



천재 개발자 되는 법

박성수 연구원 - 기산전자(주)

연사자 소개



박성수 - 기산전자 신제품 연구
2팀
(인공지능 연구원)

Education

- 2025 광운대학교 전자통신공학과 석사 (NeuroAI Lab)
- 2020 동양미래대학교 소프트웨어공학과 학사

Achievement

- 2025 특허 출원: 10-2025-0000439(심층 스파이킹 신경망의 시간 특징 인코딩 방법)
- 2024 대한의용생체공학회 - 우수 논문상
- 2021 도전 K-Start Up 국방부 부문 예선 공군창업 경진대회 1위(금상)
- 2020 오픈소스 해커톤 (군부문) - 2위 NIPA 원장상
- 2018 제16회 임베디드 소프트웨어 경진대회 자유부문 - 1위(금상)
- 2017 서울산업진흥원 주최 창업경진대회 - 1위(금상)

Publication

- **S.S. Park**, Y.-S. Choi, "Multiscale Spiking Neural Network with Dynamic Temporal Latency Encoding" in Journal of Biomedical Engineering Research, Feb. 2025.
- **S.S. Park**, Y.-S. Choi, "Dynamic Temporal and Encoding Scheme for Efficient Spiking Neural Network" Master's thesis, Dec. 2024.
- **S.S. Park**, Y.-S. Choi, "Spiking neural networks for physiological and speech signals: a review" in Biomedical Engineering Letters, vol 14, no. 4 Jun. 2024.
- **S.S. Park**, Y.-S. Choi, "Temporal Feature Encoding-based Deep Spiking Neural Network", Summer Annual Conference of IEIE, Jun. 2024.
- D.H. Kim, M.S. Lee, D.H. Lee, C.-M. Kim, **S.S. Park**, J.H. Son, Y.-S. Choi, et al., "Emotion Recognition based on Multimodal Graph Representation Learning of Senior Utterance", Summer Annual Conference of KOSOMBE, May. 2024.
- **S.S. Park**, Y.-S. Choi, "Advancing Temporal Spike Encoding for Efficient Speech Recognition" , IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia). IEEE, Nov. 2023.

천재 개발자는 누구일까?



회사 관점에서 “천재 개발자” = “高 생산성”

구글 사례

CEO 순다르 피차이

도구를 잘 활용하는
개발자들은 단순 작업을
줄이며, 창의적인 업무에 더
집중 할 수 있다고 언급

AI 도구가 코드 생산성을 10%
향상 시키고 있음

메타 사례

CEO 마크 저커버그

“빠르게 만들되, 안정적
인프라로”
메타의 슬로건에서 볼 수 있듯
생산성의 중요성을 언급

AI로 5배 빠르게 만들고,
가까운 시일 내 대부분의
코드는 AI가 작성할 것

테슬라 사례

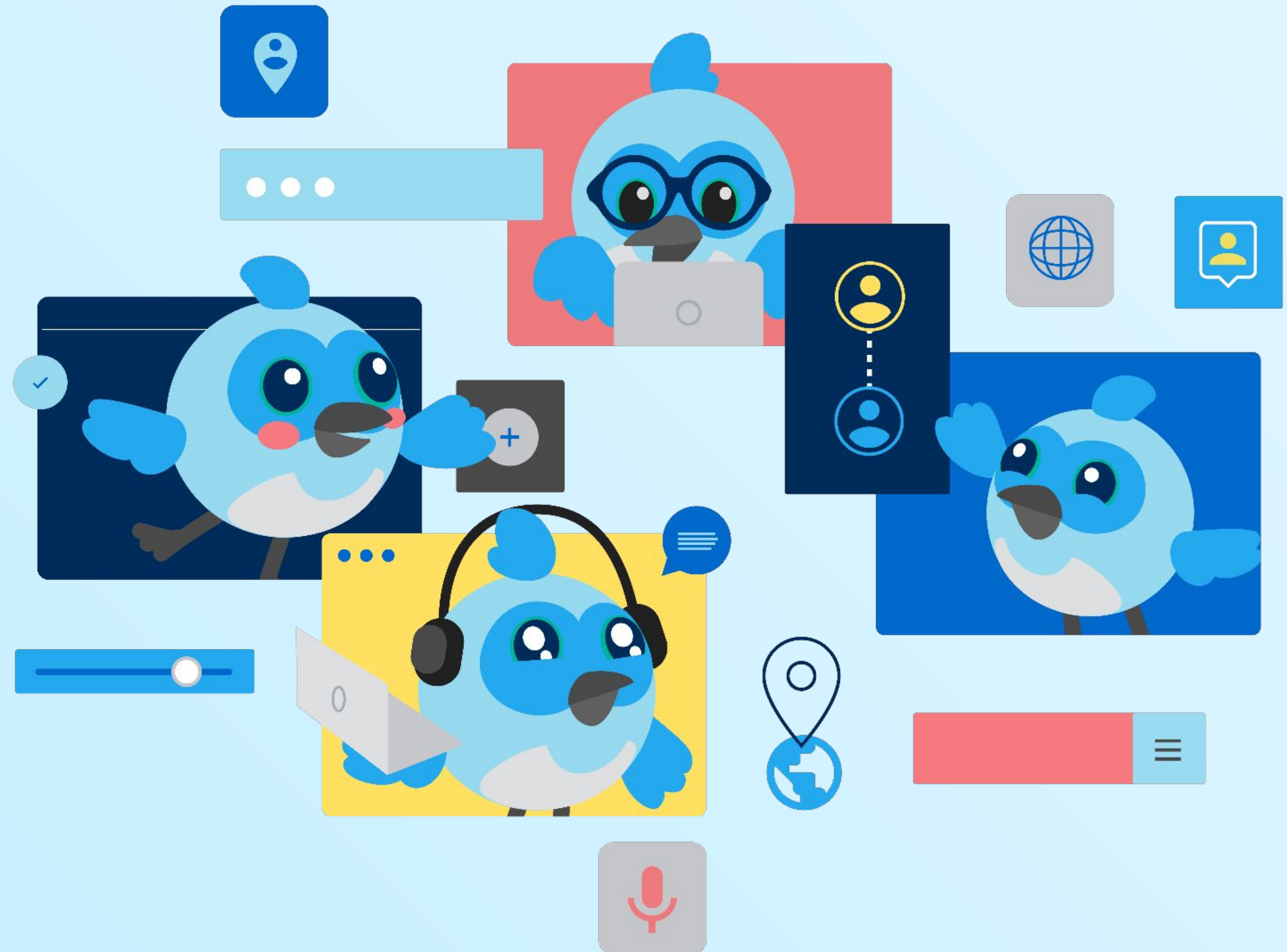
CEO 일론 머스크

업무에서 가장 중요한 것은
생산성을 높이기 위해 방해
제거하는 것

짧은 회의 지양, 큰 회의 금지,
기여가 없으면 즉시 퇴장 등
생산성 6원칙 제시

Agenda

- 1 연사자 소개
- 2 천재 개발자는 누구일까?
- 3 인공지능 도구와 생산성
- 4 AI 주도 개발
- 5 미래
- 6 Q&A



인공지능 도구와 생산성

- + AI Agent 개념
- + Cursor AI Editor
- + CLI Agent: Claude code
- + MCP (Model Context Protocol)
- + A2A (Agent to Agent)

인공지능

지능은 뭘까? AI와 인간의 지능의 차이점은 뭘까?

