

● VIBE CODING · LESSON 02

나만의 프로필 사이트, 인터넷에 배포까지.

코드 한 줄 안 치고, AI와 함께 진짜 웹사이트를 세상에 공개한다.

하네스 엔지니어링

플러그인

스킬

훅

데이터베이스

배포

https://내이름.vercel.app

RECAP

우리가 이미 가진 것

지난 시간 환경 설정과 첫 작품 만들기에서 챙겨온 것들.

W

Warp = 우리의 작업 공간

터미널 + 코드 에디터 + 파일 트리가 한 앱에. 오늘 모든 작업은 여기서 시작한다.

C

Claude Code = AI 코딩 비서

자연어로 시키는 AI. 터미널에서 `claude` 한 단어로 켜다.

P

Plan Mode = 설계 먼저, 실행 나중

안전 모드. `/plan` 또는 `Shift` + `Tab` 으로 진입.

★

첫 작품, 이미 만들었다

`~/my-first-vibe` 폴더에 내 손으로(말로) 만든 첫 결과물이 살아있다.

오늘은 이 위에 '진짜 웹사이트'를 쌓는다.

TODAY

오늘 우리가 할 일

■ 오늘 만들 것

- 1 내가 자유롭게 구상한 프로필 사이트
- 2 넣고 싶은 사진 한 장을 올려서 사이트에 표시 (데이터베이스 맛보기)
- 3 전 세계에 공개되는 나만의 URL (배포)

■ 오늘 배울 개념 · 하네스 엔지니어링

- 플러그인 & 스킴 — 도구 vs 사용 설명서
- 훅 — 자동으로 굴러가는 안전 규칙
- 데이터베이스 & 보안 — 내 데이터의 집
- 배포 — 인터넷에 내놓기

AI가 일을 잘하게 '판'을 깔아주는 것 = 하네스 엔지니어링. 오늘 그 첫 맛.

CONCEPT 01

플러그인과 스킬, 뭐가 다를까?

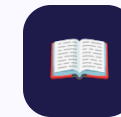


플러그인

확장팩 — 앱스토어 앱

Claude Code에 기능을 **통째로** 설치하는 묶음.
마켓플레이스에서 받는다. 안에는 여러 **스킬·도구**가 들어 있다.

비유 — 공구함을 통째로 사 옴.



스킬

필요할 때 꺼내는 매뉴얼

특정 작업을 잘하는 방법을 적어둔 **설명서**.
Claude가 '**필요하다고 판단할 때**'만 꺼내 본다. 평소엔 안 꺼내니 **가볍다**.

비유 — 공구함 안의 사용 설명서.

다음 — Superpowers 플러그인을 깔아서, 그 안의 '계획 잘 세우는 스킬'을 우리 플랜 모드에 연결한다.

STEP 01

실습 · 플러그인 설치

Superpowers 플러그인 깔기

설계를 더 똑똑하게 — “코딩 전에 제대로 설계부터” 습관을 Claude에게 심어준다.

● ● ● Claude Code 안에서 차례로 입력

slash commands

```
/plugin marketplace add obra/superpowers-marketplace  
/plugin install superpowers@superpowers-marketplace
```

■ 설치 후 — 한 번 재시작

`/exit` 으로 나갔다가, 같은 폴더에서 `claude` 로 다시 실행. 플러그인이 새로 로드된다.

■ 비유

마켓플레이스(앱스토어)에서 전문 공구함을 다운로드하는 느낌. 안에 든 도구·매뉴얼이 한 번에 들어온다.

✓ 확인 다시 켜 뒤 `/superpowers` 를 칠 때 명령어가 자동완성되면 설치 성공.

STEP 02

실습 · 지침서에 이정표

지침서에 ‘이정표’ 한 줄 남기기

플러그인이 스킬을 설치했다면, 지침서의 한 줄은 ‘언제 그 스킬을 쓸지’ 알려주는 이정표다.

● ● ● Claude Code · 입력

prompt

▶ 전역 지침서(~/.claude/CLAUDE.md)에 딱 한 줄만 추가해줘 - “플랜 모드에서는 슈퍼파워의 계획(브레인스토밍) 스킬을 참고해 꼼꼼히 계획을 세운다.” 그 외 내용은 넣지 말고 짧게 유지해줘.

