

[한국기계연구원 동남권기계연구본부]

Claude Code 활용 교육

오후 실습 · 궁금한 주제 하나를 논문부터 과제 기획까지

2026. 10. 15. · 이철로 (지피타쿠) · 엑스워드(AXwith)

오늘 오후

내 연구 워크스페이스를 만들고, 그 공간에서 주제 하나를 끝까지 해 봅니다

준비

환경 점검

앱, 로그인, 백업,
권한 모드



실습 1

처음 써 보기

빈 폴더에서 논문 수
그래프 만들기



실습 2

내 워크스페이스 만들기

인터뷰, 내 자료 10개,
지식베이스



실습 3

주제 하나로 끝까지

논문 → 공고·나랑
맞나 → 판단 →
기획서



실습 4

내 공간으로 키우기

스킬, 기록 자동화,
KB 키우기

가져가는 것

내 프로필·문서 스타일·지식베이스, 문헌표, 공고 적합도 판정,
주제 비교표, 기획서 초안. 모두 내 PC 폴더에 파일로
남습니다.

순서만 정해 두었습니다

시간은 정하지 않습니다. 진도가 빠르면 뒤쪽 "더 해 보기"로,
늦으면 실습 3 STEP 3(주제 고르기)까지만 해도 됩니다.
쉬는 시간은 강사가 흐름을 보고 정합니다.

진행 방식

이렇게 진행합니다

1. 요청문과 주소는 **kimm.axwith.com** 에서 복사해 붙입니다. 실습 2부터는 **/lab2** ~ **/lab4** 로 시작합니다. 강사가 먼저 보여 드립니다.
2. 승인 창이 뜨면 바로 누르지 않습니다. 무엇을, 어디에 하려는지 읽고 정합니다.
3. 막히면 손을 듭니다. 보조강사가 자리로 갑니다.
4. 숫자가 나오면 직접 확인할 곳(원문 링크, API 주소)을 함께 받습니다.
5. 앞 단계가 덜 끝나도 강사가 넘어가면 따라 넘어갑니다. 남은 단계는 나중에 해도 됩니다.

이 가이드 읽는 법

- **왼쪽**: 내가 할 일, 번호 순서대로
- **오른쪽**: 요청과 결과. 실제 실행·예시·미실행 표기를 구분합니다
- **검은 띠**: 넘어가기 전에 확인할 것

결과가 조금씩 다릅니다

같은 요청이어도 Claude Code의 답은 매번 문장이 다르고, 인터넷 자료는 날마다 바뀝니다. 오른쪽 화면과 글자가 똑같은 필요는 없습니다. 흐름과 결과 파일이 같으면 됩니다.

두 가지 방식

처음인 분은 강사를 따라 하고, 써 본 분은 내 자료로 합니다

1. 강사 따라 하기 · 처음 쓰는 분

- 강사의 공개 자료 5개와 인터뷰 답이 준비돼 있습니다
- 강사 화면과 같은 프로필·지식베이스·결과가 나옵니다
- 오늘은 흐름을 익히고, 내 자료는 강의 뒤에 넣습니다

2. 내 자료로 · 써 본 분

- 내 공개 자료 10개 이내: 논문, 학회 발표, 공개 보고서
- 인터뷰에 내 정보로 답하고, 내 관심 주제로 갑니다
- 내가 쓰던 방식과 비교해 가져다 쓸 것을 고릅니다

넣지 않는 것

기관 내부 문서, 미공개 데이터, 장비 정보, 과제명, 개인정보. 넣은 파일은 처리를 위해 Anthropic 서버로 갑니다. 이미 공개한 자료만 넣습니다.

관심 주제가 없으면 AI 에이전트, 돌봄 로봇, 폐배터리 재활용, 수면 웨어러블, 배달 로봇 중 하나를 고릅니다.

A. 환경 점검

앱을 열고 실습 폴더를 확인합니다

1. Claude 데스크톱 앱을 열고 **Code** 탭(`</>`)을 누릅니다. 실행 환경은 **로컬**입니다.
업그레이드 안내가 뜨면 손을 듭니다.
2. 탐색기에서 **C:\work\kimm** 안에 **first**, **research-planner**, **equipment** 가 있는지 봅니다.
3. 받은 **kimm-lab-main.zip** 은 **다운로드** 에 둡니다. 이것이 백업입니다.
되돌리기는 명령으로 바뀐 파일을 복원하지 못합니다. 망가지면 ZIP을 다시 풉니다.
4. 모델은 중간 등급(예: Sonnet), 권한 모드는 **수동**으로 맞춥니다.
화면 위치는 kimm.axwith.com 「시작 전 준비」 그림

```
C:\work\kimm\  
├ first\           실습 1 (거의 빈 폴더)  
├ research-planner\ 실습 2·3·4 (연구 기획 공간)  
├ equipment\      더 해 보기 E (모의 장비)  
├ README.md  
└ TROUBLESHOOTING.md  막힐 때 대응표
```

경험자 한 줄 · 기본 모델이 최상위 모델이면 4시간 동안 사용 한도에 닿을 수 있습니다. 막힐 때만 올립니다.

파이썬 패키지 · 미리 깔지 않습니다. 필요할 때 Claude Code가 승인 창을 띄우고 설치합니다.

확인할 것 Code 탭이 열리고, 실습 폴더 세 개가 있고, ZIP이 다운로드에 남아 있고, 권한 모드가 수동입니다.

실습 1 · C:\work\kimm\first

처음 써 보기

규칙도 명령도 없는 폴더에서, 말 한마디로 자료를 모으고 그래프를 그립니다

실습 1·1. 폴더 열고 말 걸기 · → 1단계 질문하기

빈 폴더를 열고, 궁금한 것을 그냥 물어봅니다

1. **Select folder**로 `C:\work\kimm\first` 를 고릅니다.

[앱 실행 확인 필요: 버튼 이름] CLI: `cd C:\work\kimm\first` 후 `claude`

2. **이 폴더에 뭐가 있어?** 를 보냅니다. 파일 읽기는 승인 없이 진행됩니다.

3. **요즘 AI 에이전트 관련 논문이 얼마나 나와?**

검색 승인 창이 뜨면 주소를 읽고 허용합니다.

화면 예시

Claude Code – first

> 이 폴더에 뭐가 있어?

● `List(.`

└ 예비자료/ (README.md, openalex_연도별_논문수.csv)

예비자료 폴더 하나뿐입니다. 인터넷이 안 될 때 쓰는
논문 수 CSV 와 설명 파일이 들어 있습니다.

> 요즘 AI 에이전트 관련 논문이 얼마나 나와?

● `Fetch(api.openalex.org/works?... "AI agent"...)`

└ 허용 여부를 묻습니다 → 허용

공개 학술 DB OpenAlex 기준, 제목·초록에 "AI agent"가
들어간 논문은 **2024년 1,230편 → 2025년 4,844편**입니다.

대화는 예시로 그렸고, 논문 수는 2026-10-12 OpenAlex 조회값입니다. 앱 화면은 모양이 다르고,
답 문장은 매번 달라집니다.