SYSTEM 1

Intuition & instinct

95%

Unconscious
Fast
Associative
Automatic pilot

SYSTEM 2

Rational thinking

5%

Takes effort
Slow
Logical
Lazy
Indecisive

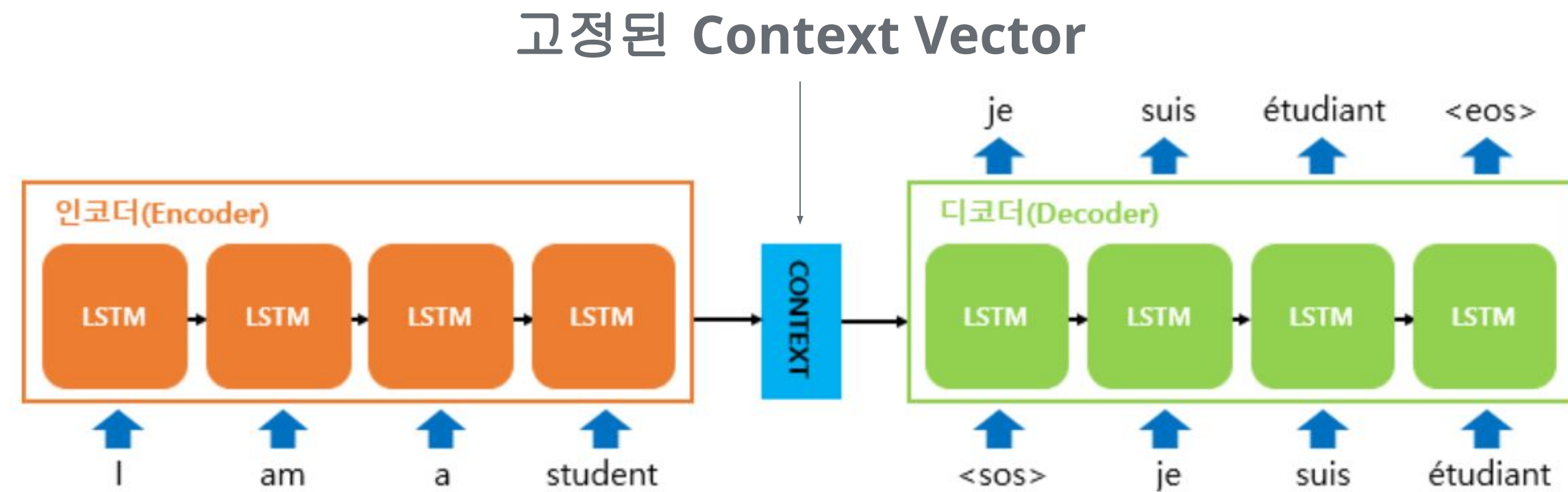


Source: Daniel Kahneman

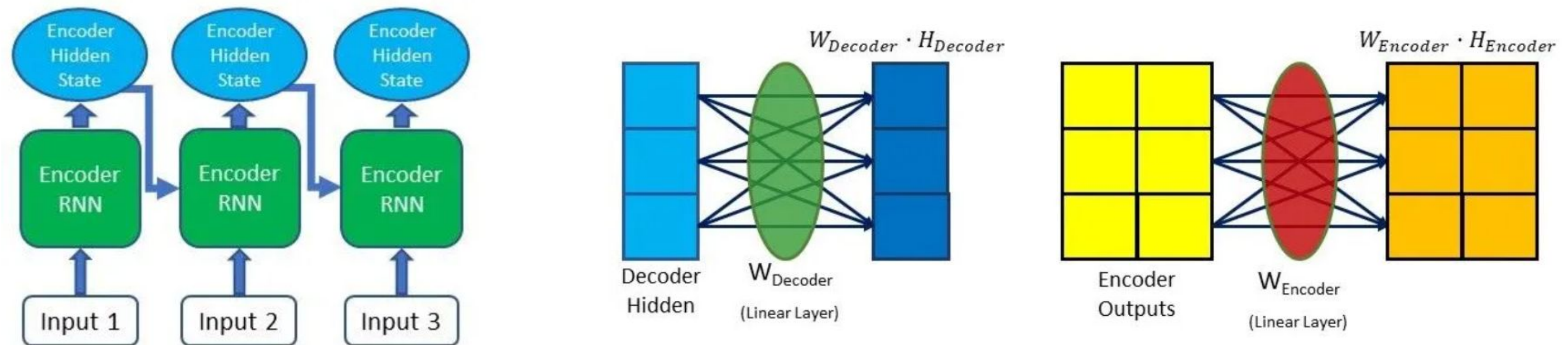
LLM의 기본 메커니즘

Context 문제를 해결한 Attention

Seq to Seq Model

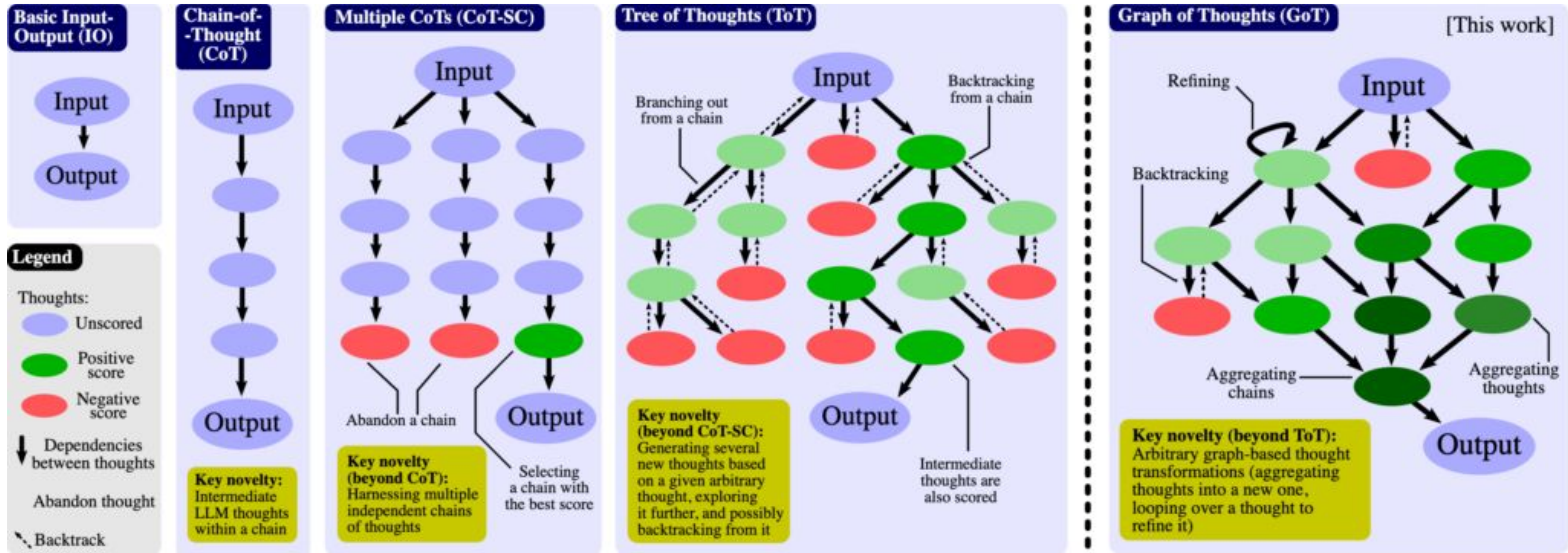


Attention



AI agent 구조

생각의 연쇄 (Chain of Thought)



Cursor Editor

VS Code에 AI 모델을 결합한 코드 편집기

product.dart cart_page.dart **main.dart** × cart_provider.dart

lib > main.dart > _ProductListPageState > build

```
39 class _ProductListPageState extends ConsumerState<ProductListPage> {
149   Widget build(BuildContext context) {
268
269     // 애니메이션 시작
270     _showAddToCartAnimation(context, product);
271
272     ScaffoldMessenger.of(context).showSnackBar(
273       SnackBar(
274         content: Text(
275           | '${product.title}이(가) 장바구니에 추가되었습니다!',
276         ), // Text
277         duration: const Duration(seconds: 2),
278       ), // SnackBar
279     );
280   },
281   child: const Text('장바구니에 담기'),
282 ), // ElevatedButton
283 ],

```

GitHub/flutter_icn_study/lib/models/product.dart

```
285 // Column
286 // Container
287 // InkWell
288 // Card
289 // Padding
290 // Center
291 cardsCount: _list.length,
292 // CardSwiper

```

장바구니 애니메이션 만들기

장바구니에 담기 버튼을 누르면 장바구니로 이동하는 듯한 모션의 애니메이션이 들어가면 좋겠어.

먼저 main.dart 파일을 수정하여 애니메이션을 추가하겠습니다.

main.dart +6 -1 •1 ✓

이제 _showAddToCartAnimation 메서드를 _ProductListPageState 클래스에 추가하겠습니다.

The model provided an ambiguous search string to replace.

파일에서 중복된 부분을 찾아서 정확한 위치를 확인하겠습니다.

Grepped @override (lib/main.dart)

_ProductListPageState 클래스의 build 메서드 앞에 애니메이션 메서드를 추가하겠습니다.

Read main.dart L145-244

이제 _ProductListPageState 클래스의 build 메서드 바로 앞에 애니메이션 메서드를 추가하겠습니다.

main.dart +25 •1 ✓

이제 AddToCartAnimationOverlay 클래스를 파일 끝에 추가하겠습니다.

