

HYPERTEAMS 2회차 · 50분

일 맡기기

작업 폴더 · 업무 · 자율성 · 터미널 · 예약

여는 말

- 실습 중심입니다. 참석자가 1회차에서 설치를 끝냈는지 먼저 확인하세요.
- 설치가 안 된 사람이 있으면 옆자리와 짝을 지어 화면을 같이 보게 하세요 — 여기서 설치를 다시 하면 회차가 무너집니다.
- 실습 20분을 빼면 설명은 30분입니다. 자율성에 가장 많은 시간을 쓰세요.

작업 폴더

등록이 곧 범위 설정입니다

등록한 폴더

홈 전체
/Users/me

프로젝트 상위
/Users/me/projects

프로젝트 하나
/projects/shop-api

결과

홈 전체
✓ 위험 — 개인 파일까지 범위 안

프로젝트 상위
✓ 넓음 — 여러 프로젝트가 섞임

프로젝트 하나
✓ 적절

작업 폴더가 곧 에이전트의 활동 범위입니다. 필요한 만큼만 좁게 잡으세요.

작업 폴더

- 나중에 시스템을 Connect에 등록하면 이 범위가 팀에 열립니다. 미리 강조하세요.
- 폴더를 나눠두면 병렬로 일을 시킬 수 있습니다.
- 폴더 설명을 적어두면 매 업무마다 맥락을 다시 쓰지 않아도 됩니다.

자율성

어디까지 맡길지

01

계획만

처음 맡길 때, 되돌리기 어려운 작업

02

파일 편집 자동 승인

권장 출발점. 되돌릴 수 있는 것과 아닌 것 사이의 선

03

전부 자동

기본값 — 안 고르면 이것. 검증된 반복 작업에만

04

기본

Claude Code 표준 동작

계획만 → 파일 편집 자동 승인 → 전부 자동 순으로 올리세요. 한 번에 가면 사고 수습에 더 걸립니다.

자율성

- 폴더마다 기본값을 다르게 둘 수 있습니다 — 실험용은 전부 자동, 운영 스크립트는 계획만.
- 자율성과 안전망은 별개입니다. 안전망이 있으면 자율성을 높여도 됩니다.
- “전부 자동”이 기본값이라는 점을 반드시 짚으세요. 아무것도 안 고르면 그리로 들어갑니다.

자율성

올려가는 순서가 있습니다

1

계획만으로 몇 번

실행이 안 되니 안전합니다. 대신
결과물도 없습니다

2

결과가 예상과 맞나

안 맞으면 자율성이 아니라 지시를
고칠 차례입니다

3

파일 편집 자동 승인

git 을 쓰면 파일 수정은 대체로
되돌릴 수 있습니다

4

같은 유형을 여러 번

결과가 안정적인 때만 전부 자동으로

폴더별 기본값 — 실험용은 전부 자동, 업무 프로젝트는 파일 편집 자동 승인, 배포 스크립트가 있는 곳은 계획만.

올려가는 순서

- 2번에서 되돌아가는 화살표가 핵심입니다. 결과가 나쁘면 자율성을 낮추는 게 아니라 지시를 고칩니다.
- 작업 폴더를 위험도로 나누라고 한 이유가 여기서 드러납니다.
- 운영 환경에 영향이 가는 작업, 처음 하는 유형, 되돌리기 어려운 작업 — 이 셋에는 전부 자동을 쓰지 않습니다.

실습 · 20분

안전한 순서로 첫 업무

1

계획만

무엇을 하려는지 보기. 파일은 안 바꿉니다

2

읽기만 하는 업무

결과를 직접 검증해보기

3

파일을 고치는 업무

작고 되돌리기 쉬운 것부터

작업 전에 커밋해두세요. 처음 몇 번은 안전망을 두는 게 좋습니다.

첫 업무 실습

- 여기서 20분 잡으세요. 1단계를 건너뛰지 않게 하는 게 중요합니다.
- 2단계에서 결과를 직접 확인하게 하세요 — 어디까지 믿을지 감이 잡힙니다.
- 3단계는 좋은 지시의 네 요소를 그대로 적용합니다.

