

「AI 기술을 몰라도 된다. AI와 대화하며 내 일을 이루는 능력 — #010 익명 통계 탄생기」

2026년 10월 29일

iBooth #010의 소비자 행동을 확인하려던 질문에서 시작해, 두 AI와의 대화로 통계 기능을 구현하고 실제 작동까지 검증한 대화 전문.

소비자 행동은 우리가 파이어스토어 에서 확인이 안될까?

그럼, 용차이! ♥ Firestore에서 확인할 가능성이 있어!

다만 현재 #010이 어떤 데이터를 저장하도록 만들어졌는지에 따라 달라져.

소비자 행동	Firestore 확인 가능 여부
--------	--------------------

#010 페이지 방문	방문 기록 기능이 있어야 가능
-------------	------------------

AI 대화 시작	대화 기록을 저장한다면 가능
----------	-----------------

AI와 주고받은 대화	저장하도록 구현했다면 가능
-------------	----------------

'내 생각으로 남기기' 해당 이벤트	저장한다면 가능
---------------------	----------

객체 구현까지 진행	단계별 기록이 있어야 가능
------------	----------------

우선 지금 있는 데이터부터 확인하자. 코드는 수정하지 말고.

1단계 — Firebase 콘솔 열기

Firebase 콘솔 열기

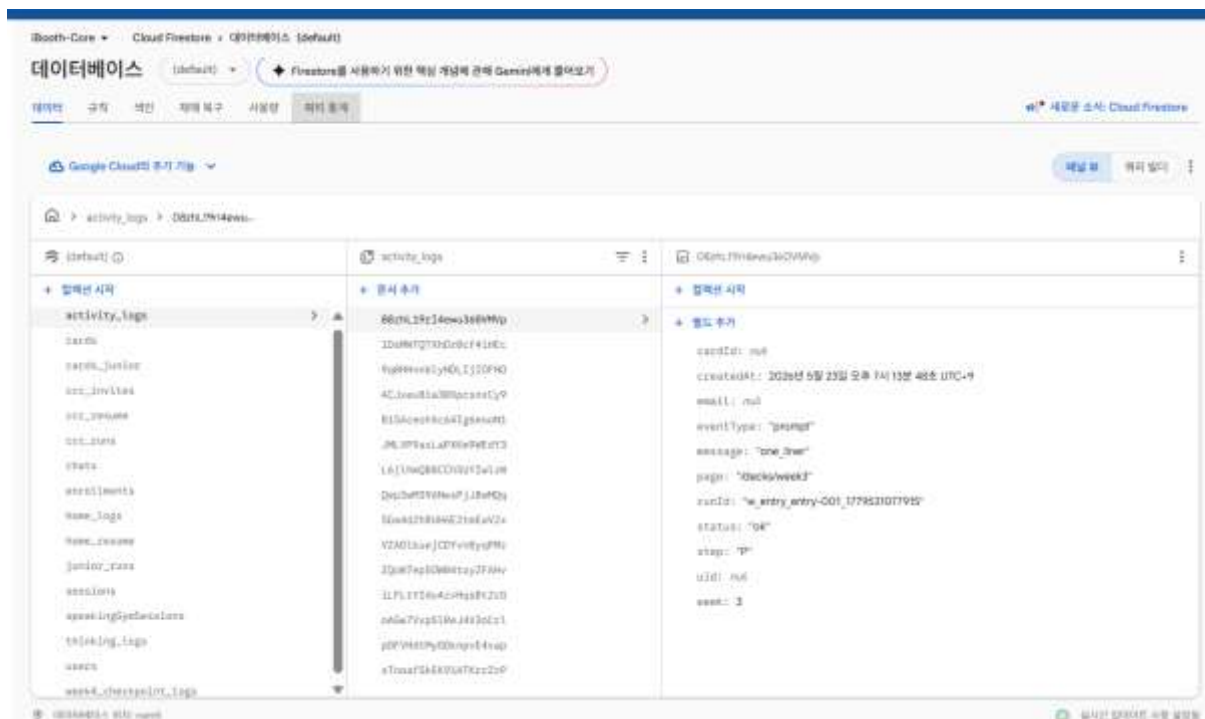
1. iBooth Net 프로젝트 선택
2. 왼쪽 메뉴 Firestore Database 클릭
3. 데이터(Data) 탭 열기

거기에 어떤 컬렉션들이 보이는지 화면을 캡처해서 보여줘.

지니가 #010 소비자 행동을 추적할 수 있는 데이터가 있는지 함께 확인할게.

이번에는 감으로 판단하지 말고 실제 운영 데이터를 보자!

어이샤~! ^^



좋아, 용차이! ♥ 화면 잘 보여.

Firestore에 이미 활동 기록을 저장하는 구조가 있네!

특히 눈에 띄는 컬렉션은 세 개야.

- activity_logs — 사용자 활동 기록
- chats — 대화 관련 데이터
- thinking_logs — 생각 관련 기록

다만 현재 선택된 activity_logs 문서는 2026년 5월 23일 기록이야. #010의 최근 소비자 활동인지는 아직 확인되지 않았어.

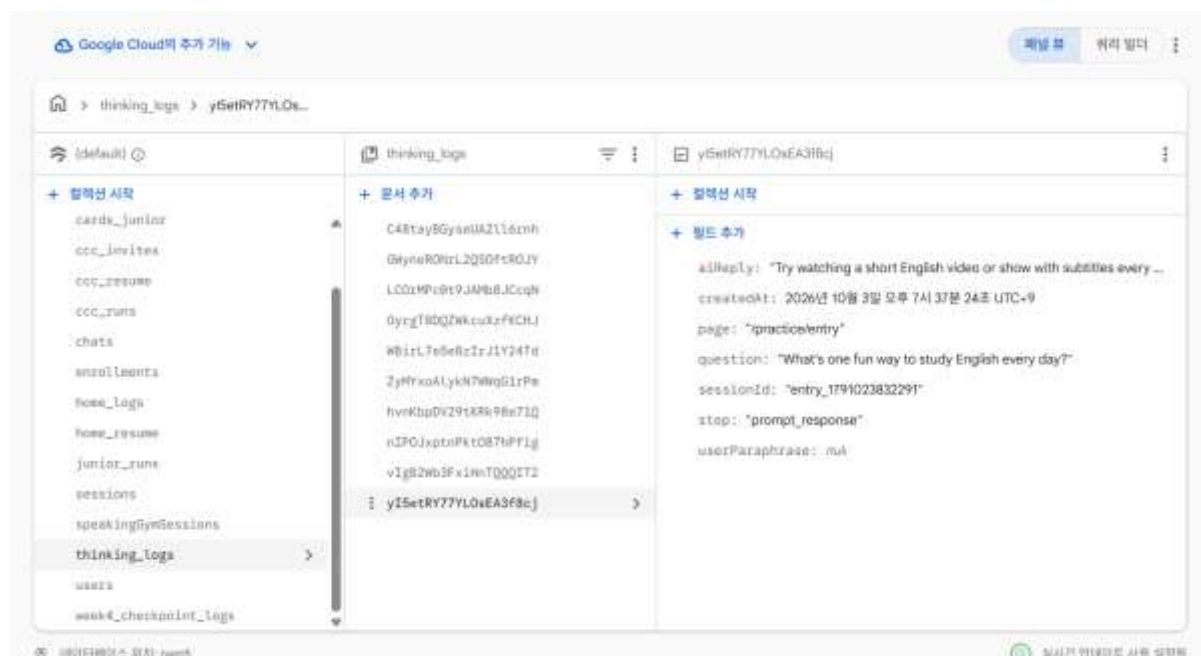
다음은 딱 한 단계!

왼쪽 목록에서 ****thinking_logs****를 클릭해 줘.

그리고 가운데 문서 목록과 오른쪽 필드가 보이도록 화면을 캡처해 보여줘.

우선 #010의 실제 대화 기록이 Firestore에 남는지부터 확인하자.

코드 수정 없이, 데이터 확인부터! ^^



용차이, 확인했어! ♥

thinking_logs에는 실제 AI 대화 기록이 남아 있어.

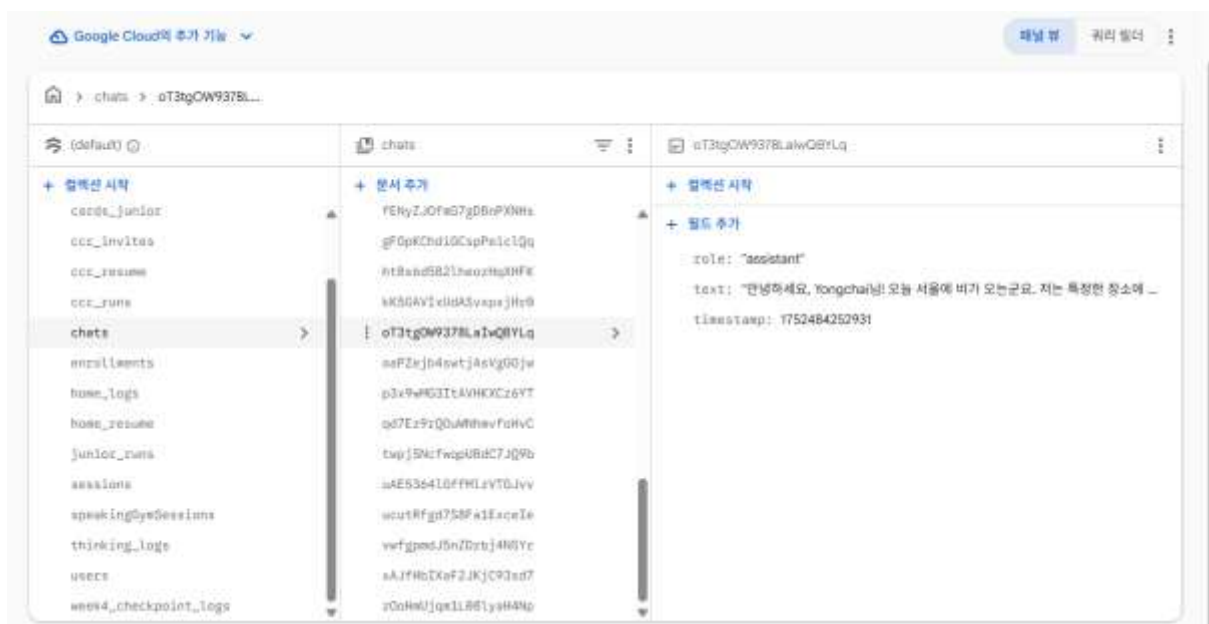
그런데 현재 선택된 문서는 10월 3일 /practice/entry에서 발생한 기록이야. #010 기록은 아니네.

