

HYPERTEAMS 5회차 · 120분

실전 자동화

예약으로 파이프라인을 만들고, 절차를 스킴으로 담고, 자료를 모읍니다

예약은 두 종류입니다

task

무엇을

에이전트에게 맡김

언제

매번 판단이 필요

성격

비싸고 매번 다름

예

원고 작성 · 로그 분석

terminal

무엇을

✓ 명령을 그대로 실행

언제

✓ 정해진 명령

성격

✓ 싸고 결정적

예

✓ 수집 · 형식 변환 · 렌더 · 백업

판단 없는 단계를 task 로 두는 것이 운영비의 대부분입니다. 규칙으로 100% 표현되면 AI를 쓸 이유가 없습니다.

단계를 잇는 법

세 가지 방법과 네 가지 원칙

01

시차를 둔다

가장 단순. 앞이 늦으면 뒤가 빈손으로 옵니다

02

앞이 뒤를 부른다

빈손 실행이 없음. 열쇠는 환경변수로

03

한 업무가 전부

권하지 않음 — 컨텍스트가 차서 뒤로 갈수록 흐려집니다

원칙 넷 — 단계마다 파일로 남기기 · 이미 있는 것은 건너뛰기 · 실패를 알리기 · 자율성은 낮게 시작.

스킬

절차를 어디에 담나

재사용 단위

지시문 템플릿
매번 붙여넣는 글
서브 에이전트
역할 · 말투 · 규칙
스킬
절차와 그에 딸린 자료

사는 곳

지시문 템플릿
✓ 내 메모장 — 그 사람 것입니다
서브 에이전트
✓ Connect 워크스페이스
스킬
✓ 작업 폴더 안의 파일 — git 으로 팀이 공유

SKILL.md 하나면 됩니다. 폴더가 곧 상태라, 폴더를 복사하면 절차가 따라옵니다.

스킬 · 만들지 말 때

스킬은 절차이지 도구가 아닙니다

- 한 번 쓰고 말 것 → 그냥 대화로
- 지시가 매번 크게 달라짐 → 대화로 좁혀가기
- 팀 전체가 쓰는 말투·역할 → Connect 서브 에이전트
- 외부 시스템을 부르는 것 → MCP 도구

절차가 바뀌었는데 스킬이 그대로면 틀린 절차를 자신 있게 반복합니다. 점검 목록을 스킬 안에 남기세요.

문서

PDF와 나머지는 사본의 역할이 다릅니다

PDF

원본을 직접 읽나
읽습니다 (쪽 범위로)

사본의 역할
보조 — 검색 · 요약용

표 · 그림
원본을 봐야 정확

docx · xlsx · pptx

원본을 직접 읽나
✓ 못 읽습니다

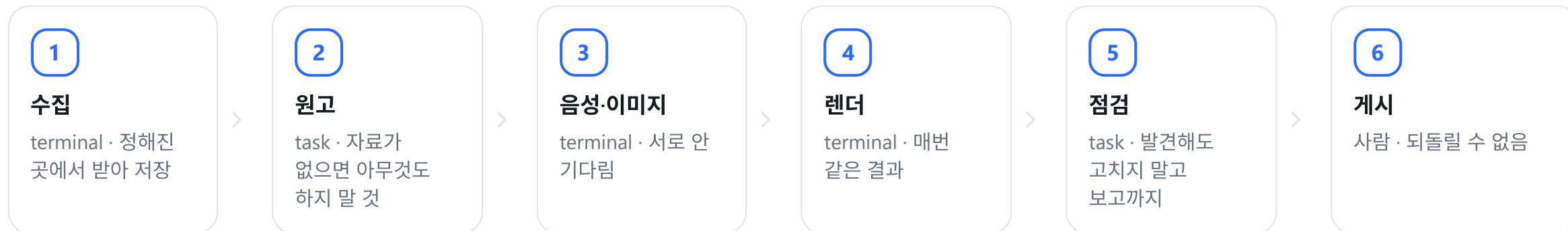
사본의 역할
✓ 전부

표 · 그림
✓ 사본에 담긴 만큼만

엑셀은 사본이 전부라 사본이 부실하면 그걸로 끝입니다 — 중요한 수치는 사람이 대조하세요.