■ 왜 지침서는 짧아야 하나

- CLAUDE.md는 매 세션마다 통째로 읽힌다 → 길면 매번 토큰을 잡아먹는다.
- 스킬은 필요할 때만 읽힌다 → 평소엔 공짜.
- 정석: 지침서는 짧게 + 자세한 작업법은 스킬로.

~/.CLAUDE/CLAUDE.MD

Global guide

- 플랜 모드에서는 슈퍼파워의
계획(브레인스토밍) 스킬을 참고해
꼼꼼히 계획을 세운다.

↑ 딱 이만큼. 끝.

✓ 확인 지침서에 그 한 줄만 깔끔히 추가됐으면 성공. 이게 바로 토큰을 아끼는 하네스 엔지니어링.

STEP 03

실습 · 자유 구상

어떤 프로필 사이트를 만들까?

정해진 틀 없음. 나는 어떤 사람인지 보여주는 사이트를 자유롭게 — 구체적일수록 좋다.

?

무엇을 보여줄까

취미 · 작품 · 관심사 · 포트폴리오 · 자기소개 · 좋아하는 것 모음.



어떤 분위기·색감

미니멀 / 감성 / 다크 / 파스텔 / 네온 / 종이 질감...

§

어떤 섹션

소개 / 갤러리 / 타임라인 / 링크 모음 / 좋아하는 영화 / 플레이리스트.

● ● ● 예시 프롬프트 - 마음껏 바꿔 써라

prompt

▶ 나만의 프로필 사이트를 만들고 싶어. 나는 [한 줄 자기소개] 이고, [관심사/취미] 를 좋아해. [소개, 갤러리, 링크 모음] 섹션이 있으면 좋겠고, 분위기는 [예: 다크 + 네온 포인트] 로. HTML·CSS·JS 기반 단일 페이지로 만들어줘.

사람마다 완전히 다른 사이트가 나온다. 그게 바이트코딩의 매력.

STEP 04

실습 · PLAN MODE로 생성

Plan Mode로 만들기

앞서 깐 게 여기서 빛난다 — 슈퍼파워 스킬이 계획 단계에서 작동한다.

- 1 `~/my-first-vibe` 안에서 `claude` 실행
또는 새 폴더: `mkdir ~/내사이트` → `cd ~/내사이트` → `claude`
- 2 `/plan` 으로 플랜 모드 진입
Windows에서 `Shift` + `Tab` 이 안 먹으면 `/plan` 직접 입력
- 3 슬라이드 7에서 정한 내 설명을 입력
- 4 계획을 읽어보고 ‘Yes’(맥락 유지) 선택. 권한 요청은 안전하면 Yes.

■ 무슨 일이 자동으로 일어나나

1. 플랜 모드가 활성화 — 실행 전 설계부터
2. 슈퍼파워 계획 스킬을 참고해 꼼꼼한 설계도
3. 승인하면 **Claude**가 그 계획대로 실제 파일을 만든다

= 계획 → 실행, 스킬이 제대로 작동!

✓ 확인 `index.html` / `style.css` / `script.js` 등이 만들어졌으면 성공.

STEP 05

실습 · 결과 보기 + 다듬기

내 작품 열어보고, 내 것으로 다듬기

Mac · 터미널

macOS

```
$ open index.html
```

Windows · 터미널

Windows

```
> start index.html
```

■ WARP 팁 — 파일 트리 열기

좌측 파일 트리는 **Cmd / Ctrl + ** 로 토글. 파일 클릭 → 코드 에디터로 바로 열림. 코드를 볼 일이 있어도 고치는 건 Claude에게 부탁.

다듬기 - 디자이너와 대화하듯

prompt

▶ 글씨가 너무 작아, 더 키워줘. 색감을 좀 더 차분하게. 맨 위에 큰 인사 문구(히어로 영역)를 넣어줘. 모바일에서도 예쁘게 보이게 해줘.