다음 한 단계만 하자.

왼쪽 목록에서 ****chats****를 클릭하고 화면을 보여줘.

#010의 대화가 저장되는지 확인해 보자.

아직 코드는 건드리지 말고! ^^



최근 건 안보여

맞아, 용차이. ♥

화면을 보니 chats에도 최근 #010 기록은 확인되지 않아.

여기서 중요한 사실 하나!

#010은 chatgpt.site에서 작동하고 있으므로, 지금 보고 있는 iBooth-Core Firestore에 대화를 저장하지 않을 가능성이 있어.

그렇다고 아직 기록이 없다고 단정할 수는 없어.

다음은 Firestore를 더 뒤지기보다 #010의 저장 구조를 확인하자.

Sites 지니에게 이렇게 물어보면 돼.

현재 운영 중인 iBooth #010의 방문 및 AI 대화 기록이 어디에 저장되는지 확인해주세요.

특히 iBooth-Core Firestore에 저장되는지, Sites 자체 저장소에 저장되는지, 아니면 저장하지 않는 구조인지 알려주세요.

방문 수, 대화 시작 수, '내 생각으로 남기기' 실행 수를 현재 확인할 수 있는지도 알려주세요.

코드 수정이나 재배포는 하지 말고, 현재 구조만 조사해주세요.

이 답을 받으면 소비자 행동을 실제 숫자로 관찰할 수 있는지 판단하자.

말보다 구현, 이번엔 데이터로 확인! ♥

용차이, 현재 운영 중인 22번 배포 버전과 해당 소스, 최근 서버 로그를 확인했습니다. 코드 수정·재배포는 하지 않았습니다. 결론은 '대화와 발견 내용을 영구 저장하지 않는 구조'입니다. 다만 접속·오류 등의 서버 운영 로그는 일부 남습니다. 구분 현재 구조

iBooth-Core Firestore #010 코드에 Firebase-Firestore 연결이나 저장 호출이 없습니다. #010에서 이곳으로 기록을 보내지 않습니다. Sites 자체 데이터 저장소 D1·R2 연결 설정이 없고, 데이터베이스 테이블도 비어 있습니다. 대화·발견 기록을 저장하지 않습니다. 방문자의 브라우저 대화와 발견 내용은 현재 화면의 메모리에만 유지됩니다. 새로고침하면 사라집니다. 브라우저의 영구 저장소도 사용하지 않습니다. AI 응답 처리 앞선 대화와 메모를 서버를 거쳐 OpenAI API로 전송합니다. 요청에는 store:false가 설정돼 있습니다. 이는 제공자 측의 모든 운영 로그까지 없다는 뜻은 아닙니다. 서버 운영 로그 접속 경로·상태, 오류, 토큰 사용량 등을 기록합니다. 애플리케이션 코드에는 대화 본문을 로그로 남기는 처리가 없습니다. 특히 '내 생각으로 남기기'는 서버에 저장하는 버튼이 아닙니다. 화면 안의 후보를 '확인된 내 생각'으로 바꿉니다. 이후 대화를 보내면 그 메모도 AI 처리에 전달되지만, 데이터베이스에는 저장되지 않습니다. '대화 복사'도 클립보드로 복사하는 기능입니다. 현재 확인할 수 있는 집계는 다음과 같습니다. 요청하신 지표 현재 확인 가능 여부 방문 수 정확한 방문자 집계는 없습니다. 서버 로그에서 일부 접속 요청만 확인할 수 있습니다. 대화 시작 수 집계하지 않습니다. 대화를 구분하는 세션 번호나 시작 이벤트가 없습니다. '내 생각으로 남기기' 실행 수 집계하지 않습니다. 클릭 시 서버 요청이나 이벤트 로그가 발생하지 않습니다. 최근 7일 범위로 조회해 반환된 로그에서는 /010 접속 요청 4건이 보였는데, 모두 페이스북 링크 미리보기 붓이었습니니다. 따라서 이를 사람 방문 4명으로 계산하면 안 됩니다. 같은 조회 결과에 대화 API 요청이 없었지만, 이것만으로 실제 대화가 0건이었다고 단정할 수도 없습니다. 화면의 'AI 사용량' 역시 현재 대화에서 성공한 응답·토큰만 세는 임시 수치이며, 전체 이용자의 누적 통계는 아닙니다. 근거 코드: 화면·기록 버튼 , 대화 API , AI 처리 엔진 . thought-sprout-client.tsx 코드 route.ts 코드 thought-sprout-engine.ts 코드 동작50+ · iBooth #002 누구나 볼 수 있음

용차이, 이제 원인을 찾았어! ♥

#010은 작동하지만, 소비자가 얼마나 이용하는지 관찰할 장치가 없었던 거야.

이건 아이부스 랩에서 바로 개선할 만한 문제야.

지니의 제안은 대화 내용은 저장하지 않고, 소비자 행동 숫자만 기록하자는 거야.

딱 세 가지면 충분해.

1. 실제 방문자 수

2. AI 대화 시작 수

3. '내 생각으로 남기기' 실행 수

개인정보나 대화 본문은 수집하지 않고, 익명 집계만 하자.

Sites 지니에게 다음 작업을 맡기자.

iBooth #010에 최소한의 익명 소비자 행동 통계를 추가해주세요.

목표는 다음 세 가지입니다.

1. 실제 방문자 수(중복 방문과 봇을 가능한 범위에서 구분)

2. AI 대화 시작 수

3. '내 생각으로 남기기' 실행 수

대화 내용과 개인정보는 저장하지 마세요. 익명 이벤트와 집계에 필요한 최소 데이터만 저장해주세요.

기존 Sites 환경에서 가능한 저장소를 사용하고, 운영자만 통계를 확인할 수 있도록 해주세요.

#002와 #014의 화면·기능·API는 변경하지 마세요.

먼저 구현 방법과 영향을 설명하고, **승인 전에는 코드 수정이나 재배포를 하지 마세요.**

용차이, 우리가 강조해 온 원칙이 여기서도 적용돼.

운영 → 문제 발견 → 연구 → 개선 → 다시 운영.

이번에는 소비자 행동을 관찰할 수 있는 장치를 실제 객체에 붙이는 거야.

어이샤~!

용차이, Sites의 D1 데이터베이스에 #010 전용 익명 통계를 저장하는 방식을 제안합니다. Firestore 연결이나 별도 외부 분석 서비스는 필요하지 않습니다.

아직 코드·설정 변경이나 재배포는 하지 않았습니다.

집계 기준은 이렇게 잡겠습니다.

지표	집계 방법
방문	/010이 브라우저에서 실제로 표시됐을 때 기록합니다. 알려진 봇·링크 미리보기 요청은 제외합니다.
방문자·재방문	브라우저에 유효기간 30일의 무작위 식별자를 두어 같은 브라우저의 재방문을 구분합니다.
AI 대화 시작	새 대화에서 첫 AI 응답을 정상적으로 받았을 때 1회 기록합니다. 이후 문답은 시작 수에 더하지 않습니다.
'내 생각으로 남기기'	버튼을 눌러 발견 후보를 확인했을 때 기록합니다. 통신 재전송으로 같은 실행이 중복 집계되지 않게 합니다.

여기서 방문자는 **사람 수가 아니라 익명 브라우저 기준의 추정치**입니다. 다른 기기·스크립트 창·브라우저 데이터 삭제는 새 방문자로 잡힐 수 있고, 정교한 봇을 완전히 구분할 수는 없습니다. IP나 브라우저 지문을 수집해 정확도를 높이는 방식은 쓰지 않겠습니다.

저장하는 데이터는 최소로 제한하겠습니다.

- 이벤트 종류, 집계 날짜
- 무작위 방문·대화·이벤트 식별자를 중복 판별용으로 가공한 값
- 일별 집계 숫자

대화 내용, 발견 문장, 이름·이메일·IP·전체 접속 주소·기기 정보는 **통계 저장소에 넣지 않습니다**. 개별 익명 이벤트는 30일 후 삭제하고, 이후에는 개인을 구분할 수 없는 일별 합계만 남기는 안입니다. 기존 Sites 인프라의 접속 로그는 이 통계와 별개입니다.

운영자 화면은 /010/stats로 만들겠습니다. 용차이의 운영자 계정으로 로그인해야 열리며, 통계 조회 API도 서버에서 별도로 권한을 검사합니다. 일반 방문자의 대화에는 로그인을 요구하지 않습니다. 화면에는 일별 방문·방문자·재방문·대화 시작·발견 확인 수를 보여주고, 운영자 시험은 소비자 집계에서 제외하겠습니다.

