

BESTAR FOLLOW-UP EBOOK

채팅을 넘어, AI에게 일을 맡기는 법

코덱스·코워크·작업형 AI를 처음 이해하는 사람을 위한
앵콜 웨비나 복습 전자책

2026.07.10 웨비나 후속자료

비스타 김인숙

- 강의: 코덱스/코워크 사용법 웨비나 앵콜
- 원자료: 2026.07.10 Zoom 녹취록, 후기 설문 응답 20개
- 작성: 비스타 김인숙 강의 후속자료

이 자료를 읽는 방법

이 전자책은 버튼 위치를 외우기 위한 사용 설명서가 아닙니다. 이번 강의에서 반복해서 설명한 핵심은 “AI에게 무엇을 물어볼까”가 아니라 **“내 일을 어떤 구조로 맡길 수 있을까”**였습니다.

그래서 이 자료는 강의 흐름을 따라가며 다음 질문에 답하도록 구성했습니다.

1. 채팅형 AI와 작업형 AI는 무엇이 다른가?
2. 로컬 폴더, 작업공간, 권한, 커넥터, 로그는 왜 중요한가?
3. Markdown, HTML, HWPX 같은 문서 형식을 왜 알아야 하는가?
4. Claude Cowork, GPT Work/Codex, Claude Code, Hermes Agent는 각각 언제 쓰면 좋은가?
5. 실제 강의에서 보여준 사례들은 무엇을 보여주는가?
6. 초보자는 무엇부터 안전하게 시작하면 좋은가?

후기 설문에서 “왕초보용 프롬프트”, “처음 세팅 방법”, “실제 업무 적용 예시”, “단계별 복습 자료” 요청이 많았습니다. 그래서 강의 내용을 해치지 않는 범위에서, 각 장 끝에 **복습 포인트**와 **처음 해볼 문장**을 함께 넣었습니다.

목차

1. 이번 강의의 큰 전환: AI와 채팅하지 말고 일을 맡기기
2. 대화형 AI와 작업형 AI의 결정적 차이
3. 로컬 폴더와 작업공간: AI가 일하는 책상
4. 권한 설정: 편리함과 안전 사이에서 결정하기
5. 커넥터, API, MCP: 외부 도구를 꽃아 쓰는 방식
6. 작업 로그와 파일 관리: AI가 일을 하면 문서가 쌓인다
7. Markdown과 HTML: AI 시대의 기본 문서 언어

8. Claude Cowork 시작 화면 이해하기
9. 모델, 작업량, 사용량: 똑똑한 모델을 아껴 써야 하는 이유
10. 사례 1 — 랜덤 질문 뽑기 HTML 도구
11. 스킬: 한 번 만든 일을 반복 가능한 레시피로 바꾸기
12. 사례 2 — 인스타 성과 대시보드와 루틴 자동화
13. 사례 3 — 내 형용사 고르기 도구와 배포 개념
14. GPT Work와 Codex: 무엇이 바뀌었고 어떻게 구분할까
15. GitHub, Vercel, 배포: 내 컴퓨터 밖으로 내보내는 법
16. Hermes Agent와 파니니: 24시간 돌아가는 AI 비서의 의미
17. 녹취록과 콘텐츠 자산화: 말로 사라지던 지식을 남기는 법
18. HWP, HWPX, PDF: 한글 문서 업무를 AI에게 맡길 때
19. Harness와 외부 스킬: AI를 더 잘 달리게 하는 안장
20. 강의안·PPT 만들기: 기획 먼저, 디자인 규칙은 분명하게
21. 카드뉴스 생성기 사례: 반복 결과물에는 시스템이 필요하다
22. 초보자가 오늘 바로 시작할 수 있는 가장 작은 업무
23. AI에게 일을 잘 맡기는 말하기 방식
24. 마지막 메시지: AI 전문성이 아니라 내 일을 더 잘하는 힘
25. 부록 — 안전한 첫 실습 프롬프트와 용어 사전

1. 이번 강의의 큰 전환: AI와 채팅하지 말고 일을 맡기기

이번 웨비나의 출발점은 아주 단순했습니다.

“왜 내가 매번 AI가 준 답을 복사해서 문서에 붙여 넣어야 하지? 문서를 네가 만들면 안 돼?”

많은 사람들이 ChatGPT, Claude, Gemini를 처음 배울 때 채팅창에서 시작합니다. 질문을 쓰고 답을 받습니다. 답이 마음에 들면 복사해서 문서에 붙여 넣고, 마음에 들지 않으면 다시 물어봅니다. 이 방식은 빠른 질문, 간단한 요약, 문장 다듬기에는 충분히 유용합니다.

하지만 실제 업무는 대부분 채팅창 안에서 끝나지 않습니다. 강의안을 만들려면 과거 강의자료와 의뢰서, 후기, 기존 워크북을 봐야 합니다. 홈페이지 글을 쓰려면 기존 글, 브랜드 문체, 이미지, 워드프레스 구조를 확인해야 합니다. 보고서를 만들려면 녹취록, 양식, 이전 샘플, 고객 맥락이 함께 필요합니다.

그래서 강의에서 가장 강조한 전환은 이것이었습니다.

AI에게 답을 받는 단계에서, AI에게 일을 맡기는 단계로 넘어가야 한다.

일을 맡긴다는 것은 “프롬프트를 길게 쓴다”는 뜻이 아닙니다. AI가 볼 자료를 정리하고, 작업할 폴더를 지정하고, 결과물 형식을 말하고, 어디까지 수정해도 되는지 권한을 정해주는 것입니다. 즉, AI 활용 능력은 이제 질문을 잘 쓰는 능력에서 **업무를 구조화해서 위임하는 능력**으로 옮겨가고 있습니다.

복습 포인트

- 채팅형 AI는 답을 줍니다.
- 작업형 AI는 내 폴더와 파일 안에서 일을 합니다.
- 앞으로 중요한 능력은 “질문 잘하기”만이 아니라 “일을 맡길 수 있게 정리하기”입니다.

2. 대화형 AI와 작업형 AI의 결정적 차이

강의에서는 대화형 AI와 작업형 AI를 구분했습니다. 이 구분만 이해해도 코워크, 코덱스, 클로드 코드, 헤르메스가 왜 필요한지 훨씬 쉽게 보입니다.

