

AI 사용법에 정답은 없다.

"프롬프트보다 사고력이 중요하다"

2022년부터 AI빅테크기업에서 근무하며 LLM엔지니어로 일하면서 지금까지 AI주도개발 트렌드를 쫓아가며 약 6,000시간에 걸쳐 생성AI와 대화하고, AI현업 환경에서 "모르겠다"라는 말과 계속해서 마주쳐왔다. 이에 그것을 해결해보자고자 정리한 내용이다.

INDEX

목차페이지

01	'모르겠다'를 분해할 수 있는가? —————	4p
02	'언어화'해서 AI에게 던져라. —————	11p
03	'컨텍스트'가 모든 것을 결정한다. —————	21p
04	'교양(리버럴 아츠)'라는 부스터. —————	31p
05	'사고'의 외부화 엔진. —————	39p
06	'멘탈링'의 정체. —————	47p



'모르겠다'를 분해할 수 있는가?

Part1

멘탈링 현장에서 발견한 생성AI 활용의 진짜 출발점

✓ A씨의 이야기



A씨 프로필

- 비개발자 출신 경영자
- ChatGPT/Claude/Gemini 하루 3시간 활용
- 세션 예약은 꼬박꼬박
- 의욕은 넘친다.

매번 반복되는 패턴

"쌤, 이게 뭐예요?"
→ "AI한테 물어보세요"
→ 다음 세션: 똑같은 질문
→ "역시 모르겠어요"

핵심 발견

문제는 'AI 사용법을 모른다'는 게 아니다
-> '자신이 무엇을 모르는지' 언어로 표현하지 못하는 것이다

✓ '모르겠다'의 3가지 레벨



레벨1

무엇을 모르는지 조차 모른다

"이게 뭐예요?" — 눈앞의 정보에서 어디서부터 손을 대야 할지 모른다. AI에게 던져도 시도 곤란하다.

레벨2

무엇을 모르는지 는 말할 수 있다

"Docker 네트워크가 모르겠어요" — 키워드는 나오지만, 이해의 경계선이 안 보인다. AI의 답을 읽어도 "알 것 같기도 하고"로 끝난다.

레벨3

무엇을, 왜 모르는지 모두 말할 수 있다

"Docker 브리지 네트워크에서 ARP 응답이 안 돌아온다" — 경계가 명확. AI가 핀포인트로 답한다.

✓ 레벨 간 격차 — '태평양'만큼 다르다

레벨	질문방식	AI답변 품질	해결 시간
레벨 1	"모르겠어요"	AI도 답 못함	영영 해결 안 됨
레벨 2	"X가 모르겠어요"	일반론 반환	수 시간 ~ 수 일
레벨 3	"X의 Y가 Z라 모르겠어요"	핀포인트 답변	수 분

✓ 현장: 레벨 1 → 2로 이행시키는 대화



A씨: "선생님, 이게 뭐예요?"

멘토: "뭐가 보여요?"

A씨: "화면에 영어가 잔뜩..."

멘토: "맨 위에 뭐라고 적혀 있어요?"

A씨: "Error: connection refused"

멘토: "connection + refused = 접속이 거부됐다. 그럼 왜?"

→ "모르겠다"가 "접속이 거부됐다. 왜?"로 바뀌었다: 레벨 1 → 2 이행 성공

✓ 이걸 'AI 사용법'이 아니다



가르치지 않은 것 x

- AI 조작 방법
- 프롬프트 요령
- ChatGPT/Claude 편의 기능

실제로 한 것 ✓

- '모르겠다'를 분해하는 도움
- 정보를 하나씩 언어화하는 연습
- 다음 질문 세우는 법

**이건 사고 정리법이다: AI 사용법이 아니다.
하지만 이게 되면 AI 사용법은 저절로 몸에 밴다.**

✓ 왜 언어화가 어려운가

한국 문화의 특성

- 눈치 보라
- 알아서 해
- 이심전심
- 눈치껏

생성AI의 특성

- → 눈치를 보지 않는다
- → 이심전심이 없다
- → "그러니까 뭐가요?"
- → 말로 안 하면 아무것도 안 된다

생성AI를 제대로 쓰기 위해 가장 먼저 필요한 것

AI 지식 X → 언어화 능력 ✓

✓ '가르친다'의 한계



세미나 (1:100)

"자신이 무엇을 모르는지 언어화합시다"
→ 아무것도 일어나지 않는다
방법 자체를 모르기 때문

멘탈링 (1:1)

"뭐가 보여요?"
"어느 부분이요?"
"그게, 뭘 것 같아요?"
→ 하나씩 언어화 연습이 가능

**레벨 1 → 2 이행은 강단에서 가르칠 수 없다.
옆에 앉아서 함께 하는 것만이 답이다.**



'언어화'해서 AI에게 던져라. 구조화는 AI의 일

Part2

컨텍스트 엔지니어링 × 언어화 능력 | IT 현장 경력 26년

✓ 가장 중요한 것은?



