



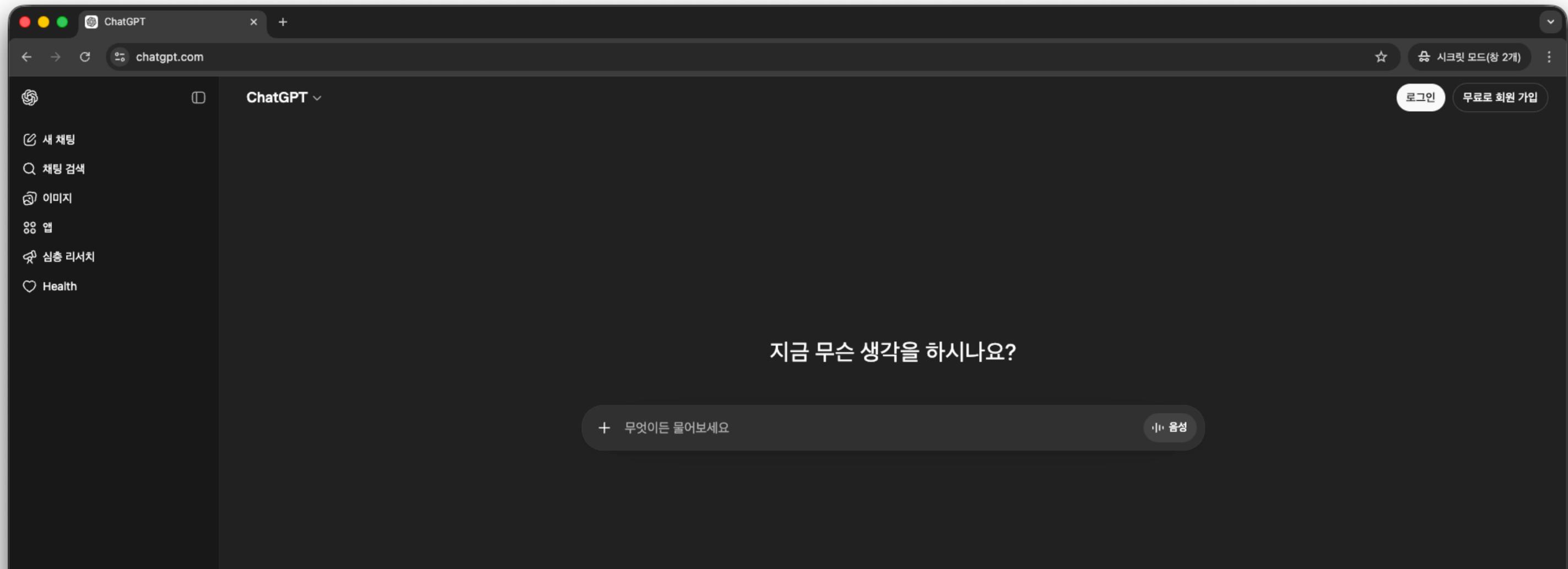
# 아직도 ChatGPT 웹에서만 AI를 사용하시나요?



Feat. 개발에서 AI 120% 활용 방법과 AI 세상에서 살아남는 방법



Kim Jun Young



# AI가 바꾼 개발/IT 패러다임

- 인공지능이 바꾼 IT 패러다임 (Feat. 나때는... )
  - 스택오버플로우로 보는 "검색 기반 개발"에서 "대화형 개발"로써의 변화
- AI와 LLM
- 어떤 LLM 모델이 좋을까?
- AI를 못 쓰는 개발자의 특징
- 효과적인 프롬프트 작성 방법 (Feat. AI는 생각보다 멍청하다)
  - 명확하게 요청하는 것
    - Context를 얼마나 주느냐에 따라 결과가 바뀐다. (목표, 상황, 문제, 제약)
  - 역할을 부여하자.
  - 검증은 필수다.
- Context 토큰 아끼기 (너무 큰 코드베이스 지양하기)
- Agent와 MCP 활용해보기
  - 어떤 Agent와 MCP를 사용해볼까?

## AI에게 일자리를 뺏기지 않는 방법

- AI는 보조인인가, 아니면 대리인인가?
- 코딩 실력으러 승부보는 시대는 이미 끝났다. 코딩 실력은 AI를 절대로 이길 수 없다.
- 대신 이해하고 검증할 수 있는 사람만이 살아남을 수 있다.

## AI를 얼마나 신뢰해야하나?

- AI는 구라를 너무나도 정성껏 깐다.
- "신뢰"의 기준이 바뀌었다.
- 신뢰하면 안되는 영역과 신뢰해도 괜찮은 영역의 3단계

## (소신발언 500%) OpenClaw, 써도 괜찮은가?

- 인공지능 비서, OpenClaw는 바이럴인가?
- Principle of Least Privilege(PoLP)은 AI에도 적용되어야 한다.
- 그리고 OpenClaw는 PoLP 개념과 상반된다.
- 내가 만약 총괄자라면..

## Github Copilot Education Plan 활용하기 (학생 혜택은 무조건 뽑아먹기)

- 너프를 먹었지만 그래도 쓸만한 수준 그 이상

AI/LLM에 대한 다소 개인적인 견해가 일부 포함되어 있을 수 있습니다.

해당 프레젠테이션 자료는 2026년 3월 기준입니다.

주제 특성상 트렌드가 매우 빠르게 변화될 수 있으니 마찬가지로 참고 바랍니다.

2026년 4월 1일, Null4U(C502)에서 계속...