

2026.10.01 · 09:00~18:00 · 8교시

[장금상선 그룹] AI agent 교육 1회차

묻는 AI에서 시키는 AI로

클로드 코드 · 1교시는 데크, 2교시부터는 노션

오늘 저녁, 내가 만든 웹페이지 주소가 휴대폰에서 열립니다.

오늘 만드는 결과물 4개

① 내 직무 대시보드

② 업무 맥락 문서

③ brd.md · prd.md

④ 웹페이지 + 배포 주소

8교시 동안 결과물 4개를 만듭니다

오늘의 대표 완성물은 ④ 웹페이지입니다.

교시	시각	무엇	결과물
1교시	09:00~09:50	시키는 AI · 바이브코딩 · 클로드 코드	
2교시	10:00~10:50	설치 · 작업 폴더 · 규칙	
3교시	11:00~11:50	파일 만들기 · 직무 대시보드	① 내 직무 대시보드
점심	11:50~13:00		
4교시	13:00~13:50	프롬프트 3기법	
5교시	14:00~14:50	문서로 일한다 · 업무 맥락	② 업무 맥락 문서
6교시	15:00~15:50	만들기 전 기획서 BRD · PRD	③ brd.md · prd.md
7교시	16:00~16:50	웹사이트 실습 ① 첫 제작	
8교시	17:00~17:50	웹사이트 실습 ② 수정 · 배포	④ 웹페이지 + 배포 주소

에이전트는 2회차부터 만듭니다

오늘은 AI에게 일을 시키는 법을 배웁니다. 2회차부터 그 위에 하나씩 붙입니다.

회차	날짜	배우는 것	만드는 것
1회차 · 오늘	10월 1일 (목)	바이브코딩 · 클로드 코드 · 프롬프트 · 기획서	대시보드 · 웹페이지
2회차	10월 8일 (목)	스킬 · MCP · 메모리 · 서브에이전트	에이전트를 이루는 4가지
3회차	10월 15일 (목)	배포 · API · 스케줄링 · 문제 정의	실전 프로젝트
4회차	10월 27일 (화)	조별교육	조별 결과물

에이전트는 스킬 · MCP · 메모리 · 서브에이전트로 만듭니다. 오늘 배우는 「시키는 법」이 그 시작입니다.

1교시 · 09:00~09:50

묻는 AI에서 시키는 AI로

바이브코딩과 클로드 코드가 무엇인지 압니다.
설치는 2교시에 노션을 보며 합니다.

시키는 AI 클로드 코드 개발을 못 해도

1 시키는 AI · 2 설치 · 3 대시보드 · 4 프롬프트 · 5 업무 맥락 · 6 기획서 · 7 웹페이지 · 8 배포

생성형 AI는 회사 3곳이 이릅니다

기업	최신 모델	강점
OpenAI	GPT-6.1 Sol / GPT-6 Astra	LLM 모델의 표준, 복잡한 논리 추론, 이미지 생성
Google	Gemini 3.1 Pro / 3.8 Flash	실시간 정보 검색, 이미지 및 영상 인식 & 생성
Anthropic	Claude Fable 5.1 / Opus 5.5 / Sonnet 5.5	장문 글쓰기, 코딩, 안전성

2026년 10월 1일 기준입니다. 모델은 자주 바뀝니다.

이 과정은 Claude 하나만 씁니다.

장문 글쓰기와 코딩이 강해서, 일을 맡기기에 맞습니다.

Claude 모델은 4가지입니다

왼쪽일수록 어려운 일에 맞고, 오른쪽일수록 빠릅니다.

Claude

Fable 5.1

가장 어려운 추론과
오래 이어지는 일

속도

●○○○ 느림

Opus 5.5로도
부족할 때 씁니다

Claude

여기서 시작

Opus 5.5

오래 걸리는 코딩과
지식 업무

속도

●●○○ 보통

어느 것을 쓸지 모르겠으면
여기서 시작합니다

Claude

Sonnet 5.5

속도와 성능을
함께 갖춘 모델

속도

●●●○ 빠름

빠르게 자주
돌릴 때 씁니다

Claude

Haiku 4.5

가장 빠른 모델

속도

●●●● 가장 빠름

가볍고 단순한 일에
씁니다

클로드 코드에서는 **/model** 을 쳐서 바꿉니다. 출처 · Claude 공식 문서 「Models overview」 · 2026년 10월 1일 기준.

