

「월간클로드」 2기 · 기초 이론서

만들고, 올린다

내 사이트는 인터넷에,
내 앱은 구글 플레이에.

다 읽고 나면
카톡으로 보낼 수 있는 내 웹사이트 주소와
플레이스토어에서 받아 설치되는 내 앱이 남습니다.
그리고 그 사이에 무슨 일이 있었는지 설명할 수 있습니다.

전석환 지음

v0.1 초고 · 2026년 9월 14일

작가의 말

초고 v0.2 — **저자 목소리로 다듬으실 자리입니다.** 저자가 직접 짚어 준 핵심(딸깍 이후 · 만든 사람의 생각을 기록해 유지보수)을 뼈대로 삼고, 나머지는 2기 강의 기획서와 강의 사이트에서 확인되는 사실만으로 채웠습니다.

유튜브에 이런 말이 돌아다닙니다.

"딸깍 하면 웹사이트가 만들어진다."

틀린 말이 아닙니다. AI에게 "내 공방 소개 페이지 만들어줘" 라고 하면 정말 금방 나옵니다. 20분이면 그럴듯한 사이트가 화면에 뜹니다. 그걸 보면 누구나 이렇게 생각합니다.

"와, 나도 이제 AI로 웹이든 앱이든 그냥 쉽게 만들 수 있겠네."

그래서 강의를 기획하면서 가장 오래 붙잡고 있던 질문이 이것이었습니니다.

"Claude한테 시키면 20분이면 나오는데, 이걸 왜 배워요?"

답은 **만들고 난 다음**에 있습니다.

사이트가 한 번 나오고 나면, 반드시 고칠 일이 생깁니다. 사진을 바꾸고 싶고, 가격이 바뀌었고, 맨 위 큰 화면이 좀 답답해 보입니다.

그런데 여기서 막힙니다. **내가 직접 만든 게 아니기 때문**입니다.

AI가 만들어 줬으니, 어디를 어떻게 건드려야 하는지 모릅니다. 사이트 맨 위에 큼직하게 들어가는 첫 화면을 **히어로(Hero)** 영역이라고 부르는데, 이 단어조차 모르는 게 당연합니다. 부를 이름을 모르니 고쳐 달라는 말도 흐릿해집니다.

그래서 이렇게만 말하게 됩니다.

"수정해줘."

AI는 고쳐 줍니다. 문제는 **어떻게** 고치느냐입니다.

AI는 이 사이트를 **처음에 왜 이렇게 만들었는지** 모릅니다. 왜 이 색을 골랐는지, 왜 메뉴를 세 개로 줄였는지, 왜 예약 버튼을 맨 위에 올렸는지 — 만든 사람의 생각이 어디에도 남아 있지 않으니까요. 그러면 AI는 **자기 생각대로** 고칩니다. 그럴듯하게, 하지만 다르게.

몇 번 "수정해줘" 를 반복하고 나면, 처음 사이트와는 **완전히 다른 느낌의 사이트**가 되어 있습니다. 고친 게 아니라 새로 만든 것에 가깝습니다.

딸깍으로 만든 사이트는, 딸깍으로 고칠 수 없습니다.

이 책과 강의가 가려는 방향은 하나입니다.

만든 사람의 생각을 기록해 두고, 그 기록을 지키면서 고친다.

무엇을 만들고 **무엇을 안 만들기로 했는지**, 왜 이 화면을 골랐는지, 무엇이 **왜 바뀌었는지**. 이것들을 글로 남겨 둡니다. 개발자들은 이런 문서를 설계 문서라고 부르고, 그 안에 담긴 것을 **철학**이라고 부르기도 합니다.

기록이 있으면 AI가 달라집니다. "수정해줘" 라고만 해도, AI는 먼저 그 기록을 읽고 **원래 생각에 맞춰서** 고칩니다. 석 달 뒤에 다시 열어도, 다른 사람이 맡아도, 사이트는 처음 느낌을 잃지 않습니다. 이것을 **유지보수**라고 합니다. 만든 것을 망가뜨리지 않고 계속 쓸 수 있게 돌보는 일입니다.

그래서 2기 첫 주는 코드를 한 줄도 쓰지 않습니다. 두 시간 내내 **정하고, 기록합니다**. 뭘 만들지, 뭘 안 만들지, 어떻게 생겼는지. 화면 시안을 보고 나서는 설계 문서를 다시 고쳐 쓰고, **뭘가 왜 바뀌었는지** 남깁니다. 느리게 보이지만, 이게 둘째 주부터의 모든 수정을 빠르게 만듭니다.