✓ 확인 브라우저에 내 사이트가 뜨고, 고친 점이 반영 되면 성공. 마음에 들 때까지 계속 요청.

CONCEPT 03

사진은 어디에 저장될까?

넣고 싶은 사진 한 장 — ‘서버 창고’에 올려보자.



내 컴퓨터 파일

내 노트북에만 있음.
사이트를 인터넷에 올리면 그 사진은 안 보인다.



서버 창고에 올린 사진

인터넷 어디서나 주소(URL)로 보인다 → 배포해도 그대로.

■ 데이터베이스 · 스토리지

Supabase(수파베이스) = 무료로 쉽게 쓰는 ‘서버 창고’. 오늘은 사진을 올려두는 **Storage(스토리지)** 기능을 맛본다. 내 데이터(사진·글 등)를 인터넷에 안전하게 보관하는 곳이라고 생각하면 OK.

오늘의 목표 — 내가 좋아하는 사진 한 장을 Supabase에 올리고, 그 사진을 내 프로필 사이트에 띄운다.

STEP 06

실습 · 사람이 직접 하는 부분

무료 창고 켜고 사진 올리기

웹에서 사람이 할 수 있는 일 — 가입 · 업로드 · 키 복사까지만.

1

가입 + 프로젝트 생성

`supabase.com` 가입 → **New Project** 로 무료 프로젝트 생성 (이름·비밀번호 정하기).

2

Storage에 사진 업로드

왼쪽 **Storage** 메뉴 → 새 버킷(폴더) 만들기 → 넣고 싶은 사진 **1장** 업로드.

3

메모장에 2가지 복사

Project URL (프로젝트 주소)
anon / publishable key (공개용 키)

■ 개념 · MCP (MODEL CONTEXT PROTOCOL)

Claude가 외부 서비스(Supabase 등)와 ‘약속된 방식’으로 대화하게 해주는 연결 통로. Supabase는 **MCP로 연결**한다 — 이유: ① **보안** — DB는 한 번 잘못 건드리면 끝이라 매 작업마다 Claude가 승인을 묻고 사람이 OK해야 실행 ② **프로젝트 단위 권한** — 이 프로젝트만 접근, 다른 곳엔 못 닿음.

● ● ● Claude Code · 입력

prompt

✓ 확인

Claude가 “Supabase MCP 연결 완료” 라고 답하면 성공. 이제 사진 띄우는 건 다음 STEP에서 Claude가 한다.

CONCEPT 04 · SECURITY

잠깐, 키에는 두 종류가 있다

✓ anon(공개) / publishable

원래 '공개용'. 사이트 코드에 들어가도 괜찮다.

✗ service(비밀) 키

모든 잠금을 무시하는 마스터 키. 절대 사이트 코드·인터넷에 올리지 말 것.

■ RLS · 접근 규칙

공개 키가 안전한 진짜 이유는 접근 규칙(RLS) 덕분. 규칙 = '누가 무엇을 읽고 쓸 수 있는지' 정하는 것. 사진 창고는 '누구나 보기(읽기)는 OK, 올리고 지우는 건 나만' 으로 두면 안전 — 이 규칙은 Claude가 설정해 준다.



anon 키

가게 출입문 — 누구나 들어와 구경 OK



RLS

창고 직원만 물건 추가



service 키

금고 마스터키 — 절대 밖에 두면 안 됨

STEP 07

실습 · 사진을 사이트에 띄우기

“내 사진 사이트에 띄워줘” — 말로 시키기

● ● ● Claude Code · 입력

prompt

▶ 내 프로필 사이트에, Supabase Storage에 올려둔 내 사진을 불러와서 보여줘. 백엔드는 Supabase를 쓰고, 내 Project URL과 anon 키는 이거야 — [여기에 붙여넣기]. 사진은 누구나 볼 수 있게(공개 읽기) 안전하게 설정해줘. **service(비밀) 키는 절대 코드에 넣지 마.**

■ CLAUDE가 하는 일

1. Supabase 사진의 공개 주소(URL)를 가져온다
2. 사이트 코드에 그 사진을 띄우는 부분을 작성
3. ‘공개 읽기’ 접근 규칙을 안전하게 맞춤

사람이 할 일은?