물어보는 AI에서 일하는 AI로

지금 · 현재 사용 방식

물어보는 AI

챗GPT, Claude에 질문하고 답을 받습니다
답을 복사해 문서에 붙입니다
정리와 작업은 사람이 직접 합니다



■ 사람이 하는 일 □ AI가 하는 일 · 왼쪽은 사람 칸이 4개, 오른쪽은 **처음과 끝 2개**뿐입니다

AI 성능이 올라가도 지금 방식이면 일은 사람 몫입니다.
묻는 AI에서 시키는 AI로. 사용 단계를 한 단계 올리는 것이 이번 과정의 목적입니다.

바이트코딩 · AI Agent

일하는 AI

복붙, 정리, 실행까지 AI가 합니다
파일 열기·생성·수정·실행을 직접 합니다
PC에서 실제 작업을 AI에 지시합니다
내가 코딩을 배우는 과정이 아닙니다



AI가 일을 뺏는 게 아니라, 일이 둘로 갈립니다

약 30% · AI에게 맡긴다

약 70% · 사람이 집중한다

사라지는 일 · 약 30%

반복, 정리, 복사·붙여넣기

엑셀 취합, 실적 수기 집계, 같은 양식 반복 작성, 메일 초안 반복.
손이 많이 가는 단순 작업입니다.

남는 일 · 약 70%

판단, 기준, 검수

무엇에 힘을 줄지, 무엇을 보고할지, 결과가 맞는지 확인.
사람의 판단이 필요한 일입니다.

앞의 30%는 AI에게 맡기고, 우리는 뒤의 70%에 집중합니다.

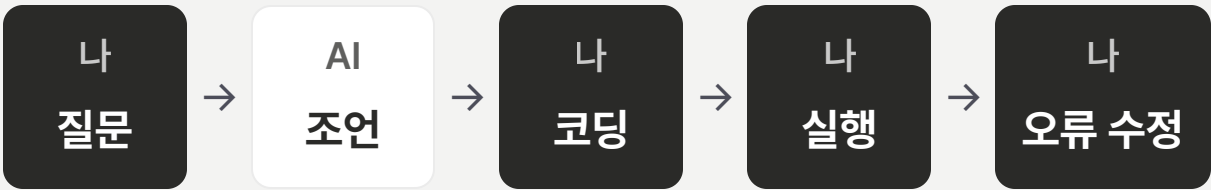
생성형 AI와 함께 코딩하는 일입니다

요즘은 에이전틱 엔지니어링이라고도 부릅니다.

과거

내가 직접

AI에게 조언이나 도움을 받고
내가 직접 코딩합니다



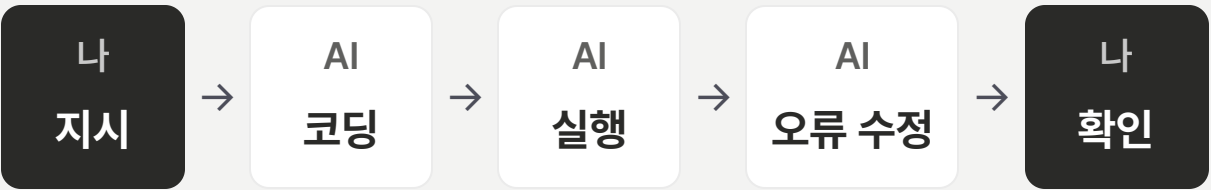
■ 사람이 하는 일 □ AI가 하는 일 · 코딩 · 실행 · 오류 수정이 **사람에서 AI로** 넘어갔습니다

활용의 자유도가 높고, AI가 내 PC를 열어 보고 다룰 수 있어서 **업무 자동화에 가장 잘 맞습니다.**
코딩을 몰라도 지시만 하면 됩니다.

현재

AI가 직접

지시하면 AI가 알아서 **직접** 코딩하고,
실행하고, 오류를 고칩니다



이름이 많아도 3가지로 나눕니다

바이브코딩 환경을 이해하기 위해 알아야 할 용어들입니다.