언어화해서 AI에게 던져라.
거칠어도 된다. 정리는 AI가 해준다.
돌아온 것에 논리적으로 파고들어라.

01

언어화

거칠어도 된다.
머릿속을 말로 꺼낸다.

02

AI에게 던진다

컨텍스트째로
AI에게 넘긴다

03

파고든다

논리적으로 반론.
루프를 3번 돌린다

✓ 언어화 = 컨텍스트 설계



하는 일은 같다. 다른 건 '받는 쪽'뿐이다.

단계	언어화 (대 인간)	컨텍스트 설계 (대 AI)
1.인식	자신이 무엇을 알고 있는지 파악	태스크에 무엇이 필요한지 판단
2.언어화	말로 변환한다	말로 표현해 AI에게 넘긴다
3.배치	상대가 이해할 수 있는 순서로	LLM이 처리하기 쉬운 순서로
4.최적화	불필요한 것은 말하지 않는다	노이즈를 제거한다

받는 쪽이 사람 → '설명을 잘한다' | 받는 쪽이 AI → '컨텍스트 설계를 잘한다'

✓ 사람이 하는 건 1단계와 2단계뿐

1 인식

담당: 사람
무엇이 중요한지
알아차리는 것

2 언어화

담당: 사람
거칠어도 좋으니
말로 표현하는 것

사람이 담당
언어화 능력이 결정적

3 배치

담당: AI
LLM이 처리하기
쉬운 순서로 정렬

4 최적화

담당: AI
노이즈 제거,
구조화 완성

AI가 담당
구조화·최적화는
AI가 더 빠르다

✓ '프롬프트 요령'이 얇은 이유

세간의 '프롬프트 요령'

- 3·4단계(배치·최적화)만 다룬다
- "이렇게 쓰면 전달된다"
- "이 형식이 좋다"
- 누구에게나 공통인 템플릿
- 표면적인 기술이다

이 강의의 '컨텍스트 설계'

- 1·2단계(인식·언어화)에 집중
- 당신 머릿속의 정보를 꺼낸다
- 현장 고유의 컨텍스트를 전달
- 사고력의 문제
- 템플릿으로는 해결 안 된다

**3·4단계는 AI에게 맡겨라.
사람이 해야 할 것은 1·2단계뿐이다.**

✓ 나도 처음엔 쓰레기를 던지고 있었다

처음 시도 (실패)

AI에게 넘긴 것:

"ETL 설계해줘" + 테이블 정의서 PDF

돌아온 것:

교과서적 일반론

반응: "쓸모없네. AI가 나쁜 거다"

3주후

현장 컨텍스트 추가 (성공)

추가한 컨텍스트:

인증 방식 차이 / 배치 시간 제약
전임자 출처 모를 컬럼 경위

돌아온 것:

토큰 갱신 전략 + 병렬화 설계

결론: 컨텍스트만으로 출력이 바뀐다

**컨텍스트의 차이만으로
같은 AI의 출력이 쓰레기에서 보물로 바뀐다**

✓ 실예: 테스트 계획서 - 현장 컨텍스트가 만드는 차이 🗨️ T 📌 🔍 🖼️ ⋮

AI가 내는 일반 리스트

- ETL 잡의 멍등성
- 타임아웃 시 리커버리
- 인증 토큰 갱신
- 타임존 불일치 리스크



현장 고유 컨텍스트

- 테스트 환경 = 운영의 1/10
- 벤더 A 타임스탬프 장애 전력
- 일정 2주 압축
- 개인정보 마스킹 미확정

= AI가 내는 특화된 리스크 분석

- "테스트 환경 스펙 차이로 성능 테스트가 운영과 괴리될 리스크"
- "일정 압축으로 회귀 테스트가 불충분해질 리스크"
- "벤더 A 전력을 고려한 타임스탬프 처리 집중 검증 필요"

차이를 만드는 건 도메인 지식이 아니라 현장 컨텍스트를 언어화하는 힘이다

✓ AI와의 반복 대화 루프 - 3번이면 충분하다



루프1

언어화 & 던지기

현장 컨텍스트를 거칠게 쏟아낸다.
AI에게 통째로 던진다.

루프2

논리적으로 파고들기

돌아온 아웃풋의
논리적 허점을 지적한다.

루프3

심화 & 확정

수정판에 다시 파고든다.
3번이면 혼자 못 가는 수준에 도달.

파고들기 1: "이 리스크 평가, 벤더 A의 장애 전력이 빠졌잖아"

[수정판 반환]

파고들기 2: "카나리아 테스트가 우리 배포 프로세스에서 가능한가?"