이 부분이 이 책의 핵심입니다.

그리고 하나 더 있습니다. **AI가 대신 해주지 못하는 구간**입니다.

개발자 계정을 만들고 신분증으로 본인 확인을 받는 일, 사이트를 인터넷 주소에 연결하는 일, 스토어 등록 양식을 한 칸씩 채우는 일. 이걸 말로 시켜서 끝나지 않습니다. **사람이 직접 가입하고, 직접 누르고, 직접 기다려야 합니다**. 딸깍 영상에는 이 부분이 나오지 않습니다.

구글 플레이에는 혼자 넘기 어려운 벽도 하나 있습니다. 개인 계정으로 앱을 정식 공개하려면 테스터 12명이 14일 동안 참여해야 합니다. 월간클로드는 매달 사람이 모이는 곳이라, 서로의 앱에 테스터가 되어줄 수 있습니다. **혼자서는 못 넘는 벽을 기수가 같이 넘습니다**.

그럼, 0장에서 뵙겠습니다.

전석환

목차

작가의 말	2
 4부 — 설계 문서 ↔ 화면 (Week 3)	
14 만들 자리, 개발 환경과 빈 화면	6
15 웹 첫 화면, 네 바퀴 — 문서 먼저 화면 나중	18
16 웹 둘째 화면과 간판 시안	31
17 앱을 내 폰에 띄우기	39
18 앱 화면과 간판, 문서대로 하나씩	49
 5부 — 갈림길과 고쳐 쓰기 (Week 3)	
19 갈림길, 저장되게 만들기 또는 화면 더	58
20 GitHub 가입, 열쇠부터 만든다	64
21 Supabase, 데이터를 둘 창고 만들기	70
22 웹과 앱을 창고에 잇기	81
23 설계 문서 고쳐 쓰기, 누가 만들어도 똑같이	90
24 올리기 전에, Vercel 가입	99
25 공공데이터포털, 가입하고 쓸 데이터 고르기	105

쪽수는 이 PDF의 실제 쪽 번호입니다. 본문의 「N.N절」 표기는 해당 장 안의 절 번호입니다.

14장 — 만들 자리, 개발 환경과 빈 화면

초고 v0.1 (2026-09-18) · 2주차 11단계

14.1 화면을 만들기 전에 자리부터

설계 문서가 준비됐습니다. 이제 만들면 될 것 같은데, 11단계는 **화면을 하나도 만들지 않습니다**. 만들 자리만 만듭니다.

집을 짓기 전에 땅을 고르고 전기와 수도를 끌어오는 것과 같습니다. 이 단계가 끝나면 폴더가 이렇게 됩니다.



그림 14-1 11단계가 끝난 작업 폴더. 지난주 문서(docs) 옆에 웹 자리(web) · 앱 자리(app) · 데이터 자리(supabase)가 생긴다. web 과 app 안에는 도구가 받아 온 재료 폴더(node_modules)가 들어 있다

상자는 넷입니다. ****하나씩 따로**** 붙여넣습니다.

상자	하는 일
11-①	웹을 무엇으로 만들지 고른다
11-②	개발 환경 을 맞춘다 (PC에 한 번만)
11-③	자리를 만들고 빈 화면 을 띄운다
11-④	Claude가 크롬을 직접 누를 수 있게 붙인다

한 상자에서 다 해 버리면, 중간에 **개발 환경 설정**이라는 게 있었다는 것을 못 보고 지나갑니다. 그래서 지침서가 상자 경계를 지키라고 따로 적어 두었습니다.

14.2 11-① 웹을 무엇으로 만들까

11-① 상자를 붙여넣습니다.

이제 진짜로 만들기 시작할게.
만들기 전에 뭘 정해야 하는지 물어봐 줘.
정하는 것까지만 하고, 만들지는 말고 기다려줘.

Claude가 하나를 묻습니다. **웹을 무엇으로 만들지**입니다.

프레임워크(framework) — 만들기 도구 세트

웹을 처음부터 맨손으로 만들 수도 있지만, 보통은 **자주 쓰는 부품과 조립 규칙이 미리 갖춰진 도구 세트**를 씁니다. 가구로 치면 이케아 조립 세트입니다. 판자·나사·설명서가 한 상자에 들어 있어서, 톱질부터 하지 않아도 됩니다.

이런 도구 세트를 **프레임워크**라고 부릅니다. 오늘 보기로 나오는 `Next.js` · `Vite` + `React` 가 그 이름입니다.