구분	OpenAI	Google	Anthropic	그 밖
생성형 AI	ChatGPT	Gemini	Claude	-
IDE (개발환경)	-	Antigravity IDE	-	VS Code, Cursor
바이브코딩 프로그램	Codex	Antigravity 2.0	Claude Code	-

이번 과정에서 쓸 도구

Claude Code (CLI)

VS Code 터미널에서 실행하고, 내 PC 작업을 직접 지시합니다.

코딩을 넘어 업무 전반에 씁니다



오늘

디지털 결과물 생성

업무 현황판, 데이터 대시보드,
미니 업무도구, 문서·양식 생성기



3회차

자동화·스케줄링

반복 업무를 자동으로
수행하도록 설정합니다



업무 Tool (ERP 등)

사내 업무 도구를
직접 만들 수 있습니다



오늘

문서 생성·편집

보고서, PPT, 문서를
자동으로 만듭니다



웹 탐색·크롤링

웹사이트에서 필요한 정보를
자동으로 모으고 작업을 수행합니다



PC 파일 열람·조작

파일 이름 변경, 정리, 변환을
자동으로 합니다

오늘은 «오늘» 2가지를 합니다. 대시보드와 기획서를 만들고, 저녁에는 **웹페이지** 하나를 만들어 배포합니다.

좋아지는 것은 3가지입니다



1 · 내 업무가 쾌적해집니다

파일 작업, 웹 검색을 모두 AI가 합니다. 필요하다면 간단한 웹페이지를 빠르게 만듭니다.



2 · 팀 자동화를 직접 만듭니다

대시보드, 자동화 프로그램을 우리 팀에 맞게 만들고 고칩니다. 협업과 소통 리소스를 줄이고, 실시간 데이터로 투명하게 결정합니다.



3 · 팀의 성공 확률이 오릅니다

직접 만든 시스템에 AI를 넣어 성능을 높입니다. 데이터로 예측해 결정의 품질을 높입니다.

셋 다 **내가 직접 만들고 고칠 수 있어서** 생깁니다.

능력이 아니라 **상황**으로 나눕니다

도구	언제	하는 일	이렇게 씁니다
클로드 챗	생각할 때	아이디어 · 초안 · 자료 요약을 묻고, 결과를 손으로 가져다 쓴다	"이 보고서 요약해줘"
클로드 코워크	가벼운 걸 맡길 때	메일 · 노션 · 시트에 붙여 글 한 편짜리 단순 작업을 끝낸다	"이 메일 정리해서 시트에 넣어줘"
클로드 인 크롬 크롬 확장 프로그램	브라우저 일을 맡길 때	크롬에서 사이트를 열고, 누르고, 입력한다 웹에서 하는 반복 작업을 대신한다	"이 사이트에서 표를 모아줘"
클로드 코드 데스크톱 앱	화면이 편할 때	클로드 코드를 앱 화면에서 쓴다 터미널에서만 되는 기능은 못 쓴다	오늘은 안 씁니다
클로드 코드 VS Code 터미널 · 오늘 여기서	만들고 실행할 때	내 PC의 파일을 읽고, 쓰고, 실행해서 작동하는 결과물 을 만든다	"이 엑셀로 대시보드 만들어줘"

기준은 한 줄입니다. 글 한 편이면 코워크, 작동하는 결과물이면 코드.

