

기초부터 스킴·커맨드·MCP까지

클로드 코드

시작하기

every._ai 지음

클로드 코드 시작하기

기초부터 스킬·커맨드·MCP까지

차례

CH 01	기초 — 챗봇이 아니라 직접 일하는 AI	1
	◆ 달라지는 것	2
CH 02	어디까지 되나 — 실제로 돌아가는 결과물	3
	◆ 달라지는 것	4
CH 03	사용법의 핵심 — 코딩이 아니라 명확한 요청	5
	◆ 달라지는 것	6
CH 04	커맨드 — 나만의 단축 지시어	7
	◆ 달라지는 것	8
CH 05	스킬 — AI 에게 주는 업무 매뉴얼	9
	◆ 달라지는 것	10
CH 06	하네스 — 스킬과 커맨드를 엮은 자동화	11
	◆ 달라지는 것	12
CH 07	실전 예시 — 전체 흐름을 스킬 하나로	13
	◆ 달라지는 것	14
CH 08	MCP — 바깥 도구를 끼우다	15
	◆ 달라지는 것	16

CHAPTER

01

기초 — 챗봇이 아니라 직접 일하는 AI

답을 주는 도구와 파일을 고치는 도구의 차이입니다.

챗봇과 클로드 코드

챗봇

답을 글로 주고 파일은 사람이
웁니다

클로드 코드

폴더를 열고 파일을 직접 고칩니
다

▲ 챗봇과 클로드 코드

STEP 01 / 08

챗봇은 질문에 답을 씁니다. 그 답을 파일에 옮기는 것은 사람이 합니다. 클로드 코드는 폴더를 열고 파일을 직접 고치고 저장까지 합니다.

	챗봇	클로드 코드
파일	사람이 옮깁니다	직접 고칩니다
범위	올린 파일만 봅니다	폴더 전체를 봅니다
결과	대화창에 남습니다	디스크에 파일로 남습니다

한국어로 “이 폴더 정리해줘”라고 쓰면 됩니다. 명령어를 외우지 않습니다. 무엇을 시킬지만 정하면 됩니다.

파일이 실제로 바뀌므로 결과를 눈으로 확인하고 다음을 시키면 됩니다.

◆ 달라지는 것

- 설명 대신 완성된 파일을 받습니다
- 코드를 아는 것보다 무엇을 시킬지 정하는 게 중요해집니다

CHAPTER

02

어디까지 되나 — 실제로 돌아가는 결과물

예제가 아니라 바로 쓸 수 있는 것이 나옵니다.

무엇을 만들 수 있나

랜딩페이지

홍보용 웹페이지가 만들어집니다

발표자료

슬라이드 파일이 나옵니다

배포된 서비스

인터넷 주소가 붙습니다

자동 리포트

정해진 시각에 만들어집니다

▲ 무엇을 만들 수 있나

STEP 02 / 08

예제가 아니라 바로 쓸 수 있는 결과물이 만들어집니다.

첫 결과물	한 문장을 쓰고 몇 분 안에 나옵니다
다듬기	"이 부분만 바꿔줘"를 몇 번 반복합니다
배포	주소가 나오기까지 보통 한 시간 안입니다

비개발자가 앱 5개와 AI 상담 서비스, SNS 홍보 자동화를 혼자 만들고 운영하는 사례가 있습니다.

✦ 달라지는 것

- 만들 수 있는지가 아니라 무엇을 만들지가 문제가 됩니다
- 외주를 맡기던 일을 직접 하게 됩니다

CHAPTER

03

사용법의 핵심 — 코딩이 아니라 명확한 요청

왜·누가·무엇을·어떻게 네 조각을 적습니다.

요청 한 문단의 네 조각

무엇 때문에 하는지
왜

무엇이 나와야 하는지
무엇을



▲ 요청 한 문단의 네 조각

STEP 03 / 08

좋은 결과는 좋은 요청에서 나옵니다. 왜·누가·무엇을·어떻게 네 조각만 적으면 결과가 달라집니다.

왜 매달 정산이 번거로워서. **누가** 나 혼자. **무엇을** 영수증 폴더를 읽어 월별 합계를 내는 표. **어떻게** 파일을 넣으면 금액과 날짜가 자동으로 채워지게.

