

바이브 메이커 · 무료 자료

클로드 워터마크 정리 가이드

클로드가 결과물에 넣는 표시가 무엇인지 알고, 오픈소스 리무버로 내 파일과 글을 점검하고 정리하는 법. 정리 전후를 비교하는 세팅과 앞으로 대비할 것까지.

앤트로픽 표시 방식 · 설치 · 서비스 켜기 · 클로드코드에서 쓰기 · 전후 비교 · 한계 · 대응

목차

한눈에	이 자료로 할 수 있는 것	03
표시 방식 앤트로픽 도움말	클로드가 넣는 표시 두 가지	04
리무버가 하는 일 README 원문	watermarks-remover 는 세 층으로 나눠 다룹니다	05
설치 README 원문	저장소를 받고 스킬을 깝니다	06
서비스 켜기 README · win...	스킬이 일을 맡길 서비스를 켭니다	07
클로드코드에서 부르기 SKILL...	점검 먼저, 정리는 그다음	08
글 정리 README 원문	글은 왜 어렵고, 무엇을 설정하나	09
전후 비교 세팅	정리 전후를 탐지기로 비교하기	10
한계	정리해도 남는 것, 지워도 모르는 것	11
대응 · 쓰는 범위	앞으로 대비할 것, 내 콘텐츠에만 쓰기	12
막히면	증상 → 할 일	13

한눈에

이 자료로 할 수 있는 것

준비물	클로드코드 · 파이썬 3.10 이상 · git (맥 · 리눅스 · 윈도 모두 됩니다)
쓰는 도구	**watermarks-remover** · 깃허브 공개 저장소 · MIT 라이선스 · 별 24,584개 (2026-10-21 기준) · 클로드코드 스킬 `/remove-ai-marks` + 내 컴퓨터에서 도는 작은 서비스
끝나면 남는 것	파일 속 표시(C2PA · EXIF · XMP · 문서 속성) 정리본 · 글 속 숨은 글자 정리본 · 정리 전후 점검 보고서

순서

1. **표시가 어떻게 들어가는지 안다**
   글에는 워터마크, 파일에는 C2PA 꼬리표. 4쪽 · 5쪽
2. **리무버를 설치한다**
   저장소 받기 → 스킬 설치. 6쪽
3. **서비스를 켠다**
   스킬은 이 서비스에 일을 맡긴다. 7쪽
4. **클로드코드에서 부른다**
   점검 먼저, 정리는 그다음. 8쪽 · 9쪽
5. **정리 전후를 비교한다**
   점검 보고서 · 공식 확인 도구. 10쪽
6. **한계와 쓰는 범위를 지킨다**
   11쪽 · 12쪽

> **먼저 알아 둘 것.** 글에 들어가는 클로드 워터마크는 **낱말 선택 자체**에 퍼져 있어서, 숨은 글자를 지우는 정리로는 사라지지 않습니다. 글을 다시 써야만 줄어들고, 그것도 결과를 보장할 수 없습니다(9쪽 · 11쪽). 파일 꼬리표 정리는 확인이 되는 작업입니다.

표시 방식 앤트로픽 도움말

클로드가 넣는 표시 두 가지

앤트로픽은 EU 인공지능법 50조 2항의 「AI 생성 콘텐츠 투명성 실천 강령」에 서명했고, 그에 따라 클로드 결과물에 기계가 읽는 표시를 넣습니다. 도움말 문서 「How Claude marks AI-generated content」에 적힌 내용입니다.

종류	어디에 · 어떻게
① 글 워터마크	글 자체에 눈에 안 보이게 넣습니다. 뜻 · 품질 · 읽기에는 영향이 없다고 합니다. **글의 일부라서 복사 · 붙여넣기 해도 따라가고, 어느 정도 고쳐도 남을 수 있습니다.** 모델 단계에서 넣기 때문에 어느 클로드 제품에서 나온 글이든 같습니다.
② 콘텐츠 자격증명 (C2PA)	PNG · JPEG 처럼 지원하는 파일을 클로드가 만들면 파일에 출처 정보가 붙습니다.