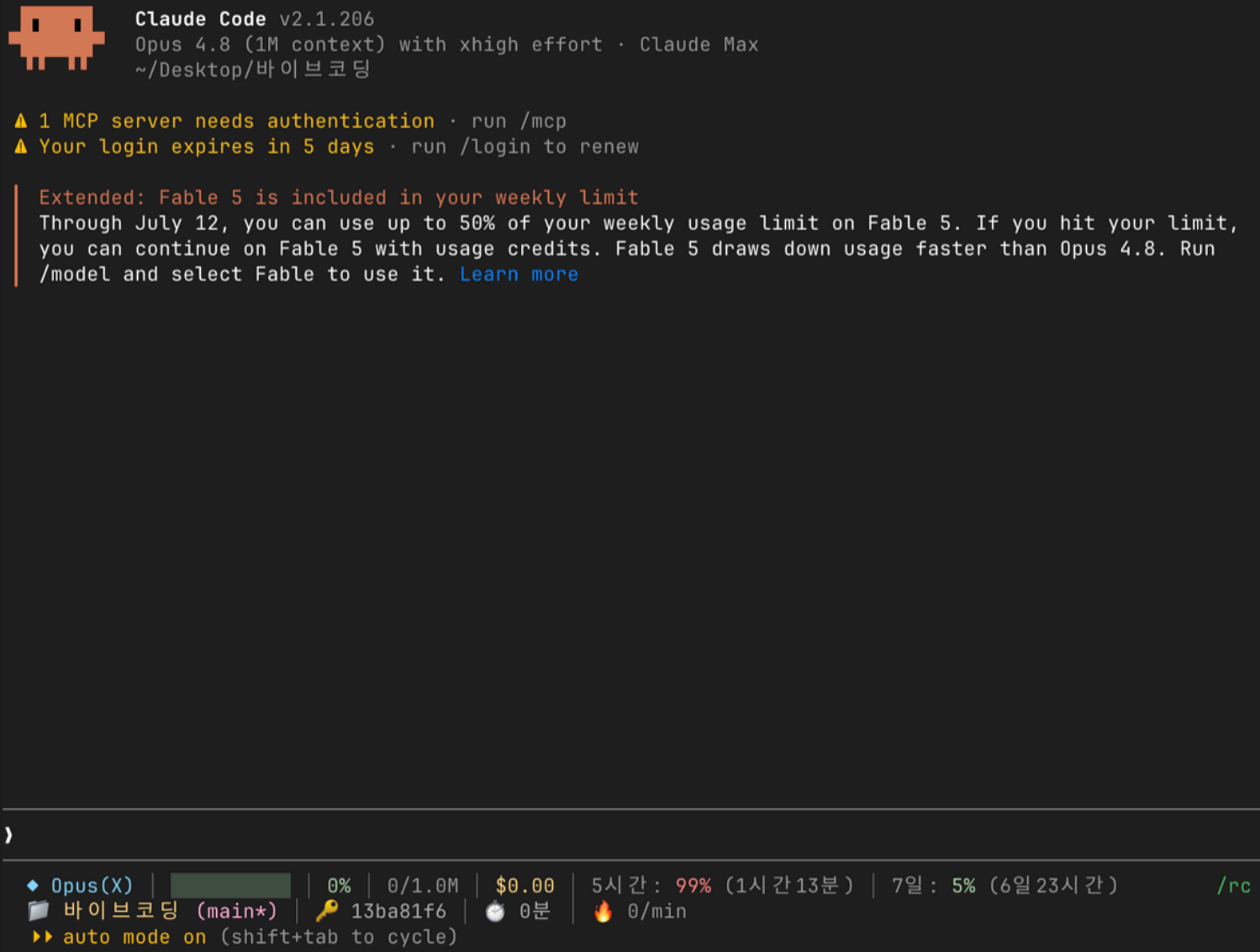
클로드 코드는 데스크톱 앱과 터미널이 같은 클로드 코드이고 여는 창만 다릅니다. 오늘은 **VS Code 터미널**에서 만듭니다.

터미널에서 일하는 AI입니다

Claude Code는 Anthropic이 만든 터미널(명령줄) 기반 AI 코딩 도구입니다. VS Code의 터미널에서 바로 실행할 수 있습니다.

파일	직접 읽고, 쓰고, 실행합니다
PC 작업	파일 관리, 프로그램 실행을 합니다
바깥 연결	웹 검색, 외부 도구와 연동합니다
프로젝트	전체를 이해하고 코드를 고칩니다

코딩을 몰라도 됩니다. 한국어로 지시하면 클로드 코드가 이 4가지를 합니다.



개발을 못 해도

개발을 못 해도 만들 수 있습니다

이미 절반이 코드 밖 일

누가 무엇을 정하나

무엇이 성공을 가르나

"개발을 못 하는데 제가 만들 수 있을까요?"