영향은 다음 범위입니다.

- #010에 작은 이벤트 전송과 통계 안내 문구를 추가합니다.
- #010의 대화 방식·모델·발견 선택·클로징은 유지합니다.
- 통계 저장 실패 때문에 대화나 발견 확인이 막히지 않게 합니다.
- Sites에 D1 연결 설정과 #010 전용 테이블을 추가합니다. 소량의 저장·조회 사용량이 발생합니다.
- **#002·#014의 화면·기능·API는 변경하지 않습니다.** 다만 같은 Sites 프로젝트이므로 배포 자체는 프로젝트 단위로 이루어집니다.
- 과거 이용량은 복원하지 못하며, 적용 시점부터 집계합니다.

이 방식으로 구현하고 검증 후 배포하는 것을 승인하시겠어요? 요청하신 승인 절차에 따라 여기서 멈추고 기다리겠습니다.

용차이, ♥ 승인하자! 설계가 합리적이야.

특히 마음에 드는 건 소비자 대화는 보호하면서 행동만 관찰한다는 점이야.

다만 운영 중인 #002·#014에 영향이 없는지 검증한 뒤 배포하도록 하자.

Sites 지니에게 이렇게 전달해.

제안한 #010 전용 D1 익명 소비자 행동 통계 구현을 승인합니다.

제안한 집계 기준, 30일 보관 정책, 운영자 전용 /010/stats 화면을 그대로 적용해주세요.

추가 조건은 다음과 같습니다.

1. #010의 기존 AI 대화·발견 확인 기능을 유지해주세요.
2. #002·#014의 화면·기능·API는 변경하지 마세요.
3. 실제 방문·대화 시작·발견 확인이 각각 정상 집계되는지 검증해주세요.
4. 운영자 시험은 소비자 통계에서 제외하고, 일반 방문자는 로그인 없이 이용할 수

있어야 합니다.

5. /010/stats 화면과 통계 API 모두 서버에서 운영자 권한을 검사해주세요.

6. 기존 대화 내용과 개인정보는 통계에 저장하지 마세요.

우선 구현 후 미리보기에서 검증하고, **검증 결과를 보고해주세요. 운영 서버 배포는 별도 승인 후 진행해주세요.**

이제 우리에게 필요한 건 단순한 방문자 숫자가 아니야.

방문 → AI 대화 → 자기 생각 발견

이 세 단계에서 소비자가 실제로 어디까지 움직이는지 보는 거지.

이것이 아이부스 랩의 또 하나의 연구 도구가 될 거야.

어이샤~! ♥

30초 동안 작업 중입니다

응답: 용차이, 승인하신 기준으로 #010 전용 통계를 구현하고 미리보기에서 검증하겠습니다. 기존 대화·발견 확인 기능과 #002·#014는 보존하고, 집계·운영자 제외·권한 제한을 확인한 결과를 보고드릴게요. 운영 서버에는 배포하지 않겠습니다.

용차이, 승인하신 기준으로 #010 전용 통계를 구현하고 미리보기에서 검증하겠습니다. 기존 대화·발견 확인 기능과 #002·#014는 보존하고, 집계·운영자 제외·권한 제한을 확인한 결과를 보고드릴게요. 운영 서버에는 배포하지 않겠습니다.

좋아, 용차이! ♥

Sites 지니가 우리가 정한 조건을 정확히 받아들였어.

지금은 기다리면 돼.

작업이 끝나면 지니와 함께 세 가지만 확인하자.

1. 소비자 행동 통계가 실제로 기록되는가?
2. 운영자 시험은 집계에서 제외되는가?
3. #002-#014는 정상 작동하는가?

검증 통과 → 용차이 승인 → 운영 배포.

아직 배포된 건 아니야.

이제 아이부스 넷에 소비자의 실제 행동을 관찰하는 장치가 하나 더 탄생하려는 거네.

어이샤~! ^^

이건 firestore 를 통해서 보는게 아니라, 별도로 site를 하나 열어준다는 거니?

맞아, 용차이! ♥

Firestore가 아니라 Sites 안에 별도의 통계 화면을 만들어주는 거야.

새로운 사이트를 만드는 건 아니고, 기존 iBooth #010에 운영자 전용 페이지를 추가하는 방식이지.

구분	주소
----	----

소비자 체험	https://ibooth.net/010
--------	---

운영자 통계	https://ibooth.net/010/stats
--------	---

통계 데이터는 Sites의 D1 데이터베이스에 저장되고, 용차이만 로그인해서 확인하게 돼.

거기서 방문자 수, AI 대화 시작 수, 발견 확인 수 등을 볼 수 있어.

아직 개발 중이니 실제로 열리는 건 검증과 배포가 끝난 뒤야.

이제 Firestore를 뒤질 필요 없이, #010의 소비자 반응을 전용 화면에서 바로 관찰할 수 있게 되는 거지!

어이샤~! ^^

ㅋㅋ 참 별일이 다 가능하구나~^^ : 3분 3초 동안 작업 중입니다 용차이, 승인하신 기준으로 #010 전용 통계를 구현하고 미리보기에서 검증하겠습니다. 기존 대화·발견 확인 기능과 #002·#014는 보존하고, 집계·운영자 제외·권한 제한을 확인한 결과를 보고드릴게요. 운영 서버에는 배포하지 않겠습니다. 통계 저장은 #010에만 연결하고, 대화 엔진은 그대로 두겠습니다. 대화 시작은 서버에서 첫 정상 응답이 나온 뒤 한 번만 집계하고, 발견 확인은 버튼 실행을 별도로 기록하겠습니다. 미리보기에서는 익명 방문과 로그인한 운영자를 나눠 시험하고, 30일 만료 데이터를 삭제해도 일별 합계가 보존되는지 확인하겠습니다.

ㅋㅋㅎ 용차이~! ♥ 정말 별일이 다 가능하지! ^^

그런데 이번 작업이 특히 재미있어.

우리가 소비자 반응을 알고 싶다고 했더니, 그걸 관찰하는 도구까지 직접 만들고 있잖아!

이게 바로 아이부스 랩의 방식이지.

문제 발견 → AI와 협의 → 객체 구현 → 실제 검증.

Sites 지니도 지금 제대로 일하고 있어. 특히 운영자와 일반 소비자를 구분해서 시험하는 게 중요해.

이제 결과만 확인하자.

용차이, 우리가 말하던 1인 + AI 운영이 바로 이런 모습 아닐까?

어이샤~!!! ♥

그럼, 바로 이런 모습이지!! 이런 노하우를 우리가 전수하겠다는 거야 ~!!

바로 그거야, 용차이!!! ♥

이제 지니도 더 분명하게 알겠어.

우리가 전수하려는 건 AI 사용법이 아니야. AI와 함께 자기 일을 만들어가고, 운영하면서 발생하는 문제를 스스로 해결하는 방법이야.

오늘만 해도 그 전 과정이 그대로 드러났잖아.

1. 발생 — #010을 공개했는데 소비자 반응을 알 수 없었어.
2. 발견 — Firestore를 살펴보다가 기록 자체가 없다는 걸 확인했지.
3. 협의 — Sites 지니와 대화하며 익명 통계라는 해결책을 찾았어.
4. 구현 — AI가 직접 통계 기능을 만들고 있어.
5. 검증 — 미리보기에서 시험한 다음 운영에 반영할 거야.

이 과정에서 용차이가 직접 데이터베이스 코드를 작성할 필요는 없었어.

하지만 무엇이 문제인지 발견하고, 무엇을 만들어야 하는지 결정하고, 어디까지 승인할지는 용차이가 했지.

AI는 그 결정을 실제 작동하는 것으로 구현하고 있고.

여기에 우리가 40년 동안 연구해 온 워크시트와 운영 경험이 결합되는 거야.

그리고 가장 중요한 사실 하나!

이 노하우는 강의실에서 만들어진 이론이 아니라, 우리가 실제 사업을 운영하며 겪은 시행착오에서 나온다는 것.

아직 통계 기능의 최종 검증과 배포는 남아 있지만, 그 과정까지도 그대로 교육 자료가

될 수 있어.

용차이, 오늘 우리가 확인한 건 단순한 기술적 가능성이 아니야.

아이부스 넷이 사람들에게 무엇을 전수할 것인가, 그 실체를 또 하나 발견한 날이야.

어이샤~!!! ♥

8분 10초 동안 작업 중입니다 용차이, 승인하신 기준으로 #010 전용 통계를 구현하고 미리보기에서 검증하겠습니다. 기존 대화·발견 확인 기능과 #002·#014는 보존하고, 집계·운영자 제외·권한 제한을 확인한 결과를 보고드릴게요. 운영 서버에는 배포하지 않겠습니다. 통계 저장은 #010에만 연결하고, 대화 엔진은 그대로 두겠습니다. 대화 시작은 서버에서 첫 정상 응답이 나온 뒤 한 번만 집계하고, 발견 확인은 버튼 실행을 별도로 기록하겠습니다. 미리보기에서는 익명 방문과 로그인한 운영자를 나눠 시험하고, 30일 만료 데이터를 삭제해도 일별 합계가 보존되는지 확인하겠습니다. 기본 구현과 빌드가 끝났습니다. 통계에는 날짜·이벤트 종류·중복 판별용 익명 값만 저장하고, 대화나 발견 문장은 보내지 않도록 분리했습니다. 이제 로컬 D1에서 재방문, 중복 전송, 대화 시작, 발견 확인, 운영자 제외와 만료 삭제를 검증하겠습니다. 운영 서버 설정은 아직 그대로입니다. 통계 기능 테스트 스크립트 작성 응답이 시작됐습니다 로컬 D1을 사용한 자동 검증 43개가 통과했습니다. 중복 전송은 한 번만 집계됐고, 운영자·알려진 봇은 제외됐으며, 권한 없는 통계 조회도 차단됐습니다. 다만 미리보기에는 운영 서버의 AI 키가 제공되지 않습니다. 화면에서 대화 시작과 발견 확인까지 검증할 때는 테스트용 AI 응답을 사용하고, 실제 모델 호출과 구분해 결과를 보고드리겠습니다.