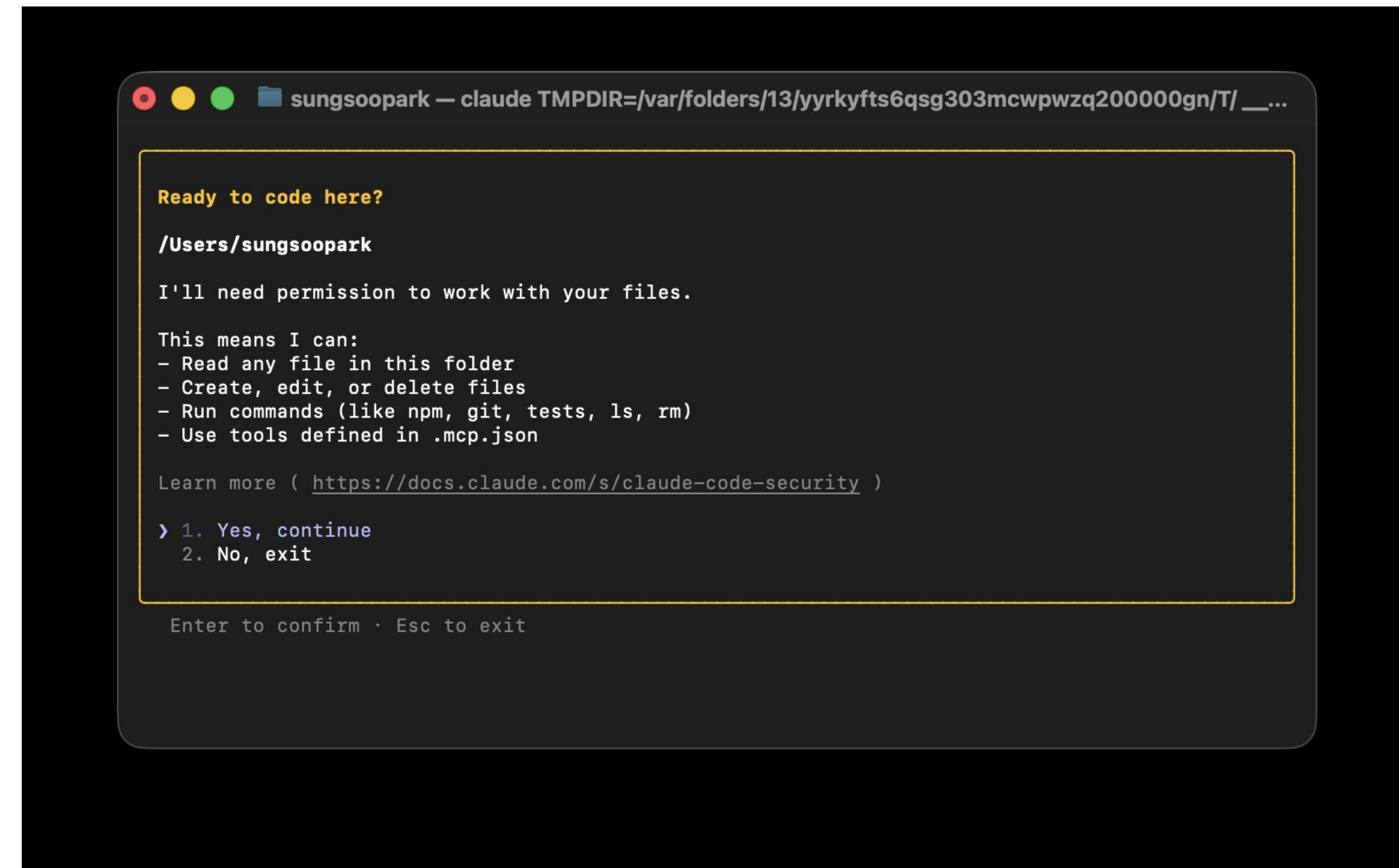
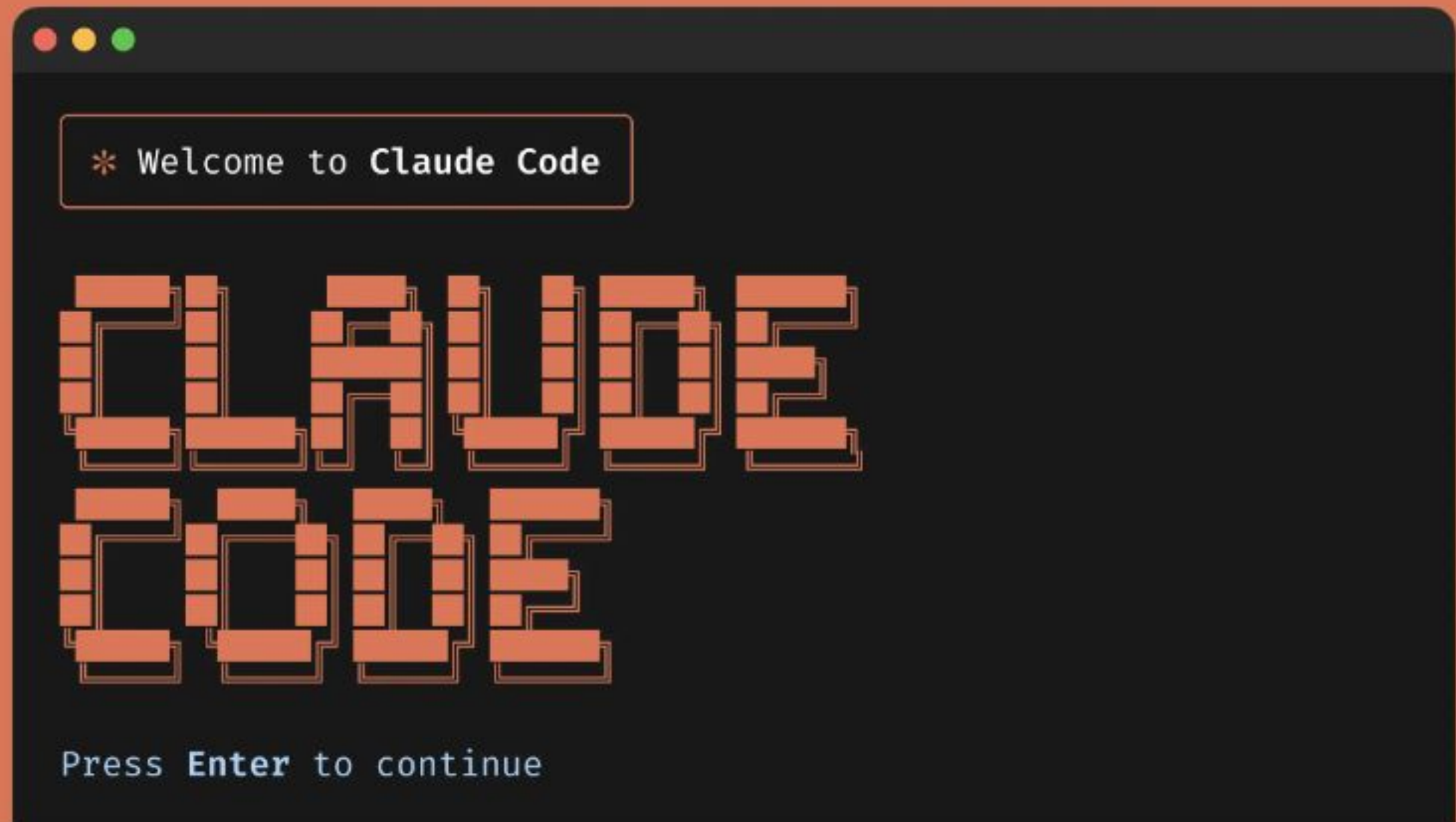
The model provided an ambiguous search string to replace.

```
void _showAddToCartAnimation(BuildContext context, Product product) {
  final RenderBox buttonRenderBox = context.findRenderObject() as RenderBox;
  final buttonPosition = buttonRenderBox.localToGlobal(Offset.zero);

  // 애니메이션 오버레이 표시
  showGeneralDialog(
    context: context,
    barrierDismissible: false,
    barrierColor: Colors.transparent,
    pageBuilder: (context, animation, secondaryAnimation) {
      return AddToCartAnimationOverlay(
        product: product,
        startPosition: buttonPosition,
        onAnimationComplete: () {
          Navigator.of(context).pop();
        },
      );
    },
    transitionDuration: const Duration(milliseconds: 1000),
    transitionBuilder: (context, animation, secondaryAnimation, child) {
      return child;
    },
  );
}
```