처음 걱정하는 세 가지

결과가 매번 조금씩 다릅니다	“이 부분만 이렇게 바꿔줘”로 다듬으면 됩니다
에러가 납니다	그 메시지를 그대로 붙여넣으면 읽고 고칩니다
한참 멈춰 있습니다	몇 분씩 걸리는 작업이 있습니다

◆ 달라지는 것

- 프롬프트를 외우지 않고 요청을 정리하게 됩니다
- 막히는 곳이 기술이 아니라 설명이라는 것을 알게 됩니다

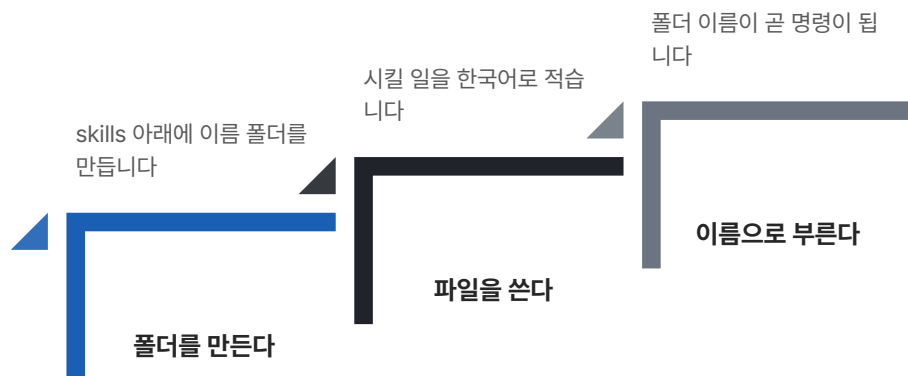
CHAPTER

04

커맨드 — 나만의 단축 지시어

자주 하는 일을 단어 하나로 부릅니다.

커맨드가 만들어지는 자리



▲ 커맨드가 만들어지는 자리

STEP 04 / 08

같은 지시를 매번 길게 쓰는 대신, 폴더에 파일 한 장을 둡니다. **이 파일이 커맨드입니다.** 폴더 이름이 곧 명령 이름이 됩니다.

```
~/claude/skills/정리/SKILL.md 모든 프로젝트에서 /정리
```

```
.claude/skills/정리/SKILL.md 이 프로젝트에서만 /정리
```

(파일 안) 받은 폴더의 파일을 날짜별로 옮기고 결과를 표로 알려줘.

이제 `/정리` 한 단어면 여러 단계가 끝납니다.

◆ 달라지는 것

- 매번 설명하던 것을 한 번만 적습니다
- 손에 익은 일이 단어 하나로 줄어듭니다

CHAPTER

05

스킬 — AI 에게 주는 업무 매뉴얼

한 번 써 두면 상황에 맞을 때 알아서 꺼내 씁니다.

스킬이 가벼운 이유

평소

이름과 한 줄 설명만 들고 있습니다

필요할 때

본문을 펴서 읽습니다

그래서

많이 만들어도 느려지지 않습니다

▲ 스킬 파일의 세 칸

STEP 05 / 08

앞 STEP의 파일에는 쓰임새를 적는 칸이 있습니다. `description` 입니다. 이 칸을 채우면 스킬이 되어, 내가 부르지 않아도 상황에 맞을 때 알아서 꺼내 줍니다.

name	폴더 이름과 같게 적습니다
description	어떤 상황에 쓰는지 적습니다. 이 칸을 보고 꺼냅니다
본문	실제로 시킬 일을 한국어로 적습니다

2단계 로딩이라 가볍습니다. 평소에는 이름과 한 줄 설명만 들고 있다가, 필요할 때 본문을 펴니다. 그래서 많이 만들어도 느려지지 않습니다.

예전에는 `.claude/commands/`의 커맨드와 `.claude/skills/`의 스킬이 따로였습니다. 지금은 둘 다 같은 방식으로 동작합니다.

✦ 달라지는 것

- 매번 시키지 않아도 같은 품질로 처리됩니다
- 설명을 파일에 두면 대화가 짧아집니다

CHAPTER

06

하네스 — 스킬과 커맨드를 엮은 자동화

여러 개를 이어 붙이면 한 번에 여러 단계가 갑니다.

하네스를 이루는 셋

전체 흐름을 사람 말로 적습니다

워크플로우

작은 조각 여럿이 안정적입니다

작은 도구들



에이전트

막히면 스스로 고쳐 다음으로 갑니다

▲ 하네스를 이루는 셋

STEP 06 / 08

스킬과 커맨드를 여러 개 이어 붙이면 한 번의 요청으로 여러 단계가 진행됩니다. 이렇게 엮은 것을 **하네스**라고 부릅니다.