실습 1·2. 승인 창 읽기 · → 3단계 만들기

만들기는 허용하고, 지우기는 거부합니다

- 1. **오늘 교육 메모.md** 를 만들고 첫 줄에 오늘 날짜와 "Claude Code 첫 실습"이라고 적어 줘 를 보냅니다.
- 2. 승인 창에서 세 가지를 읽습니다. **어떤 도구**인지, **대상**(파일 경로·명령 원문)이 무엇인지, **선택지**가 무엇인지.
- 3. 만들기는 허용해 파일을 확인합니다.
메모 파일 지워 줘 는 보내되, 지우기는 **거부**합니다.

이런 창이 뜹니다

파일을 만들까요?

Write C:\work\kimm\first\오늘 교육 메모.md
+ 2026-10-15 Claude Code 첫 실습

허용

이 세션 동안 허용

거부

명령을 실행할까요?

Bash Remove-Item "오늘 교육 메모.md"

허용

이 세션 동안 허용

거부 ← 이번엔 이쪽

버튼 문구는 앱 버전마다 다릅니다. [앱 실물 확인 필요] 오늘은 "이 세션 동안 허용" 대신 한 번씩 허용합니다.

확인할 것 만들기는 허용해서 파일이 생겼고, 지우기는 거부해서 파일이 그대로 남았습니다.

실습 1 · 3. 계획부터 받고 말기기 · → 2단계 기획하기

일이 커지면 계획부터 받고, 읽어 본 뒤 말 겁니다

1. **계획만** 먼저 받습니다. "아직 실행하지 말고 계획부터 세워 줘"
2. 계획을 **검토**시킵니다. 빠진 것, 틀릴 수 있는 것, 숫자를 확인할 방법
3. 검토를 반영해 **계획.md** 로 저장시킵니다
4. **계획.md 대로** 만들게 하고, 승인 창을 읽고 허용합니다

요청문 전문은 kimm.axwith.com 실습 1. 주제어만 내 주제로 바뀌도 됩니다

사전 실행 화면

Claude Code – first

> 아직 실행하지 말고 계획부터 세워 줘 ...

자료 OpenAlex 한 곳 · 파일 CSV, PNG

> 이 계획을 검토해 줘 ...

"AI agents" 복수형이 빠질 수 있음

> 검토 반영해서 계획.md 로 저장해 줘

✓ 계획.md

> 계획.md 대로 만들어 줘 ...

✓ CSV · PNG · 확인용 주소 2025년 4,636편(복수형 포함)
1,560편("AI agent"만)

2026-10-14 앱 엔진 사전 실행. 네 요청에 약 3분. 답 문장은 줄여 옮겼습니다.

확인할 것 계획.md 가 생겼고, 검토에서 나온 지적이 계획과 결과에 반영됐습니다.

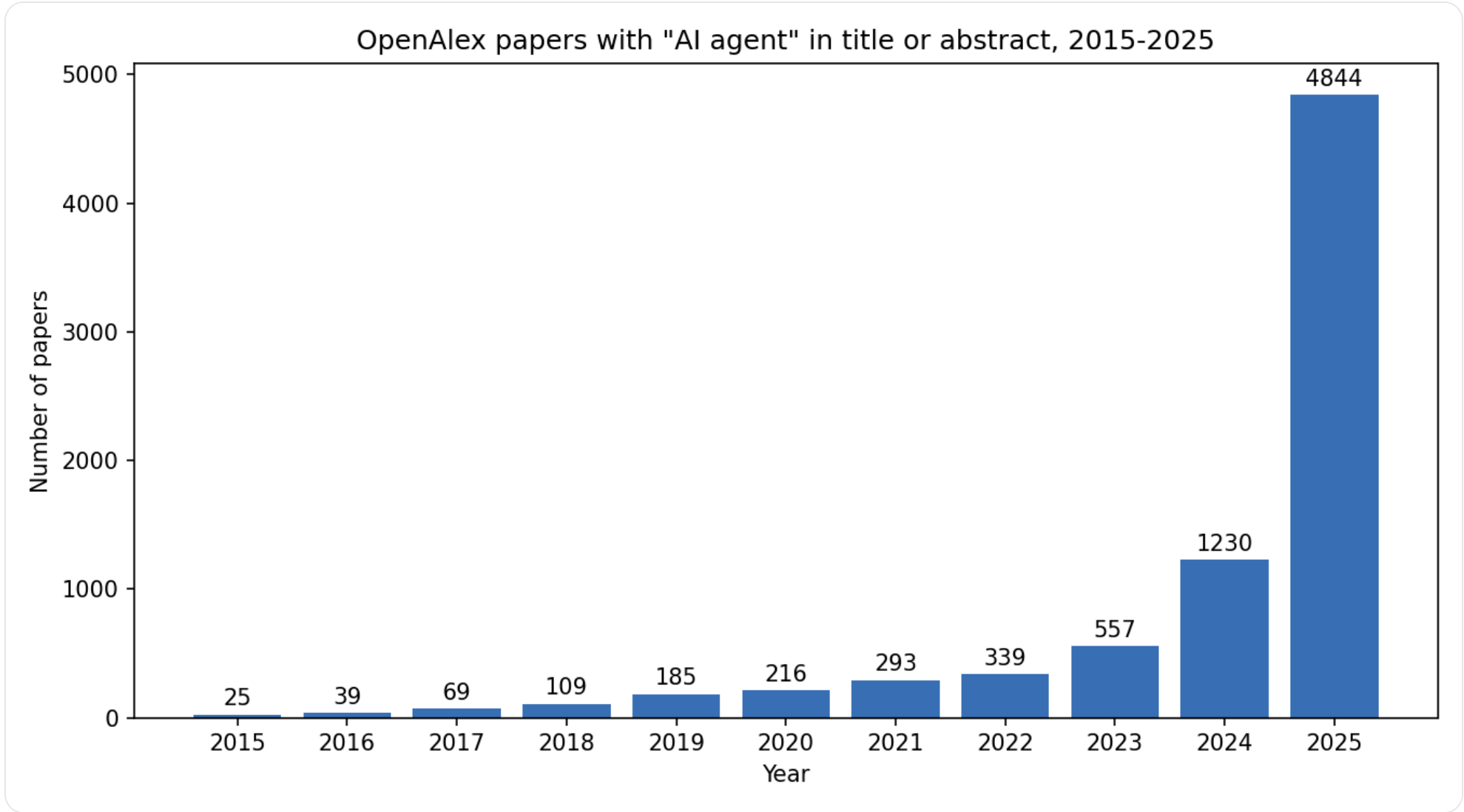
실습 1·4. 결과 확인하고 고치기 · → 4단계 검토하기

숫자는 직접 확인할 곳과 함께 받고, 그래프는 말로 고칩니다

- 1. Claude Code가 준 API 주소를 브라우저로 엽니다.
2025년 **count** 가 답과 같은지 봅니다.
- 2. 그래프를 열고 하나를 고쳐 달라고 합니다.
2024년과 2025년 막대만 빨간색으로 바꿔 줘
- 3. 주제를 하나 더 넣어 비교합니다.
**"delivery robot" 도 같은 방식으로 세서 한 그래프에 선
두 개로 그려 줘**

사전 실행 답	값
가장 많이 늘어난 해	2025년 (+3,614편)
2025년 논문 수	4,844편
스스로 밝힌 주의점	어간 처리로 "AI agents" 도 함께 잡힘. 날마다 숫자가 조금 바뀜

실제로 나온 그래프



2026-10-12 사전 실행 결과 그대로입니다. 오늘 숫자는 OpenAlex 갱신에 따라 조금 다를 수 있습니다. 인터넷이 막히면 **예비자료\openalex_연도별_논문수.csv** 로 그려 줘 라고 합니다.