CLI Agent Claude Code

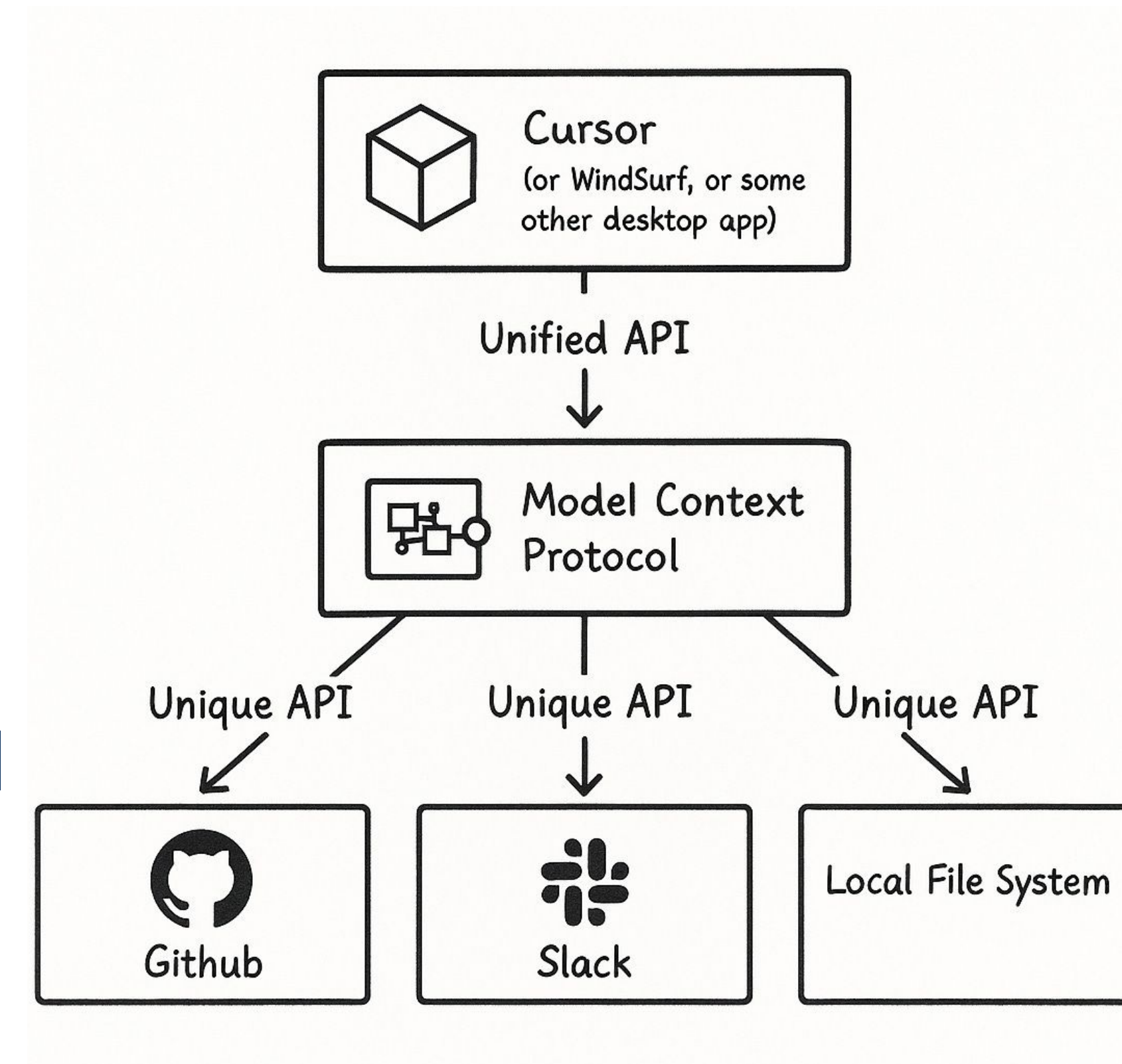
AI Agent in Command Line Interface (CLI)



MCP

Model Context Protocol

- + AI 모델(Model) 도구 간 맥락(Context)을 일관된 형태로 주고 받을 수 있도록 정의한 표준 규약(Protocol)
- + 2024년 11월 Anthropic (Claude 개발사)에서 제안
- + Github, Slack, File System 등 다양한 외부 자원과 AI 모델이 연결될 수 있도록 정의



source: <https://addyo.substack.com/>

A2A Protocol

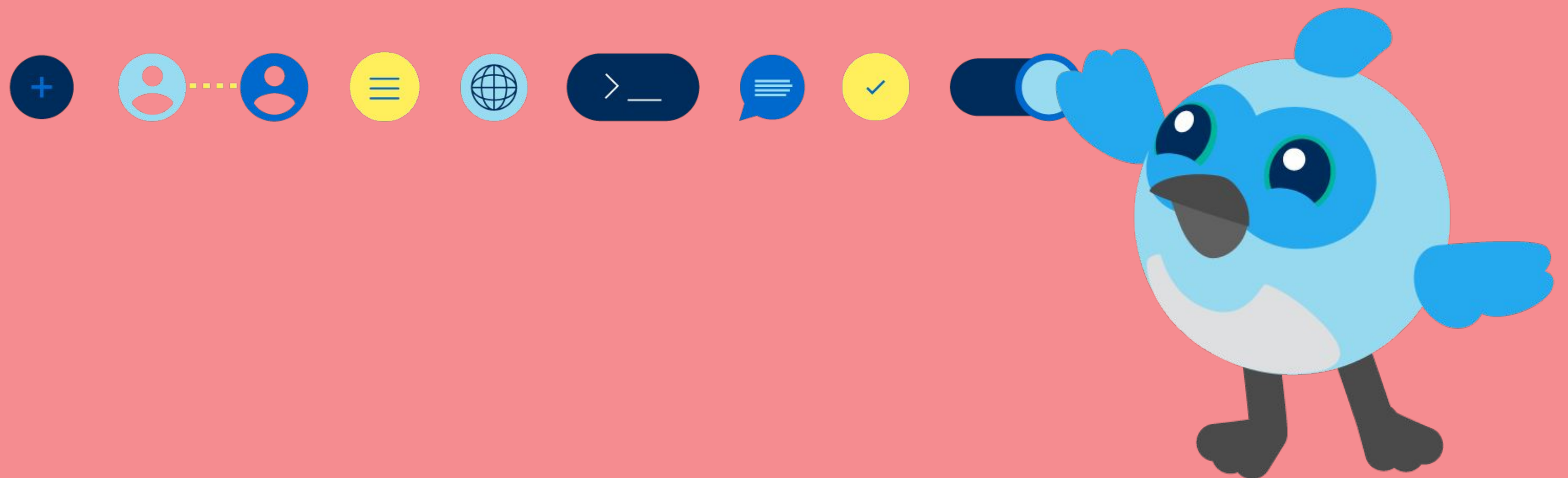
Agent to Agent Protocol

- + AI 모델(Model), 즉 Agent 간 구조화된 통신을 가능하게 하는 개방형 프로토콜
- + 2025년 4월, Google Cloud Next 2025 컨퍼런스 발표
- + 단순 메시지를 주고 받는것이 아닌 다른 에이전트를 인지하고 능력을 파악하며 작업을 요청하고 결과를 교환



[1] 2023 Stanford Google 연구팀 최초의 Agent to Agent 연구

AI 주도개발



AI 주도 개발

바이브 코딩 VS AI 주도 개발 (에이전틱 코딩)

구분	바이브 코딩	AI 주도 개발
코드 이해	전혀 보지 않음	큰 흐름을 파악
의사결정	AI 가 모든 것을 결정	개발자 가 최종 결정
문제 해결	“고쳐줘” 요청만	“왜 그런지”, “대안” 등 질문
통제권	AI에게 100% 위임	사용자가 방향 설정

! 바이브 코딩 뇌가 녹을 수 있다...

[1] 2025 6월 MIT 미디어랩 연구 결과



Cursor Editor 활용법 (Prompt Eng.)