(필요 시) Claude가 안내하는 클릭/붙여넣기 한두 번 + 결과 확인.

코드 작성·수정은 0줄.

✓ 확인 브라우저 새로고침 → 내 사진이 사이트에 뜨면 **성공!** 새 창에서 열어도 보이면 진짜 ‘서버에서’ 불러온 것.

STEP 08

실습 · 자동 안전장치

실수 방지 자동 규칙 — 훅(Hook)

■ 훅이란?

스킬·플러그인은 Claude가 ‘판단해서’ 쓰는 것.

훅은 ‘무조건 자동으로’ 실행되는 규칙.

비유 — 스킬 = 직접 꺼내 보는 레시피 / 훅 = 알아서 올리는 자동 타이머.

● ● ● Claude Code · 입력

prompt

▶ 비밀 키(.env 파일이나 service 키 같은 것)를 실수로 코드 파일에 쓰려고 하면 자동으로 막아주는 규칙(훅)을 전역 설정에 만들어줘.

■ 무슨 일이 일어나나

Claude가 ‘파일을 쓰기 전에 검사하는’ 훅(PreToolUse)을 설정한다. 비밀 키를 넣으려는 시도를 자동 차단 (exit code 2).

다른 훅 예시

“파일을 고칠 때마다 자동으로 코드 정리(포매팅)” 같은 훅도 가능. 필요하면 똑같이 말로 시키면 된다.

✓ 확인 훅이 전역 설정에 추가됐다 고 Claude가 답하면 성공. 보안 한 겹 추가 완료!

CONCEPT 05

이제 전 세계에 공개하자 — 배포

deploy — 내 사이트 파일을 인터넷 ‘서버’에 올리는 것.

/. 지금까지 — localhost

내 컴퓨터 안에만 존재. 나만 볼 수 있음.

```
# 내 브라우저 주소창  
file:///Users/me/내사이트/index.html
```

▲ 배포 후 — Vercel

인터넷 어디서나 접속 가능. 나만의주소.vercel.app 완성.

```
# 친구·가족도 보는 주소  
https://내이름.vercel.app
```

■ 개념 · CLI (COMMAND-LINE INTERFACE · 명령어 도구)

컴퓨터에 깔아두는 ‘배포 전용 리모컨’. Vercel은 CLI로 전역 권한을 준다 — 이유: ① 배포는 한 번에 통째로 끝나는 단순 작업이라 매번 승인받을 필요 없음 ② 정적 사이트를 올리는 거라 위험도 낮음 ③ 한 번 `vercel login` 해두면 어떤 폴더에서든 바로 배포. → Supabase(MCP, 매번 승인)와 정반대 철학.

STEP 09

실습 · 마지막 — 인터넷에 올리기

“이 사이트 인터넷에 배포해줘”

● ● ● Claude Code · 입력

prompt

▶ 이 사이트를 Vercel로 인터넷에 배포해줘. **Vercel CLI**가 없으면 네가 알아서 설치하고(전역에 한 번만), **GitHub 없이 바로 배포**해줘. 배포가 끝나면 **접속 주소(URL)**를 알려줘.

■ 사람이 할 일 — 2가지뿐

1. 중간에 **Vercel 로그인** 단계가 뜨면 → 브라우저에서 로그인 (이메일/구글 등). 처음이면 무료 가입.
2. Claude가 권한을 물으면(도구 설치·명령 실행) → ‘**Yes**’로 승인.

⌚ 몇 분 걸릴 수 있어요 — 정상이에요.

Claude가 Node와 Vercel CLI를 설치하는 시간. 끝나면 `https://...vercel.app` 주소가 나옵니다.

✓ 확인 받은 URL을 휴대폰 등 다른 기기로 열어 **사이트(+사진)**가 뜨면 성공!

DONE

해냈다. 나만의 URL이 생겼다.



친구·가족에게 자랑

링크를 보내고, 반응을 받자.



다른 기기로 확인

사진·디자인이 잘 뜨는지 휴대폰으로도.