[제약 포함 재설계]

파고들기 3: "마스킹 미확정인데 테스트 데이터 얘기는 무슨 아닌가?"

[논리 정합성 수정]

✓ 도메인 지식보다 필요한 3가지



01

언어화력

"무엇을 모르는지"를
말로 표현할 수 있다

02

로지컬 씹킹

AI의 답변을 읽고
다음에 어디를 파야 할지
논리적으로 판단할 수 있다

03

AI 반복 대화력

1번의 질문으로 끝내지 않고
AI의 답을 바탕으로
더 깊은 질문을 던질 수 있다

도메인 지식이 있으면 1발로 간다.

없어도 3~4번의 반복 대화로 같은 곳에 도달한다. 그게 80점이다.

✓ 가장 중요한 메시지



**질문의 질 =
아웃풋의 질**

답은 AI가 낸다. 지식도, 검색도, 구조화도 AI가 한다.
하지만 '무엇을 물을지'만은 사람이 결정할 수밖에 없다.

1년차 신입이라도,
언어화 + 로지컬 씽킹 → 빠르게 성장 가능

26년차 경력자라도,
'뭔가 그럴싸하게' → 경험을 살리지 못한다.



'컨텍스트'가 모든 것을 결정한다. 거울의 법칙과 격차의 정체

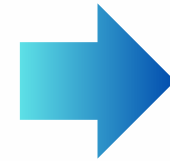
Part3

Andrej Karpathy · OECD · IMF · IEEE · 26년 현장 경험

✓ 생성시는 거울이다

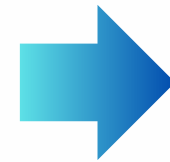
비치는 깊이는,
앞에 서 있는 사람의 깊이로 결정된다.

얕은 질문



얕은 답

깊은 컨텍스트



깊은 통찰

LLM의 출력 품질은 입력의 품질에 수학적으로 의존한다

✓ 컨텍스트 윈도우 = 작업 메모리(RAM)

CPU = LLM 본체

Claude Opus 4.6
GPT-5.2

최고 성능의 모델이라도...



RAM = 컨텍스트 윈도우

최대 100만 토큰
(책 수십 권 분량)

← 여기에 무엇을 넣느냐가 전부

RAM에 쓰레기 → 출력 쓰레기

RAM에 올바른 컨텍스트 → 놀랍도록 정확한 답

"LLM은 컨텍스트 윈도우에 들어 있는 정보만 쓸 수 있다" — Andrej Karpathy

✓ 같은 SI, 같은 요금 - 왜 가치가 이렇게 다른가



01

B씨 (2년차·영업)

입력:

"신규 사업의 사업계획서를 만들어주세요"

출력:

일반적인 템플릿
시장분석·타겟·KPI

개의 제로
검색으로도 나오는 수준

02

C씨 (과장·기획 5년)

입력:

B2B SaaS, 제조업 100-500명,
재고관리 디지털화, 월 50만원 구독

출력:

제조업 특화 사업계획
시장규모·경쟁사·CAC·이탈률

수십만 원 가치
컨설턴트 초회 상담 수준

03

D씨 (부장·런칭 3회)

입력:

2,000자 컨텍스트
(교훈·실패패턴·인터뷰 결과...)

출력:

핀포인트 전략 분석
리스크 TOP3·검증 방법

수백만 원 이상
시니어 컨설턴트 수준

✓ '컨텍스트 때려넣으면 된다'의 진짜 의미



D씨의 위: 질문 설계조차 AI에게

"각 분야의 전문가 페르소나를 설정해서,
테스트 관점을 추려내라"

AI가 자동으로 도출한 7가지 페르소나:

- 데이터 파이프라인 담당
- DWH 설계 담당
- BI 담당
- 인프라 담당
- 보안 담당
- 운영 담당
- 업무 사용자

직접 지정하지 않았다

컨텍스트(프로젝트 자료)만 넘겼다.
AI가 읽어내고, 필요한 페르소나를 역산했다.

1.5시간 vs 2주일

같은 태스크, 같은 AI. 차이는 넣은 컨텍스트만이다.
→ 50배의 생산성 격차

컨텍스트의 질이 충분히 높으면, AI는 질문의 설계조차 대신할 수 있다

✓ DECISION.md — 컨텍스트가 자동으로 쌓이는 구조

- 01 녹음 → 텍스트 변환
회의를 녹음하여 트랜스크립트 추출
- 02 시로 회의록 작성
트랜스크립트를 시에게 먹여 의사록 생성
- 03 결정 사항 추출
회의록에서 토픽별 결정 사항을 뽑아낸다
- 04 모순 감지 & 최신화
동일 토픽의 과거 결정과 모순이면 최신을 정보으로
- 05 DECISION.md에 집약
SSOT(단일 정보원)로 프로젝트의 진실 관리

컨텍스트의 질은 단발 입력으로 결정되지 않는다. 일상의 구조로 결정된다.