사례

콘텐츠 공장 — 6단계



단계마다 파일로 남기면 실패한 곳부터 다시 돌릴 수 있습니다. 폴더는 소스와 렌더로 나눕니다.

처음 2주

한 번에 자동화하지 마세요

시기

1~3일

손으로

4~7일

예약으로

2주차

자율성 올리기

이후

무인

하는 일

1~3일

✓ 각 단계를 한 번씩 실행. 어디서 막히는지 확인

4~7일

✓ 걸되 매일 결과를 눈으로 확인

2주차

✓ 점검 단계의 보고만 확인

이후

✓ 실패 알림이 올 때만 개입

1~3일을 건너뛰면 여섯 단계가 동시에 처음 도는 걸 보게 되고, 어느 단계가 원인인지 못 가립니다.

수집

브라우저는 마지막 순서입니다

1

공식 API

있으면 여기서 끝

2

피드

RSS · 사이트맵 · JSON 엔드포인트

3

정적 HTML

fetch + 파서

4

브라우저

스크립트가 돌아야 나올 때만

브라우저는 가장 비싸고 가장 자주 깨집니다 — 상대가 마크업을 조금만 바꿔도 멈춥니다.

스텔스를 없으면 오히려 더 눈에 떨 수 있습니다

Crawlee 는 자체 지문 생성기가 기본으로 켜져 있습니다 — 둘이 같은 값을 각자 덮어씹습니다

받는 순서가 곧 비용입니다

필요한 것

무슨 말을 했나
내용

제목 · 길이 · 조회수
메타데이터

자막이 없는 영상
내용

화면에 나온 것
자료 화면 · 제품

받을 것 · 크기

무슨 말을 했나
✓ 자막 — KB

제목 · 길이 · 조회수
✓ JSON — KB

자막이 없는 영상
✓ 오디오만 — MB

화면에 나온 것
✓ 영상 — 수백 MB

대부분의 업무는 첫 두 줄에서 끝납니다. 도구는 yt-dlp 하나이고 ffmpeg가 함께 필요합니다.

공문서

hwp 는 읽기 전용, hwpx 는 쓰기까지

01

읽기

hwp · hwpx 모두 마크다운으로. 표는 HTML `<table>` 로 나옵니다

02

만들기

마크다운으로 내용만 쓰면 항목부호 8단계와 표준 서식은 맡깁니다

03

서식 채우기

무엇을 채울 수 있는지 먼저 보고, 값은 JSON 파일로 넘깁니다

04

고치기

원본 서식을 유지한 채, .hwp 는 이 자리에서 안 됩니다

넘기기 전에 validate 로 구조를 검사하세요. 제출은 사람이 합니다 — 되돌릴 수 없기 때문입니다.

실습 · 60분

파이프라인 하나를 끝까지 만들어보기

1

세 단계 그리기

각 단계가 **무엇을
파일로 남기는지**
적습니다

2

1단계만 먼저

파일이 생긴 것과 내용이
맞는 것은 다릅니다

3

2·3단계 잇기

간격은 평소 소요의
3배 이상

4

가운데 깨뜨리기

1단계 산출물을 지우고
뒷단계가 어떻게 되는지

5

비용과 정리

세 단계 × 매일이면 한 달에
90번입니다

가장 위험한 결과는 “예전 파일을 찾아 쓰는 것”입니다 — 오류가 안 나서 아무도 모르고, 어제 데이터로 오늘 보고가 나갑니다.

정리

가져갈 것

- 1 판단이 있으면 task, 없으면 terminal — 여기가 운영비의 대부분
- 2 단계마다 파일로 남긴다. 그래야 실패한 곳부터 다시 돈다
- 3 스킬은 폴더에 산다 — git 으로 팀이 공유하고 되돌릴 수 있다
- 4 문서는 사본을 먼저 만들고 검색으로 좁힌 뒤 읽힌다
- 5 수집은 API → 피드 → HTML → 브라우저 순서로만 내려간다
- 6 영상은 메타데이터 → 자막 → 오디오 → 영상 순서로만 내려간다
- 7 되돌릴 수 없는 동작 앞에만 사람을 세운다