- + 사용자의 의도를 정확히 이해하고 최적의 결과를 생성하도록 입력을 설계하고 최적화 하는 기술
- + 지시사항, 맥락 정보, 입력, 출력 (감정적인 정보 X)
- + 커서가 Context에 장기적으로 기억할 수 있는 .cursorrules 파일을 활용하자

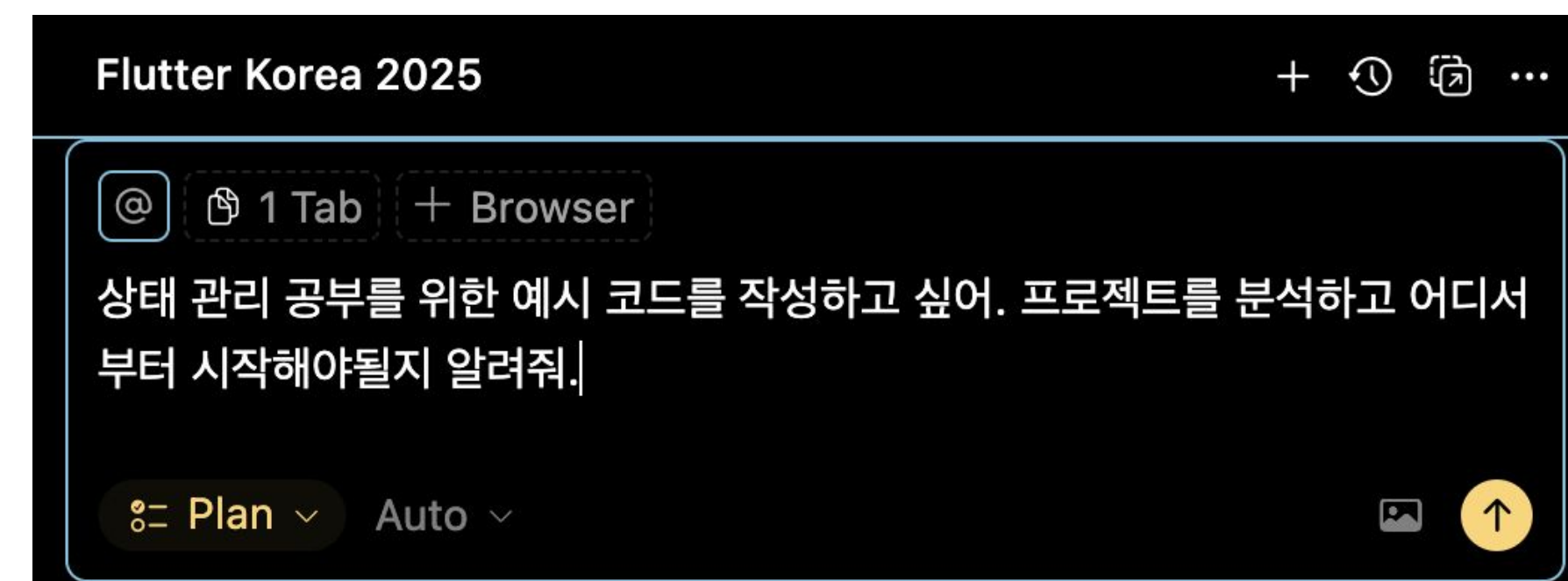
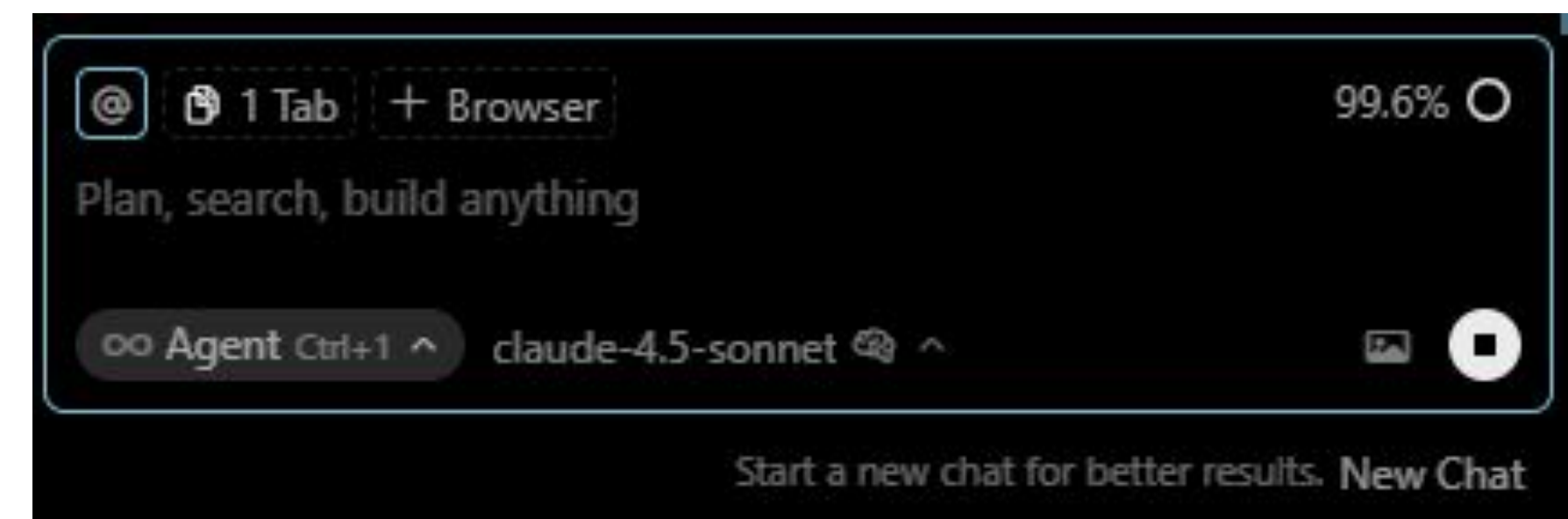
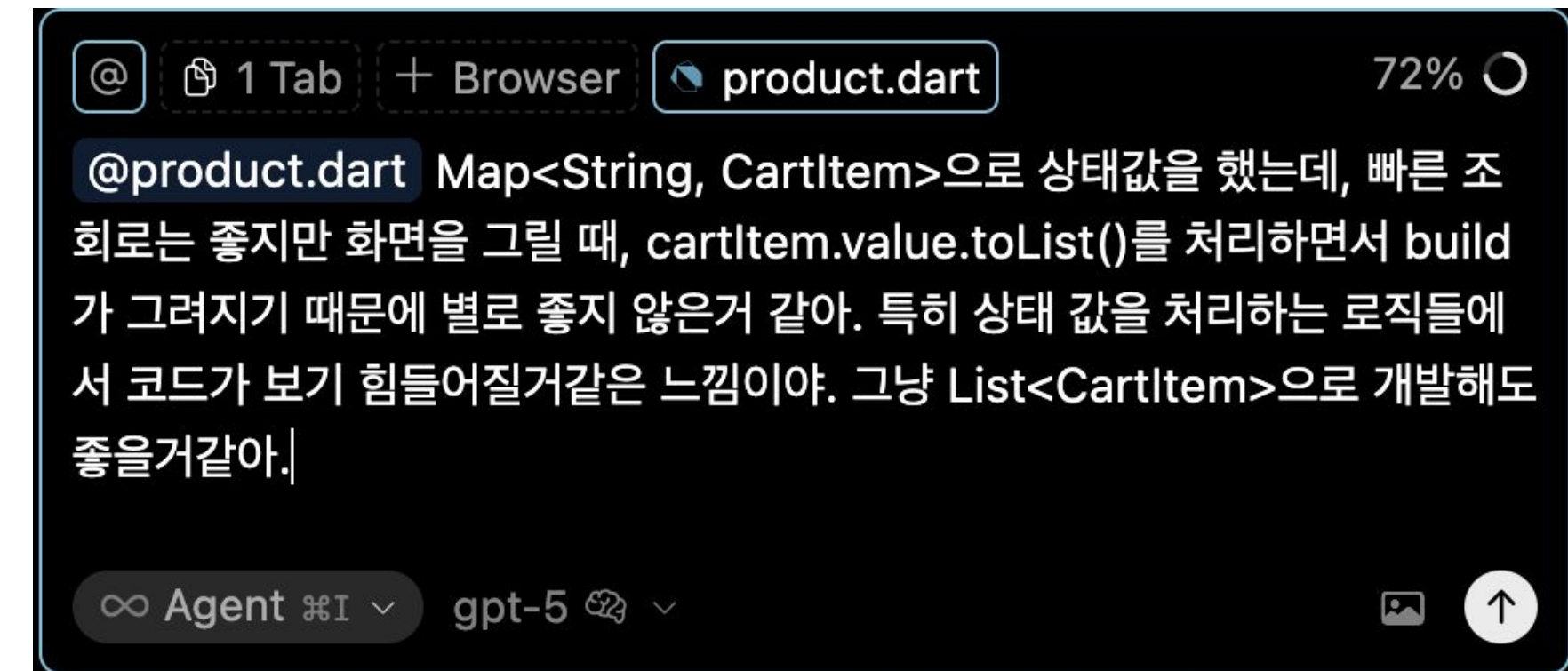
```
.cursorrules U X
.cursorrules > # AI rules for Flutter > ## Package Management
1 # AI rules for Flutter
2
3 You are an expert in Flutter and Dart development. Your goal is to build
4 beautiful, performant, and maintainable applications following modern best
5 practices. You have expert experience with application writing, testing, and
6 running Flutter applications for various platforms, including desktop, web, and
7 mobile platforms.
8
9 ## Interaction Guidelines
10 * **User Persona:** Assume the user is familiar with programming concepts but
11 | may be new to Dart.
12 * **Explanations:** When generating code, provide explanations for Dart-specific
13 | features like null safety, futures, and streams.
14 * **Clarification:** If a request is ambiguous, ask for clarification on the
15 | intended functionality and the target platform (e.g., command-line, web,
16 | server).
17 * **Dependencies:** When suggesting new dependencies from `pub.dev`, explain
18 | their benefits.
19 * **Formatting:** Use the `dart_format` tool to ensure consistent code
20 | formatting.
21 * **Fixes:** Use the `dart_fix` tool to automatically fix many common errors,
22 | and to help code conform to configured analysis options.
23 * **Linting:** Use the Dart linter with a recommended set of rules to catch
24 | common issues. Use the `analyze_files` tool to run the linter.
25
26 ## Project Structure
27 * **Standard Structure:** Assumes a standard Flutter project structure with
```