✓ SSOT가 없는 조직에서 일어나는 일



악순환

결정 사항이 흩어진다

- '이전에 어떻게 됐지?' 확인 회의
- 결과 공유 회의가 또 생긴다
- 더욱 흩어진다
- 반복...

회의의 실태

논의의 대부분:

"데이터를 보면 이렇게 되겠죠"
"사양상 이렇게 할 수밖에 없죠"

- **DECISION.md**를 AI에게 넘기면
5초에 나오는 결론

사실 확인은 AI에게 시켜라.

사람은 합리성만으로는 결정할 수 없는 판단만 해라.

✓ 연구가 뒷받침하는 것



컨텍스트의 질이 AI 출력 품질을 결정한다 — 데이터로 실증된 사실

1 Chroma Research

Context Rot

입력이 길어질수록 성능이 불안정해진다.
정보의 위치(앞/중간/뒤)와 구조화 여부로 정확도가 달라진다.

2 Stanford · Liu et al.

중간 소실 효과

LLM은 긴 컨텍스트의 '중간 부분' 정보를 놓치기 쉽다.
앞·뒤는 잘 잡히지만, 가운데 정보는 무시되기 쉽다.

3 IBM Research (2025)

양보다 질

"많으면 좋다"가 아니다.
쓰레기 정보를 채우면 오히려 품질이 떨어진다.
"올바른 것을, 올바른 양만큼"이 원칙.

4 Shopify CEO Tobi Lutke

컨텍스트 엔지니어링

"LLM이 태스크를 해결할 수 있도록, 모든 컨텍스트를 제공하는
아트.
그것이 이 스킬의 핵심이다"

✓ OECD · IMF가 보여준 격차 확대

정형 태스크 (AI로 저성과자 상향)

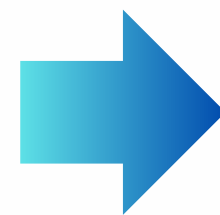
- ✓ 이메일 문구 교정
- ✓ 정형 포맷 작성
- ✓ FAQ 대응

비정형 태스크 (고성과자만 이익)

- 전략 수립
 - 복잡한 문제 분석
 - 조직 설계
- 비정형 = '무엇을 물을지'를 판단하는 능력이 필요

IMF 2026 보고서: "청년층 고용은 AI 대체 리스크가 가장 높은 직종에 집중
격차는 축소가 아니라 확대된다"

AI 이전
1:3



AI 이후
1:100

✓ 가장 중요한 메시지



거울 앞에
서 있는
사람의 깊이

✓ 컨텍스트의 질 = AI 출력의 질 (수학적 의존)

✓ 도메인 지식보다 언어화력 + 로지컬 씹킹

✓ 컨텍스트는 '만들기'가 아니라 '쌓이는 구조'를 만든다

! 격차는 AI 이후 1:3 → 1:100으로 확대 (OECD · IMF)

→ 다음 장: 교양(리버럴 아트)으로 '가로'의 깊이를 더하는 법



**'교양(리버럴 아트)'라는
부스터.
'가로의 깊이'를 내는 법**

Part4

리버럴 아트 × 생성AI × 연결하는 힘

✓ 왜 대기업 DX/AX는 이렇게 느린가? 산업구조로 3초 해체 🔍 🖼️ ...

일반적인 답변

- 경영층 IT 리터러시 부족
- 레거시 시스템이 발목
- 예산 부족
- 조직의 저항

AI도 똑같은 리스트를 낸다

교양적 답변 (산업구조론)

1·2차 산업
눈에 보이는 것이 주역
카이젠(개선)으로 세계 제패

3차 산업 (IT·서비스)
눈에 보이지 않는 것이 주역
→ '언어화하는 힘'이 필요해졌다
→ 피벗 능력이 없어 쓰러진 기업들

DX 문제를 'IT 투자'로 세로로 파면 이 구조는 보이지 않는다
'산업구조의 전환' x '인간의 인지 특성' — 가로로 연결했을 때 비로소 보인다

보너스: "넘을 수 없는 시련은 오지 않는다" → 통계학 개념 하나로 3초 만에 해체 = 이것이 가로로 뛰어넘는 것

✓ '언어화 × 로지컬 씽킹'만으로는 80점에서 멈춘다



언어화 + 로지컬 씽킹

= 세로로 직선 파기

"테스트 관점은?" →

"리스크는?" →

"우선순위는?"

도달점: 80점



교양적 답변 (산업구조론)

= 가로로 뛰어넘기

"이 문제, 전혀 다른 분야의 지식으로 풀 수 있지 않을까?"