보기는 셋입니다.

보기	좋은 점	대가
A. <code>Next.js</code> (추천)	다음 주에 인터넷에 올릴 때 제일 쉽다 . 화면이 늘어나도 잘 버틴다	처음 설치가 조금 길다 (5분쯤)
B. <code>Vite</code> + <code>React</code>	더 가볍고 설치가 빠르다 (1~2분). 만드는 동안 화면이 빨리 바뀐다	다음 주 올릴 때 손이 한 번 더 간다
C. 순수 HTML	제일 단순하다. 파일 하나로 시작	화면이 서너 개만 넘어도 지저분해지고, 데이터를 붙일 때 손이 많이 간다

모르겠으면 **추천대로** 라고 하면 됩니다. A로 갑니다.

앱과 데이터는 **문지 않습니다**. 앱은 `Expo`, 데이터는 `Supabase` 로 정해져 있습니다. 폰에서 돌리는 현실적인 길이 하나뿐이고(17장), 데이터를 둘 곳도 이유가 있어서 정해져 있습니다(21장). **고를 게**

없는 것을 고르라고 하면 사람은 멈춥니다.

💡 웹만 고르게 하는 이유

웹은 길이 여러 갈래이고, **고르면 3주차에 인터넷에 올리는 방법까지 달라집니다.** 그래서 이것만은 알고 고르게 합니다.

한 번 고르면 다시 묻지 않습니다. 세부 설정(글자 규칙, 꾸미는 방식 같은 것)은 Claude가 정합니다. 우리가 알아야 할 선택은 **하나뿐**입니다.

고른 것은 PRD에 적힙니다(14.6절). 3주차의 Claude가 그걸 보고 올리는 방법을 정합니다.

14.3 11-② 개발 환경 설정 — PC에 한 번만

☐ 11-② 상자입니다. 제작prompt에는 상자 앞에 이런 설명이 붙어 있습니다.

잠깐 — 이게 뭔가요? 프로그램을 만들려면 **재료를 받아오고 조립해주는 도구가 내 PC에 있어야** 합니다. 워드 문서를 쓰려면 워드가 깔려 있어야 하는 것과 같아요. **PC가 알아서 해주지 않습니다.**

개발 환경부터 맞춰줘.

Node 있는지, npm 도는지 확인하고 없으면 깔아줘.

윈도우면 실행 정책도 풀어줘. 맥이면 맥에 맞게 해줘.

다 되면 뭐가 됐는지 한 줄로 알려줘.

☐ 개발 환경(development environment)

프로그램을 만들 수 있게 내 PC에 **갖춰 두는 도구와 설정 전부.** 부엌으로 치면 가스레인지·칼·도마입니다. 레시피(코드)가 아무리 좋아도 부엌이 없으면 요리를 못 합니다.

한 번 갖춰 두면 **다음 요리에도 그대로** 씁니다. 그래서 PC에 한 번만 하면 됩니다.

이 상자가 확인하는 건 세 가지입니다.

① Node — 1장에서 깐 실행기

1장에서 Node.js를 깔았습니다. 자바스크립트로 쓴 코드를 **실제로 돌려 주는 실행기**였습니다. Claude는 그게 잘 깔려 있는지, 너무 오래된 판은 아닌지 봅니다.

없으면 Claude가 깔니다. 윈도우에는 `winget` 이라는 **설치 심부름꾼**이 들어 있어서, 사이트에 가서 내려받고 다음 · 다음 누르는 일을 명령 한 줄로 대신합니다.

② npm — 재료를 받아 오는 배달부

📦 npm(엔피엠) — 재료 배달부

프로그램을 만들 때 쓰는 **남이 만들어 둔 부품**(패키지라고 부릅니다)을 인터넷에서 받아다 폴더에 넣어 주는 도구. 요리로 치면 **식자재 배달**입니다. "밀가루, 버터 가져다줘" 하면 부엌에 갖다 놓습니다.

Node.js를 깔면 같이 딸려 옵니다. 받아 온 재료는 `node_modules` 라는 폴더에 쌓입니다. 이 폴더에는 **파일이 수만 개** 들어갑니다.

Claude는 npm이 제대로 도는지 확인합니다. 숫자(판 번호)가 나오면 된 겁니다.

③ 실행 정책 — 윈도우의 잠금장치

여기가 오늘 처음으로 **PC 설정을 바꾸는** 자리입니다.