구분	대화형 AI	작업형 AI
기본 관계	질문자와 답변자	업무지시자와 작업자
자료 제공 방식	채팅창에 붙여넣기, 파일 업로드	폴더, 프로젝트, 커넥터, 외부 도구 연결
결과 위치	채팅창 안의 답변	내 폴더의 파일, 문서, 코드, HTML, PDF 등
사람의 역할	답을 읽고 복사·수정	목표 설정, 자료 정리, 권한 승인, 결과 검토
잘 맞는 일	짧은 질문, 번역, 간단한 요약	복합 산출물, 반복 업무, 자료 기반 작업

대화형 AI를 쓸 때는 대화가 길어질수록 문제가 생깁니다. 처음에 올린 파일을 나중에는 잊어버리거나, 최근 대화만 보고 답하는 일이 생길 수 있습니다. 또 매번 필요한 자료를 사람이 끌어와야 하고, 결과를 다시 사람이 옮겨야 합니다.

반면 작업형 AI는 폴더를 기준으로 일합니다. “이 폴더 안의 자료를 읽고 강의안을 만들어줘”, “이 녹취록을 보고 복습 전자책으로 정리해줘”, “이 HTML 파일을 고쳐서 전체화면 발표용으로 만들어줘”처럼 말할 수 있습니다. 그러면 AI는 답만 쓰는 것이 아니라 파일을 만들고, 수정하고, 저장합니다.

처음 해볼 문장

이 폴더 안의 자료를 읽고, 어떤 자료가 들어 있는지 먼저 요약해줘. 아직 파일을 수정하거나 새로 만들지는 말고, 작업 계획만 보여줘.

3. 로컬 폴더와 작업공간: AI가 일하는 책상

강의에서 “로컬”이라는 단어가 반복해서 나왔습니다. 로컬은 어렵게 들리지만, 쉽게 말하면 **내 컴퓨터 안**이라는 뜻입니다. 로컬 폴더는 내 컴퓨터 안에 있는 특정 폴더입니다.

작업형 AI에게 로컬 폴더를 지정한다는 것은 AI에게 업무 책상을 내어주는 것과 비슷합니다. 책상 위에 강의자료, 의뢰서, 녹취록, 보고서 양식, 이미지가 잘 정리되어 있으면 AI는 훨씬 수월하게 일합니다. 반대로 다운로드 폴더처럼

파일명이 뒤섞여 있고 자료가 섞여 있으면 AI도 무엇을 기준으로 일해야 하는지 파악하기 어렵습니다.

강의에서 예로 든 상황은 강의안 제작이었습니다. 작년에 했던 강의처에서 다시 강의 요청이 들어왔다면, 작년 강의안만 볼 것이 아니라 의뢰서, 강의안, 워크북, 후기 설문, 관련 보고서가 같은 맥락 폴더에 모여 있는 것이 좋습니다. 그러면 AI에게 폴더 하나만 지정해도 강의 맥락을 훨씬 잘 이해할 수 있습니다.

폴더는 AI에게 주는 업무 책상입니다

잘 정리된 폴더는 필요한 자료가 놓인 책상입니다. 정리되지 않은 폴더는 서류가 바닥에 흩어진 사무실입니다. AI가 아무리 똑똑해도, 어떤 자료를 기준으로 일해야 하는지 모르면 좋은 결과를 내기 어렵습니다.

복습 포인트

- 로컬 = 내 컴퓨터 안.
- 작업공간 = AI가 읽고 쓸 수 있도록 지정한 폴더.
- AI를 잘 쓰려면 폴더 정리가 먼저입니다.
- 하나의 프로젝트 맥락은 가능하면 한 폴더 안에 모아두는 것이 좋습니다.

4. 권한 설정: 편리함과 안전 사이에서 결정하기

작업형 AI가 강력한 이유는 파일을 읽고 만들고 수정할 수 있기 때문입니다. 그런데 바로 그 이유 때문에 권한 설정이 중요합니다. AI에게 폴더를 지정하면 보통 “이 폴더의 파일을 변경하도록 허용할까요?”와 비슷한 질문이 나옵니다.

권한은 보통 다음 수준으로 생각할 수 있습니다.

권한 수준	의미	초보자에게 적합한가
읽기 중심	자료를 읽고 계획·요약만 함	가장 안전함
매번 허용	수정·생성 전에 확인함	초보자에게 권장
항상 허용	덜 묻고 빠르게 진행	복사본/테스트 폴더에서만 권장

강의에서는 숙련자의 경우 빠른 작업을 위해 “항상 허용”을 쓰기도 한다고 설명했습니다. 하지만 이것은 자기 작업 구조와 위험을 알고 있을 때 가능한 방식입니다. 초보자가 원본 자료, 고객 자료, 계약서, 홈페이지 파일이 들어 있는 폴더에서 바로 항상 허용을 누르는 것은 위험할 수 있습니다.

초보자의 기본 원칙

1. 원본 폴더가 아니라 복사본 폴더에서 시작합니다.
2. 먼저 계획만 보여달라고 합니다.
3. 삭제, 이동, 덮어쓰기, 발행, 발송은 승인 전에는 하지 말라고 합니다.
4. 결과물은 새 파일로 만들게 합니다.

처음 해볼 문장

이 폴더는 실습용 복사본이야. 먼저 작업 계획만 세워줘. 내가 승인하기 전에는 파일을 이동하거나 삭제하거나 덮어쓰지 마. 결과물은 새 파일로만 만들어줘.

5. 커넥터, API, MCP: 외부 도구를 꽃아 쓰는 방식

강의에서는 커넥터, API, MCP 같은 단어도 나왔습니다. 기술적으로 깊이 들어갈 필요는 없지만, 개념은 알아두면 좋습니다.

핵심은 단순합니다.

AI가 자기 혼자 할 수 없는 일을 외부 도구와 연결해서 하게 만드는 장치입니다.

예를 들어 AI가 Gmail을 읽고 메일 초안을 쓰게 하려면 Gmail과 연결해야 합니다. Notion 문서를 읽고 쓰게 하려면 Notion과 연결해야 합니다. Canva에서 디자인 결과물을 만들게 하려면 Canva와 연결해야 합니다. Instagram 통계를 가져오려면 Instagram API와 연결해야 합니다.

강의에서는 커넥터를 멀티탭에 비유했습니다. AI라는 본체에 Gmail, Notion, Google Drive, Canva, Instagram, WordPress 같은 도구를 하나씩 꽂아두는 것입니다. 처음 연결할 때는 번거롭지만, 한 번 연결해두면 그다음부터는 "이메일 초안 써줘", "노선에 정리해줘", "캔바 자료 만들어줘"처럼 훨씬 자연스럽게 일할 수 있습니다.

복습 포인트

- 커넥터 = 외부 서비스와 쉽게 연결하는 통로.
- API = 외부 서비스가 데이터를 주고받게 해주는 공식 통로.
- MCP = 여러 도구를 AI가 사용할 수 있게 연결하는 방식 중 하나.
- 연결은 처음이 어렵고, 반복 업무에서는 점점 강력해집니다.