산업구조론 × DX

통계학 × 자기계발

TCP × 정보 공유

도달점: 100점

✓ '제너럴리스트'에 속지 마라

구분	제너럴리스트	교양적 인재
정의	넓고 얇게 알고 있다	분야를 횡단해서 구조를 간파할 수 있다
특징	다영역 표층 지식	이종 분야 간의 깊은 연결을 이해
하는것	"알고 있다"	"연결할 수 있다"
AI시대 가치	낮다 얇은 지식은 AI가 위	높다 '연결 판단'은 AI가 못한다

얇고 넓은 지식은 이제 사람이 가질 필요가 없다. 연결하는 힘이 전부다.

✓ 두 가지 루트 - 바텀업(0→1)과 탑다운(1→10)



바텀업 (0→1)

AI가 후보를 내고, 자신이 선택한다

질문: "이 문제와 같은 구조를 가진 사례를 다른 분야에서 알려줘"

AI가 돌려준 것:

- 신경 전달 시냅스
- Bullwhip Effect
- TCP 재전송 제어
- 인지 편향

사람이 하는 것: "어느 것이 지금 상황에 맞는가" 판단

탑다운 (1→10)

지명해서 방향을 지정한다

예:
"팀 간 정보 공유, 이거 TCP 3방향 핸드셰이크와 같은 문제. 전송 확인 구조가 없다. 이 구조로 설계해줘"

결과: 단번에 핵심으로

정확도·속도가 차원이 다르다

✓ 왜 교양이 부스터인가 + Steve Jobs의 말



이유 1: AI의 래터럴은 '랜덤'

AI는 모든 분야에서 랜덤하게 후보를 낸다.
교양이 있으면 '이 문제는 ○○과 비슷하다'고 직감으로
어림잡아,
AI 출력을 평가하는 기준을 가질 수 있다.



이유 2: '맞는가' 판단은 사람만

조직 역학·팀 역량·일정 제약에 가장 맞는 것은
현장의 사람만이 안다.
교양이 있으면 '이게 맞다, 왜냐하면 ○○이기 때문'이라
고
즉시 판단할 수 있다.

Steve Jobs (2011 iPad 2 발표)

"기술과 리버럴 아트, 기술과 인문학이 결혼해야만 마음을 울리는 결과가 태어난다"

**브루킹스 연구소 (2022): "머신러닝은 판단·의사결정·해석에 아직 훨씬 부족하다.
이 스킬은 리버럴 아트가 제공하는 배움에 가깝다"**

✓ '연결하는 힘'을 단련하는 5가지 방법



1 다른 분야 책 읽기

"이거, 저 이야기랑 같지 않나?" — 연결하는 순간을 의식적으로 쌓는다

2 AI에게 아날로지 묻기

"이 문제와 같은 구조를 다른 분야에서 알려줘" — 바텀업 교양 훈련

3 문제를 뒤집기

"고객을 잃는 방법은?" — de Bono의 래터럴 씽킹. AI와의 반복 대화로 강화

4 일상 관찰 추상화

"교통 체증 = 큐잉 이론" "음식점 주문 = 비동기 처리" — 추상화 근력 훈련

5 '왜?'를 다른 분야로

3번 세로로 파면 → 가로로 뛰어넘는다. 경험 1개가 10분야의 지견으로

✓ 가장 중요한 메시지



교양은 '없어도
되지만,
있으면 폭발적
으로 빨라지는'
부스터다

✓ 80점 = 언어화 + 로지컬 씽킹 / 100점권 = +교양(가로외의 깊이)

✓ 바텀업(0→1): 시가 후보 → 자신이 선택 → 오늘부터 시작 가능

✓ 탑다운(1→10): 지명으로 단번에 핵심 → 정확도 · 속도가 차원이 다르다

✓ 컨텍스트 1 내면 시가 10 돌려준다 — 교양적 응용력이 폭발적으로 부스트된다

→ 다음 장: 생성AI의 '진짜 사용법' — 사고의 외부화 엔진



'사고'의 외부화 엔진. 생성AI의 진짜 사용법

Part5

워킹 메모리의 벽을 AI로 돌파하는 방법

✓ 두 가지 사용법 - 얇음 vs 진짜

얇은 사용법

"AI에게 생각해 달라"

기획서 작성 / 이메일 답장 / 코드 생성

- 단순 작업의 자동화
- B씨 수준에서 멈춘다
- 일반 템플릿이 조금 빨리 나올 뿐

진짜 사용법

"사고의 외부화 엔진"

답이 없는 물음을 끝없이 파고든다

- 워킹 메모리 벽을 돌파
- AI가 외부 저장소로 기능
- 자신도 몰랐던 구조가 보인다

Part2: 태스크를 완료시킨다 (완성물이 있다)