📋 실행 정책(execution policy)

윈도우가 "**모르는 프로그램이 멋대로 돌지 못하게**" 걸어 둔 잠금장치. 아파트 현관의 공동 출입문과 같습니다. 안전하지만, 내가 부른 배달부(npm)까지 문 앞에서 막힙니다.

그래서 **내 계정에서만** 문을 조금 열어 줍니다. "인터넷에서 온 건 서명이 있어야 들어오고, 내 PC에서 만든 건 들어와도 된다" 정도로요. 관리자 권한은 필요 없습니다.

맥에는 이 잠금장치가 없습니다. 맥이면 Claude가 아무 말 없이 넘어갑니다.

이때 화면에 「**이 명령을 실행할까요**」 창이 뜹니다(2장의 권한 창). Claude가 PC 설정을 바꾸려 하니 물어보는 겁니다. Claude가 무엇을 바꾸는지 먼저 말해 주고, 우리는 **허용**을 누릅니다.

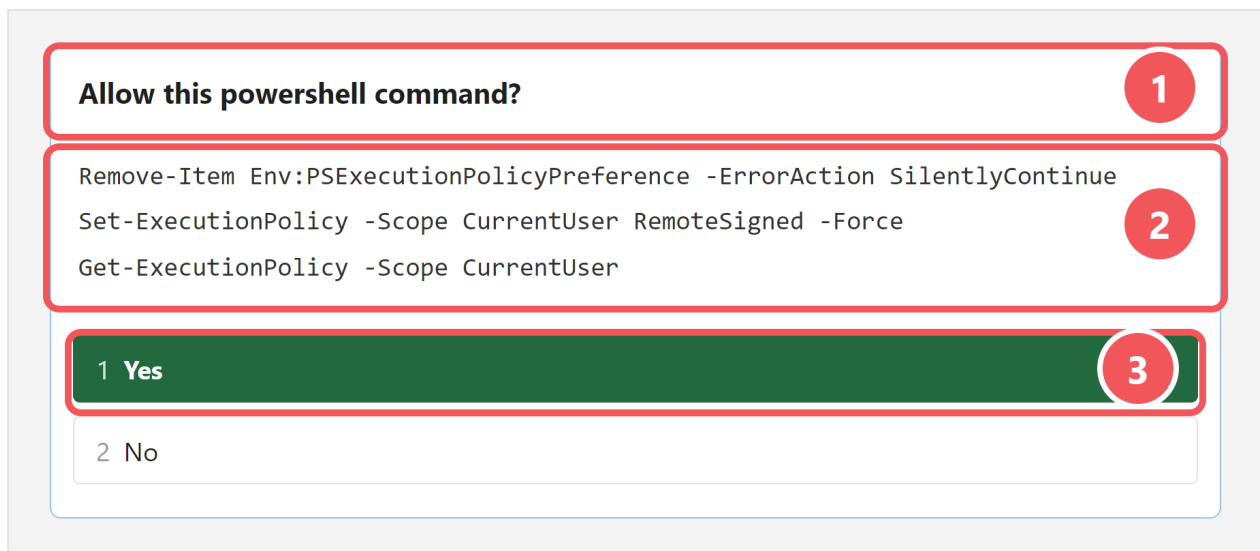


그림 14-2 실행 정책을 바꾸기 전에 뜨는 권한 창. ① 이 명령을 실행해도 되는지 묻는다 ② Claude가 실행하려는 명령 세 줄 — 내 계정에서만 잠금을 푼다 ③ 허용(Yes)

끝나면 Claude가 이렇게 말합니다.

"환경 설정 끝났습니다. **이건 이 PC에 한 번만 하는 거예요** — 다음 프로젝트에서는 안 하셔도 됩니다. 혹시 나중에 혼자 만드시다가 `npm ... cannot be loaded` 라는 빨간 글씨를 보시면, **이걸 안 해준 겁니다.**"

💡 오늘은 안 막히는데 왜 푸냐

사실 수업 중에는 이것 때문에 막힐 일이 거의 없습니다. Claude Code가 명령을 칠 때는 이 잠금을 **그 창에서만 잠깐 비켜 가도록** 되어 있기 때문입니다.

막히는 건 **나중에 혼자 할 때**입니다. 내가 직접 창을 열어 명령을 치거나, 다른 도구로 새 프로젝트를 시작하면 첫 줄에서 빨간 글씨가 나옵니다. 그리고 혼자서는 원인을 못 찾습니다.