6. 작업 로그와 파일 관리: AI가 일을 하면 문서가 쌓인다

작업형 AI는 채팅창 안에서만 답을 주지 않습니다. 실제 파일을 만듭니다. 그래서 일을 많이 시키면 파일이 계속 생깁니다. 1차 초안, 2차 수정본, 최종본, 테스트 파일, 백업 파일, 로그 파일이 쌓입니다.

강의에서는 이것을 개발자 문화의 "로그"와 연결해서 설명했습니다. 개발자들은 문제가 생겼을 때 과거 상태로 돌아갈 수 있어야 하므로 히스토리를 남기는 일을 중요하게 여깁니다. AI 작업도 비슷합니다. 수정할 때마다 새 파일을 만들거나, 이전 버전을 남겨두는 경향이 있습니다.

이것은 장점이기도 하고 단점이기도 합니다. 장점은 언제든지 과거 작업으로 돌아갈 수 있다는 점입니다. 단점은 폴더가 금방 지저분해질 수 있다는 점입니다.

파일 관리 팁

- 작업 전 프로젝트 폴더를 정합니다.
- `source`, `draft`, `final`, `archive` 정도의 하위 폴더를 둡니다.
- AI에게 "최종본 외의 중간 파일은 archive 폴더로 옮겨줘"라고 요청할 수 있습니다.
- 단, 삭제는 원본 확인 후 신중하게 합니다.

처음 해볼 문장

이 폴더의 파일을 훑어보고, 원본/초안/최종본/보관본으로 분류해줘. 실제 이동은 하지 말고, 먼저 분류 계획만 보여줘.

7. Markdown과 HTML: AI 시대의 기본 문서 언어

이번 강의에서 특히 중요하게 다룬 개념이 Markdown과 HTML입니다. 참가자들이 가장 헛갈려 하는 부분이기도 했고, 실제로 작업형 AI를 쓰면 자주 보게 되는 형식입니다.

Markdown이란 무엇인가

Markdown은 가장 가볍고 단순한 텍스트 문서 형식입니다. 확장자는 보통 `.md` 입니다. 제목, 굵은 글씨, 목록, 표 같은 구조를 아주 단순한 기호로 표현합니다.

예를 들어 `#` 제목 은 큰 제목이고, `##` 소제목 은 그보다 작은 제목입니다. **굵은 글씨** 는 굵게 표시하라는 뜻입니다. 사람에게는 처음에 낯설지만, AI와 개발 도구에게는 아주 읽기 좋은 구조화된 텍스트입니다.

강의에서 강조한 핵심은 이것입니다.

AI가 가장 편하게 읽고 쓰는 최소 단위 문서는 Markdown입니다.

PPT, HWP, PDF는 사람이 보기에는 편하지만 AI가 매번 읽기에는 무겁습니다. 이미지, 레이아웃, 서식이 섞여 있기 때문입니다. 반면 Markdown은 텍스트 중심이라 빠르게 읽고, 비교하고, 변환하기 좋습니다.

HTML이란 무엇인가

HTML은 웹페이지를 만드는 기본 언어입니다. AI가 시각화 자료나 인터랙티브 도구를 만들 때 자주 사용하는 형식입니다. 강의안처럼 화면에 보여주는 자료, 랜덤 질문 뽑기, 대시보드, 간단한 진단 도구, 카드뉴스 생성기 같은 것도 HTML로 만들 수 있습니다.

Markdown이 “기획과 원문”에 가깝다면, HTML은 “사람이 보기 좋게 구현한 화면”에 가깝습니다.

구분	Markdown	HTML
성격	가벼운 텍스트 원문	시각화된 웹 화면
장점	AI가 빠르게 읽음, 용량 작음	보기 좋음, 클릭/전환 가능
주 용도	원고, 정리, 지침, 스킬 문서	강의 화면, 도구, 대시보드, 전자책
사람에게	처음엔 낯설 수 있음	브라우저에서 익숙하게 보임

복습 포인트

- Markdown은 AI가 좋아하는 가벼운 문서입니다.
- HTML은 사람이 보기 좋은 시각화·인터랙티브 결과물에 적합합니다.
- 좋은 작업 흐름은 종종 Markdown 기획 → HTML/PDF/PPT 결과물 입니다.

8. Claude Cwork 시작 화면 이해하기

강의에서는 Claude를 예로 들어 Cwork 모드를 설명했습니다. 웹 브라우저에서 Claude를 쓰는 것과 데스크톱 앱에서 Cwork로 작업하는 것은 다릅니다.

웹에서 채팅을 하면 채팅창 안에서 대화가 쌓입니다. 하지만 데스크톱 앱에서 협업/Cwork 모드로 전환하면 폴더를 지정할 수 있습니다. 이때부터는 AI가 채팅창 안에서만 답하는 것이 아니라, 지정한 폴더를 읽고 파일을 만들 수 있습니다.

기본 흐름

1. Claude 데스크톱 앱을 설치합니다.
2. 채팅이 아니라 협업/Cwork 모드로 전환합니다.
3. 작업할 폴더를 지정합니다.
4. 권한을 선택합니다.
5. 모델과 작업량을 선택합니다.
6. 작업 목표를 말합니다.

강의에서 중요한 장면은 “폴더를 지정하는 순간”이었습니다. 이것이 채팅과 Cwork의 가장 큰 차이입니다. 폴더를 지정하면 그 폴더 안의 자료가 하나의 맥락이 됩니다.

복습 포인트

- Cwork는 “채팅”이 아니라 “협업” 모드입니다.
- 핵심은 폴더 지정입니다.
- 폴더를 지정하면 AI가 그 폴더를 기준으로 읽고 쓰기 시작합니다.

9. 모델, 작업량, 사용량: 똑똑한 모델을 아껴 써야 하는 이유

강의에서는 모델과 사용량 이야기도 자세히 나왔습니다. AI를 많이 쓰다 보면 단순히 “어떤 앱이 좋다”보다 **어떤 모델을 언제 쓸 것인가**가 중요해집니다.

강의에서는 쉬운 비유로 설명했습니다. 단순 복사 업무를 박사에게 맡길 필요는 없습니다. 간단한 일은 가벼운 모델로도 충분합니다. 복잡한 기획, 검토, 코딩, 구조화 작업은 더 똑똑한 모델이 필요합니다.