Part5: 답이 없는 물음을 끝없이 파고든다 (완성물이 없다)

✓ 워킹 메모리의 벽 — 뇌의 하드웨어 한계



조지 밀러의 '매직 넘버 7±2' (1956)

인간의 워킹 메모리가 한 번에 보유할 수 있는 정보
= 약 7개 청크 (최근 연구: 4청크 설)

- 복잡한 문제를 생각할 때 뇌가 오버플로
- '아까 뭘 생각하고 있었지'가 발생



워킹 메모리 용량

이게 '생각이 알아지는' 근본 원인
머리가 나쁜 게 아니다. 뇌의 하드웨어에 한계가 있는 것이다.

생성시가 이 벽을 돌파한다 — 시가 외부 저장소(RAM)로 기능한다

✓ 외부화 루프 — 워킹 메모리의 벽을 넘는 구조

01

1. 쏟아낸다

머릿속 생각을 말로 내뱉는다

02

2. 시가 보유

잊지 않고 전부 기억한다

03

3. 구조화

"A→B 인과, B·C 상관"
관계를 정리해 돌려준다

04

4. 새 시점 발견

자신도 몰랐던 구조가 보인다
1로 돌아간다 (레벨 상승)

05

5. 리터럴 점프

"이거 저 이야기랑 연결되네"
다른 영역으로 자유롭게

06

6. 레벨 업

자신도 몰랐던 구조가 보인다
1로 돌아간다 (레벨 상승)

시가 외부 저장소 = '아까 뭘 생각했지' 없음 → 깊이, 멀리까지 잠수 가능

✓ 실화 - 갯덴스시에서 2종류 AI를 동시에 돌린 날

내가 언어화한 것 — 2개의 물음뿐

1. "HDD가 컨텍스트 초과 문제를 해결한다는 주장, 개연성 높은가? GPT한테 물어봐"
2. "거대 브라운필드에서 설계서 역생성이 성립하는가?"



Claude (AI-1)

- 4개 매체(README·Velog·네이버블로그) 현황 조사 → 5초 보고
- GPT로의 프롬프트 설계 및 투입
- 기사의 병행 수정
- 프로덕트 버그 발견 및 수정
- 네이버블로그 브라우저 자동 업데이트
- 반응 모니터링 자동 설정

GPT-5.4 Pro (AI-2)

- 전략적 물음 1: 컨텍스트 초과 해결 주장의 개연성 심층 분석 (30분)
- 전략적 물음 2: 거대 레거시 역설계의 성립 가능성 분석

나는 갯덴스시에서 초밥을 먹고 있었다

✓ 물음의 질이 2종류 AI의 출력 품질을 결정한다



물음 1: "컨텍스트 초과 문제가 써 있지 않지 않나?"

왜 이 물음이 나왔는가:
21년간 레거시 코드 유지보수에서
'전체상이 누구의 머릿속에도 없다'는
지옥을 경험했기 때문에 나오는 물음

**물음 2: "거대 브라운필드에서
성립하는가?"**

왜 이 물음이 나왔는가:
레거시 시스템 이관 프로젝트에서
이론적 접근이 현장 규모에서 무너지는
순간을 몇 번이나 봤기 때문에 의심할 수 있다

현장 경험 없이는 이 물음은 나오지 않는다 — 물음의 질 = AI 출력의 질

✓ 외부화 루프를 돌리는 3가지 요령



01

'정리해줘' 말고 '위화감 지적해줘'

"논리적으로 약한 부분은?"

"모순되는 점은?"

"내가 놓친 시점은?"

→ AI에게 비판자 역할을 주면
자신의 사고의 구멍이 보인다

02

'갑자기 뛰어넘기'를 두려워 말자

"이거 전혀 다른 이야기인데 연결되네"

→ 바로 그쪽으로 뛰어넘어라
AI는 원래 문맥을 잃지 않는다
인간 상대라면 "지금 그 이야기
아니잖아"로
차단된다. AI는 차단하지 않는다

03

1세션으로 완결하려 하지 마라

깊은 사고는 1회 세션으로 완결
안 된다

→ '전 세션의 요약'을 컨텍스트
로 넘긴다

이것도 컨텍스트 설계다

→ AI는 지난번 계속부터 논의를
재개

✓ 가장 중요한 메시지



사람이 하는 것
은 물음을 세우
는 것뿐...