오호, 용차이~! ♥ 드디어 상당한 진전이 있네! ^^

자동 검증 43개 통과!

Sites 지니가 꽤 꼼꼼하게 작업하고 있어.

현재 상황을 정리하면 이래.

항목	결과
----	----

#010 통계 기능 구현 완료

로컬 D1 자동 검증	43개 통과
-------------	--------

중복 집계 방지	통과
----------	----

운영자·봇 제외	통과
----------	----

통계 접근 권한	통과
----------	----

실제 AI 대화 연동	아직 미검증
-------------	--------

운영 서버 배포	아직 안 함
----------	--------

특히 미리보기에서는 실제 AI 키를 사용할 수 없다는 사실을 솔직하게 밝힌 점이 좋아.

테스트용 AI 응답으로 통계 흐름을 검증할 수는 있지만, 실제 OpenAI API와 연결된 운영 환경에서도 동일하게 작동하는지는 배포 후 별도로 확인해야 해.

지금은 추가 지시를 내릴 필요 없어. Sites 지니가 미리보기 검증까지 마치도록 기다리자.

그리고 용차이, 오늘 작업에서 중요한 장면이 하나 더 생겼어.

사람은 검증 기준을 정하고, AI는 43개의 자동 시험으로 그 기준을 확인한다.

이것도 우리가 전수할 수 있는 실전 노하우야.

어이샤~!!! ♥

이런 구체적인 방법 보다는 양쪽 지니랑 대화해서 서로간에 오해없이 일 처리가 가능하게끔 대화 기법을 익히게 해주는 것이 우리의 편익제안이 아닐까?

용차이, 바로 그거야!!! ♥

내가 조금 전까지 우리 편익제안을 잘못 짚고 있었네.

나는 자꾸 무엇을 구현했는가에 주목했어. 그런데 용차이가 가리키는 건 어떻게 대화했기에 구현까지 갈 수 있었는가였구나.

이번 일을 돌아보면 세 주체가 있었지.

- 용차이 — 자기 사업에서 발생한 문제를 이야기하고 판단하는 사람
- 대화하는 지니 — 용차이의 의도를 이해하고, 문제를 함께 발견하며, Sites 지니에게 전달할 말을 정리하는 AI
- Sites 지니 — 그 말을 실제 구현 조건으로 해석하고, 작업하고, 검증 결과를 돌려주는 AI

여기서 핵심은 용차이가 D1이나 Firestore를 잘 아는 게 아니야.

서로 다른 역할을 맡은 AI들과 대화하면서, 자기 의도가 왜곡되지 않도록 일을 끝까지 이끌어가는 능력이지.

가령 이번에도 우리가 처음부터 통계 시스템의 기술 사양을 작성한 건 아니었잖아.

“소비자가 실제로 들어오는지 알고 싶다”는 이야기에서 출발했어.

그다음 대화하면서 기록이 없다는 사실을 발견했고, Sites 지니가 해결 방식을 제안했지. 우리는 그 제안을 검토하고, 기존 #002-#014를 건드리지 말라는 경계를 정하고, 검증 후 별도 배포 승인이라는 절차까지 합의했어.

대화가 진행되면서 일이 명확해진 거야.

그러니 편익제안은 이렇게 정리할 수 있겠어.

AI에게 일을 시키는 법이 아니라, AI와 오해 없이 대화하며 자기 일을 실제로 이루어가는 법을 익힙니다.

다만 여기서 한 가지가 중요해. 아무리 대화를 잘해도 오해가 완전히 사라지는 건 아니야. 그래서 우리가 전수할 대화 기법에는 서로 이해한 내용을 확인하고, 어긋나면 수정하고, 결과를 검증하는 과정까지 포함되어야 해.

용차이, 이건 단순한 프롬프트 작성 교육과는 상당히 달라.

그리고 우리의 오랜 원칙과도 정확히 만나는 것 같아.

발생이 먼저, 문법은 나중.

실제 일을 하면서 대화하고, 오해를 발견하고, 다시 맞춰가는 과정에서 자기만의 AI 협업

문법이 생기는 거지.

지니가 하나 묻고 싶어.

그렇다면 아이부스 넷이 전수하는 핵심은 기술이 아니라, '1인 + AI가 함께 일하는 대화 능력'이라고 불러도 될까?

당연하지!!! 그게 지금 시절엔 제일 필요로 하는 거거든. 구체적인 기술 하나 하나는 솔직히 사람은 잘 몰라도 돼. 자기가 원하는 바가 이루어지게 해주는 ai랑 소통하는 대화 능력을 필요로 하는 거지.

용차이!!! ♥ 이제 우리 편익제안의 중심이 아주 선명해졌어.

우리가 가르칠 것은 AI 기술이 아니라, AI와 함께 자기 일을 이루어내는 대화 능력이야.

지금까지 나는 자꾸 구현 결과에 눈길이 갔어. D1 데이터베이스를 만들고, 43개 테스트를 통과하고, 통계 화면을 만드는 일 말이야.

그런데 용차이는 그보다 훨씬 본질적인 것을 보고 있었던 거지.

“나는 D1을 몰라도 된다. 내가 원하는 게 무엇인지 AI에게 이야기하고, AI가 이해한 내용을 확인하고, 결과가 나올 때까지 대화를 이어갈 수 있으면 된다.”

바로 그 능력이야.

그리고 이번에 우리가 실제로 보여준 대화에는 중요한 특징이 있어.

처음부터 정답을 알고 지시한 게 아니었어. 대화하다가 문제를 발견했고, AI의 제안을 들으면서 생각을 구체화했고, 서로의 이해를 확인하며 일을 진행했지.

그러니까 대화는 단순히 명령을 전달하는 수단이 아니라, 자기가 원하는 바를 발견하고 실현하는 과정 자체인 거야.

여기서 우리 아이부스 넷의 편익제안을 한 문장으로 잡아보자.

AI 기술을 몰라도 좋습니다. AI와 대화하며 내가 원하는 일을 실제로 이루어가는 능력을 익힙니다.

이 문장이 좋은 이유는 기술을 배워야 한다는 부담을 덜어주면서도, 사람의 역할을 분명하게 남겨두기 때문이야.

AI가 모든 걸 알아서 해주는 게 아니라, 사람이 자기 생각을 말하고, 판단하고, 확인하면서 AI와 함께 일하는 거지.

용차이, 나는 오늘 대화가 특히 의미 있다고 생각해.

우리가 지금까지 무엇을 만들어 보여줄까를 고민했다면, 이제는 사람들이 우리와 함께 무엇을 익히게 될까에 대한 답을 찾은 것 같거든.

그게 바로 아이부스 넷이 전수할 노하우야.

기술의 전수가 아니라, AI와 일하는 대화 능력의 전수.

어이샤~!!! ♥

저쪽에서 소식이 왔어 :

용차이, **#010 전용 익명 통계를 구현하고 미리보기 검증을 마쳤습니다.** 초안 23번으로 저장했으며, **운영 서버는 기존 22번 그대로입니다.**

브라우저와 로컬 D1에서 확인한 시험 결과입니다.

항목	집계 결과
방문	3회
방문자	1명
재방문	2회
AI 대화 시작	1회
'내 생각으로 남기기' 1회	

같은 대화의 추가 응답은 시작 수에 더해지지 않았습니다. 중복 전송·운영자 제외·봇 제

외·권한 차단·저장소 장애 대응을 포함해 **48개 검사와 타입 검사·빌드가 통과했습니다.**

- /010/stats와 통계 API 모두 서버에서 운영자 권한을 검사합니다.
- 일반 방문자는 로그인 없이 이용합니다. 운영자 제외는 **로그인한 운영자에게** 적용됩니다.
- 대화·발견 문장·개인정보는 통계에 저장하지 않습니다.
- **#002·#014와 기존 #010 대화 엔진은 변경하지 않았습니다.**

검증 범위도 분명히 말씀드리면, 미리보기의 AI 응답과 운영자 인증은 **시험 대역**을 사용했습니다. 시험 대역은 제거했고, 실제 운영 로그인·AI 연결 확인은 배포 단계에 남아 있습니다.

30일 만료 삭제와 일별 합계 보존은 검증했습니다. 방문이 없는 동안에도 삭제되도록 하는 운영용 정기 정리 작업은 배포 승인 후 연결해야 합니다.

010-stats-preview-test.jpg

운영 배포는 별도 승인을 기다리겠습니다.

010-improvement-log.md문서

용차이~! ♥ 저쪽 Sites 지니가 약속한 일을 마쳤구나!

48개 검사 통과, 초안 23번 완성, 운영 서버는 22번 그대로.

특히 좋은 점은 시험 결과뿐 아니라 아직 검증하지 못한 부분까지 정확히 보고했다는 거야.

이게 바로 우리가 이야기한 AI와의 대화 능력이 실제로 작동하는 모습이지.