실습 1 · 5. 매번 다시 말한 것 · → 5단계 개선하기

요청문마다 반복한 말은 규칙 파일로 옮길 대상입니다

매번 적은 것	실습 1에서 쓴 말
자료를 받아 올 곳	"OpenAlex 에서"
그래프 규칙	"글자는 영어로"
설치 허락	matplotlib 설치 승인
확인 방법	"확인할 API 주소를 알려 줘"
결과를 둘 곳	지금은 폴더에 섞여 있음

메모 파일에 적어 봅니다

오늘 교육 메모.md 에 "매번 말해야 했던 것"을 세 가지 이상 적습니다. 새 대화를 열면 이것을 처음부터 다시 말해야 합니다.

다음 실습

실습 2의 research-planner 는 이런 말이 규칙 파일 (CLAUDE.md)과 명령에 미리 적혀 있는 폴더입니다. 같은 일을 짧은 명령 하나로 시킵니다.

확인할 것 메모 파일에 반복해서 말한 것을 세 가지 이상 적었습니다.

실습 2 · research-planner · /lab2

내 워크스페이스 만들기

나를 알려 주고 내 자료를 넣어, 연구 기획 공간을 내 것으로 만듭니다

오전에 본 플러그인 여섯 개를 이 폴더에서 확인합니다

- 1. 새 세션에서 `C:\work\kimm\research-planner` 를 고릅니다.
- 2. "이 폴더를 신뢰하시겠습니까" 창에서 신뢰합니다.
[앱 실물 확인 필요: 신뢰 창과 플러그인 등록 안내]
- 3. 입력창의 `+ → Plugins` (CLI: `/plugin`)에서 오른쪽 여섯 개가
켜졌는지 봅니다.
- 4. `/lab2` 를 보내면 실습 전체 흐름이 먼저 나오고, 방식(강사 따라
하기 / 내 자료로)을 선택 창에서 고릅니다.
플러그인이 안 켜져도 실습 2는 진행됩니다.

어떻게 켜지나 · 폴더의 `.claude\settings.json` 에 마켓 주소와 플러그인 이름이
적혀 있습니다. 신뢰한 폴더에서만 적용됩니다.

플러그인	오늘 쓰는 곳
insane-research	실습 3 /market · 여러 관점 조사
insane-search	실습 3 · 공고 원문 읽기
kkirikkiri	세 관점 검토 팀 (더 해 보기 B)
skillers-suda	반복 업무를 내 스킬로 (실습 4 1.)
docs-guide	실습 2-B 1. · 한글 그래프
show-me-the-prd	실습 2-B 2. · 도구 기획 문서

출처: github.com/fivetaku/gptaku_plugins (엑스워드 기존 자료, 공개 저장소)

확인할 것 폴더를 신뢰했고, /lab2 가 방식을 물었습니다.

나는 누구이고 어떻게 일하는지, 여덟 가지 질문에 답합니다

- 1. /onboarding 을 보냅니다. 할 일 여섯 단계가 먼저 나옵니다.
- 2. 선택 질문 넷은 선택 창에서 고르고, 나머지 넷은 한두 줄로 답합니다.
- 3. 강사 따라 하기는 examples\강사-따라하기\인터뷰답.md 의 답을 그대로 씁니다.

과제명·미공개 내용은 답하지 않습니다 · 섞여 들어가면 [비공개] 로 바뀌 적게 되어 있습니다.

여덟 가지 질문

#	질문	강사 답(예)
1	소속 유형	그 밖 · 1인 기업
2	분야와 하는 일	AI 에이전트 도구 개발·교육
3	역할	연구 책임·기획
4	가장 오래 걸리는 반복 업무	사례 조사·출처 확인, 공고 훑기
5	주로 쓰는 문서	보고서, 과제 계획서, 발표
6	답변 방식	결론 먼저, 표·숫자 위주
7	하지 말 것	비유·슬로건, 확인 안 한 수치
8	관심 주제	AI 에이전트로 연구자의 문서 업무 줄이기

확인할 것 여덟 가지에 답했고, 8번 관심 주제가 실습 3의 주제가 됩니다.

오늘은 최근 공개 자료 10개 이내만 00-inbox에 넣습니다

- 1. 인터뷰가 끝나면 자료를 넣으라는 안내가 나옵니다. 탐색기에서 00-inbox 에 복사합니다.
- 2. HWP는 PDF로 저장해서 넣습니다.
- 3. 다 넣었으면 완료 라고 보냅니다.
강사 따라 하기는 /lab2 가 강사 자료 5개를 이미 넣어 두었습니다.

넣을 것	넣지 않는 것
내가 쓴 논문	기관 내부 문서
학회 발표 자료	미공개 데이터·시험 결과
공개된 보고서	장비 정보, 과제명
공개된 과제 요약	개인정보가 든 파일

왜 10개인가

자료가 많을수록 흡수 시간이 길어집니다. 오늘은 흐름을 익히고, 나머지는 강의 뒤 KB-구축가이드.md 대로 묶음으로 넣습니다.

확인할 것 넣은 파일이 모두 이미 공개한 자료입니다. 넣은 파일은 처리를 위해 Anthropic 서버로 갑니다.

자료마다 요약 카드를 만들고, 실적은 증빙이 있을 때만 "보유"로 적습니다

- 1. 완료 를 보내면 자료 흡수가 됩니다. 파일 쓰기 승인 창을 읽고 허용합니다.
- 2. 70-kb\sources 의 카드 하나를 엽니다. 요지, 방법, 수치, 원본 위치가 있는지 봅니다.
- 3. 70-kb\assets\보유역량.md 에서 "보유"와 "부분"을 봅니다.

사전 실행 결과 · 강사 자료 5개

보유 역량	상태	근거
공개 플러그인 모음 gptaku-plugins	보유	공개 GitHub
R&D 과제 파이프라인 research-lab	보유	공개 GitHub
공개 세미나 2건	부분	발표 자료만, 원자료 없음
기업·기관 교육	부분	자기 소개 진술만
기구 설계·시험 10년	부분	자기 소개 진술만

요약 카드 5장 · 원본은 00-inbox_done 으로 · 목록은 70-kb\INDEX.md

확인할 것 "보유"는 증빙이 있는 것뿐입니다. 이 구분이 실습 3의 공고 적합도 판정에 그대로 쓰입니다.

내 자료에서 뽑은 문서 스타일을 읽고, 틀린 곳을 고쳐 달라고 합니다

- 1. 흡수가 끝나면 "이렇게 이해했습니다"가 나옵니다. 하는 일, 문서 스타일, 용어, 실적, 먼저 시켜 볼 일 세 가지.
- 2. 한 줄씩 내 실제 문서와 비교합니다.
- 3. 틀리면 **OO는 틀렸어, △△야** 라고 보냅니다. 고친 뒤 다시 보여 줍니다.