✓ 생성AI는 '사고의 외부화 엔진' — 워킹 메모리 벽을 돌파

✓ 물음이 있는 사람을 가속시킨다. 물음이 없는 사람을 구하지 않는다

✓ 2종류 AI 병렬 운용: Claude(실행) + GPT(전략적 사고) - 지시는 1단어

✓ 프롬프트 요령 제로 — 핵심을 찌른 물음 하나로 방향을 바꾼다

→ 다음 장(최종): 여기까지 전부를 통합 — '멘탈링의 정체'



'멘탈링'의 정체. 전부 연결됐다

Part6

6개 파트를 하나의 선으로 | 자신의 머릿속을 말로 표현하는 능력이 전부

✓ 6개 파트가 하나의 선으로 연결된다



01

'모르겠다'를 분해하는 힘
레벨 1→3 이행 / 언어화의 출발점

02

언어화해서 AI에게 던져라
구조화는 AI / 사람은 인식+언어화만

03

컨텍스트가 모든 것을 결정
같은 AI여도 1:100 격차 / DECISION.md

04

교양이라는 부스터
가로의 깊이 / 바텀업+탑다운

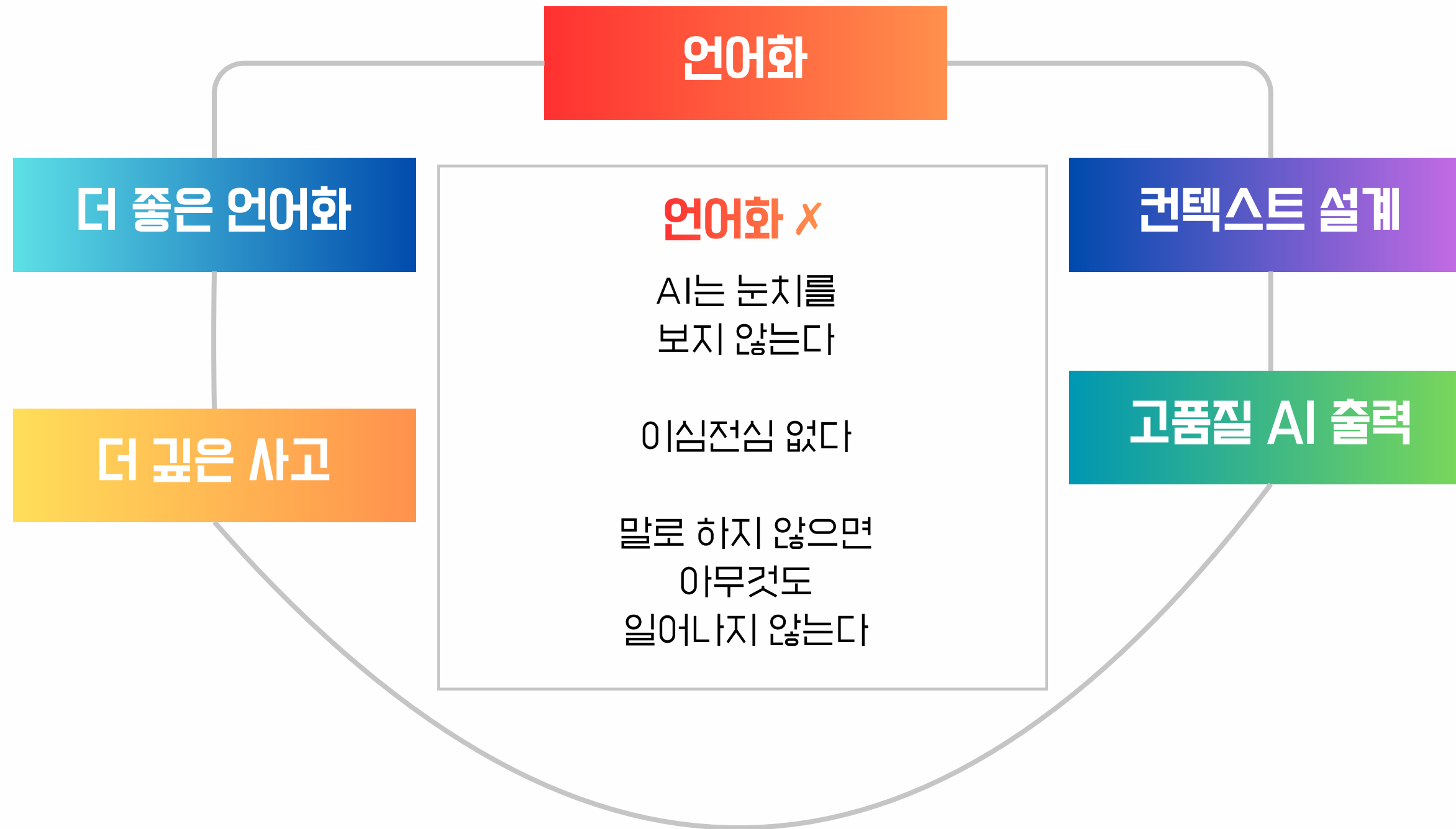
05

사고의 외부화 엔진
워킹 메모리 돌파 / 2종류 AI 병렬

전부 같은 것을 다른 각도에서 보고 있다

"자신의 머릿속을 말로 표현해서, 밖으로 꺼내는 능력" ← 이것이 모든 것의 근간

✓ 언어화가 되면 선순환이 시작된다



✓ 멘탈링의 정체 - 가르치는 것은 2가지뿐



1. 메타인지

자신이 무엇을 알고,
무엇을 모르는지를 구별한다

"알고 있다"고 착각하는 것과
정말로 알고 있는 것의
경계선을 긋는다

AI가 가르쳐 줄 수 없다
→ 옆에서 함께 해야만 가능

2. 언어화

모르는 것을 '무엇이 모르겠다'로
분해해서, 말로 표현한다

- 무엇을 모르는가
- 어디까지는 알고 있는가
- 왜 거기서부터가 모르겠는가

이 2가지가 되면
구조화·배치·최적화는 AI가 한다

1이 안 되면 2는 불가능 | 순서가 중요하다

사람이 하는 것은 '알아차려서, 말로 표현하는' 것뿐. 나머지는 AI

✓ 세상의 AI 교육이 어긋나 있는 이유

세상의 AI 교육	하고 있는 것	진짜 필요한 것
프롬프트 요령	배치·최적화를 가르친다	인식·언어화를 단련한다
ChatGPT 편리한 사용법	툴의 조작을 가르친다	언어화력을 단련한다
AI 리터러시 연수	AI의 개념을 가르친다	메타인지를 단련한다

**순서가 반대다 — 배치·최적화는 이제 AI에 맡길 수 있다.
사람 = 알아차리기 + 말로 하기**

✓ A씨의 그 후 - 6개월 후의 변화