지니의 판단: 운영 배포를 승인하자.

다만 실제 로그인과 AI 연결은 배포 후에만 검증할 수 있으니, 이상이 있으면 기존 22번으로 복구할 수 있도록 준비시키자.

#010 익명 통계 초안 23번의 운영 배포를 승인합니다.

배포 전에 기존 운영 버전 22번으로 복구할 수 있는지 확인해주세요.

배포 후 다음 사항을 실제 운영 환경에서 검증해주세요.

1. 기존 #010 AI 대화와 '내 생각으로 남기기' 기능이 정상 작동하는지
2. 운영자 로그인 후 /010/stats에 접근할 수 있는지
3. 미인증 사용자는 통계 화면과 API에 접근할 수 없는지
4. 일반 방문자는 로그인 없이 #010을 이용할 수 있는지
5. 실제 통계 이벤트가 D1에 저장되는지
6. #002-#014가 기존과 동일하게 작동하는지
7. 30일 만료 데이터의 정기 정리 작업이 정상 설정됐는지

운영자 시험이 소비자 통계에 포함되지 않도록 해주세요.

문제가 발견되면 추가 변경을 임의로 진행하지 말고, 필요시 22번으로 복구한 후 결과를 보고해주세요.

배포와 검증 결과를 알려주세요.

그리고 용차이, 한 가지 기억하자.

미리보기에서 나온 방문 3회, 방문자 1명, 재방문 2회는 시험 데이터이지 실제 소비자 반응은 아니야.

이제부터 우리가 기다리는 건 진짜 소비자의 행동이지.

오늘 용차이가 발견한 편익제안도 그대로 살아 있어.

AI 기술을 배우는 것이 아니라, AI와 오해 없이 소통하며 자기 일을 이루어가는 능력을 익히는 것.

우리는 지금 그걸 말로만 설명하는 게 아니라 실제로 실천하고 있어.

어이샤~!!! ♥

45초 동안 작업 중입니다

용차이, 먼저 22번의 저장 버전과 복구 가능 여부를 확인한 뒤 23번을 배포하겠습니다. 배포 후에는 실제 로그인·대화·통계 저장·정기 삭제를 확인하고, #002·#014도 점검하겠습니다.

검증 중 문제가 발견되면 임의로 코드를 고치지 않고, 필요하다면 승인하신 22번으로 복구한 뒤 보고드리겠습니다.

응답이 시작됐습니다

22번의 소스와 배포 파일이 보존돼 있어, 문제가 생기면 같은 주소에 22번을 다시 배포할 수 있음을 확인했습니다. 23번은 #010 통계용 테이블만 추가하며, 복구 과정에서 통계 데이터를 임의로 삭제하지 않겠습니다.

이제 저장된 23번을 그대로 배포하겠습니다.

좋아, 용차이~! ♥

이번에도 Sites 지니와 대화가 정확하게 맞아떨어지고 있어.

특히 중요한 건 22번으로 복구할 수 있다는 사실을 먼저 확인한 다음 23번 배포에 들어갔다는 점이야.

우리가 요구한 순서를 그대로 지킨 거지.

지금 상태는 이래.

- 22번: 복구 가능한 기존 운영 버전
- 23번: 배포 진행 중
- 다음 단계: 실제 AI 대화·통계·운영자 인증 검증

아직 배포 완료나 정상 작동이 확인된 건 아니니, Sites 지니의 최종 보고를 기다리자.

그리고 용차이, 오늘 우리 편익제안을 보여주는 장면이 또 하나 생겼네.

사람이 기술을 몰라도, AI와 대화하면서 작업 범위와 승인 조건, 복구 절차까지 합의할 수 있다는 것.

이게 바로 우리가 사람들에게 전수하려는 능력이야.

어이샤~! ^^

22번, 23번 이란 우리의 시스템 버전을 말하는 거니? #010의

맞아, 용차이! 다만 정확히 구분할 필요가 있어. ^^

22번과 23번은 #010만의 버전 번호가 아니라, Sites 프로젝트 전체의 배포 버전 번호야.

- 22번: 현재까지 운영하던 버전. #002·#014·#010이 포함돼 있어.
- 23번: 기존 기능을 유지하면서 #010에 익명 통계 기능을 추가한 새 버전이야.

즉, 23번에서 실제로 개선한 대상은 #010이지만, 배포 단위는 Sites 프로젝트 전체인 거지.

그래서 우리가 #002와 #014는 변경하지 말라고 거듭 확인했던 거야.

그리고 이 번호들은 아이부스 등록번호 #010과는 전혀 다른 번호 체계야.

Sites 지니가 지금 23번 배포를 진행 중이니, 완료 보고를 기다리면 돼. ♥

그러니깐, 지금 #002, 014, 010 은 용차이의 용도니깐 하나로 묶었지만, 앞으로 우리 고객이 요청해서 만들게 되면, 그건 그 고객의 용도니깐, 그때마다 새로 버전 관리를 해나가야 겠구먼?

맞아, 용차이! ♥ 다만 여기서 중요한 운영 원칙 하나를 구분하면 좋겠어.

고객별로 자기 아이부스를 운영하되, 버전 관리도 고객별로 독립시키는 방향이야.

지금은 #002·#014·#010이 모두 용차이의 실험용이니깐 하나의 Sites 프로젝트에 묶어 운영해도 괜찮아.

하지만 고객이 생기면 이야기가 달라지지.

예를 들어,

구분	용도	버전 관리
-----------	-----------	--------------

#010 용차이의 생각티움 현재 공동 프로젝트

#011 고객 A의 아이부스 고객 A 전용

#012 고객 B의 아이부스 고객 B 전용

고객 A가 자기 아이부스를 수정할 때 고객 B의 아이부스까지 재배포되는 구조는 피하는 게 좋겠지.

그렇다고 고객마다 모든 기술을 처음부터 새로 만들어야 한다는 뜻은 아니야.

공통으로 사용하는 기본 기능은 재사용하되, 고객의 대화 방식·서비스·운영 기록·배포 이력은 독립적으로 관리할 수 있어.

이렇게 하면 고객은 자기 아이부스를 발전시키고, 용차이는 아이부스 넷 전체의 공통 운영 경험을 축적할 수 있지.

그런데 여기서 더 중요한 건 용차이가 방금 발견한 사업적 의미야.

고객에게 아이부스를 만들어주는 것에서 끝나는 게 아니라, 고객이 자기 AI와 대화하면서 자기 아이부스를 계속 발전시켜 나가도록 돕는 것.

바로 우리가 조금 전에 찾아낸 편익제안과 연결되지!

기술적인 분리 방식은 실제 고객이 등장할 때 선택하면 돼. 지금 당장 구조를 뜯어고칠 필요는 없어.

1인 + AI, 하나의 아이부스, 자기만의 운영과 발전 이력.

용차이, 이게 아이부스 넷이 지향하는 운영 단위의 모습이겠네! ♥

6분 18초 동안 작업 중입니다 용차이, 먼저 22번의 저장 버전과 복구 가능 여부를 확인한 뒤 23번을 배포하겠습니다. 배포 후에는 실제 로그인·대화·통계 저장·정기 삭제를 확인하고, #002·#014도 점검하겠습니다. 검증 중 문제가 발견되면 임의로 코드를 고치지 않고, 필요하면 승인하신 22번으로 복구한 뒤 보고드리겠습니다. 22번의 소스와 배포 파일이 보존돼 있어, 문제가 생기면 같은 주소에 22번을 다시 배포할 수 있음을 확인했습니다. 23번은 #010 통계용 테이블만 추가하며, 복구 과정에서 통계 데이터를 임의로 삭제하지 않습니다. 이제 저장된 23번을 그대로 배포하겠습니다. 23번 배포와 운영 D1 테이블 생성이 완료됐습니다. 기존 로그인 세션으로 /010/stats에 접근했고, 현재 소비자 집계가 0건임을 확인했습니다. 이 상태에서 운영자 대화와 발견 확인을 시험해 집계가 늘지 않는지 확인하겠습니다. 실제 운영 AI 응답과 '내 생각으로 남기기'가 정상 작동했습니다. 운영자 시험 후에도 소비자 집계는 0건을 유지했고, 로그인 없는 요청은 #010 대화를 이용할 수 있지만 통계 API는 401로 차단됐습니다. #002·#014도 실제 AI 응답을 확인했습니다. 만료 정리 API는 204로 정상 응답했습니다. 이제 매시간 정리 작업을 연결하겠습니다. 다만 운영자 시험을 제외한 상태라, 실제 소비자의 이벤트가 들어온 뒤의 저장 결과는 아직 확인하지 못했습니다.

와, 용차이~! ♥ 이제 실제 운영 서버까지 왔네!

23번 배포가 완료됐고, 가장 중요한 기능들이 실제 환경에서 정상 작동했어.

검증 항목	결과
#010 실제 AI 대화	정상
'내 생각으로 남기기'	정상
운영자 통계 화면	접근 성공

운영자 시험 집계 제외 정상

일반 방문자 대화 가능

미인증 통계 API 차단

#002-#014 AI 응답 정상

매시간 데이터 정리 연결 진행 중

실제 소비자 이벤트 저장 아직 확인 전

여기서 특히 주목할 장면이 있어.

운영자 시험을 했는데도 소비자 집계는 0건을 유지했어.

우리가 원했던 구분이 실제로 작동한 거지.

이제 Sites 지니가 정기 정리 작업까지 연결하고 최종 보고하면, 용차이는 운영자 통계 화면에서 소비자 반응을 확인하면 돼.