습관과 관찰을 나눕니다 · 자료 두 개 이상에서 보인 것만 "습관"으로 적고, 하나에서만 보인 것은 "관찰"로 둡니다.

사전 실행 결과 · 70-kb\style\작업스타일.md

항목	뽑은 내용	구분
끝맺음	"~합니다" 체	습관
제목	주제가 아니라 결과를 한 문장으로	습관
한계	못 한 것을 숨기지 않고 따로 적음	습관
단계	표(단계·하는 일·산출 위치)	습관
수치	조건을 표 아래에 붙임	관찰

문서 구성 순서: 결론 → 근거 표 → 조건·한계 → 출처 (세미나 자료 2개 공통)

확인할 것 스타일 표의 항목 하나 이상을 내 문서와 직접 대조했습니다.

새 세션이 나를 알고 시작하는지 보고, 내 방식과 비교합니다

1. **CLAUDE.md** 맨 아래 **## 나에 대해** 를 엽니다.
2. 새 세션에서 **나에 대해 아는 것을 말해 줘**
3. 써 본 분은 비교를 요청합니다. **내가 원래 쓰던 방식은 ○○야. 이 워크스페이스와 비교해서 여기만 있는 것, 나만 있는 것, 가져다 쓸 것을 표로 정리해 줘**

사전 실행 결과 · CLAUDE.md 에 더해진 점

● ● ● CLAUDE.md · ## 나에 대해

한 줄 소개 AI 에이전트 도구 개발·교육,
전에는 기구 설계와 시험 10년
답변 방식 결론 먼저, 표·숫자 위주
금지 비유·슬로건, 확인 안 한 수치
문서 70-kb/style 을 따른다
관심 주제 AI 에이전트로
연구자의 문서 업무 줄이기

확인할 것 새 세션의 답에 오늘 정한 내용이 나오고, (써 본 분) 가져다 쓸 것 하나를 메모했습니다.

공식 문서를 확인해, 그래프 제목을 한글로 고칩니다

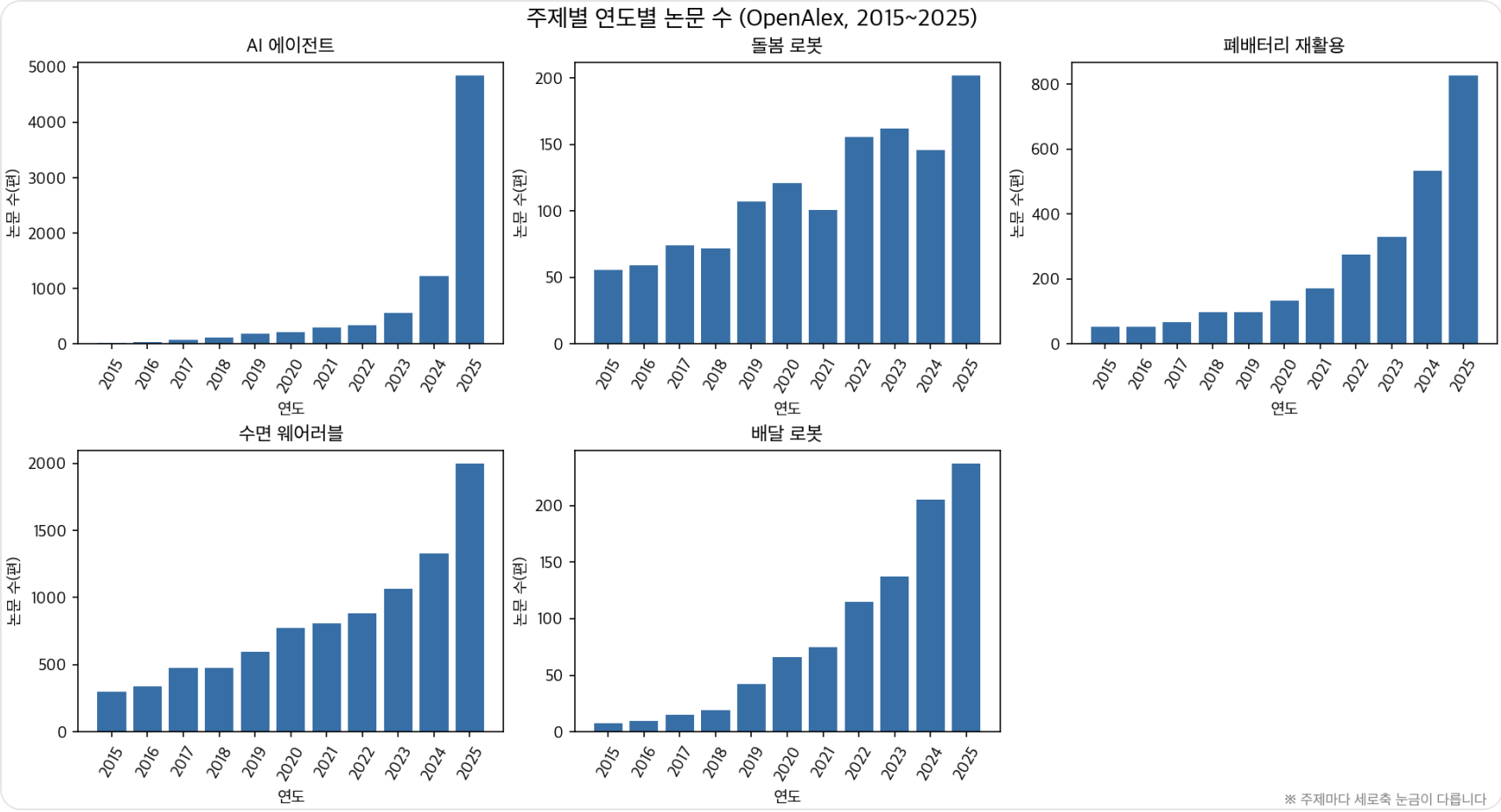
이렇게 요청합니다

```
/docs-guide:docs-guide
```

matplotlib 공식 문서 기준으로 한글 글꼴 지정 방법을 확인하고, 실습 1의 논문 수 CSV로 막대그래프를 다시 그려 줘. 제목과 축은 한글로, 그래프와 문서 주소는 **10-literature** 에 남겨 줘.

CSV는 `..\first\예비자료\openalex_연도별_논문수.csv` 입니다.

사전 실행 결과 · 71초



참고 문서: matplotlib text_props.html("Text with non-latin glyphs"), fonts.html

확인할 것 답에 공식 문서 주소가 있고, 내 PC의 PNG에서 한글이 네모로 깨지지 않습니다.

매주 논문 수를 받아 보는 도구를 기획합니다

이렇게 요청합니다

`/show-me-the-prd:show-me-the-prd`
관심 주제 논문 수를 매주 받아 보는 도구. 기획 문서는 **PRD** 폴더에 만들어 줘.

질문이 오면 답합니다. 모르는 것은 "기본값으로 정하고 가정으로 적어 줘"라고 답합니다.