어디서 나온 결과물에 붙나

- Claude Platform(API) · Claude · **Claude Code** · Claude Cowork · Claude Tag, 그리고 클로드를 제공하는 모든 곳 · 전 세계
- 2026년 8월 2일 이후 나온 모델은 처음부터 표시합니다. 그 전 모델에도 표시를 넣는 중이고, 도움말의 모델 표에는 Opus 4.5 · Sonnet 4.5 · Haiku 4.5 까지 글 워터마크가 표시돼 있습니다.
- 파일 꼬리표(C2PA)는 최신 모델(Opus 5.5 · Sonnet 5.5 · Haiku 5.5 · Fable 5.1 등)에 표시돼 있습니다.

누가 확인할 수 있나

파일에 클로드 콘텐츠 자격증명이 있는지는 누구나 무료 「Claude Content Checker」로 봅니다. **글 워터마크 탐지**는 아직 비공개 미리보기라, EU 법이 정한 기관(규제 기관 · 수사 기관 · 언론 · 팩트체커 · 연구자 · 교육 기관 등)과 같은 의무가 있는 기업만 씁니다.

복사해서 그대로 쓰세요

```
https://claude.com/check-content
```

리무버가 하는 일 README 원문

watermarks-remover 는 세 층으로 나눠 다룹니다

여러 회사의 AI 출처 표시를 글과 파일에서 걷어 내는 오픈소스입니다. 저장소 첫 줄이 용도를 밝힙니다. **「내가 가진 콘텐츠의 개인정보 · 위생을 위해」**. 클로드코드 스킬은 얇은 손잡이고, 실제 일은 내 컴퓨터에서 도는 파이썬 서비스가 합니다.

층	무엇을	확실한가
A · 숨은 글자	보이지 않는 유니코드 · 특이한 공백 · 방향 제어 문자 · 태그 문자	**확인됨** 규칙대로 지우고 개수를 셉니다
파일 꼬리표	C2PA · EXIF · XMP · 문서 속성	**확인됨** 무엇을 지웠는지 보고서에 남습니다
B · 다시 쓰기	낱말 선택에 퍼진 통계적 워터마크를 다른 모델로 다시 써서 흐리게	**최선 노력** 통과를 보장 못 합니다

다루는 파일 21종

PNG · JPEG · WebP · AVIF · HEIC · BMP · GIF · TIFF · SVG · PDF · DOCX · XLSX · PPTX · EPUB · ODT · HTML · Markdown · MP4/MOV/M4A/M4V · WAV · MP3 · FLAC

> **PDF 는 도구가 더 필요합니다.** 서버 쪽에 `exiftool` 과 `qpdf` 가 있어야 제대로 정리되고, 문서 안 그림에 든 정보까지 지우려면 `ghostscript` 도 필요합니다. 없으면 보고서에 「일부만」이라고 나옵니다.

범위 밖 SKILL.md

오디오 워터마크 · 영상 SynthID · **C2PA 소프트 바인딩**(꼬리표를 지워도 그림 속 신호로 원래 정보에 다시 연결되는 방식) · 비밀 키 탐지기 · 학습 단계 백도어는 다루지 않습니다. 꼬리표를 지운 뒤에도 회사 도구가 미디어 워터마크를 찾을 수 있습니다.

설치 README 원문

저장소를 받고 스킬을 깝니다

서비스가 저장소 안에서 돌기 때문에 **저장소는 어느 경로로 설치하든 받아 둡니다.** 터미널(윈도는 PowerShell)에서 차례로 칩니다.