걱정하시는 것

코드를
써야 한다

명령어를 외우고, 문법을 배우고, 틀리면 어디가 틀렸는지 찾아야 한다고 생각하십니다.

오늘 실제로 하는 것

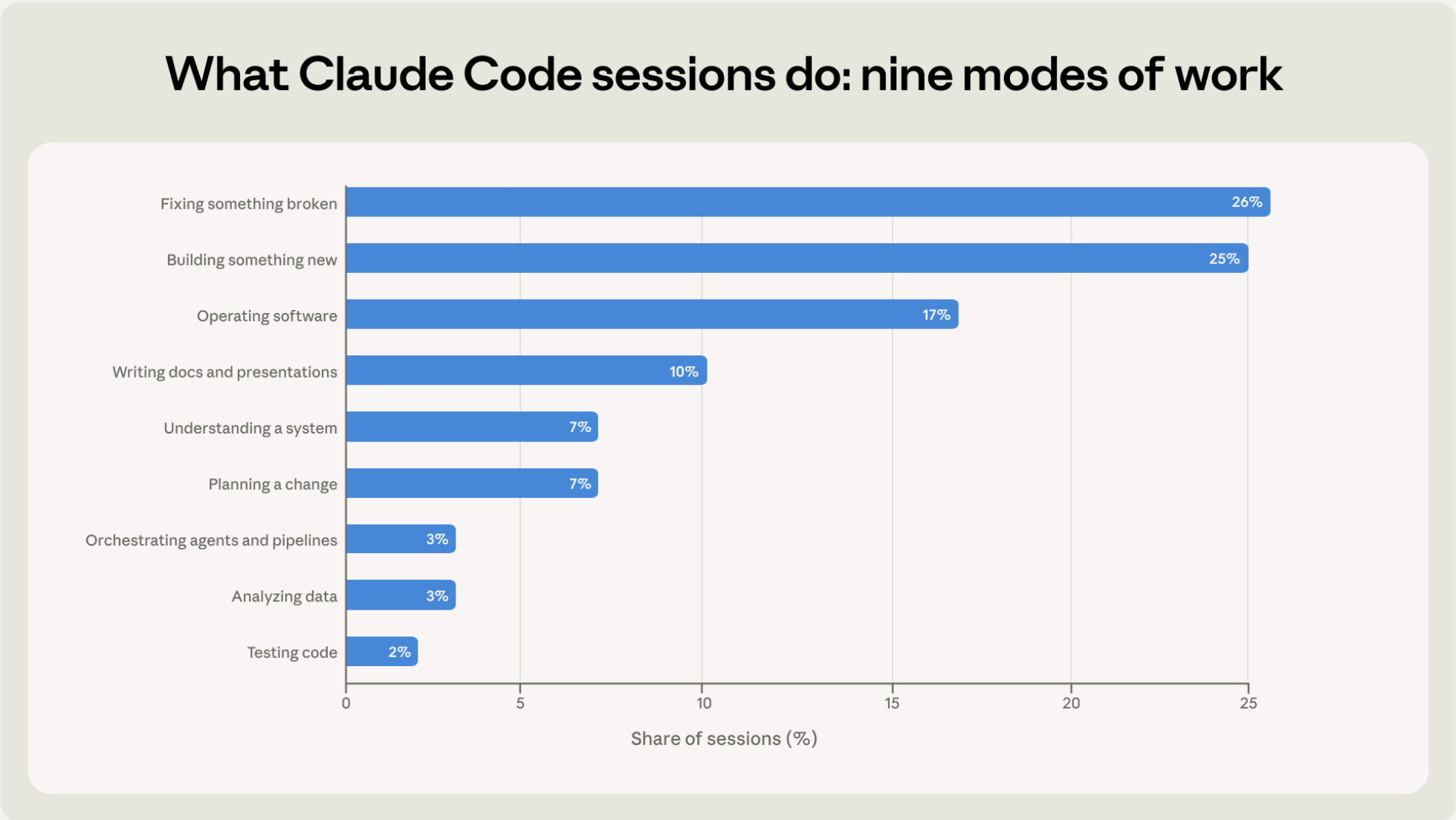
한국어로
시킨다

오늘 여러분이 입력하는 것은 전부 **한국어 문장**입니다. 코드는 클로드가 씁니다.