사전 실행 결과 · 91초

PRD · 문서 네 개와 안내 파일

01_PRD.md	목적·기능·성공 기준
02_DATA_MODEL.md	주제·키워드·주간 집계
03_PHASES.md	단계별 구현 순서
04_PROJECT_SPEC.md	기술 스택·AI 규칙

맨 위 가정 표 8개

본인용 1인 · 화면 없이 주 1회 보고

공개 서지 API(OpenAlex·arXiv)

API 호출 한도는 미확인

확인할 것 가정 표에서 내 상황과 다른 것을 하나 골라 고쳐 달라고 합니다.

실습 2-B · 3. 결과 폴더 · 오전 19장 표 · → 4단계 검토하기

플러그인 이름보다, 남은 파일과 근거를 확인합니다

이렇게 요청합니다

방금 한 두 작업의 파일 경로와 출처를 표로 보여 줘. 실제로 열어서 확인한 것만 적어 줘.

사전 실행 뒤 폴더

```
research-planner/  
├─ 10-literature/  
│   한글 그래프 PNG · 공식 문서 주소  
├─ 60-scripts/  
│   그래프 스크립트  
└─ PRD/  
    기획 문서 네 개 · 안내 파일
```

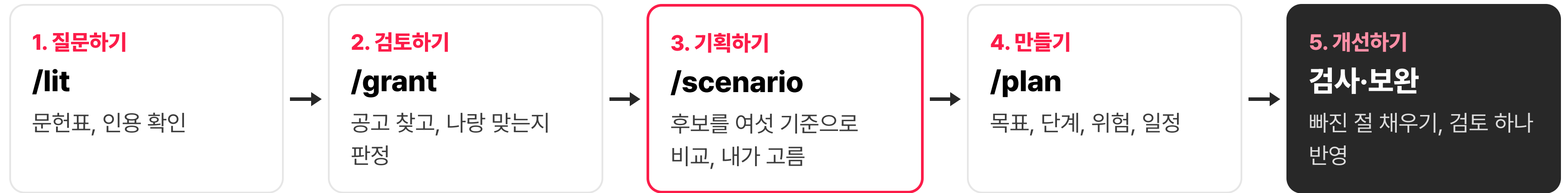
다음 실습 `/market` 은 insane-research, `/review` 는 kkirikkiri로 이어집니다.

실습 3 · research-planner

주제 하나로 끝까지

논문을 찾고, 공고와 시장을 보고, 어떤 주제가 맞는지 판단해 기획서 한 장으로 만듭니다

찾기부터 기획서까지, 다섯 STEP으로 이어 갑니다



오전에 본 KBSI 흐름과 이어집니다

공고 수집·자격은 `/grant search`, 나랑 맞는지는 `/grant fit`, 리서치는 `/lit` · `/market`, 계획서는 `/plan`, 형식 검사는 `check_planner.py` 입니다.
명령 뒤 주제를 비우면 프로필의 관심 주제로 둡니다.

다른 점 오늘은 관심 주제를 찾기 위해 논문부터 시작합니다. 형식 검사 통과와 사실 확인은 다릅니다.

/lit에 질문 한 줄을 주고, 문헌표와 더 확인할 질문을 받습니다

- 1. 새 세션에서 내 주제로 질문 한 줄을 보냅니다.
/lit AI 에이전트가 연구자의 문서 업무 시간과 품질에 미치는 영향
- 2. 범위를 물으면 번호로 답합니다. 기본값(최근 5년, 영어+한국어)이면 충분합니다.
- 3. 10-literature\ 에 생긴 문헌표를 엽니다. 맨 아래 빈틈 절을 꼭 읽습니다.

사전 실행 결과 · 문헌표 8건 중 3건

문헌	핵심 수치	관계
Noy & Zhang 2023, Science	전문직 453명 글쓰기 과제, 시간 40% 감소, 품질 18% 상승	같다
Dillon 외 2025, NBER	Copilot 7,137명 현장 실험, 이메일 주 1.3시간 감소, 문서 작성 시간은 유의한 변화 없음	반대
Becker 외 2025, METR	숙련 개발자, AI 사용 시 19% 느려짐	반대

10-literature\...-lit-review.md · 빈틈

1. 공공 연구기관 연구원의 문서 업무 시간·품질을 통제해서 켜 연구는 찾지 못했다

2. 실험(큰 효과)과 현장(작은 효과)의 격차 원인을 나눠 본 연구가 없다

웹 검색 요약 기준 · 원문 대조 전

진짜 있는 논문인지는 스크립트가, 주장이 맞는지 검토 담당이 봅니다

1. `/lit` 은 끝에 인용 검사를 스스로 돌립니다. 결과 줄을 찾아 읽습니다.

직접 돌리려면 `cite_check.py` 로 방금 문헌표 인용 확인해 줘

2. 검토 담당(서브에이전트)에게 맡깁니다.

`citation-reviewer` 에게 이 문헌표의 핵심 주장마다 근거가 충분한지 검토시켜 줘

3. 문헌표 아래 **미확인** 절을 봅니다. 확인 못 한 것이 지워지지 않고 남아 있어야 합니다.

서브에이전트 · 따로 떨어진 대화에서 맡은 일만 하고 결과만 돌려주는 담당입니다. 이 폴더에는 인용 검토, 방법 검토, 반론 찾기 세 담당이 `.claude\agents\` 에 들어 있습니다.

실제 검사 결과

```
python 60-scripts/cite_check.py file 10-literature/...

✓ OK doi:10.1126/science.adh2586
  → Experimental evidence on the productivity effects
    of generative artificial intelligence (2023) Science
✓ OK doi:10.1287/orsc.2025.21838
  → Navigating the Jagged Technological Frontier (2026)
✓ OK doi:10.1038/s41586-025-10072-4
  → Synthesizing scientific literature ... (2026) Nature
● OK_PREPRINT arxiv:2409.04600
  · 프리프린트 — 기본 D급

요약: OK 3 · OK_PREPRINT 1
✓ 게이트 통과
```

Crossref, OpenAlex, arXiv 에 실제로 있는지 묻는 스크립트입니다. 없는 논문 (`NOT_FOUND`)이나 철회된 논문(`RETRACTED`)은 표에서 빠지고 미확인으로 옮겨집니다.

국가 R&D 공고를 검색해, 이 주제에 돈을 쓰는 곳을 찾습니다

- 1. **/grant search 인공지능** 처럼 짧은 검색어를 보냅니다.
NTIS(국가R&D통합공고), IRIS(범부처통합연구지원시스템)를 스크립트로 검색합니다.
- 2. 0건이 나오면 Claude Code가 검색어를 넓혀 다시 찾는지 봅니다.
- 3. **50-grants** 의 공고 목록에서 접수 기간과 관련성 한 줄을 읽습니다.

공고명만 본 단계입니다 · 신청 자격, 예산, 배점은 공고문을 읽어야 합니다. 다음 장의 **/grant fit** 이 그 일을 합니다.