그래서 오늘 **이름을 붙여서, 보이게** 팝니다. 조용히 처리해 버리면 "개발 환경 설정" 이라는 게 있다는 것조차 모르고 넘어갑니다. 이 장에서 가져갈 몇 안 되는 **혼자 풀 수 있는 것** 가운데 하나입니다.

14.4 11-③ 자리 만들기 · 빈 화면 보기

☐ 11-③ 상자입니다.

이제 만들 자리 만들어줘. 웨이랑 앱 폴더.
다 되면 빈 화면 띄워서 보여줘.

여기서 **설치가 돕니다. 5분쯤 걸립니다.** 글씨가 주르륵 올라가는데 정상입니다. **아무것도 누르지 않습니다.**

그 5분 동안 일어나는 일은 이렇습니다.

1. `web/` 폴더에 웹의 뼈대를 만든다 — 11-① 에서 고른 도구 세트로
2. `app/` 폴더에 앱의 뼈대를 만든다 — Expo로
3. `supabase/` 폴더는 이름만 만들어 둔다 — 쓰는 건 21장
4. 각각 npm이 재료를 받아 온다 — 여기가 오래 걸린다

설치 도중에 도구가 "이름은 뭘로 할래?", "어느 틀로 시작할래?" 하고 묻기도 합니다. **그건 Claude가 답합니다.** 명령을 칠 때 답을 미리 넣어 두어서 묻는 창이 뜨지도 않게 합니다. 우리에게 넘어오지 않습니다.

⚠️ 작업 폴더가 구글드라이브·원드라이브 안이면 여기서 깨집니다

1장에서 "구글드라이브 폴더는 피하라" 고 했습니다. 이유가 바로 여기서 나옵니다. npm은 `node_modules` 에 파일을 **수만 개** 만들었다 지웠다 합니다. 그 폴더가 동기화 폴더 안에 있으면, 동기화 프로그램이 파일을 붙잡고 올리려 하다가 설치가 깨집니다(`EPERM` , `EBADF` 같은 빨간 글씨).

그래서 Claude는 설치 **전에** 폴더 경로를 먼저 봅니다. 동기화 폴더 안이면 `C:\workspace\` 같은 곳으로 **폴더째 옮기고** 거기서 이어 갑니다. 1장대로 `C:\workspace\` 에 만들었다면 아무 말 없이 지나갑니다.

설치가 끝나면 Claude는 웹을 **띄웁니다.** 그리고 브라우저에 이런 화면이 뜹니다.

