做只選一個缺料或衝突，確認不產假答案。

6. 保存分析Skill、CSV、規格與測試狀態交給開發。最後驗算：美式171杯損益-15、172杯20；更新199杯-36、200杯0。說明這些是模型計算，不是實店營運結果。

人工備案：四則運算與向上取整可用計算機完成；逐列清理報告與規格可手寫。若沒有代理輸出，操作欄照實標示。

問題與反思

1. 為什麼不能把兩品項固定成本6000加成12000？
2. $6000 / 35 \approx 171.43$ ，為何要172？
3. 8個報名席數可以直接當8杯銷售嗎？需要哪些額外資料？

學生作答：我手算_____；實際對照_____；錯誤停止_____；尚未驗證_____。

完成檢查：單位正確，毛利率除售價；共享固定不相加；損平向上；新完整資料仍用相同格式；一個缺料 / 衝突能回報；不把席數當銷售。

第 6 週：互動決策網頁、換資料與交接

交付：[decision-tool.html](#)、[README.md](#)、[analysis-tests.md](#)、[handoff.md](#)、[workflow.md](#)及新資料同儕重跑紀錄；[build-tool-SKILL.md](#)為反覆變更與回歸驗收選做。

六站跟做

1. 寫需求：本機可開啟；改數字、讀CSV、比較兩情境、檢核與錯誤；每項有可查核預期。無需外部帳號或服務。
2. 測試表先列原案、更新、錯誤、固定0、毛利0及反覆匯入。原案172與150；更新200與172；不累加舊列、不保留過期數字。
3. 整理單次開發規格，讀已確認分析Skill，列小型HTML、CSV、錯誤、README與測試；有重複改版需求再選做開發驗收Skill。
4. 在Codex App明確讀自己的Skill和材料，完成本週一項必要任務。先顯示成果等人核對，保存實際輸出；下面是任務指令。

先讀[.agents/skills/cafe-analysis/SKILL.md](#)及已核對開發規格；這次是單次開發任務。若我另有自建[cafe-build-tool](#)，再明確讀取並用於重複變更與回歸驗收。

依Skill、[analysis-spec.md](#)與已核對手算，製作[decision-tool.html](#)與[README.md](#)。

網頁可本機開啟，支持改數字、讀CSV、比較情境、檢核與錯誤提示。

非法資料不補0，不輸出有效損平；毛利 ≤ 0 停止，固定0允許損平0。

保持單位、共享固定與單品假設，標記所有資料為模擬。

替換CSV須重新計算，不累加舊列。勿連真實信箱、勿部署。

附操作與測試步驟，提供檔案等我實際開啟核對；未測功能寫未驗證。

5. 以瀏覽器開你自己的檔案。真的讀更新CSV並查200杯，選一筆缺料或衝突核對；四筆錯誤與0邊界為教師示範 / 選做；保存實際畫面或輸出位置與結果。[#lab/analysis](#)的匯出只記示範工具使用，不能冒稱自己完成Codex生成。
6. 改教師[handoff](#)與[workflow](#)範本：情報Skill交來源與活動限制→營運Skill交表、草稿、核准與失敗→分析Skill交已核對規格與模擬CSV→單次開發任務交HTML與README；重複開發驗收Skill選做。每步列材料、權限、輸出、人工核對與失敗返回。營運只有報名，不能作為營業杯數。

同儕實測補充：同學只照README讀更新檔，重跑公式與錯誤。你修正不清楚說明並記錄改動。展示可開檔案，不只封面截圖。

額度不足：交完整開發規格、介面草圖、手算及lab規則測試，[decision-tool.html](#)寫尚未生成；日後取得額度再補生成與實測。

本機小前端選項：教師先準備[local-codex-starter.zip](#)與已登入CodexCLI。你在同一項必要Codex任務只改少量中文UI與任務設定，保留連接與權限，不做DB、登入、付款、真實寄信或部署。驗一條真實任務：選Skill→開始→狀態→開實際產物；未連接、未登入或額度不足要顯示不能執行，不假成功。草稿核准仍綁版本，決策HTML可另開。無前端環境則先完成App與檔案流程。

問題與反思

1. 更新後哪些數字一定要變？怎樣找到保留舊數的bug？
2. 同學在哪一步追問？README需要補什麼？
3. 哪部分已驗證、哪部分僅是需求或預期？

學生作答：我的交付_____；同學實際換料結果_____；未測功能_____；下一步_____。

100分驗收規準

紀錄 01

面向：情報來源與品質

配分：20

滿分需要的證據：來源主張與限制10；同Skill換新資訊及一個缺漏 / 衝突品質10。

待補的情況：無來源、兩批混合、已讀誤當已證明。

紀錄 02

面向：營運理解與核對

配分：35

滿分需要的證據：原信理解與草稿15；新批格式與品質10；缺漏補問與人工核對10。

待補的情況：更新當新增、缺漏占位、過期核准、假稱寄信。

紀錄 03

面向：分析與手算

配分：25

滿分需要的證據：公式和原案 / 更新15；單位、來源與一個缺料回報10。

待補的情況：固定相加、四捨五入、缺料補0、席數當銷售。

紀錄 04

面向：作品與交接

配分：20

滿分需要的證據：可開檔與功能10；README與同儕換料10。

待補的情況：只有截圖、功能未測、將lab當代理成果。

三份短Skill能以白話解釋、換料仍維持品質即可；不以自寫schema、SHA、CLI或進階故障測試給分。
每項依據自己的檔案與紀錄判斷。人工演練具方法證據，代理未執行保留狀態，由老師安排後續補測；

不能為了得分寫假結果。網站勾選與筆記保存在瀏覽器，沒有自動繳交，請依老師指定方式交檔，自己留完整副本。