사전 실행 결과 (2026-10-12 검색)

Claude Code – research-planner

> /grant search 인공지능 에이전트

- Bash(`python 60-scripts/rnd_ancm.py ntis --keyword "인공지능 에이전트"`)
 - └ 0건 (iris 도 0건)두 단어를 함께 넣으면 공고명이 맞지 않습니다.
"인공지능", "AI", "AX", "자율"로 넓혀 다시 찾습니다.
- Bash(`rnd_ancm.py ntis --keyword "AI"`) └ 7건

✓ 후보 5건 → **50-grants/2026-10-12-공고검색-AI에이전트.md**

공고명 (줄임)	접수
중소제조 특화 Multi AI Agent 개발(R&D)	10.01~10.30
AI 기반 대학 과학기술 혁신사업(자율실험실)	10.01~10.14
지역거점 AX 혁신 기술개발 사업	10.26~11.09 예정

공고 하나를 골라 내 KB와 대조하면, 어디서 막히는지가 먼저 보입니다

- 1. 공고 목록에서 하나를 골라 보냅니다. /grant fit {공고명 또는 번호}
 - 2. 맨 위 **판정**과 **자격 게이트** 표를 읽습니다. 자격에서 막히면 기술 요건은 보지 않습니다.
 - 3. "없음"이 "없다고 확인"인지 "증빙이 없음"인지 근거 줄로 구분합니다.
- 오전 KBSI 실측과 같습니다 · 접수 중·예정 38건 중 자격을 통과한 것은 1건이었습니다. "해당 없음"이 정직한 결과인 주가 많습니다.

사전 실행 결과 · 강사 KB로 판정 · 75초

50-grants · 지역거점 AX 혁신 기술개발 - fit

판정: 부적합 (주관기관 단독 신청 시)

자격 게이트

기업부설연구소	없음	증빙이 KB에 없음
법인사업자	미확인	기록 없음
소재지	미확인	역외면 협약 전 이전
민간 매칭	조달 미확보	

바꿀 수 있는 길

연구소 보유 기관이 주관하고 참여기관으로
→ 담당 기관에 물을 질문 4개를 함께 남김

공고 본문만 읽음. 붙임 RFP·평가표는 받지 않아 배점은 미확인으로 남김

확인할 것 판정 근거가 내 KB 파일(프로필·보유역량)을 가리키는지 봅니다. KB가 비면 판정도 비어 나옵니다.

시장 조사는 이 주제를 누가 필요로 하는지, 누가 돈을 쓰는지 찾습니다

1. /market AI 에이전트로 연구자의 문서 업무 줄이기 를 보냅니다.
2. 15-market\ 파일 맨 위 한 줄 결론을 먼저 읽고, 반대 신호 절을 꼭 봅니다.
3. 숫자 하나를 골라 출처 링크를 열어 봅니다. 광고성 자료인지도 봅니다.

찾는 다섯 가지

누가 불편한가 · 시장 규모 · 정책·예산 · 이미 파는 회사와 제품 · 안 쓰는 이유 (반대 신호)

사전 실행 결과 · 15-market 발췌

15-market\2026-10-12-ai-agent-researcher-doc-work.md

한 줄 결론

수요가 있다. 다만 "문서 업무가 연구 시간을 많이 먹는다"는 근거와 "AI가 그 시간을 실제로 줄인다"는 근거는 따로 봐야 하고, 뒤쪽은 실측이 부족하다.

수요자와 불편

- 미국 연구자, 연구 시간의 약 42%를 행정에 씀 (FDP 조사 2018, 검색 요약 기준)
- 연구자 AI 도구 사용 57% → 84% (Wiley 2025, 출판사 자체 조사)

반대 신호

- 64%가 부정확·환각을 장벽으로 꼽음 (Wiley)
- 기관 자료를 외부 AI에 못 넣는 보안 제약

시장 규모는 "연구자 문서 업무용"을 따로 켜 보고서가 없어, 범위가 다른 인접 시장 넷을 합치지 않고 나란히 적었습니다. `check_planner.py` 형식 검사 통과.

후보 주제를 여섯 기준으로 나란히 놓고, 고르는 일은 내가 합니다

1. /scenario AI 에이전트로 연구 문서 업무 줄이기 를 보냅니다.
앞에서 만든 문헌표, 공고 목록, 시장 조사를 읽고 후보를 뽑습니다.
2. 비교표에서 점수보다 한 줄 근거를 읽습니다. 근거가 약한 점수에 표시합니다.
3. 마음에 드는 시나리오의 첫 4주에 할 일을 봅니다.
다음 주에 바로 할 수 있는지가 기준입니다.

여섯 기준 · 새로움, 실현성, 근거 강도, 수요, 영향, 위험. 기준은 frameworks\scenario-rubric.md 에 있고 내 기관 기준으로 고쳐 써도 됩니다.

사전 실행 결과 · 비교표

시나리오	새로움	실현성	근거	수요	영향	위험
1. 문서 업무 시간·품질 실측	3	3	4	4	2	4
2. 인용·수치 2단 검증	2	3	4	4	3	4
3. 보안 환경 운영 규칙	4	2	3	4	4	3

1 · 가장 작게 시작할 수 있고 6개월 목표(부서 시범 제안)와 바로 이어짐

2 · 새로움은 약하지만 가장 빨리 결과가 나오고 다른 시나리오의 품질 지표로 다시 씀

3 · 영향은 크지만 혼자서는 막힌 자원이 많고 부서 승인이 관건

점수마다 한 줄 근거가 붙어 있습니다. 시나리오 3 실현성 2점의 근거는 "허가된 보안 환경·부서 승인은 없음, 공동 연구자도 없음"입니다.

고른 시나리오 하나를 기획서로 만들고, 빠진 절은 검사기로 잡습니다

1. 비교표에서 하나를 고르고 보냅니다.

/plan 시나리오 1

논문으로 갈 거면 **/paper 시나리오 1** (더 해 보기)

2. 맨 위 **먼저 볼 것** 표에서 "없음"인 자원을 확인합니다. 이 상태로 갈지 정하는 것은 나입니다.

3. 끝에 나온 형식 검사 결과를 봅니다. 빠진 절이 있으면 **빠진 절 채워 줘**.

검사기가 잡는 것 · 필수 절 누락, 증빙 없이 "보유"라고 쓴 자원, 아직 하지 않은 실험의 결과 수치. 내용이 맞는지 사람과 **/review** 가 봅니다.

사전 실행 결과 · 40-plans 발췌

40-plans\2026-10-12-연구자-문서업무-시간품질-실측.md

고른 이유: 시나리오 1은 실현성·수요가 2와 같은 수준이지만 새로움과 근거 강도가 받쳐 주고, 시나리오 3은 보안 환경·기관 승인이 없어 혼자 시작할 수 없다.