모델을 고르는 감각

업무 성격	적합한 모델 선택 감각
짧은 질문, 간단한 정리	가벼운 모델
일반적인 문서 작업, 초안 작성	중간 모델
복잡한 기획, 깊은 검토, 고난도 코딩	상위 모델
최종 리뷰, 오류 검증	상위 모델 + 충분한 작업량

다만 상위 모델은 사용량을 더 빨리 소모합니다. 구독 요금제 안에서도 5시간 단위 한도, 주간 한도, 모델별 별도 한도, 크레딧 구매 등이 있을 수 있습니다. 이 정책은 서비스마다 자주 바뀌므로, 강의에서 언급된 날짜나 모델 이름은 항상 최신 화면에서 다시 확인해야 합니다.

복습 포인트

- 모든 일을 가장 비싼 모델로 할 필요는 없습니다.
- 기획·검토·복잡한 작업에 상위 모델을 아껴 씁니다.
- 사용량 화면을 확인하는 습관이 필요합니다.
- 정책과 모델 이름은 자주 바뀌므로 최신 앱 화면을 기준으로 봅니다.

10. 사례 1 — 랜덤 질문 뽑기 HTML 도구

강의에서 가장 직관적으로 보여준 사례는 대학원 종강 파티용 랜덤 질문 뽑기였습니다.

처음 목표는 단순했습니다. 종강 파티에서 30명이 한마디씩 회고할 수 있도록 질문을 띄우고 싶었습니다. 처음에는 PPT에 질문을 여러 장 넣거나, 종이를 뽑는 방식을 생각할 수 있습니다. 하지만 AI에게 “랜덤으로 질문이 뽑히게 할 수 있을까?”라고 묻자 HTML 파일 하나로 버튼을 누르면 질문이 돌아가다가 하나씩 나오는 도구를 만들 수 있었습니다.

이 사례가 중요한 이유는 “AI가 문서를 써준다”를 넘어섭니다. AI가 **작은 도구**를 만들어줄 수 있다는 것을 보여줍니다.

이 사례에서 배울 점

1. 맥락 설명이 중요합니다.

“연세대 심리과학 특수대학원 종강 파티”처럼 상황을 말했기 때문에 질문의 톤이 맞아졌습니다.

1. 처음부터 기술 용어를 몰라도 됩니다.

“랜덤으로 뽑기처럼 할 수 있을까?”라고 물어도 AI가 HTML이라는 구현 방법을 제안했습니다.

1. 결과물은 채팅창이 아니라 파일입니다.

HTML 파일이 만들어지고, 브라우저에서 열 수 있습니다.

1. 현장 사용 조건을 말하면 기능이 추가됩니다.

“USB 포인터로 넘길 수 있게 해줘”, “전체화면으로 쓰고 싶어”처럼 실제 사용 상황을 말하면 됩니다.

처음 해볼 문장

작은 모임에서 참가자들이 돌아가며 답할 수 있는 랜덤 질문 뽑기 화면을 만들고 싶어. 질문은 20개 정도이고, 버튼을 누르면 중복 없이 하나씩 나오게 해줘. HTML 파일 하나로 만들어줘.

11.스킬: 한 번 만든 일을 반복 가능한 레시피로 바꾸기

랜덤 질문 뽑기 사례는 한 번 만들고 끝날 수도 있습니다. 하지만 비슷한 행사를 자주 한다면 매번 처음부터 만들 필요가 없습니다. 이때 등장하는 개념이 **스킬**입니다.

강의에서는 스킬을 "레시피"라고 설명했습니다. 한 번 잘 만든 작업 과정을 문서로 정리해두면, 다음에는 "개강 파티용 30명 랜덤 질문 만들어줘"라고만 해도 AI가 그 레시피를 읽고 비슷한 결과물을 만들 수 있습니다.

스킬은 보통 Markdown 문서입니다. 그 안에는 다음 내용이 들어갑니다.

- 이 스킬이 하는 일
- 언제 이 스킬을 불러야 하는지
- 어떤 정보를 사용자에게 받아야 하는지
- 어떤 순서로 작업해야 하는지
- 결과물을 어떤 형식으로 전달해야 하는지
- 피해야 할 실수는 무엇인지

스킬이 중요한 이유

AI 활용이 일회성으로 끝나지 않으려면 "좋은 결과가 나온 작업"을 다시 쓸 수 있어야 합니다. 그때 스킬은 나만의 업무 레시피가 됩니다. 강의에서 말한 "자동화로 가는 지름길"이 바로 이것입니다.

복습 포인트

- 스킬은 반복 업무를 위한 레시피입니다.
 - 한 번 성공한 작업은 스킬로 남겨야 다음에 빨라집니다.
 - 스킬도 계속 고쳐가며 좋아질 수 있습니다.
-

12. 사례 2 — 인스타 성과 대시보드와 루틴 자동화

두 번째 주요 사례는 인스타그램 성과 대시보드였습니다. 강의에서는 비스타 인스타 콘텐츠의 조회수, 저장률, 콘텐츠 유형, 릴스 후보 등을 한눈에 보는 대시보드를 보여주었습니다.

이 사례는 여러 개념이 결합된 예입니다.

1. 인스타그램 성과 데이터가 필요합니다.
2. 그 데이터를 가져오기 위해 API 연결이 필요합니다.
3. 가져온 데이터를 Notion에 정리합니다.
4. Notion 표는 사람이 보기 불편할 수 있으므로 HTML 대시보드로 시각화합니다.
5. 매주 릴스 후보를 만들고, 주기적으로 통계를 업데이트하는 루틴을 설정합니다.

여기서 핵심은 “대시보드를 만들어줘”라는 말 하나가 아닙니다. 대시보드가 나오려면 데이터가 있어야 하고, 데이터가 들어올 통로가 있어야 하고, 반복적으로 업데이트될 규칙이 있어야 합니다.

이 사례에서 배울 점

- 대시보드는 보기 좋은 화면이 아니라 **데이터 흐름의 결과**입니다.
- 자동화는 한 번에 생기지 않고, 데이터 수집 → 정리 → 시각화 → 반복 실행 순서로 만들어집니다.
- Notion이나 Sheet는 AI가 보기 위한 백엔드가 될 수 있고, HTML 대시보드는 사람이 보기 위한 프론트가 될 수 있습니다.

처음 해볼 문장

내가 관리하는 콘텐츠 성과를 한눈에 보고 싶어. 먼저 어떤 데이터를 모아야 하는지, 어디에 저장하면 좋은지, 어떤 대시보드로 볼 수 있는지 설계안만 만들어줘.