맥 · 리눅스

복사해서 그대로 쓰세요

```
git clone https://github.com/guillaumemeyer/watermarks-remover.git
cd watermarks-remover
python3 install_skill.py --skill remove-ai-marks --target claude-code
```

윈도

복사해서 그대로 쓰세요

```
git clone https://github.com/guillaumemeyer/watermarks-remover.git
cd watermarks-remover
py install_skill.py --skill remove-ai-marks --target claude-code
```

1. **개인 설치 `--target claude-code`**
   ~/.claude/skills/ 에 들어가 모든 프로젝트에서 씁니다

2. **이 프로젝트에만 `--target claude-project --project-dir 경로`**
   그 프로젝트의 .claude/skills/ 에 들어갑니다

3. **재시작 없이 바로**
   클로드코드에서 `/skills` 를 치면 불러온 스킬 목록에 보입니다

플러그인으로 깔 때

클로드코드 안에서 두 줄을 칩니다. 이 경우 스킬 이름이 `/watermarks-remover:remove-ai-marks` 로 붙고, 다른 스킬이 같은 이름을 안 쓰면 `/remove-ai-marks` 로도 됩니다.

복사해서 그대로 쓰세요

```
/plugin marketplace add guillaumemeyer/watermarks-remover
/plugin install watermarks-remover@watermarks-remover
```

> 파이썬 표준 라이브러리만 씁니다(3.10 이상). 글 다시 쓰기(B 층)에 쓰는 모델 연결은 선택입니다(9쪽).

서비스 켜기 README · windows-autostart

스킬이 일을 맡길 서비스를 켭니다

스킬에는 코드가 없습니다. `http://127.0.0.1:8765` 에서 도는 서비스에 파일을 보내고 결과를 받아 옵니다. **서비스가 꺼져 있으면 스킬이 아무것도 못 합니다.** 서비스는 내 컴퓨터 안(127.0.0.1)에서만 열리니, 글 다시 쓰기(B 층)에 외부 모델을 연결하지 않는 한 파일이 밖으로 나가지 않습니다.

켜기 · 저장소 폴더에서

복사해서 그대로 쓰세요

```
make serve
```

`make` 가 없는 윈도에서는 이렇게 켭니다.

복사해서 그대로 쓰세요

```
py service/scripts/server.py --host 127.0.0.1 --port 8765
```

켜졌는지 확인

복사해서 그대로 쓰세요

```
curl -s http://127.0.0.1:8765/health
```

`{"ok": true, "version": ...}` 가 나오면 됩니다. 윈도 PowerShell 에서는 `Invoke-RestMethod http://127.0.0.1:8765/health`.

윈도에서 로그인할 때 자동으로 켜기

1. **저장소를 자리 잡을 곳에 받는다**
   예: C:\watermarks-remover · 아래 경로를 전부 같게
2. **창 없이 켜는 스크립트 `start-service.vbs` 를 만든다**
   안에서 `python service\scripts\server.py --host 127.0.0.1 --port 8765` 를 숨긴 창으로 실행
3. **작업 스케줄러에 등록 `Register-ScheduledTask`**
   로그인 때 실행 · 관리자 권한 없이 일반 PowerShell 로 됩니다. 원문 명령은 저장소 docs/windows-autostart.md

> 서비스 주소를 바꿨다면 `WATERMARKS_SERVICE_URL` 에 새 주소를 넣습니다. 서비스에 API 키를 걸었다면 `WATERMARKS_SERVER_API_KEY` 도 함께.

클로드코드에서 부르기 SKILL.md

점검 먼저, 정리는 그다음

스킬의 규칙은 **「먼저 점검하고, 짐작하지 말고 결정한다」**입니다. 점검(`/inspect`)은 무엇이 들어 있는지 보고만 하고, 정리(`/clean`)는 새 파일을 만듭니다. 원본은 그대로 두고 `이름.cleaned.확장자` 로 저장하는 게 기본입니다.

① 파일 점검

복사해서 그대로 쓰세요

/remove-ai-marks ./images/banner.png 를 점검만 해 줘. 숨은 글자 개수, C2PA · AI 메타데이터가 있는지, 각 항목의 확신 수준을 표로 보여 주고 정리는 내가 「진행」이라고 하면 해 줘.