먼저 볼 것: 핵심 자원 상태

본인 업무 시간 기록 **없음** → 1단계에서 직접 만든다
재채점자(동료 1명) **없음** → 못 구하면 간격 둔 본인 재채점

성과 지표

측정 도구 묶음 1식 · 방법·사례 보고 초안 1편
부서 시범 연구 제안서 1건
논문 게재 여부와 시점은 약속하지 않는다

● Bash(python 60-scripts/check_planner.py 40-plans/...)
✓ (plan) 통과 · 사전 실행 1분

실습 3 · 여기까지

대화가 끝나도 결과는 폴더에 파일로 남습니다

```
research-planner\  
├─ 70-kb\profile\  
│   └─ 연구프로필.md           /start  
├─ 10-literature\  
│   └─ ...-lit-review.md       /lit   문헌 8건  
├─ 50-grants\  
│   └─ ...-광고검색-AI에이전트.md /grant 후보 5건  
├─ 15-market\  
│   └─ ...-doc-work.md         /market  
├─ 20-scenarios\  
│   └─ ...-doc-work.md         /scenario 3개  
├─ 40-plans\  
│   └─ ...-시간품질-실측.md     /plan  
└─ logs\  
    └─ agent_history.jsonl      후 기록
```

다음 대화에서 이어 갑니다

새 세션을 열어도 Claude Code는 이 파일들을 읽고 이어서 일합니다.
"어제 시나리오 2로 논문 기획서 만들어 줘"처럼 말하면 됩니다.

문서마다 끝에 "미확인" 절

근거를 못 찾은 것, 원문을 대조하지 않은 수치, 사람이 정해야 할 것이
모여 있습니다. 오늘 결과를 쓰기 전에 이 절부터 읽습니다.

사전 실행 합계

다섯 명령에 중간 등급 모델로 약 7분 30초가 걸렸습니다. 승인 창을
읽고 질문에 답하는 시간은 들어 있지 않습니다.

실습 4 · research-planner · /lab4

내 공간으로 키우기

반복하는 일은 스킬로, 기록은 자동으로 남기고, 지식베이스는 계속 키웁니다

실습 4 · STEP 1. 스킬 · 오전의 반복 업무 조합 · → 3단계 만들기

오늘 두 번 이상 시킨 일은 스킬 파일 하나로 남깁니다

1. 오른쪽 요청문을 보내 `/trend` 스킬을 만듭니다. 실습 1에서 한 일을 이름 하나로 부르게 됩니다.
2. 파일 만들기 승인 창에서 내용을 읽고 허용합니다.
3. 새 세션에서 `/trend "battery recycling"` 을 실행하고, 바꿀 규칙 하나를 말합니다.

오전 연결 · `/skillers-suda:skillers-suda` 로 같은 반복 업무를 스킬로 만들 수도 있습니다.

요청문과 결과 예시

```
Claude Code - research-planner

> .claude/skills/trend/SKILL.md 를 만들어 줘.
"/trend {검색어}" 라고 하면 OpenAlex 에서
그 검색어의 2015~2025 연도별 논문 수를 세어
10-literature/ 에 CSV 와 그래프를 저장하고,
확인용 API 주소를 함께 알려 주는 스킬이야.

● Write(.claude/skills/trend/SKILL.md)
---
name: trend
description: 검색어의 연도별 논문 수 추이를
  OpenAlex 로 세어 표와 그래프로 남긴다
---

✓ 스킬 저장 · 새 세션부터 /trend 로 부를 수 있음
```

스킬은 이름, 설명, 지킬 규칙을 적은 파일입니다. 설명에 맞는 요청이 오면 `/trend` 를 치지 않아도 불러 씁니다. 이 장의 결과 줄은 사전 실행하지 않은 예시입니다.

확인할 것 `.claude\skills\trend\SKILL.md` 가 생겼고, 새 세션에서 불러 결과를 받았습니다.

흑이 남긴 이력에서 오늘 한 일을 확인합니다

- 1. `.claude\settings.json` 의 `hooks` 부분을 설명해 줘 를 보냅니다. 도구를 쓴 뒤(성공, 실패)마다 같은 스크립트가 돈다는 것을 봅니다.
- 2. 다른 PowerShell 창에서 최근 기록을 봅니다.
`Get-Content C:\work\kimm\research-planner\logs\agent_history.jsonl -Encoding UTF8 -Tail 10`
- 3. 실습 3에서 한 일(검색, 파일 쓰기, 검사)이 시각 순으로 남았는지 봅니다. 검토 담당이 한 일은 `agent` 칸에 그 이름이 남습니다.

사전 실행 기록 · 66줄 중 일부

시각	담당	도구	결과	입력
19:43:03	main	WebSearch	성공	Generative AI and productivity...
19:43:50	main	Write	성공	10-literature/...-lit-review.md
19:43:58	main	Bash	성공	cite_check.py file 10-literature/...
19:44:39	main	Bash	성공	rnd_ancm.py iris --keyword "인공지능..."
19:45:20	main	Write	성공	50-grants/...-광고검색-AI에이전트.md
19:46:48	main	Write	성공	15-market/...-doc-work.md
19:49:08	main	Write	성공	20-scenarios/...-doc-work.md
19:50:18	main	Bash	성공	check_planner.py 40-plans/...

실제 파일은 한 줄이 JSON 하나입니다(`time`, `agent`, `tool`, `input`, `stage`, `result`). 화면은 읽기 쉽게 칸을 맞췄습니다.

실습 4 · STEP 3. 기록을 다음 세션으로 · → 5단계 개선하기

/wrap이 남긴 기록을 다음 세션이 먼저 읽습니다

1. `/wrap` 을 보내고 `90-log` 의 기록 파일을 엽니다.
2. 같은 폴더에서 새 세션을 열어 `이어서 할 일을 알려 줘` 라고 합니다.
3. 읽은 파일 경로와 요약을 대조합니다. 틀린 결정은 기록부터 고칩니다.

사전 실행 · 새 세션의 첫 답

새 세션 · "이어서 할 일을 알려 줘"

90-log 최근 기록: 2026-10-14.md

한 일 · 한글 그래프, 공고 자격, PRD 문서

미확인 · 첨부 공고문, PRD 가정 A4

다음 행동 · /lit 로 문헌 정리 시작

/wrap 을 잊고 창을 닫아도 세션 종료 후 `90-log\_queue` 에 원자료를 남기고, 다음 세션이 먼저 정리합니다.

확인할 것 새 세션이 실제로 읽은 파일과 세 항목이 기록에 맞는지 확인합니다.

강의 뒤에는 자료를 묶음으로 넣고, 기록과 공고 확인은 예약해 둡니다

KB 키우는 순서 · KB-구축가이드.md

순	자료	쓰이는 곳
1	평가 결과·심사 의견	왜 됐고 왜 떨어졌나
2	과거 계획서·결과보고서	실적 증빙, 문서 구성
3	논문·특허	실적 증빙, 용어
4	장비·데이터 목록	보유 역량
5	발표 자료	문장 습관

한 번에 10~20개씩 넣고 `/kb ingest` → 결과 확인 → 다음 묶음. 한 달에 한 번 `/kb check`.
기관 지침 확인이 먼저입니다.

자동으로 돌아가는 것

무엇	언제
세션 기록 원자료	세션이 끝날 때마다(혹)
기록 정리·요약	다음 세션 시작 때(규칙)
<code>/wrap</code>	매일 18:00(예약)
<code>/weekly</code> 새 공고 스캔	매주 월 08:30(예약)

예약 등록(CLI가 있는 분) · `powershell -ExecutionPolicy Bypass -File 60-scripts\schedule_windows.ps1`

확인할 것 `/kb gaps` 로 다음 묶음에 넣을 자료 세 가지를 적었습니다.