34% vs **29%**
개발자 개발이 아닌 직군

엔트로픽이 40만 건을 들여다봤더니 **끝까지 되는 비율이 5%p 밖에 차이 나지 않았습니다.**
개발을 배웠느냐는 거의 상관없었습니다.

이미 절반이 코드 밖 일입니다



코드 작업이 절반, 나머지 절반이 운영 17% · 문서와 발표자료 10% · 데이터 분석입니다. 문서·데이터·운영 업무도 이미 여기 들어 있습니다.

무엇을 할지는 사람이, 어떻게 할지는 클로드가

사람이 정합니다

무엇을 할지

무슨 일을 자동화할지

어떤 순서로 할지

어디서 멈춰 물어볼지

클로드가 정합니다

어떻게 할지

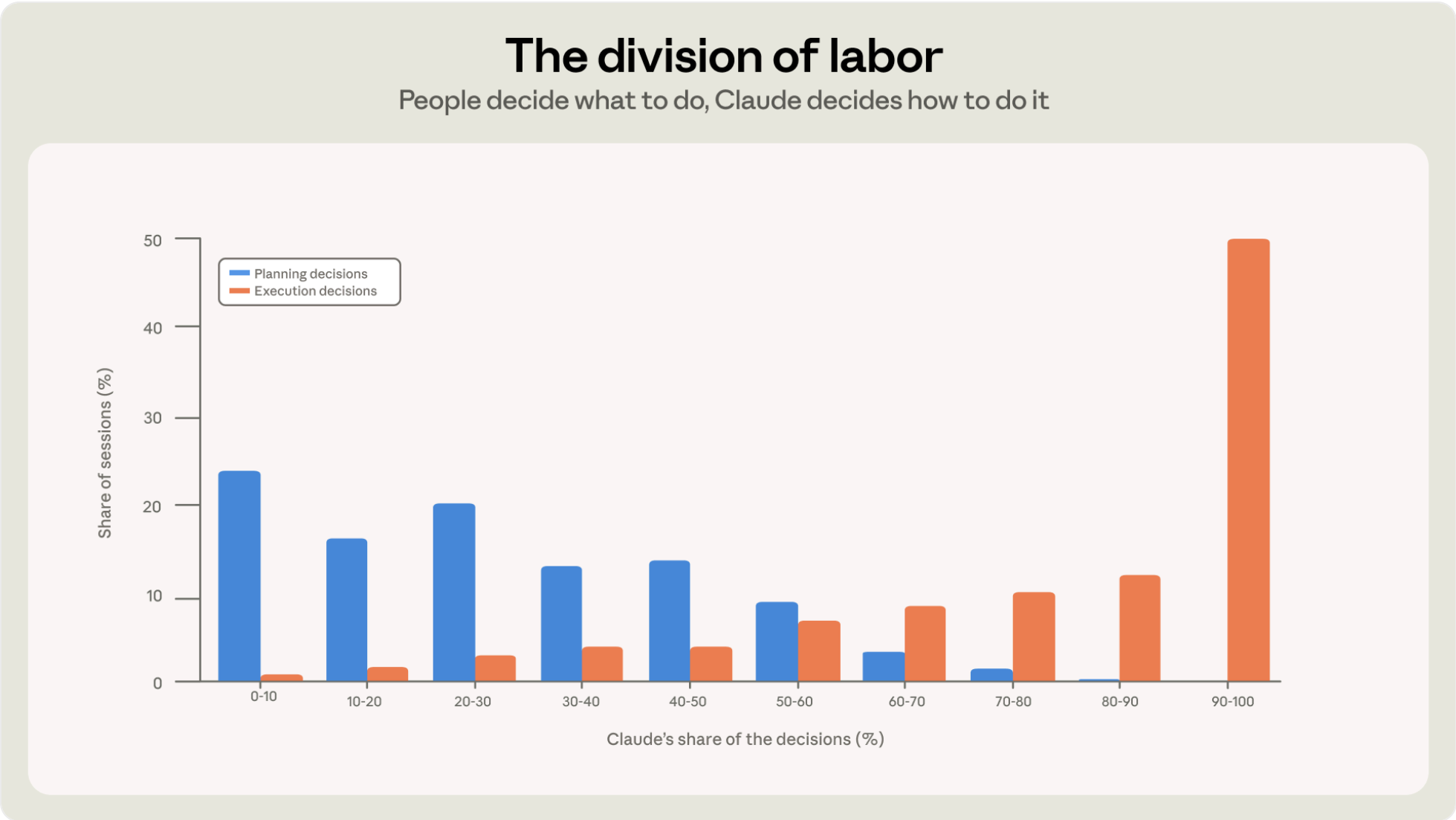
그 순서를 어떤 방법으로 처리할지

어떤 도구를 쓸지

그래서 필요한 것은 개발 실력이 아니라 **내 업무를 아는 것**입니다

여러분은 해운·물류 업무를 해 온 사람입니다. 클로드는 그 업무를 해 본 적이 없습니다.

기획은 왼쪽, 실행은 오른쪽에 몰립니다



가로축은 클로드가 내린 결정의 비율입니다. 기획 결정은 사람 쪽에, 실행 결정은 클로드 쪽에 뚜렷이 몰립니다. **방향을 잡는 힘이 곧 전문성입니다.**

전문성을 가져올수록 클로드가 더 많은 일을 제대로 합니다