② 정리 + 다시 점검

복사해서 그대로 쓰세요

방금 점검한 파일을 정리해서 원본 옆에 .cleaned 파일로 저장하고, 정리한 파일을 다시 점검해서 남은 것(still_has_c2pa · still_has_ai_metadata)을 정리 전과 나란히 보여 줘. 보고서는 부풀리지 말고 그대로.

③ 폴더 한꺼번에

복사해서 그대로 쓰세요

/remove-ai-marks ./export 폴더 안 파일 전부를 점검해서, 표시가 남은 파일만 목록으로 보여 줘. 정리는 아직 하지 마.

입력별로 이렇게 갑니다

넣은 것	가는 길
사진 · 그림 (PNG · JPEG · WebP …)	메타데이터 정리. 설정 없이 바로 됩니다
SVG · PDF · DOCX · EPUB · ODT	문서 꼬리표 정리
.md · .html	앞머리 · 메타 태그 · 주석 속 출처 정리 + 숨은 글자
붙여넣은 글 · .txt	임시 파일로 저장 → 점검 → 정리. **글 정리는 B 층 설정이 있어야 합니다(9쪽)**

글 정리 README 원문

글은 왜 어렵고, 무엇을 설정하나

README 의 설명입니다. 글 워터마크는 **낱말 자체**에 있어서 거의 모든 문장에 신호가 조금씩 퍼져 있습니다. 그래서 문단 순서를 바꾸거나 제목을 고치는 정도로는 거의 안 움직이고, 글의 상당 부분을 문장 단위로 다시 써야 합니다. 숨은 글자를 지우는 A 층은 이 워터마크를 지우지 않습니다.

글 정리를 하면 생기는 일

서비스는 글(.txt)에 A 층 다음 B 층 다시 쓰기를 자동으로 붙입니다. 기본 전략은 `paraphrase@0.8,mlm@0.2` 이고, **쓸 모델이 설정돼 있지 않으면 400 오류로 거절**합니다.

환경변수	넣는 값
`WATERMARKS_REWRITE_BACKEND`	`ollama` 또는 `openai-compatible` (기본 print-prompt = 프롬프트만 출력)
`WATERMARKS_REWRITE_MODEL`	다시 쓸 모델 이름 (예: llama3.2)
`WATERMARKS_REWRITE_BASE_URL`	모델 주소. 올라마 기본은 내 컴퓨터
`WATERMARKS_REWRITE_API_KEY`	키는 이 환경변수로만 받습니다
`WATERMARKS_REWRITE_ALLOW_REMOTE=1`	내 컴퓨터 밖 주소를 쓸 때만

`mlm` 단계는 `transformers` 와 `roberta-large` 모델이 따로 필요합니다.

> **클로드 글을 클로드로 다시 쓰지 않습니다.** 원래 모델로 다시 쓰면 표시가 다시 찍힐 수 있어, README 는 다른 회사 모델을 쓰라고 합니다.

B 층을 건너뛰는 게 나을 때

README 가 직접 묻습니다. 어차피 더 싼 모델로 다시 쓸 거면 처음부터 그 모델로 쓰는 게 더 싸고 결과도 같거나 낫다고. **글 품질이 중요하면** 원문은 그대로 두고 A 층 숨은 글자 정리와 파일 꼬리표 정리만 하는 쪽을 권합니다. 글만 다루는 별도 스킬 `clean-user-facing-text` 는 서비스 없이 A 층만 돌립니다.

전후 비교 세팅

정리 전후를 탐지기로 비교하기

「정리했다」는 느낌이 아니라 숫자로 봅니다. 세 가지를 순서대로 겹쳐 씁니다.