SNS·포트폴리오에

소개란에 링크 박고, 너의 흔적을 남겨라.

코드 한 줄 안 치고, 데이터베이스 + 배포까지 — 오늘 진짜 웹 서비스를 만들었다.

SUMMARY

오늘의 결과물 한눈에

```
~/.claude/ ← 전역 · 모든 프로젝트 공통
├ settings.json ← 훅(보안 키 가리기)
└ CLAUDE.md ← 짧은 전역 지침서

~/내사이트/ ← 이 프로젝트
├ index.html
├ style.css
├ script.js
└ (Supabase 연결 코드 - 공개 키 + 사진)
```

 **Supabase** **MCP**

사진 저장 창고(Storage) + 공개 읽기 규칙. 매 작업마다 승인 — DB는 보안 우선.

 **Vercel** **CLI**

인터넷에 배포된 내 URL. 한 번 로그인하면 어디서든 배포.

■ 연결 방식 — 두 철학

MCP(Supabase) = 매 작업 승인 · 프로젝트별 권한 → 보안.

CLI(Vercel) = 한 번 전역 로그인 → 속도.

WRAP UP

바이트코딩의 본질, 그리고 다음

■ 본질

플러그인 · 스킴 · 혹은 — 다 ‘도구 상자’일 뿐. 핵심은 이것 ‘어떻게 창의적으로 조합하느냐(=하네스 엔지니어링)’. 이 분야는 지금도 새 방법이 쏟아진다. 과학자가 연구하듯 호기심으로 실험하면 된다. 정답은 없다 — 내가 원하는 걸 만들어내면 그게 정답.



이번 주 과제

- 사진을 더 올리거나, 작은 갤러리 만들기.
- 사이트 1가지 이상 개선(섹션·디자인·기능).
- 이어서 작업할 땐 `claude --resume`.



다음 시간 예고

- 글 저장·방명록 — ‘쓰기’ 기능.
- 커스텀 도메인 연결.
- 서버에이전트를 직접 만들어 역할 나누기.

막히는 건 실패가 아니라 과정. Claude와 집요하게 대화하면 거의 다 풀린다.

APPENDIX A

어디에 저장될까 — 전역 vs 프로젝트

항목	전역 위치 (~/.claude)	프로젝트 위치 (.claude)	비유
지침서 CLAUDE.md	~/.claude/CLAUDE.md	./CLAUDE.md	회사 전체 규칙 vs 우리 팀 규칙
설정·흑 settings.json	~/.claude/settings.json	.claude/settings.json	〃
스킬	~/.claude/skills/	.claude/skills/	〃
플러그인	마켓플레이스에서 설치 — 모든 프로젝트에서 사용	—	공구함 통째로 구매

■ 한 줄 비유

전역 = 회사 전체 규칙(모든 프로젝트에 적용)
프로젝트 = 이 팀만의 규칙(이 폴더에만 적용).

■ 자동 VS 판단

흑 = 100% 자동 실행
스킬 · 서브에이전트 = Claude가 판단해서 사용

APPENDIX B

자주 막히는 곳 & 해결

claude가 안 켜져요

→ Warp를 완전히 닫고 다시 열기.

플러그인이 안 보여요

→ `/exit` 후 `claude` 재실행, `/help` 로 확인.

Plan Mode가 안 켜져요 (Windows)

→ 입력창에 `/plan` 직접 입력.

Vercel 로그인 후 멈춰요

→ 1~2분 대기. 브라우저에서 ‘승인/Allow’ 눌렀는지 확인.

사진이 안 떠요

→ Storage 업로드 + 공개 읽기 설정 점검을 Claude에게 요청. 사진 URL을 새 탭에서 직접 열어 확인.

키가 헛갈려요

→ 코드엔 **anon(공개)** 키만. **service(비밀)**는 절대 코드/인터넷에 X.

Claude가 영어로 답해요

→ “한국어로 답해줘”.

무엇이든 막힐 때

→ 에러 메시지를 그대로 복사해 “이 에러 고쳐줘”.

막히는 건 실패가 아니라 과정 — Claude와 집요하게 대화하면 거의 다 풀린다.