+25%

7개월 동안 한 작업이 만들어내는 값이 평균 25% 올랐습니다.

오류 잡는 데 쓰던 비중은 절반으로 줄고, 그 자리를 분석·문서·실행이 채웠습니다.

모든 주요 직군이 소프트웨어 엔지니어와 거의 같은 성공률이었습니다.

코딩 훈련이 아니라 **내 일을 아는 정도**가 결과를 갈랐습니다.

클로드는 여러분의 전문성을 대신하지 않습니다. 더 많이 가져올수록 더 좋은 결과가 나옵니다.

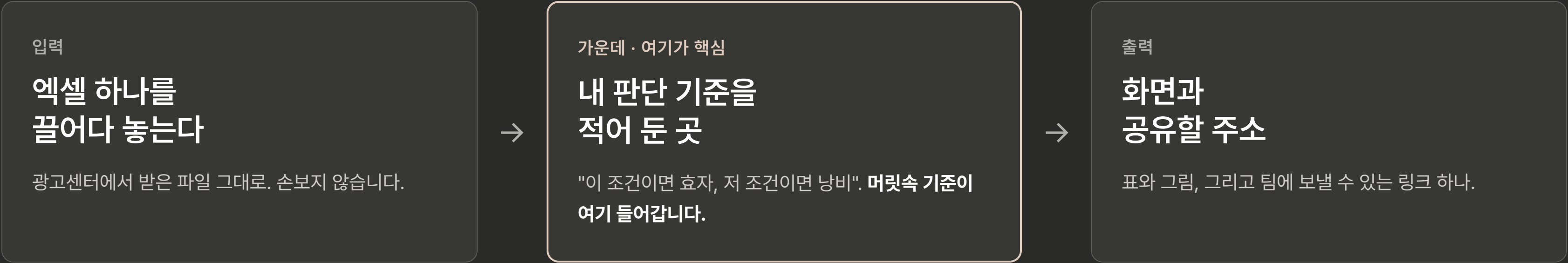
제가 매주 돌리는 것 5개입니다

지금 보시는 이 강의 자료도 4번째 것으로 만들었습니다. **형광 2개**는 잠시 뒤 화면으로 보여 드립니다.

무엇	어떻게	걸리던 시간
광고 리포트 분석기	웹앱 하나	1시간 → 3초
광고 통합 리포트	에이전트 하나	2시간 → 5분
영상 자동 제작	에이전트 여럿을 이어서	40시간 → 30분
강의 자료 제작	서브에이전트 여럿 + 메인 에이전트	40시간 → 1시간
링크드인 · 스레드 자동 배포	에이전트 7개가 매일 한 바퀴	매일 1회 · 자동

5개 다 제 업무를 제가 제일 잘 알아서 만들 수 있었습니다. 개발을 배운 적은 없습니다. **1번째가** 오늘 여러분이 만드는 웹페이지와 같은 종류입니다.