1 **리무버 자체 점검 · 정리 전과 후** 누구나

숨은 글자 개수 · C2PA · AI 메타데이터 여부 · 확신 수준(confirmed · probable · informational · likely_false_positive). 정리 뒤 보고서의 `still_has_c2pa` · `still_has_ai_metadata` 가 남은 것을 말해 줍니다

2 **공식 · 공개 확인 도구로 파일 꼬리표 보기** 누구나

Claude Content Checker(claude.com/check-content) · Content Credentials 검증(contentcredentials.org/verify) · c2patool

3 **탐지기 점수 · 정리 전과 후** 설정한 사람만

서비스에 탐지기를 붙였을 때만 나옵니다. `/capabilities` 가 탐지기를 보여 줄 때 `detect_before` · `detect_after` 를 켭니다

3번을 켜는 요청 SKILL.md

복사해서 그대로 쓰세요

/remove-ai-marks 먼저 서비스의 /capabilities 를 확인해서 탐지기(text_detectors 또는 synthid 점수기)가 있는지 알려 줘. 있으면 이 파일을 options {"detect_before": true, "detect_after": true} 로 정리하고, 정리 전후 점수를 표로 나란히 보여 줘. 없으면 탐지기 없이 점검 보고서만 비교해 줘.

점수를 읽을 때 지킬 것

- 붙일 수 있는 텍스트 탐지기(MarkLLM)는 **연구용**이고 같은 설정으로 넣은 워터마크만 잽니다. 회사 공식 탐지기가 아닙니다.
- 클로드의 글 탐지기는 아직 일반 공개 전입니다. 그래서 **클로드 글 워터마크가 공식 탐지에서 빠지는지는 누구도 지금 확인할 수 없습니다.**
- 구글은 2026년 8월 API 의 SynthID 글 탐지기를 내렸습니다(SKILL.md).
- SKILL.md 의 원칙: 내 컴퓨터 탐지기를 회사 공식 탐지기인 것처럼 말하지 않는다.

한계

정리해도 남는 것, 지워도 모르는 것

도구가 스스로 밝힌 한계 README · SKILL.md

- **「공식 검사를 통과한다」는 누구도 정직하게 보증할 수 없습니다.** 회사가 공개 탐지기와 키를 내놓기 전까지는요.
- 보고서는 **확인된 것**(지운 글자 수 · 메타데이터 처리)과 **최선 노력**(B 층 다시 쓰기)을 나눠 적습니다. 둘을 섞어 말하지 않습니다.
- 다시 쓰면 글이 상합니다. 말투 · 목소리 · 정확도가 다시 쓴 모델 수준으로 내려갑니다. 마케팅 · 고객 납품 글에서는 눈에 띕니다.
- 남는 위험은 **짧고 뻔한 글은 낮고, 길고 개성 있는 글은 높습니다.**

표시가 있어도, 없어도 단정할 수 없다 앤트로픽 도움말

표시가 나왔다고 해서	표시가 없다고 해서
클로드가 원작자라는 뜻은 아닙니다. 사람이 쓴 글을 클로드로 교정 · 번역 · 요약 · 변환했을 수도 있고, 표시가 붙은 뒤 사람이 고쳤을 수도 있습니다.	AI 가 안 썼다는 뜻이 아닙니다. 표시 지원 전 모델 · 많이 고치거나 바꿔 쓰거나 번역한 글 · 아주 짧은 글 · 변환 · 다시 저장 · 스크린샷으로 메타데이터가 빠진 파일 · 표시를 지원하지 않는 기능에서 나온 결과물이면 안 잡힐 수 있습니다.

> **정리하면.** 파일 꼬리표는 지우고 확인할 수 있습니다. 글 워터마크는 다시 써야 줄고, 줄었는지 공식적으로 확인할 방법은 아직 없습니다. 이 가이드의 전후 비교는 그 사실을 숫자로 보는 세팅이지, 통과를 약속하는 세팅이 아닙니다.