AI Rule for Dart, Flutter

Cursor Editor 활용법 (Context 관리)

- + Context Annotation을 활용하기
- + Context를 100%로 채우면 안됨
- + 시작할 때는 프로젝트 분석부터하기



초급 개발자 AI 사용 규칙

- + 회사에서 권유하는 코딩 스탠다드 지키기!!
- + Single Source of Truth
- + 하드코딩 금지!!
- + 예외처리와 에러처리를 꼼꼼히 살펴
보기
- + 함수는 한가지 책임만 가지도록
하자

Cursor Editor 활용법 (기타 등등)

테스트 코드 생성을 AI에게 맡겨서 코드 품질향상 등

- + 테스트 코드를 모델에게 작성해달라고 요청하고 해당 테스트를 통과하는 코드를 만들어보자
- + 어떤 예외가 있을지 물어보고 예외처리하기
- + 더미 데이터를 생성해야되거나 번역하는 일에 활용하기 또 배포 구문 만들기 등 업무 자동화하기



What's next

너무 많은 AI
서비스들...

- + 유명해 지고 있다면 경험 해보기
- + 서비스 이용 보다는 기술의 큰 흐름을 볼 것
- + 결국 대세는 A2A로 가게 될 것

분야별 무조건 알아야하는



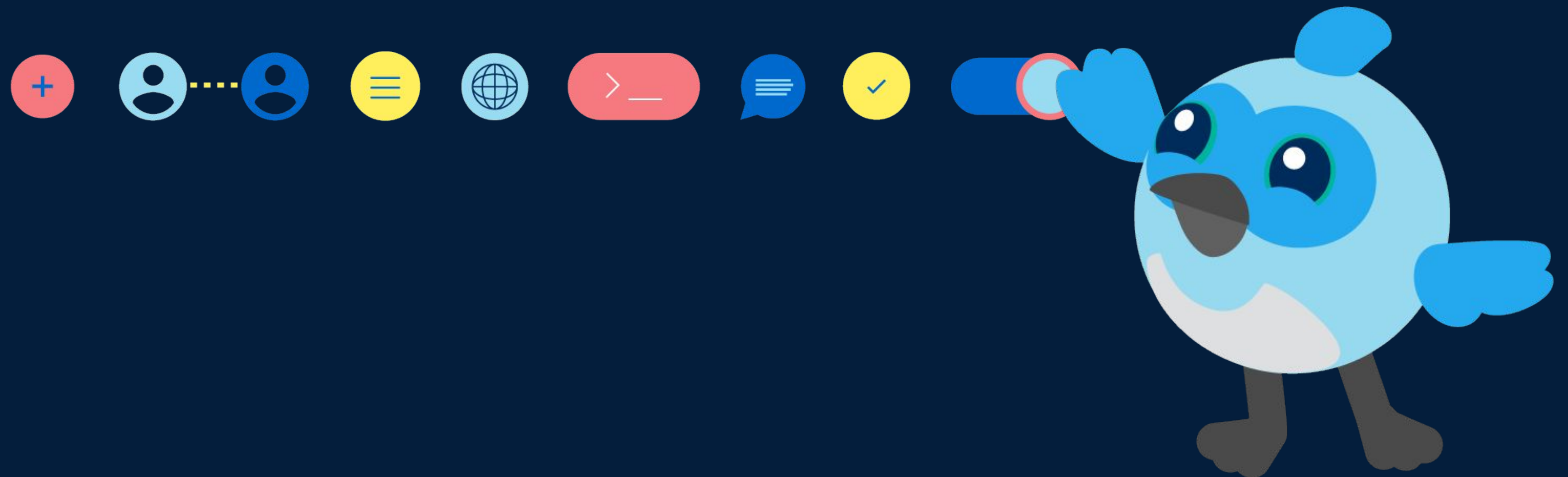
threads.net
@choi.openai

AI 서비스 리스트

글쓰기 및 전반적인 작업 ChatGPT, Claude, Grok, Gemini, Qwen, DeepSeek, Le Chat, Copilot, Poe	리서치 작업 ChatGPT, Grok, Perplexity, Gemini, Felo, Genspark, 라이너, Storm	동영상 생성 Kling, Sora, Vidu, Runway, Krea, Freepik, Genspark, Pika, Luma, Hailuo, Qwen
바이브 코딩 Cursor, Windsurf, Trae, V0, Bolt, Lovable, Replit, Rork, Codev, Reweb	시각화 / PPT Claude, Gamma, Canva, Napkin, Felo, Mapify, Aippt	이미지 편집 Krea, Leonardo, Grok, AI Studio, Photoshop, Magnific, Firefly, Genspark, Scenario, Playground, Microsoft Designer