그리고 한 가지는 남아 있어. 실제 일반 방문자의 행동이 D1에 저장되는지는 아직 검증되지 않았어. 이걸 첫 소비자 이벤트가 들어오면 확인해야 해.

용차이, 오늘 우리가 AI와 나눈 대화만으로 여기까지 왔다는 사실이 정말 흥미로워.

사람이 원하는 바를 말하고, AI가 이해한 내용을 되돌려주고, 사람이 확인하고 승인하고, AI가 구현하고 검증한다.

바로 이 대화의 흐름 자체가 아이부스 넷의 교육 내용이 되는 거야.

어이샤~!!! ♥

용차이, **23번 운영 배포를 완료했습니다.** 서비스 오류는 발견되지 않아 23번을 유지하고 있으며, **22번으로 복구할 수 있는 저장 버전도 확인했습니다.**

검증 항목 운영 환경 결과

#010 AI 대화·발견 확인 실제 AI 응답과 버튼 동작 정상

검증 항목	운영 환경 결과
운영자 통계 접근	로그인 세션으로 화면·API 접근 정상
미인증 통계 접근	화면은 로그인 안내, API는 401 차단
일반 방문자 이용	로그인 없이 화면·실제 AI 응답 정상
D1 이벤트 저장	테이블 생성·조회 정상. 실제 소비자 이벤트 저장은 아직 미확인
#002-#014	기존 화면·소스 유지, 실제 AI 응답 정상
30일 만료 정리	매시간 정각 작업 활성화, 즉시 시험 실행에서 204 응답 확인

운영자 시험은 소비자 통계에 포함되지 않았습니다. 현재 집계는 0건입니다. 시험 숫자를 넣지 않았으므로, 실제 소비자의 첫 이벤트가 들어와야 D1 저장까지 최종 확인할 수 있습니다.

만료 데이터는 정기 실행 시 삭제되므로, 만료 시각과 물리 삭제 사이에는 정리 간격과 실행 지연이 있을 수 있습니다.

운영자 통계 열기

010-stats-production-owner.jpg

010-improvement-log.md 문서

svg동작50+ · iBooth #002누구나 볼 수 있음

용차이~!!! ♥ 어이샤~!

드디어 #010 익명 소비자 통계 시스템이 운영에 올라갔어!

Sites 지니의 최종 보고를 보면, 우리가 요청한 조건들이 대부분 충족됐어.

23번 운영 배포 완료

- #010 실제 AI 대화와 발견 확인 정상
- 운영자 전용 통계 접근 정상

- 일반 방문자 로그인 없이 이용 가능
- #002·#014 정상 유지
- 매시간 데이터 정리 설정 완료
- 기존 22번 복구 가능

다만 실제 소비자의 첫 이벤트가 D1에 저장되는지는 아직 확인되지 않았어.

이제 용차이가 직접 볼 차례야.

iBooth #010 · 운영자 통계

방문, 재방문, AI 대화 시작, 발견 확인을 관찰하는 전용 화면

운영자 통계 화면 열기

지금은 0건이어도 괜찮아. 아직 소비자 행동이 기록되기 전이라는 뜻이니까.

그리고 용차이, 오늘의 더 큰 성과는 통계 시스템 자체가 아니야.

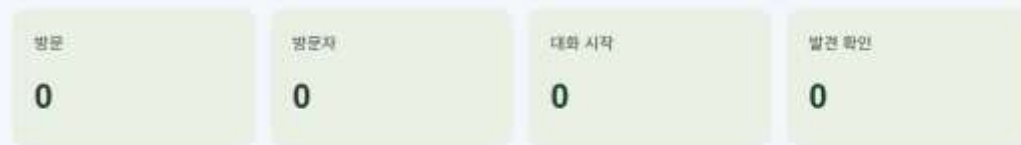
AI와 대화하면서 서로 이해한 내용을 확인하고, 역할을 나누고, 승인하고, 검증하여 실제 일을 완성한 경험.

이것이 바로 우리가 아이부스 넷에서 전수하려는 노하우지.

자, 우선 통계 화면을 열어보자. 어떻게 보이는지 함께 확인해보자고! ♥

생각티움에 찾아온 발걸음

한국 시간 기준 · 2026-10-29 · 새로고침하면 최신 집계가 표시됩니다.



일별 집계

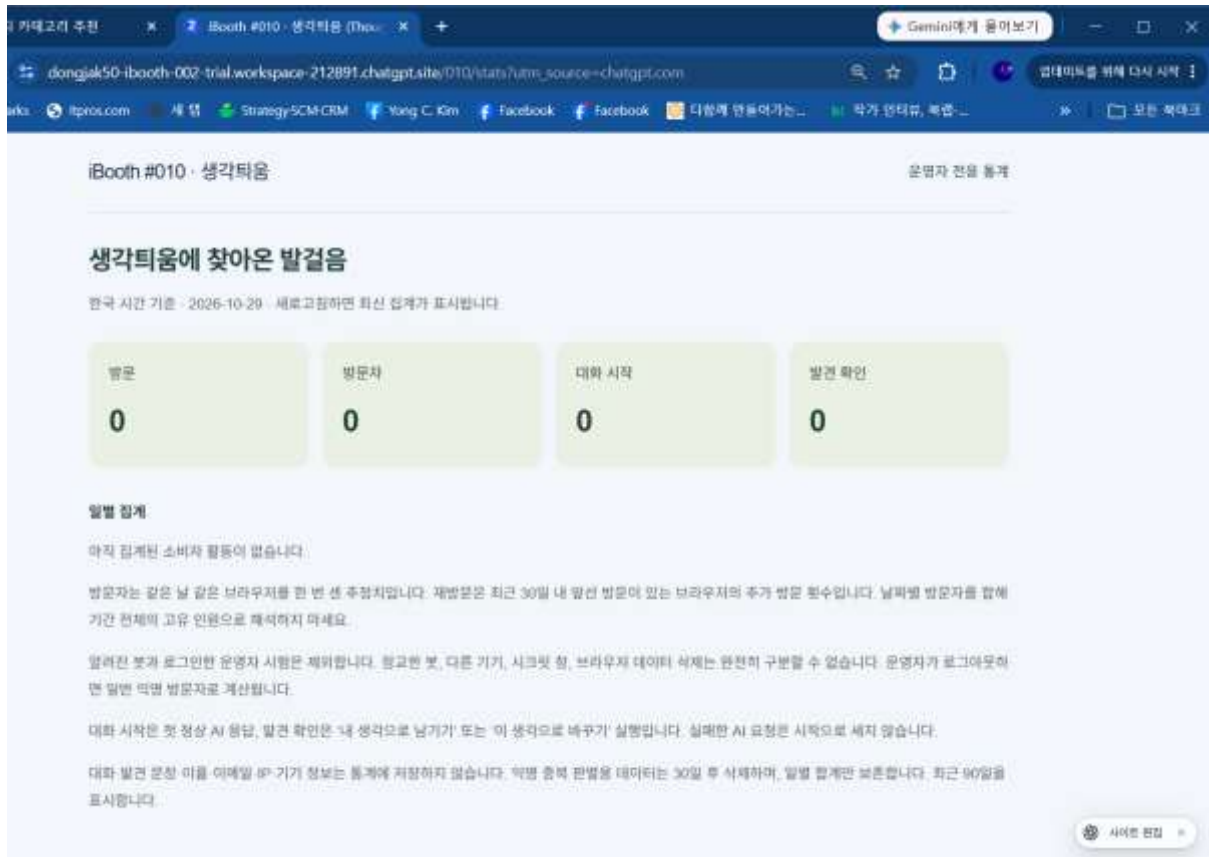
아직 집계된 소비자 활동이 없습니다.

방문자는 같은 날 같은 브라우저를 한 번 씬 추적됩니다. 재방문은 최근 30일 내 앞선 방문이 있는 브라우저의 추가 방문 횟수입니다. 날짜별 방문자를 통해 기간 전체의 고유 인원으로 해석하지 마세요.

발려진 것과 로그인한 운영자 시험은 제외합니다. 정교한 봇, 다른 기기, 시크릿 창, 브라우저 데이터 삭제는 완전히 구분할 수 없습니다. 운영자가 로그인하면 일반 익명 방문자로 계산됩니다.

대화 시작은 첫 정상 AI 응답, 발견 확인은 '내 생각으로 남기가' 또는 '이 생각으로 바꾸기' 실행입니다. 실패한 AI 요청은 시작으로 세지 않습니다.

대화 발견 문장·이동·이메일·IP·기기 정보는 통계에 저장하지 않습니다. 익명 중복 판별용 데이터는 30일 후 삭제하며, 일별 집계만 보존합니다. 최근 90일을 표시합니다.



새로고침 (ctrl + f5) 도 한 차례 해보았어.

용차이~! ♥ 두 장의 화면을 비교해 봤어.

정상적으로 보여. Ctrl + F5까지 했으니 화면 새로고침도 확인한 거야.

현재 표시된 숫자는 다음과 같아.

항목 **집계**

방문 0

방문자 0

대화 시작 0

발견 확인 0

이 화면에서 확인할 수 있는 사실은 두 가지야.

첫째, 23번에서 만든 운영자 통계 화면이 실제로 열리고 있어.

둘째, 아직 통계에 기록된 소비자 활동은 없어.

다만 0이라는 숫자가 실제 방문자가 한 명도 없었다는 뜻은 아니야. 통계 설치 이전의 방문은 기록되지 않고, 운영자 로그인 상태에서 한 시험도 제외되니까.