대응 · 쓰는 범위

앞으로 대비할 것, 내 콘텐츠에만 쓰기

쓰는 범위 SKILL.md Ethics

- **내가 가진 콘텐츠에만** 씁니다. 용도는 개인정보 · 위생 · 연구입니다.
- 정리한 결과를 **「사람이 쓴 것」이라고 내세우지 않습니다.**
- 과제 · 시험처럼 AI 사용을 밝혀야 하는 자리, 법이 공개를 요구하는 자리에서 표시를 지우는 데 쓰지 않습니다. 스킬도 그런 요청에는 경고합니다.

앞으로 대비할 것

1. **파일을 넘기기 전에 점검하는 습관**
   내 사진 · 문서에 촬영 정보 · 편집기 정보 · 출처 꼬리표가 무엇이 붙어 있는지 먼저 봅니다. 클로드와 상관없는 개인정보도 여기서 같이 보입니다

2. **글은 처음부터 내 말로 다듬는다**
   클로드는 초안 · 생각 정리에 쓰고, 최종 문장은 직접 고쳐 쓰면 품질과 내 목소리를 둘 다 지킵니다. 다시 쓰기 도구로 뭉개는 것보다 낫습니다

3. **밝혀야 할 곳은 밝힌다**
   클로드를 내 서비스에 넣어 쓰는 쪽은 50조가 무엇을 요구하는지 스스로 판단해야 한다고 앤트로픽 도움말이 적고 있습니다

4. **탐지 공개 소식을 챙긴다**
   앤트로픽은 탐지 API 접근을 넓혀 가겠다고 했습니다. 공개되면 지금 「확인 불가」였던 부분을 그때 다시 잽니다

내 콘텐츠를 점검하는 한 줄

복사해서 그대로 쓰세요

/remove-ai-marks 이번에 올릴 ./upload 폴더를 점검해서, 파일마다 붙어 있는 정보(AI 출처 · 촬영 위치 · 편집기 · 작성자)를 표로 보여 줘. 무엇을 지울지는 내가 고를게.

막히면

증상 → 할 일

증상	할 일
스킬이 「서비스에 연결할 수 없다」	저장소 폴더에서 `make serve` (윈도 `py service/scripts/server.py --host 127.0.0.1 --port 8765`) → `/health` 확인 (7쪽)
`/remove-ai-marks` 가 안 보임	`/skills` 로 목록 확인. 플러그인으로 깔았으면 `/watermarks-remover:remove-ai-marks`
글 정리가 400 오류	B 층 모델이 설정되지 않은 것. 환경변수를 넣거나(9쪽) 글은 A 층 · `clean-user-facing-text` 로
파일 정리가 400 · `unknown`	알아볼 수 있는 확장자가 아님. 파일 이름에 맞는 확장자를 붙여 다시 (예: notes.txt)
PDF 보고서에 「일부만」	`exiftool` · `qpdf` 설치. 문서 안 그림까지면 `ghostscript` 도
정리 뒤에도 `still_has_c2pa`	그 형식에서 못 지운 것. 보고서의 남은 항목을 그대로 확인. 소프트 바인딩은 범위 밖 (5쪽)
설치가 파이썬 오류로 멈춤	`python3 --version` 이 3.10 이상인지. 윈도는 `py` 로
원본이 바뀔까 걱정	기본은 `.cleaned` 새 파일. 덮어쓰기는 직접 요청할 때만
탐지기 점수가 안 나옴	정상. 탐지기를 붙이지 않았으면 `detect_before` · `detect_after` 는 비어 있습니다 (10쪽)

출처: 앤트로픽 도움말 「How Claude marks AI-generated content」 · github.com/guillaumemeyer/watermarks-remover (README · skills/remove-ai-marks/SKILL.md · docs/windows-autostart.md). 이 자료는 해당 프로젝트와 제휴 관계가 없고, 2026-10-21 공개 문서를 기준으로 썼습니다.

더 묻고 싶으면 네이버 카페 「바이브메이커」 자료실 글 댓글로.