한 번 요청하면	여러 단계가 이어서 진행됩니다
중간에 실패해도	원인을 찾아 다시 시도합니다
결과가 흔들리면	순서를 파일에 더 자세히 적습니다

판단과 실행을 나누는 것이 핵심입니다. 무엇을 할지는 사람이 정하고, 실행은 파일에 적힌 순서를 따르게 합니다. 그래야 매번 같은 결과가 나옵니다.

◆ 달라지는 것

- 한 번 요청하면 여러 단계가 이어집니다
- 결과가 매번 달라지는 문제가 줄어듭니다

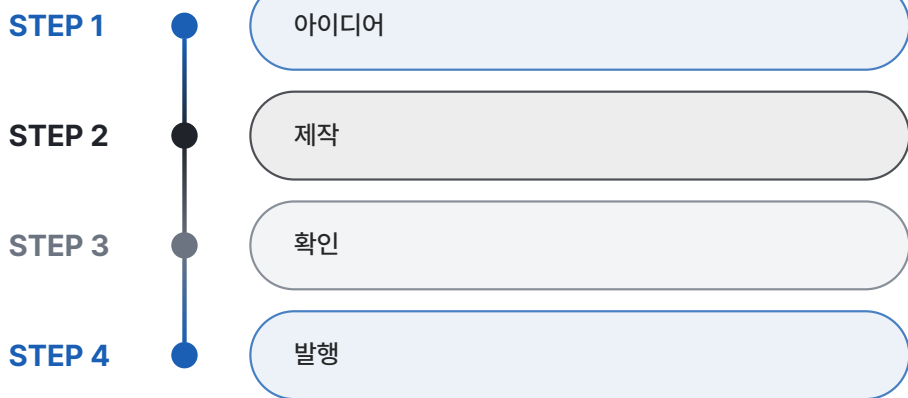
CHAPTER

07

실전 예시 — 전체 흐름을 스킬 하나로

콘텐츠 발행 과정 전부를 스킬로 짜 둔 사례입니다.

사람이 손대는 지점



▲ 아이디어에서 발행까지

STEP 07 / 08

콘텐츠를 만들어 올리는 과정 전부를 **스킬**로 짜 두면, 한 줄 아이디어에서 발행까지 이어집니다.

아이디어 한 줄 → 대본 → 화면 → 목소리
→ 사람이 한 번 확인 → 영상 → 자막 → 발행

사람이 손대는 곳은 가운데 한 번뿐입니다. 워크플로우는 코드가 아니라 사람 말로 적힌 순서입니다.

◆ 달라지는 것

- 사람이 확인하는 지점만 남기고 나머지를 넘깁니다
- 정해진 시각을 붙이면 내가 없어도 돌아갑니다 (맥 `launchd`, 윈도우 작업 스케줄러)
- 순서를 바꾸려면 스킬 파일만 고치면 됩니다

CHAPTER

08

MCP — 바깥 도구를 끼우다

내 컴퓨터 밖의 도구와 자료에 연결하는 규격입니다.

MCP 로 연결하는 것

깃허브

코드와 이슈를 읽고 씁니다

데이터베이스

자료를 직접 조회합니다

모니터링 도구

오류와 지표를 가져옵니다

디자인 도구

디자인 변경을 반영합니다

▲ MCP 로 연결하는 것

STEP 08 / 08

지금까지는 내 컴퓨터 안의 파일이었습니다. **MCP** 는 바깥의 도구와 자료에 연결하는 규격입니다.

API	한 가지 서비스를 빌려 씁니다
MCP	여러 도구를 한 규격으로 연결해 꺼내 씁니다

연결하면 붙여넣지 않아도 직접 읽고 씁니다. 데이터베이스에 질문하고, 이슈를 가져오고, 여러 시스템을 한 대화에서 다룹니다.

```
claude mcp add --transport http <이름> <주소>
```

◆ 달라지는 것

- 붙여넣던 자료를 직접 읽어 옵니다
- 도구를 늘리면 할 수 있는 일이 늘어납니다

클로드 코드 시작하기 — 기초부터 스킴·커맨드·MCP까지

지은이 every_ai

초판 1쇄 발행 2026-09

펴낸곳 bookforge · 조판 bookforge 자동 조판 파이프라인

본문 서체 Pretendard · 표제 서체 Pretendard

이 책의 내용은 조사 시점 기준이며, 인용·수치는 본문 표기 출처를 따릅니다.