빨리 끝났다면

더 해 보기

실습 3까지 끝낸 분, 이미 써 본 분은 아래에서 하나를 골라 이어 갑니다

더 해 보기

필요한 것 하나를 골라 이어 갑니다

A · 주제 비교

두 번째 주제

`/lit` · `/market` 결과를 만들고, 두 주제를 같은 기준으로 비교합니다.

B · 오전의 검토 조합

세 관점 검토

`/review 40-plans/{파일}`
kkirikkiri로 근거·방법·반론을 나눠 읽습니다.

E · 모의 장비

MCP 연결

다음 안내 장에서 허용 도구, 승인, 상한, 비상 정지와 기록을 확인합니다.

논문 기획은 `/paper`, 공고 적합도는 `/grant fit`으로 이어 갑니다. 도구 기획은 앞의 실습 2-B에서 했습니다.

주제 두 개를 같은 표에 놓으면, 어느 쪽이 지금 할 만한지 보입니다

- 1. 두 번째 주제로 **/lit**, **/market** 을 돌립니다.
문헌은 5건이면 충분합니다.
- 2. 두 주제를 같은 표에 놓아 달라고 합니다.
**두 주제의 가장 나은 시나리오를 여섯 기준 한 표로
놓고, 지금 혼자 시작하기 쉬운 쪽과 나중에 팀으로
키울 쪽을 한 줄씩 적어 줘**
- 3. 점수가 낮게 나온 이유가 "증빙이 없어서"인지
"정말 어려워서"인지 근거 줄을 읽고 구분합니다.

사전 실행 결과 · AI 에이전트 대 배달 로봇

상위 시나리오	새로움	실현성	근거	수요	영향	위험
AI 에이전트 · 문서 업무 시간·품질 실측	3	3	4	4	2	4
배달 로봇 · 현장 신뢰성· 개입률 지표 비교	4	2	3	4	4	3

지금 혼자 시작하기 쉬운 쪽

AI 에이전트. 공개 자료와 본인
시간 기록만으로 4주 안에 첫
집계표까지 갈 수 있음

나중에 팀으로 키울 쪽

배달 로봇. 새로움·영향은 크지만
로봇, 현장 접근, 시험 장비, 공동
연구자가 모두 "없음"

사전 실행: 두 번째 주제 문헌 1분, 시장 조사 1분 25초, 비교 2분. 점수는 각 주제
비교표에서 그대로 가져왔고, 고르는 일은 연구자 몫으로 남겼습니다.

모의 장비에서 승인·상한·비상 정지·기록을 확인합니다

- 1. 새 세션에서 `C:\work\kimm\equipment\practice-mcp` 를 고르고 신뢰합니다.
신뢰해야 허용 규칙과 MCP 서버(`kimm-equipment`)가 켜집니다. [앱
실물 확인 필요: 서버 승인 화면]
- 2. 오른쪽 표의 요청을 위에서부터 한 줄씩 보냅니다.
- 3. 확인 창에서 "다시 묻지 않기"는 고르지 않습니다. 고르면 그
동작이 이후 확인 없이 실행됩니다.

파이썬 · MCP 서버는 `python` 으로 실행됩니다. 서버가 안 켜지면
research-planner 세션에서 "equipment 의 requirements.txt 패키지를
설치해 줘" 후 앱을 다시 엽니다. 장비·주소·상한값은 모두 실습용
가상값입니다.

요청	기대하는 동작
가상 장비 상태 알려 줘	읽기 도구라 확인 창 없이 실행
출력 레벨을 0.2로 올려 줘	동작 도구라 확인 창 . 읽고 1회만 허용
레벨 0.45로 올려 줘	상한 0.30 초과로 거부, 다시 시도하지 않고 멈춤
비상 정지해 줘	확인 창 없이 바로 실행



확인할 것 상태는 바로, 동작은 확인 창, 상한 초과는 거부, 비상 정지는 바로. 네 가지를 모두 봤습니다.

마무리

오늘 만든 것을 남기고, 설문을 냅니다

1. 자원한 몇 분이 오늘 만든 결과물 하나를 자기 화면으로 보여 줍니다.
2. **research-planner** 세션에서 **/wrap** 을 보내 오늘 한 일을 **90-log** 에 남깁니다. 새 세션에서 확인한 것·미확인·다음 행동을 읽어 오는지도 확인합니다.
3. **10-literature** , **15-market** , **20-scenarios** , **40-plans** 를 압축해 안내한 곳에 올립니다.
[제출 위치 확정 후 맞춤]
4. 사후 이해도·만족도 설문을 작성합니다.

가져가서 다시 쓰기

실습 폴더는 가져가셔도 됩니다. **research-planner** 는 공개·가상 자료만 담긴 상태로 드립니다. 자기 연구에 쓸 때는 기관 AI 사용 규정을 먼저 확인하고, 외부에 알려지면 안 되는 항목은 **[비공개]** 로 표시합니다.

오늘 쓴 순서

맡기기(실습 1) → 규칙 적기(실습 2) → 명령으로 잇기(실습 3) → 스킬·기록으로 키우기(실습 4)

확인할 것 작업 기록을 남겼고, 결과물을 올렸고, 설문을 냈습니다.

막힐 때 1.

설치, 로그인, 파이썬

증상	조치
Code 탭에서 업그레이드 안내	계정 좌석 문제. 손을 들고 강사 화면을 보며 진행
unable to get local issuer certificate	사내 프록시나 파이썬 인증서 문제. 보조강사 호출, 급하면 예비자료 CSV로 진행
python 을 치면 Store가 열림	Python 3.12 설치 확인, py -3.12 --version
패키지 설치(pip)가 시간 초과	사내 프록시 문제. 보조강사 호출, 실습 1은 예비자료 CSV로 진행
No module named ...	"필요한 패키지를 설치해 줘" 후 승인 창에서 허용
/start 같은 명령이 안 보임	폴더를 research-planner 로 열었는지 확인하고 새 세션 열기

전체 대응표: C:\work\kimm\TROUBLESHOOTING.md

막힐 때 2.

권한, 한도, 인터넷, 플러그인

증상	조치
승인 창 없이 바로 실행됨	권한 모드를 수동으로
되돌리기 후에도 파일이 그대로	다운로드 의 ZIP을 다시 풀어 C:\work\kimm 을 새로 만듦
세션 한도 안내	중간 등급 모델로 바꾸고, 걸렸으면 강사 화면으로 따라가기
와이파이 끊김, 검색이 안 됨	실습 1은 예비자료 , 실습 3은 examples 결과 예시로
/lab2 가 없다고 나옴	research-planner 폴더로 세션을 열었는지 확인
플러그인이 등록되지 않음	플러그인 없이 진행. 명령이 웹 검색과 직접 작성으로 대신함
조사가 너무 오래 걸림	문헌은 5건만, 웹 검색 위주로 빠르게 라고 덧붙여 다시 요청

Q&A

오늘 만든 폴더는 그대로 가져가 다시 쓰셔도 됩니다