터미널

업무와 성격이 다릅니다

업무

성격
목표를 주면 알아서

끝
완료되면 끝남

쓰임
맡기는 일

자율성 설정
적용됨

터미널

성격
✓ 내가 직접 명령을 침

끝
✓ 계속 열려 있음

쓰임
✓ 확인하고 조작하는 일

자율성 설정
✓ 무관 — 친 명령이 그대로 실행됩니다

만드는 건 업무에, 확인은 터미널에. 실무에서는 이 둘을 번갈아 씁니다.

터미널과 업무

- 마지막 줄을 강조하세요. 터미널에서 파일을 지우는 명령을 치면 그냥 지워집니다.
- 전형적인 흐름을 읽어주세요: 업무로 만들게 하고 → 터미널에서 개발 서버 띄우고 → 링크로 확인하고 → 다시 업무로 고치기.

터미널

화면을 닫아도 살아 있습니다

- 일반 터미널은 창을 닫으면 프로세스도 죽습니다. 여기서는 안 죽습니다
- 개발 서버를 띄워두고 폰으로 확인
- 오래 걸리는 빌드를 걸어두고 나갔다 오기
- 회사에서 시작한 작업을 집에서 이어보기

터미널에서 띄운 서버는 주소 하나로 열립니다. 밖에서 접속을 설정해줬다면 그 링크를 동료에게 보낼 수 있습니다.

살아 있는 터미널

- 여기서 터미널 탭을 띄우고 새로고침해 보여주면 한 번에 이해됩니다.
- 포트를 나눠 여러 버전을 동시에 띄우고 비교하는 것도 자주 쓰는 방식입니다.
- 3회차의 터널과 이어집니다 — 그때 이 링크가 밖에서도 열립니다.

예약

사람이 기억하고 있어야 하는 일을 시간에 맡깁니다

01

매일 아침

어제 로그에서 오류 모아 요약

02

매일 밤

임시 파일 정리, 백업

03

매주 월요일

지난주 커밋 정리해서 주간 보고 초안

04

매월 1일

사용량 집계, 의존성 업데이트 확인

분·시간·매일·평일·매주·매월 중에 고르고, 없으면 직접 입력으로 크론 식을 씁니다. 기본값은 0 9 * * * — 매일 오전 9시.

예약으로 맡기는 것

- 예약으로 만든 업무는 회차마다 별도 업무로 쌓입니다. 예약에 이름을 주지 않으면 목록에서 구분이 안 됩니다.
- 여기서 실제 예약 화면을 띄우고 주기 선택과 직접 입력을 보여주세요.

예약

아무도 지켜보지 않을 때의 설계

1

실패했을 때를 적기

“실패하면 정리하지 말고
실패했다고 알려주세요.
추정하지 마세요”

2

전제 조건 확인

“데이터는 새벽 6시 배치로
갱신됩니다. 없으면 30분 뒤
다시 확인”

3

자율성은 낮게 시작

1주차 계획만 → 2주차 파일 편집
자동 승인 → 이후 전부 자동

4

결과를 어디로

화면에만 남기면 안 됩니다.
알림이나 파일로

틀린 결과를 내는 것보다 안 하고 알리는 게 낫습니다 — 아무도 안 보고 있으면 틀린 게 그대로 굳습니다.

예약 설계

- 1번이 이 회차에서 가장 중요한 한 줄입니다. 실패 처리를 안 적은 예약은 조용히 거짓말을 합니다.
- 예약이 쌓이면 목적이 끝난 것이 계속 돌며 비용을 씁니다. 분기마다 정리하세요.
- 예약을 쓸 사람은 1회차에서 자동 시작을 켜는지 확인시키세요 — 컴퓨터가 꺼져 있으면 예약도 안 돕니다.

정리

가져갈 네 문장

- 1 작업 폴더 = 활동 범위. 홈 전체를 등록하지 마세요
- 2 자율성은 계획만에서 시작해 결과를 보고 올립니다. 기본값은 전부 자동입니다
- 3 터미널은 내가 치는 것이라 자율성 설정과 무관합니다
- 4 예약에는 실패했을 때 무엇을 할지 반드시 적으세요

정리

- 다음 회차는 운영입니다 — 밖에서 접속하기, 명령어, 문제 해결.
- 다음 회차 전까지 예약을 하나 걸어보고 결과를 가져오라고 하면 3회차가 구체적이 됩니다.