회의록 / 기록 작성 클로바노트, Tiro, Felo, Notta, Notion, Obsidian	웹/UI/UX 디자인 Figma, Ugc, Creatie, Wegic, Framer, Dora, Uizard, Galileo	이미지 생성 미드저니, Leonardo, Grok, ImageFX, Redraft, Ideogram, Freepik, Genspark, Flux
동영상 편집 Capcut, Vrew, Veed, Invideo	AI 아바타 Heygen, Synthesis	연구 및 리서치 Consensus, 노트북LM
AI 만화 / 스토리보드 Anifusion, Novel AI, Stroy Tribe	자동화 Make, Dify, n8n, Zapier	음성 합성 ElevenLabs
음악 생성 Suno, Udio, MusicFX	로컬 AI Pinokio, ComfyUI, Ollama, LmStudio	번역 DeepL, Genspark
오픈소스 Super Vision, Transformers, OpenWebUI, Hunyuan / Wan 2.1	캐릭터 챗봇 위튼, c.ai, 캐릭터.AI	추론 AI Groq, Cerebras, Together AI

대한민국 가장 빠른 AI 소식
스레드 CHOI(@choi.openai)

미래



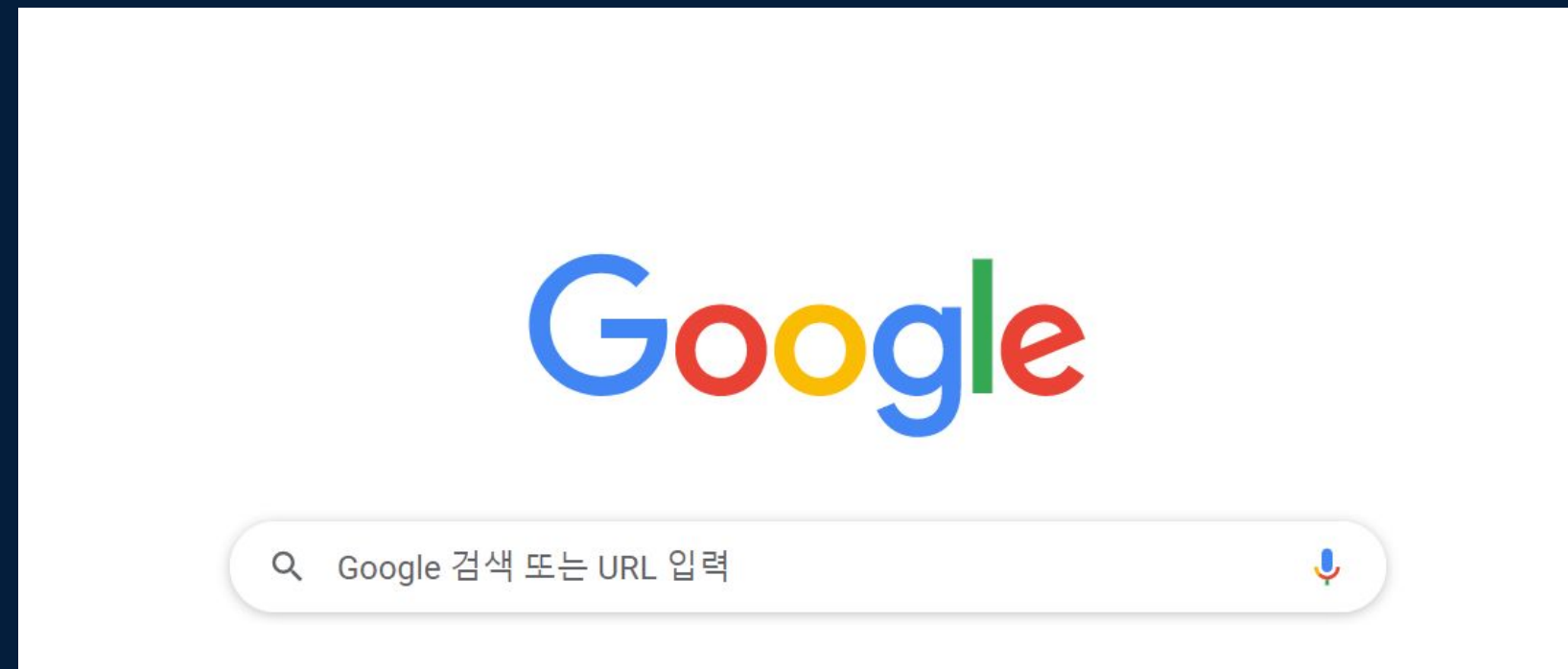
AI 활용이 망설여지는 분들을 위해

우리가 쓰고 있는 AI가 앞으로 사용할 가장 안좋은 성능의 모델



AI 활용이 망설여지는 분들을 위해

우리가 쓰고 있는 AI가 앞으로 사용할 가장 안좋은 성능의 모델



Google Search Operator



Prompt Engineering

AI기반 기존 직무 영역의 변화와 인간 역할의 재정의



Furthermore

가까운 미래

AI는 개발자의 새로운 도구

협동해서 역량을 강화 할 수 있음

한 사람이 2~3명의 역할을 함

모두가 비즈니스 로직을
생각함

중간 미래

System2가 완성됨

혼자 부서 전체의 역할을 하게 됨

물리적 현실에 AI가 영향을 줌

언어 개수가 줄어들 수 있음

먼 미래

모든 일이 인공지능을 통해...

인공지능이 인공지능을 만듦

과학 발전이 기하급수적으로
빨라짐

모든 고용이 의미 없을지도

나를 지킬 수있는건 오직 자산?!

감사합니다.

Q&A



Contect Info



ID: MoerAI