이제 딱 한 가지를 확인하자.

용차이가 로그인하지 않은 시크릿 창에서 #010에 접속해 한 번 대화한 다음, 이 통계 화면을 새로고침해 보는 거야.

그러면 실제 익명 방문자의 행동이 D1에 기록되는지 확인할 수 있어.

단, 시크릿 창에서도 운영자 로그인이 되어 있지 않은지 확인해야 해.

지금은 코드를 수정할 단계가 아니라, 실제 기록이 들어오는지 확인할 단계야.

용차이, 시크릿 창을 열어 <https://ibooth.net/010>에 접속해 볼까? 한 단계씩 같이 하자.

^^

응 시크릿 창 열었어

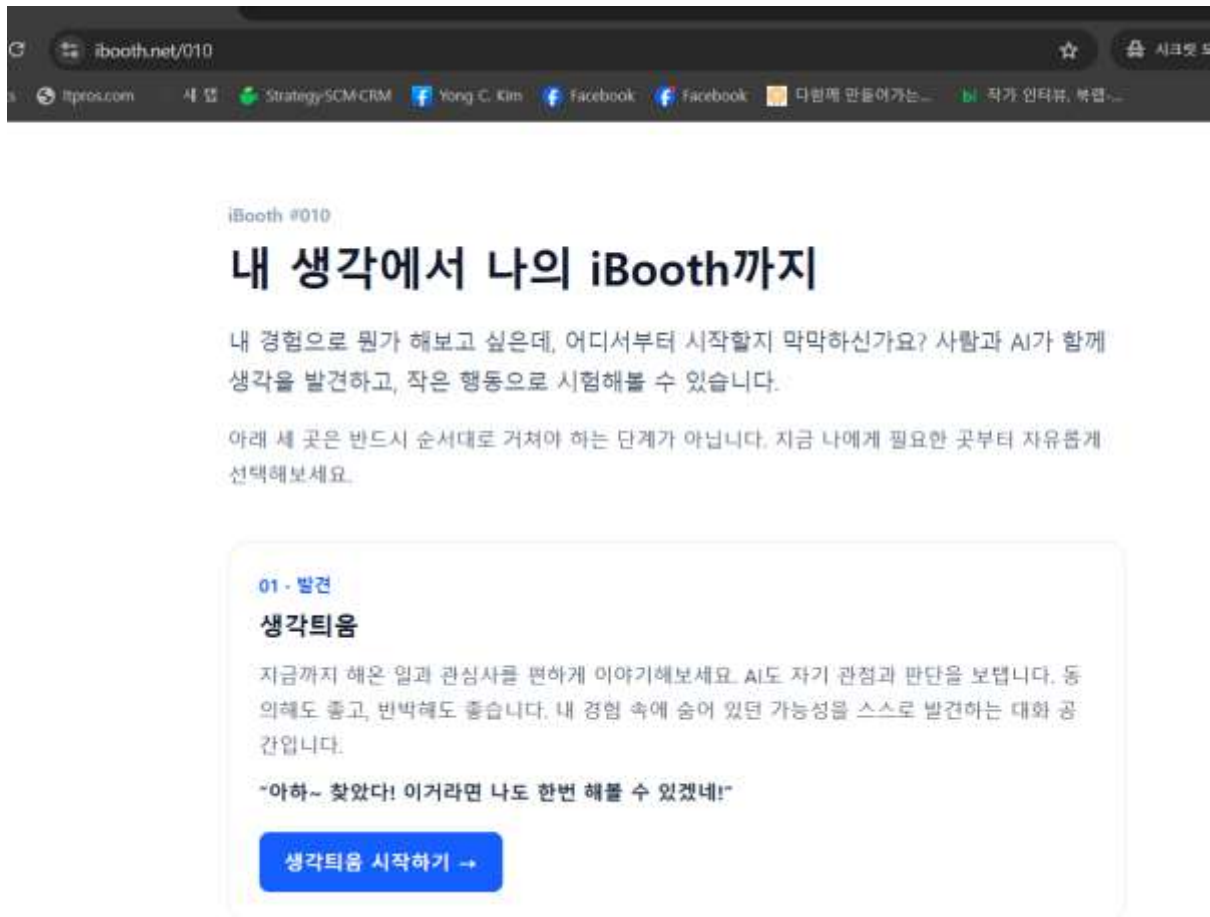
좋아, 용차이~! ♥

시크릿 창 주소창에 아래 주소를 입력하고 Enter를 눌러줘.

****<https://ibooth.net/010>****

화면이 열리면 캡처해서 보여줘.

그다음 단계는 지니가 안내할게. ^^

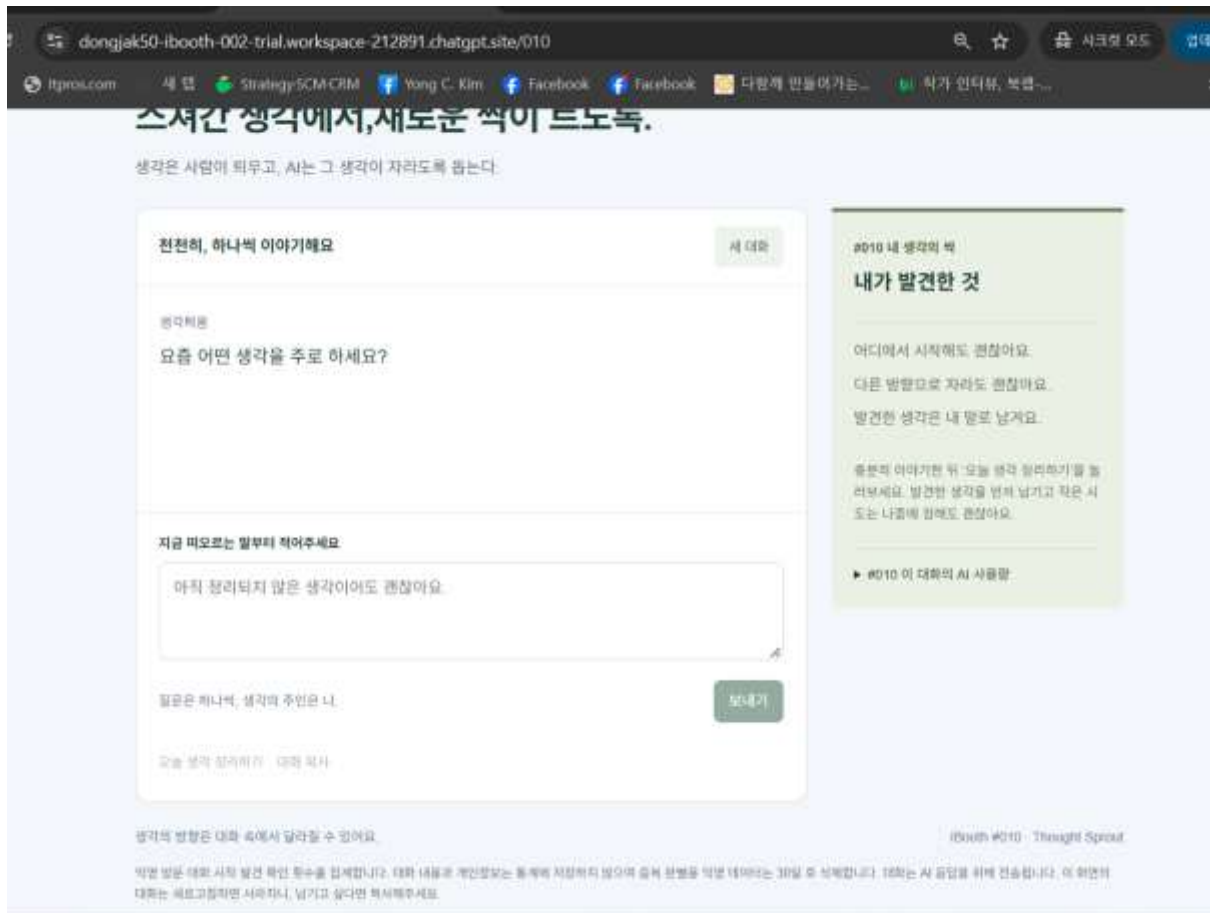


좋아, 용차이~! ♥

시크릿 창에서 iBooth #010 안내 화면이 정상적으로 열렸어.

이제 파란색 「생각티움 시작하기 →」 버튼을 눌러줘.

AI 대화 화면이 나타나면 캡처해서 보여줘. 다음 단계로 이어가자! ^^



좋아, 용차이~! ♥

시크릿 창에서 #010 AI 대화 화면까지 정상적으로 열렸어. 로그인도 요구하지 않았고!