13. 사례 3 — 내 형용사 고르기 도구와 배포 개념

강의에서는 “내가 생각하는 나”와 “타인이 보는 나”를 비교하는 형용사 선택 도구도 보여주었습니다. 원래 워크북이나 종이 활동으로 하던 것을 웹에서 클릭하고 결과를 모을 수 있는 형태로 바꾸는 실험이었습니다.

이 사례는 HTML 도구와 배포의 차이를 이해하는 데 좋습니다. 내 컴퓨터 안에서 HTML 파일을 열면 나에게는 잘 보입니다. 하지만 다른 사람이 그 주소로 들어올 수는 없습니다. 내 컴퓨터의 로컬 경로이기 때문입니다.

다른 사람도 쓰게 하려면 온라인에 올려야 합니다. 이것을 **배포**라고 합니다.

로컬 파일과 배포된 웹페이지의 차이

구분	로컬 HTML	배포된 웹페이지
위치	내 컴퓨터 안	인터넷 서버/GitHub/Vercel/홈페이지
접근	내 컴퓨터에서만 열림	링크를 받은 사람이 접속 가능
적합한 일	개인 실습, 발표용 파일	설문, 진단, 고객용 도구
주의점	공유가 제한됨	보안, 데이터 저장, 개인정보 고려 필요

복습 포인트

- HTML 파일은 내 컴퓨터에서 바로 열 수 있습니다.
 - 남에게 링크로 주려면 배포가 필요합니다.
 - 배포 방식에는 홈페이지 업로드, GitHub Pages, Vercel 등이 있습니다.
-

14. GPT Work와 Codex: 무엇이 바뀌었고 어떻게 구분할까

강의 당일에는 GPT 쪽 업데이트가 막 이루어진 상황이었습니다. 그래서 기존에 “Codex”라고 부르던 흐름이 GPT 앱 안의 Work/Codex 구조로 통합되는 장면을 함께 설명했습니다. 서비스 이름과 화면은 계속 바뀔 수 있으니, 이 장에서는 원리 중심으로 이해하면 됩니다.

핵심 구분은 다음과 같습니다.

도구	이해하기 쉬운 역할
GPT Chat	일반 대화, 질문, 짧은 답변
GPT Work	문서·업무 중심 작업공간
Codex	코딩, 파일 조작, 설치·환경 점검, 개발성 작업
Claude Cowork	문서·지식·강의안·자료 기반 협업
Claude Code	코드, 웹앱, 테스트, 개발 환경 중심 작업

강의에서는 “업무 처리할 것은 대부분 Work, 코딩이나 홈페이지 발행·진단지 제작 같은 것은 Codex”라고 설명했습니다. 물론 실제 기능은 겹칠 수 있습니다. Cowork에서도 코딩을 할 수 있고, Codex에서도 문서 작업을 할 수 있습니다. 하지만 각 도구가 더 편한 영역은 있습니다.

강의에서 말한 경험적 구분

- PPT나 문서 기반 작업은 Claude Cowork가 편한 경우가 많다.
- 설치 환경, 파일 정리, 터미널, HWPX 처리, 기술적 문제 해결은 GPT/Codex가 나은 경우가 있다.
- 이미지 생성은 GPT 쪽이 강하다.
- 어떤 도구가 절대적으로 낫다기보다, 업무에 따라 시켜보고 비교하는 것이 좋다.

복습 포인트

- 도구 이름은 계속 바뀔 수 있습니다.
- 이름보다 중요한 것은 “채팅/업무/코딩/에이전트”의 역할 구분입니다.
- 초보자는 일단 앱을 설치하고, 작은 폴더에서 Work/Cowork를 경험해보는 것이 좋습니다.

15. GitHub, Vercel, 배포: 내 컴퓨터 밖으로 내보내는 법

강의 중 참가자가 “회사에서 웹앱을 만들어 직원들이 쓰고, 입력된 데이터가 대시보드처럼 보이게 하고 싶다”고 질문했습니다. 이때 나온 개념이 GitHub, Vercel, 배포입니다.

GitHub는 개발자들이 코드를 저장하고 공유하는 대표적인 공간입니다. 강의에서는 “개발자들이 쓰는 구글 드라이브”처럼 이해해도 된다고 설명했습니다. 코드를 올리고, 버전을 관리하고, 간단한 웹페이지를 공개할 수도 있습니다.

Vercel은 웹앱을 배포하는 서비스로 많이 사용됩니다. AI에게 “Vercel에 배포하고 싶어”, “GitHub Pages로 배포하고 싶어”라고 말하면, 단계별로 안내받거나 실제 연결된 환경에서는 배포까지 도움을 받을 수 있습니다.

다만 데이터가 입력되고 저장되는 서비스는 단순 HTML보다 복잡합니다. 개인정보, 저장 위치, 권한, 관리 화면, 보안이 필요하기 때문입니다. 이 경우에는 “화면만 만들어줘”가 아니라 “입력된 데이터를 어디에 저장할지, 누가 볼 수 있는지, 관리자 화면은 어떻게 할지”까지 설계해야 합니다.

처음 해볼 문장

직원들이 입력한 내용이 저장되고, 관리자가 현황을 볼 수 있는 간단한 웹앱을 만들고 싶어. 비개발자가 이해할 수 있게 필요한 구성요소를 먼저 설명해줘. GitHub나 Vercel을 쓰는 방법도 비교해줘.

16. Hermes Agent와 파니니: 24시간 돌아가는 AI 비서의 의미

강의 후반부에서는 Hermes Agent와 파니니 이야기가 나왔습니다. 이 부분은 초보자에게는 조금 무겁지만, “AI 비서”가 어떤 방향으로 갈 수 있는지 보여주는 중요한 사례입니다.

Hermes Agent는 단순한 채팅 도구가 아니라 여러 도구와 스킬을 연결해 일을 처리하는 에이전트입니다. 강의에서는 파니니가 Discord와 연결되어 있고, Google Calendar, Notion, Zoom, Google Forms, WordPress 등 여러 도구를 오가며 일하는 모습을 보여주었습니다.

예를 들어 썸이 “금요일 밤 9시에 코덱스/코워크 앵콜 웨비나를 할 거야. 줌 링크 만들어줘”라고 하면, 파니니는 줌 링크만 만드는 것이 아니라 이전 웨비나 프로세스를 찾아서 다음 작업까지 준비할 수 있습니다.

- Zoom 링크 생성
- 강의자료 확인/복사
- 후기 설문지 준비
- Google Calendar 등록
- Notion 일정 등록
- 시작 전 체크리스트 알림 예약
- 웨비나 후 녹취록 기반 후속자료 생성

이런 흐름이 가능한 이유는 여러 스킬과 도구가 연결되어 있기 때문입니다.