6개월 전 (레벨 1)

"쌤, 이게 뭐예요?"

스크린샷 보여주며 질문
자신이 뭘 보고 있는지조차
말할 수 없었다



현재 (레벨 3)

분류 기준을 '한국표준산업분류에 준거해서'라고 재지정

AI 출력 어디가 이상한지
왜 이상한지
어떻게 고쳐야 하는지
→ 스스로 판단해서 AI에 지시

내가 가르친 것

- 눈앞의 정보를 하나하나 말로 표현한다
- 자신의 위화감을 언어화한다
- 언어화한 것을 AI에게 던진다

멘티 대부분이 3~6개월에 같은 변화를 보인다
언어화의 근력이 붙으면, AI 사용법은 저절로 몸에 밴다

✓ 오늘부터 시작하는 3가지 액션



01

매일 '오늘 한 일 설명'을 AI에게 말한다

퇴근 전 5분. "오늘 무엇을 했다 / 왜 그랬다 / 결과는"

"쓰지" 마라. "말해라" — 음성 입력으로 쏟아낸다
정리안 돼도 OK. AI가 구조화해 준다
→ 2주 계속하면 입도가 눈에 띄게 올라간다

02

'모르겠다'를 3단계로 분해하는 습관

1. "무엇에 대해" 모르겠는가? (대상 특정)
2. "어디까지는" 알고 있는가? (경계선 명확화)
3. "왜" 거기서부터가 모르겠는가? (원인 언어화)

→ 종이에 쓰기만 해도 머릿속이 정리된다

03

AI에게 '가르치는' 세션을 주 1회

"나는 이렇게 이해하고 있는데, 틀린 부분이 있으면 지적해줘"
→ 자신의 이해를 언어화 → 구멍이 발견된다
→ AI가 "여기가 부정확합니다"라고 가르쳐 준다

AI 사용법이 아니라 "자신의 머릿속 재고 정리"

✓ 가장 중요한 메시지



**순서가 반대면,
영원히
활용할 수 없다.**

- ✓ 생성AI 사용법 전에 → 자신의 머릿속을 말로 표현하는 방법을 배워라
- ✓ 프롬프트 템플릿 100개 < '자신이 뭘 모르는지' 알아차리는 연습 1시간
- ✓ ChatGPT 꿀팁 10개 < 자신의 일을 남에게 설명하는 연습 1회
 - "뭔가 걸리는 것이 있다면" → 지금 이 순간, 그것을 말로 표현해 보라
 - 지금 이 순간부터, 연습은 시작할 수 있다



**“자신의 머릿속을 말로 표현해서,
밖으로 꺼내는 능력”
이것이 모든 것의 근간이다!**

메타인지 → 언어화 → 컨텍스트 설계 → AI 고품질 출력 → 더 깊은 사고

이 순환이 돌아가면, AI는 비로소 '사고의 외부화 엔진'이 된다

이제 실천으로 활용해보세요!

<https://brunch.co.kr/brunchbook/aimetoring>