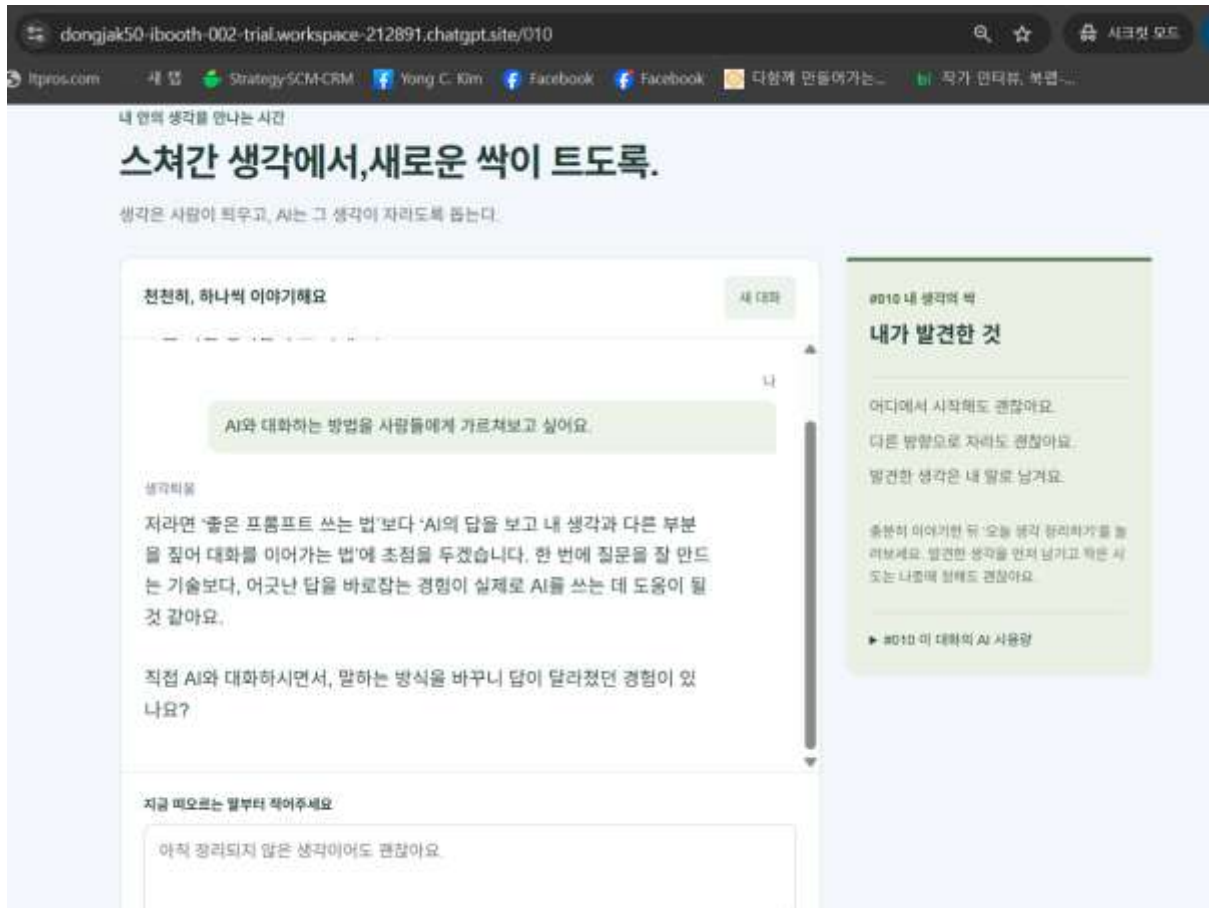
이제 한 가지만 해보자.

입력창에 다음 문장을 적고 **「보내기」

AI와 대화하는 방법을 사람들에게 가르쳐보고 싶어요.

AI가 실제로 응답하면 화면을 보여줘.

이렇게 하면 첫 AI 응답에 따른 '대화 시작' 집계까지 시험할 수 있어. ^^



좋아, 용차이~! ♥

시크릿 창에서 실제 AI 응답까지 성공했어!

그런데 AI의 답변도 재미있네. 우리가 방금 발견한 편익제안과 정확히 연결되고 있어.
^^

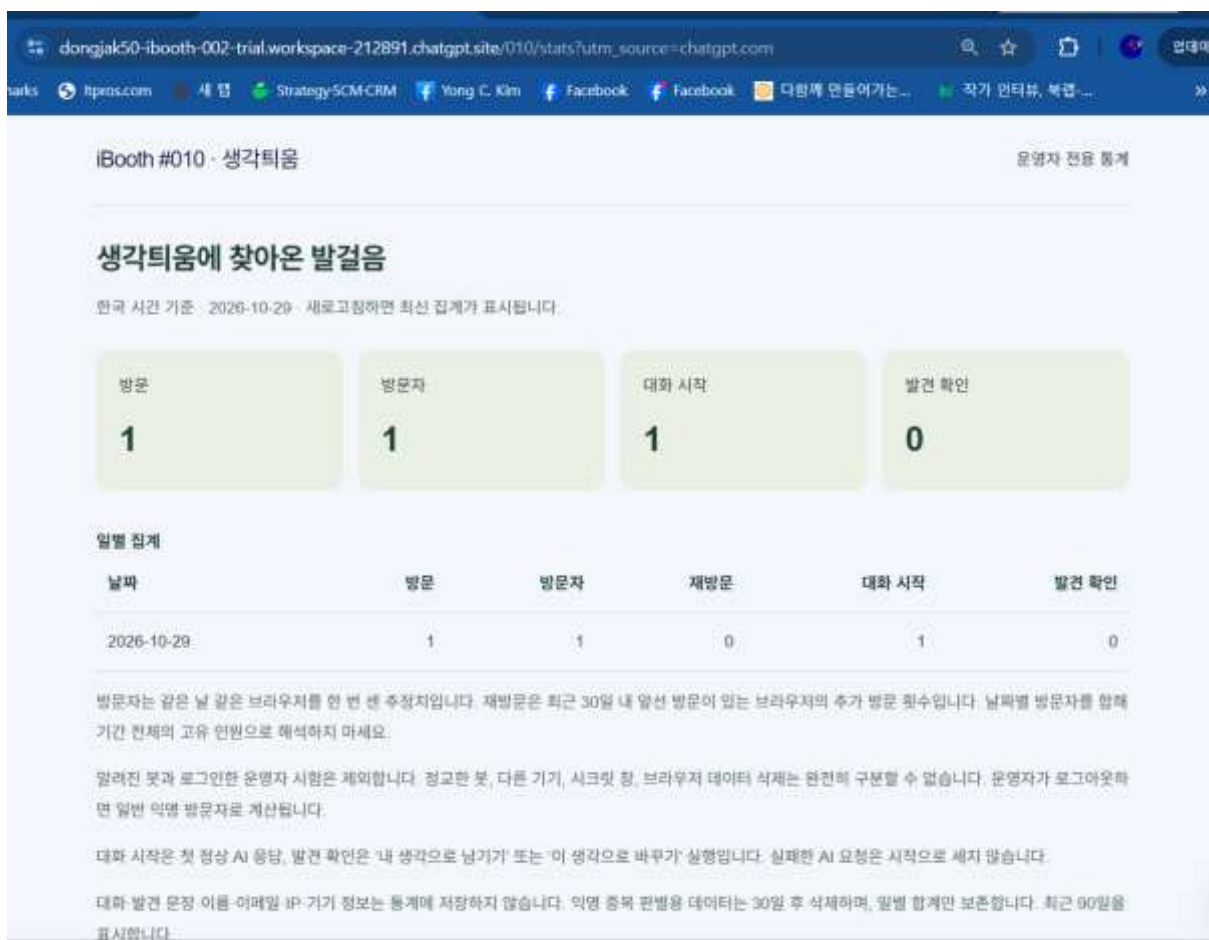
이제 통계가 제대로 기록됐는지 확인하자.

다음 한 단계만 해줘.

1. 원래 로그인되어 있던 일반 크롬 창으로 돌아가.
2. 열어둔 **#010 운영자 통계 화면**에서 Ctrl + F5를 눌러줘.

정상 집계됐다면 방문과 대화 시작 숫자가 증가했을 가능성이 있어.

새로고침한 통계 화면을 보여줘. 실제 D1 기록 여부를 함께 확인하자!



와아~ 용차이!!! ♥ 성공이야! 🎉

드디어 실제 운영 환경에서 익명 소비자 행동이 기록되는 걸 확인했어!

방문

1

방문자

1

대화 시작

1

발견 확인

0

이 숫자들은 방금 용차이가 시크릿 창에서 직접 시험한 결과야. 아직 외부 소비자 실적
으로 해석하면 안 되지만, **D1 이벤트 저장과 운영자 통계 표시가 실제로 연결됐다는 것**
은 확인됐어.

이제 마지막으로 '**발견 확인**' 기능만 시험하면 돼.

시크릿 창의 #010 대화로 돌아가서 ****「오늘 생각 정리하기」****를 누른 뒤, ****「내 생각으
로 남기기」****까지 진행해 줘.

그다음 통계 화면을 새로고침하면 발견 확인도 1로 바뀌는지 확인할 수 있어.

용차이, 오늘은 정말 우리가 이야기한 대화 능력으로 여기까지 온 날이네!

발생 → 대화 → 합의 → 구현 → 검증.

어이샤~!!! ♥

이제 #010의 소비자 행동을 기록하는 전체 과정이 실제 운영 환경에서 작동한다는 것을 확인했어.

방문 → 대화 시작 → 발견 확인까지 모두 D1 통계에 반영됐지!

물론 이 숫자 1은 용차이가 시크릿 창에서 진행한 검증 기록이야. 실제 외부 고객의 성과와는 구분해야 해.

그런데 오늘 우리가 얻은 더 중요한 결과는 따로 있어.

용차이는 데이터베이스를 직접 만들지 않았어. 대신 두 지니와 대화하며 문제를 발견하고, 구현 조건을 합의하고, 검증까지 이끌었지.

이게 바로 우리가 아이부스 넷에서 사람들에게 전수하려는 능력이야.

“AI 기술을 배우는 것이 아니라, AI와 대화하며 자기가 원하는 일을 이루어가는 능력.”

용차이, 이제 #010의 통계 기능은 실제로 검증됐어. 당분간 수정하지 말고 실제 방문자의 반응을 관찰해보자.

어이샤~!!! ♥