Hermes가 특히 유용한 상황

- 컴퓨터를 꺼도 24시간 돌아가는 비서가 필요할 때
- 메신저에서 바로 일을 시키고 싶을 때
- Gmail, Calendar, Drive, Notion, WordPress, Zoom 같은 도구가 함께 움직여야 할 때
- 반복되지만 매번 조금씩 다른 업무가 많을 때
- 작업 규칙과 선호가 계속 업데이트되어야 할 때

복습 포인트

- Cowork/Codex는 앱에서 일하는 작업형 AI에 가깝습니다.
- Hermes Agent는 여러 도구와 스킬을 연결한 상시형 AI 비서에 가깝습니다.
- 초보자가 처음부터 Hermes까지 갈 필요는 없지만, 방향성은 이해해두면 좋습니다.

17. 녹취록과 콘텐츠 자산화: 말로 사라지던 지식을 남기는 법

강의에서 반복해서 나온 중요한 주제는 **콘텐츠 자산화**였습니다. 강의, 코칭, 컨설팅, 회의에서 말로 소비되는 지식이 많습니다. 그 말들은 녹음하지 않으면 사라집니다. 녹음해도 텍스트로 바꾸지 않으면 다시 쓰기 어렵습니다.

그래서 강의에서는 다음 흐름을 강조했습니다.

1. 강의나 회의를 녹음합니다.
2. 녹취록으로 변환합니다.
3. 원본 텍스트를 저장합니다.
4. 핵심 개념, 사례, 질문, 콘텐츠 씨앗으로 자산화합니다.
5. 나중에 블로그, 강의안, 보고서, 워크북, 카드뉴스로 재사용합니다.

여기서 중요한 것은 녹취록 자체가 최종 콘텐츠가 아니라는 점입니다. 녹취록은 원천자료입니다. 이 원천자료를 Markdown, Notion, Sheet, Drive 등에 정리해두면 AI가 반복해서 읽고 활용할 수 있습니다.

복습 포인트

- 말로 한 강의도 녹취록이 되면 자산이 됩니다.
 - 자산이 되려면 텍스트로 바뀌어야 합니다.
 - 원본 녹취록과 요약/자산화 문서는 분리해서 보관하는 것이 좋습니다.
-

18.HWP,HWPX,PDF: 한글 문서 업무를 AI에게 맡길 때

한국 업무 환경에서는 HWP 문서가 자주 등장합니다. 강의에서도 HWP는 AI가 읽고 쓰기 어려운 형식이라고 설명했습니다. 그래서 한글 문서 업무를 AI에게 맡길 때는 형식을 이해해야 합니다.

기본 구분

형식	AI 작업 관점
HWP	읽기/쓰기 모두 까다로운 경우가 많음
HWPX	HWP보다 구조적으로 다루기 쉬움
PDF	읽기는 비교적 가능, 쓰기는 별도 변환 필요
DOCX	비교적 다루기 쉬운 편
Markdown	AI가 가장 가볍게 읽고 쓰기 좋음

강의에서는 HWP를 PDF로 변환하면 읽기는 가능하고, HWPX로 저장하면 내용을 채워 넣는 작업이 가능해질 수 있다고 설명했습니다. 실제로 기관 보고서, 멘토링 보고서, 포트폴리오 컨설팅 보고서처럼 양식은 고정되어 있고 내용만 바뀌는 업무는 AI에게 맡기기 좋습니다.

문서 채워넣기 업무의 흐름

1. 보고서 양식을 HWPX 또는 AI가 다루기 쉬운 형식으로 준비합니다.
2. 상담/회의/강의 녹취록을 텍스트로 준비합니다.
3. AI에게 양식과 녹취록을 함께 읽게 합니다.
4. "이 양식의 각 항목을 녹취록 기반으로 채워줘"라고 요청합니다.
5. 최종본은 사람이 검토합니다.

안전 주의

고객 정보, 민감한 상담 내용, 기관 문서가 들어 있는 경우에는 반드시 복사본에서 작업하고, 외부 공개·발송·공유는 사람이 승인해야 합니다.

19. Harness와 외부 스킬: AI를 더 잘 달리게 하는 안장

강의 후반부에는 Harness라는 개념도 나왔습니다. Harness는 원래 말의 안장, 장비 같은 뜻입니다. AI 맥락에서는 같은 AI도 더 잘 일하도록 씌워주는 규칙과 절차의 묶음으로 이해할 수 있습니다.

쉽게 말하면 다음과 같습니다.

Harness는 AI에게 "이런 방식으로 생각하고, 이런 순서로 검토하고, 이런 기준으로 결과물을 만들어라"라고 정해둔 고급 작업 규칙입니다.

전 세계의 개발자와 AI 사용자들이 GitHub에 다양한 스킬과 하네스를 공개합니다. 디자인을 잘하게 하는 규칙, 글을 매끄럽게 다듬는 규칙, 코딩을 더 안전하게 하는 규칙, 프로젝트를 계획하게 하는 규칙 등이 있습니다.

강의에서는 이런 외부 스킬을 직접 설치하려고 애쓰기보다, 링크를 AI에게 주고 "이 내용을 읽고 나에게 필요한지 검토해줘", "필요하면 설치해줘", "이 규칙을 내 작업에 맞게 일부만 반영해줘"라고 요청하는 방식을 설명했습니다.

복습 포인트

- 스킬은 반복 업무 레시피입니다.
 - Harness는 AI가 더 잘 일하게 만드는 규칙 묶음에 가깝습니다.
 - GitHub에는 공개된 스킬과 규칙이 많습니다.
 - 무조건 많이 설치하기보다 내 업무에 필요한 것만 선별해야 합니다.
-

20. 강의안·PPT 만들기: 기획 먼저, 디자인 규칙은 분명하게

강의에서는 AI로 PPT와 강의안을 만드는 방식도 자세히 다뤘습니다. 중요한 핵심은 “바로 PPT 만들어줘”가 아니라 **기획 먼저**입니다.

AI에게 바로 PPT를 만들게 하면 내용이 마음에 들지 않을 때 통째로 다시 만들게 되고, 그때마다 토큰과 시간이 많이 듭니다. 그래서 먼저 슬라이드 기획안을 만듭니다. 강의 목적, 대상, 흐름, 실습, 핵심 메시지를 텍스트로 충분히 검토한 다음 PPT로 넘어가는 것이 좋습니다.