3칸이 전부입니다



엑셀만 바꾸면 여러분 업무의 분석기가 됩니다. 구조가 같기 때문입니다.

제가 친 코드는 0줄입니다

한국어로 지시하고, 나온 것을 검수했습니다. 그게 전부입니다.

순서	무엇을 했나	실제로 친 말
1	불편한 데를 적었다	"숫자가 수백 줄인데 효자와 낭비를 못 가르겠다"
2 ★	자료를 보여줬다	엑셀을 폴더에 넣으니 칸과 구조를 스스로 읽었다
3	한 줄로 시켰다	"6개 화면으로 만들어줘. 효자와 낭비를 자동으로 갈라줘"
4 ★	하나씩 보탤다	"지면 비교 화면 하나 더" · "급한 것부터 하는 목록도"
5	링크로 만들었다	끌어다 놓으니 팀에 보낼 주소가 나왔다

실제로는 이렇게 생겼습니다

```
쿠팡 광고 분석 에이전트/  
├ CLAUDE.md           규칙 파일  
├ src/  
│   ├── App.jsx       화면 연결  
│   ├── components/   화면 6개  
│   └── utils/  
│       └── adAnalytics.js 분석 엔진  
├ .claude/skills/      사용 안내  
├ 로우파일/           원본 엑셀  
└ dist/               배포 결과
```

노란 파일 하나에 제 판단 기준이 다 들어 있습니다.
"이 조건이면 효자, 저 조건이면 낭비."

★ 2번과 4번이 오늘 실습에서 그대로 나옵니다. 자료를 먼저 주고, 초안부터 만든 뒤 하나씩 쌓습니다.

에이전트 7개가 매일 한 바퀴 돕니다

에이전트마다 한 가지 일만 맡고, 결과물을 다음 에이전트에게 넘깁니다. 수집부터 발행까지 55분입니다.

순서	하는 일	언제	무엇을 하나
1	수집	07:00	뉴스 · 블로그 · 유튜브에서 48시간 분량
2	주제 선정	07:05	이미 쓴 주제를 거르고 원문을 읽는다
3 ★	글쓰기	07:05	작성 스킬과 playbook.md 를 읽고 쓴다
4	검증	07:05	사실 5항목과 서식. 못 미치면 다시 쓴다
5	발행	08:00	예약. 같은 글을 2번 올리지 않게 조회
6	발행 확인	08:20 · 09:10	올라갔는지 2번 확인한다
7 ★	결과 분석	매주 월 06:30	성과를 보고 playbook.md 를 다시 쓴다

실제로는 이렇게 생겼습니다

```
claude-watch/  
├── L00P.md           전체 순서  
├── collect.sh        ① 수집  
├── content-pipeline.mjs  ② ③ ④  
├── posted-topics.txt  이미 쓴 주제  
├── playbook.md       글쓰기 기준  
├── buffer_schedule.py  ⑤ 발행  
├── publish-watch.sh   ⑥ 발행 확인  
└── review-pipeline.mjs  ⑦ 결과 분석
```

노란 파일 하나가 매주 다시 쓰입니다.
7번이 고치고, 다음 날 3번이 읽습니다.

★ 3번과 7번이 이어져 있어서 매일 조금씩 나아집니다. 댓글 응답과 멈춘 발행을 되살리는 일은 사람이 맡습니다.

그래서 결론

해운·물류 업무를 가장 잘 아는 여러분이 **가장 잘 만듭니다**

"이 일을 잘하는 사람의 머릿속"을 글로 옮기는 것, 그게 바이브코딩의 전부입니다.

코드는 클로드가 씁니다. 여러분은 무엇을 · 왜 · 어떤 기준으로 할지를 압니다.

오늘은 그 머릿속을 꺼내 웹페이지 하나로 만듭니다.

쉬는 시간 · 09:50~10:00

10:00에 다시 시작합니다

다음 · 2교시 · 설치 · 작업 폴더 · 규칙 (노션으로 진행합니다)

1 시키는 AI · **2 설치** · 3 대시보드 · 4 프롬프트 · 5 업무 맥락 · 6 기획서 · 7 웹페이지 · 8 배포