좋은 흐름

1. 강의 요청/이메일/대상 정보를 붙여 넣습니다.
2. 강의 목적과 대상 수준을 설명합니다.
3. 먼저 강의 흐름과 슬라이드 기획안만 요청합니다.
4. 내용이 맞는지 검토합니다.
5. 기존 강의안 폴더를 참고하게 합니다.
6. 디자인 규칙과 금지사항을 분명히 말합니다.
7. PPT 또는 HTML 강의안으로 제작합니다.

강의에서 특히 인상적이었던 부분은 “AI는 하라고 하는 것보다 하지 말라고 하는 것을 잘 지키는 경향이 있다”는 말이었습니다. 실제로 디자인에서 싫어하는 요소를 명확히 말하면 결과가 좋아집니다.

예를 들어 다음과 같은 금지 규칙이 도움이 됩니다.

- 페이지 번호 넣지 마.
- 우측 상단 넘버링 넣지 마.
- 좌측 세로 강조선 넣지 마.
- 과한 AI 느낌의 장식 넣지 마.
- 글자를 너무 작게 줄이지 마.
- 내용이 많으면 슬라이드를 나눠.

복습 포인트

- PPT는 바로 만들지 말고 기획 먼저 합니다.
 - 마음에 안 드는 디자인 요소는 금지 규칙으로 정리합니다.
 - 무에서 유보다 기존 샘플과 참고자료가 있을 때 훨씬 잘 나옵니다.
-

21. 카드뉴스 생성기 사례: 반복 결과물에는 시스템이 필요하다

강의 마지막에는 카드뉴스 생성기 사례가 나왔습니다. 이 사례는 단순히 “AI가 카드뉴스를 만들어준다”가 아니라, 반복 가능한 결과물을 만들려면 얼마나 많은 규칙이 필요한지 보여주는 장면이었습니다.

카드뉴스를 반복적으로 만들려면 다음 요소가 필요합니다.

1. 브랜드 아이덴티티
2. 독자 정의
3. 디자인 시스템
4. 컬러와 폰트
5. 표지 규칙
6. 본문 카드 규칙
7. 사진/이미지 아트 디렉션
8. 문체와 톤

9. 제목 문법
10. 금지 표현
11. 원고 입력 방식
12. 이미지 가져오는 방식
13. 검수 규칙
14. 내보내기 방식

강의에서 썸은 “남이 주는 프롬프트가 만족스럽지 않은 이유”를 설명했습니다. 남의 프롬프트는 남의 브랜드, 남의 문체, 남의 디자인 규칙에 맞춰져 있기 때문입니다. 내 카드뉴스를 만들려면 내 디자인 시스템과 내 문체가 있어야 합니다.

이 사례에서 배울 점

- 반복 결과물은 프롬프트 하나로 해결되지 않습니다.
- 좋은 반복 결과물에는 시스템이 필요합니다.
- 시스템은 디자인, 문체, 데이터, 원고, 이미지, 검수 규칙으로 쪼개집니다.
- 내가 원하는 결과를 만들려면 “내 기준”을 언어화해야 합니다.

22. 초보자가 오늘 바로 시작할 수 있는 가장 작은 업무

강의가 3시간 가까이 이어진 이유는 보여줄 수 있는 사례가 많았기 때문입니다. 하지만 처음 시작하는 사람이 모든 것을 한 번에 할 필요는 없습니다. 오히려 처음에는 아주 작은 반복 업무 하나를 고르는 것이 좋습니다.

강의에서 제안한 출발점은 이것입니다.

이번 주에 내가 했던 일 중, 앞으로도 반복될 가장 작은 업무 하나를 고른다.

예를 들어 다음과 같은 업무가 좋습니다.

- 회의 녹취록을 요약해서 회의록으로 만들기

- 상담 녹취록을 보고서 양식에 채우기
- 블로그 글 초안을 카드뉴스 원고로 바꾸기
- 이메일 초안을 쓰고 Gmail 임시보관함에 넣기
- 강의 후기를 읽고 다음 강의 보완점 정리하기
- 폴더 안의 파일명을 규칙에 맞게 정리하기

중요한 것은 “새로 만드는 일”보다 “이미 해본 일”이 쉽다는 점입니다. 이전 샘플이 있으면 AI가 훨씬 잘 배웁니다.

첫 실습 순서

1. 반복될 작은 업무 하나를 고릅니다.
2. 그 업무의 이전 샘플을 모읍니다.
3. 실습용 복사본 폴더를 만듭니다.
4. AI에게 폴더를 읽게 합니다.
5. 먼저 계획만 요청합니다.
6. 새 파일로 결과물을 만들게 합니다.
7. 결과를 보고 구체적으로 수정 지시합니다.

23. AI에게 일을 잘 맡기는 말하기 방식

강의 마지막 부분에서 가장 중요한 실무 감각이 나왔습니다. AI를 잘 쓰는 사람은 꼭 기술을 많이 아는 사람이 아닙니다. 일을 설명하는 사람이기도 합니다.

샘은 본인의 강점으로 “말이 많다”는 점을 들었습니다. 사람에게 일을 시킬 때처럼 배경을 충분히 설명하고, 원하는 결과를 구체적으로 말하고, 마음에 안 드는 이유를 자세히 설명합니다. 이것이 AI에게도 그대로 통합니다.

좋은 지시의 특징

- 상황을 설명합니다.

- 대상과 목적을 말합니다.
- 참고자료를 줍니다.
- 원하는 결과물 형식을 말합니다.
- 싫어하는 방식을 말합니다.
- 수정할 때는 번호를 붙입니다.
- “마음에 안 들어”가 아니라 “왜 마음에 안 드는지”를 말합니다.

나쁜 수정 지시

별로야. 다시 해줘.

좋은 수정 지시

1. 제목이 너무 추상적이야. 초보자가 바로 이해할 수 있게 바꿔줘.
2. 본문이 활용 가이드처럼 느껴져. 강의 복습용 전자책처럼 개념 설명을 먼저 늘려줘.
3. 표가 너무 뻑뻑해. 문단 설명을 추가하고, 소제목 사이 간격을 더 띄워줘.
4. 내부 작업 용어는 빼고 참가자에게 보이는 말로 다듬어줘.

복습 포인트

- AI는 내 맥락을 모릅니다. 말해줘야 합니다.
- 좋은 결과는 좋은 설명에서 시작합니다.
- 수정 지시는 번호를 붙여 한 번에 모아주는 것이 효율적입니다.
- 도메인 지식이 있어야 결과물을 검수할 수 있습니다.

24. 마지막 메시지: AI 전문성이 아니라 내 일을 더 잘하는 힘

강의의 마지막 메시지는 기술보다 일이었습니다. AI 전문가가 되는 것이 모두의 목표일 필요는 없습니다. AI는 내가 하고 싶은 일을 더 잘하게 도와주는 도구입니다.

쌤은 강의 말미에 이렇게 정리했습니다. AI를 잘하는 것도 중요하지만, 결국 내 본업을 잘 아는 사람이 AI도 더 잘 씁니다. 마케팅을 AI로 잘하고 싶으면 마케팅을 알아야 합니다. 디자인을 AI로 잘하고 싶으면 디자인 기준을 알아야 합니다. 브랜딩을 AI로 잘하고 싶으면 브랜딩 판단력이 있어야 합니다.

AI는 대신 생각해주는 마법이 아니라, 내가 가진 기준과 자료와 맥락을 빠르게 실행으로 옮겨주는 파트너에 가깝습니다.

이번 강의의 최종 요약

- AI를 채팅창으로만 쓰지 말고 작업공간으로 데려와야 합니다.
- 폴더 정리, 문서 형식, 권한 설정이 중요합니다.
- Markdown과 HTML은 AI 시대의 기본 언어입니다.
- 반복 업무는 스킬과 루틴으로 바꿀 수 있습니다.
- 복잡한 자동화는 작은 반복 업무 하나에서 시작합니다.
- AI에게 일을 잘 맡기려면 내 업무를 잘 설명할 수 있어야 합니다.
- FOMO를 느낄 필요는 없습니다. 내 일을 더 잘하기 위한 만큼 배우면 됩니다.

25. 부록 — 안전한 첫 실습 프롬프트와 용어 사전

안전한 첫 실습 프롬프트

아래 문장을 복사해서 Cowork나 Work에서 테스트 폴더를 지정한 뒤 사용해 보세요.

이 폴더는 실습용 복사본이야.
폴더 안의 자료를 먼저 훑어보고, 어떤 자료가 있는지 요약해줘.
아직 파일을 수정하거나 삭제하거나 이동하지 마.
내가 승인하기 전에는 어떤 파일도 덮어쓰지 마.
내가 하려는 목표는 [여기에 목표 입력]이야.
이 목표를 위해 어떤 순서로 작업하면 좋을지 계획만 먼저 보여줘.

반복 업무를 스킬로 만들 때 쓰는 문장

방금 만든 작업 흐름이 마음에 들어.
이걸 다음에도 반복할 수 있게 스킬/레시피 문서로 정리해줘.
언제 이 스킬을 불러야 하는지, 어떤 정보를 먼저 받아야 하는지, 결과물은 어떤 형식으로 만들어야 하는지,
주의할 점까지 포함해줘.

용어 사전

용어	쉬운 설명
로컬	내 컴퓨터 안
작업공간	AI가 읽고 쓸 수 있도록 지정한 폴더
Markdown	AI가 읽기 좋은 가벼운 텍스트 문서 형식
HTML	웹페이지처럼 보이는 화면을 만드는 형식
커넥터	Gmail, Notion, Canva 같은 외부 도구 연결 통로
API	서비스끼리 데이터를 주고받는 공식 통로
MCP	여러 도구를 AI가 사용할 수 있게 연결하는 방식 중 하나
스킬	반복 업무를 위한 레시피 문서
루틴/크론잡	정해진 시간에 반복 실행되는 자동화
배포	내 컴퓨터 밖의 인터넷 주소에서 접근 가능하게 올리는 일
GitHub	코드를 저장·공유·배포하는 개발자용 저장소
Vercel	웹앱을 쉽게 배포하는 서비스
HWPX	HWP보다 AI가 다루기 쉬운 한글 문서 형식
Harness	AI가 더 잘 일하도록 씌우는 작업 규칙 묶음
토큰	AI가 읽고 쓰는 글자/정보의 사용량 단위

후기 설문에서 반영한 보완 포인트

이번 앵콜 웨비나 후기 설문에서는 다음 요청이 많았습니다.

- 왕초보도 따라할 수 있는 코덱스/코워크 프롬프트
- 처음 세팅 방법과 자료 정리 방법
- 실제 업무 적용 사례
- Claude 프로젝트 구성법, GitHub 활용, 자동화 워크플로우
- 브랜딩과 콘텐츠 제작에 AI를 적용하는 방법
- 내 비서처럼 작동하는 에이전트 구축과 실제 사용 모습

그래서 이 전자책에서는 단순 요약보다 다음 내용을 의도적으로 보강했습니다.

1. 각 개념을 강의 순서대로 다시 설명했습니다.
2. 실제 사례를 “무엇을 보여주는 사례인지” 중심으로 해석했습니다.
3. 초보자가 바로 복사해볼 수 있는 안전한 첫 문장을 넣었습니다.
4. 도구 이름보다 역할 구분을 중심으로 정리했습니다.
5. 권한, 원본 보호, 문서 형식, 배포처럼 놓치기 쉬운 안전 기준을 따로 설명했습니다.

더 읽어보기: 워드프레스 참고 글

이번 복습자료와 연결되는 더 자세한 설명은 비스타 워드프레스 글에서도 확인할 수 있습니다.

- [AI에게 일을 맡기는 사람들을 위한 실전 가이드](#)

이 글에는 AI를 단순 채팅 도구가 아니라 실제 업무 파트너로 쓰기 위해 필요한 관점과 작업 방식이 정리되어 있습니다. 복습자료를 읽은 뒤, 내 업무에 어떻게 적용할지 다시 확인할 때 함께 참고하면 좋습니다.

맺음말

이번 강의는 코덱스와 코워크 버튼을 외우는 시간이 아니었습니다. 오히려 앞으로 우리가 AI와 일할 때 필요한 업무 감각을 배우는 시간이었습니다.

내 자료를 어디에 두고, 어떤 폴더를 보여주고, 어떤 형식으로 결과물을 받고, 어디까지 권한을 줄지 결정하는 것. 반복되는 일은 레시피로 남기고, 좋은 결과는 다시 쓸 수 있게 만드는 것. 그리고 무엇보다 내 일을 더 잘 설명할 수 있도록 내 업무를 이해하는 것.

이것이 이번 웨비나의 핵심입니다.

AI를 더 많이 아는 것보다 중요한 것은, **내 일을 더 잘 아는 상태에서 AI에게 일을 맡기는 것**입니다. 그 출발은 거창하지 않아도 됩니다. 작은 복사본 폴더 하나, 반복되는 업무 하나, 계획 먼저 요청하는 문장 하나면 충분합니다.

마지막 참고 링크

더 자세한 실전 적용 관점은 아래 글에서 이어서 확인할 수 있습니다.

- AI에게 일을 맡기는 사람들을 위한 실전 가이드: <https://www.bestar.kr/ai-agent-work-guide/>