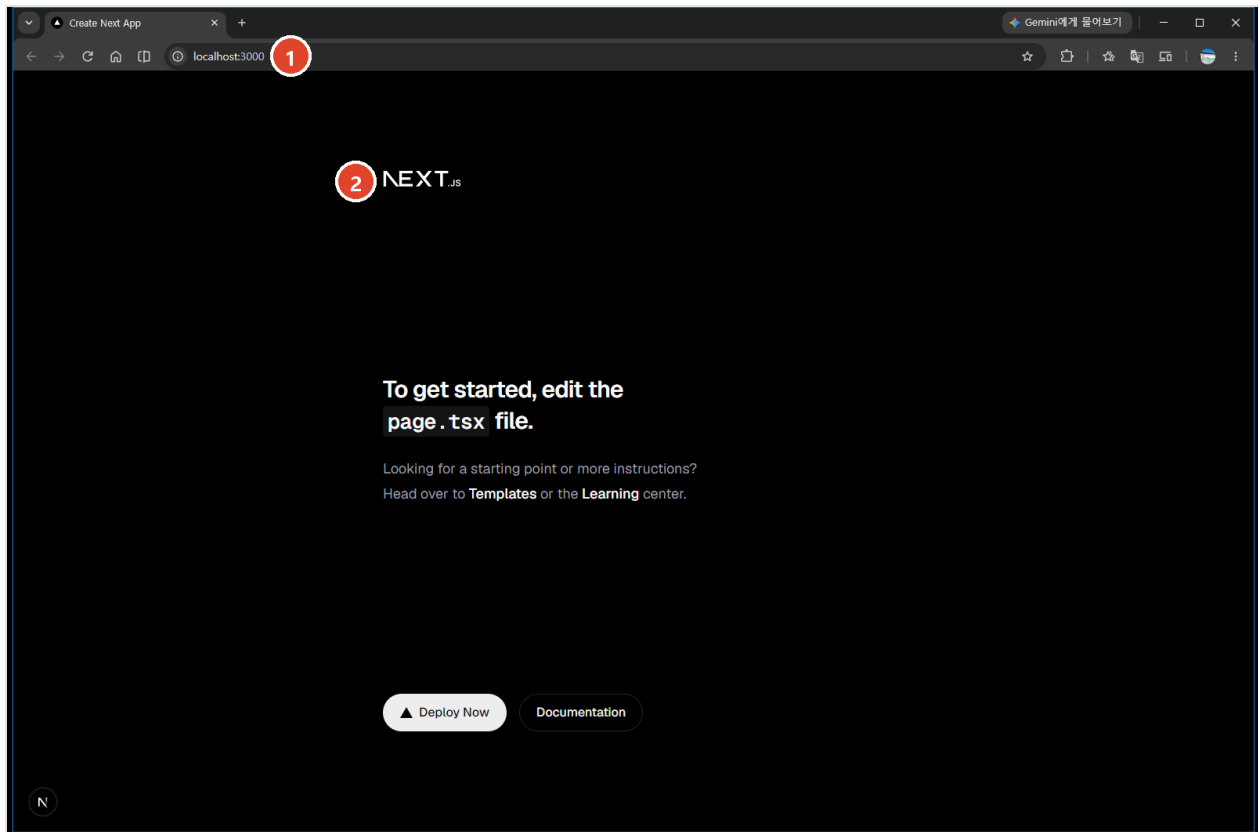


그림 14-3 빈 웹이 뜬 크롬. ① 주소창의 localhost:3000 ② 도구 세트에 원래 들어 있는 기본 화면(까만 바탕에 Next.js 글자)

>  **localhost(로컬호스트)와 포트** > > `localhost` 는 **"바로 이 컴퓨터"** 를 가리키는 주소입니다. 인터넷 어딘가가 아니라 > 내 책상 위 PC에서 도는 웹사이트를 보고 있다는 뜻입니다. 그래서 **나만** 볼 수 > 있습니다. 옆 사람 PC에서 같은 주소를 쳐도 안 뜹니다. > > 뒤의 `:3000` 은 **포트(port)** 입니다. 한 건물(내 PC) 안의 **호수**라고 보면 됩니다. > 웹은 3000호에 산다는 뜻입니다. 3000호에 이미 누가 살고 있으면 3001호로 갑니다. Claude는 이렇게 말합니다.

"자리가 생겼습니다. 지금 보시는 게 아직 아무것도 없는 빈 웹이에요. 주소창에 localhost:3000 이라고 쓰여 있죠. 이게 지금 내 컴퓨터에서 도는 웹사이트입니다. 다음 주에 이걸 인터넷에 올립니다."

"뒀습니다" 는 확인한 뒤에만

Claude는 "뒀습니다" 를 그냥 말하지 않습니다. 순서가 정해져 있습니다.

순서	하는 일
1	웹을 뒤에서 조용히 돌린다. 까만 창을 띄우지 않는다
2	웹이 제대로 대답하는지 Claude 쪽에서 먼저 찢러 본다
3	크롬을 열어 준다
4	크롬이 실제로 받아 갔는지 기록을 본다
5	받아 갔을 때만 "뒀습니다"

강사가 준비하면서 실제로 틀린 적이 있기 때문입니다. 크롬을 여는 명령이 **아무 말 없이 실패**했는데 "열어뒀습니다" 라고 말해 버렸습니다. 확인하지 않은 말은 하지 않는다 — 지침서의 규칙입니다.

크롬이 저절로 안 열리면 주소창에 `localhost:3000` 을 직접 치면 됩니다. 주소는 **Claude가 알려줍니다**. 보이면 `보여` 라고만 답합니다.

14.5 11-④ 크롬 붙이기 — Claude가 직접 눌러 보여주게

이제 화면을 만들 차례인데, 그 전에 하나를 더 붙입니다.

오늘 화면을 하나 만들 때마다 "**진짜 되나**" 를 확인합니다. 검색칸에 이름을 쳐 보고, 버튼을 눌러 봅니다(15장). 이걸 **Claude가 크롬에서 직접** 할 수 있으면, 우리는 보고만 있으면 됩니다.

그런데 VS Code 안의 Claude 창에서는 Claude가 **화면을 띄워 줄 수는 있어도 대신 눌러 줄 수는 없는** 경우가 많습니다. 그래서 창을 하나 따로 띄웁니다.

터미널(terminal) — 까만 창

마우스 대신 **글자로 컴퓨터에 명령하는 창**. 대개 까만 바탕에 흰 글씨입니다. 은행 창구 대신 ARS에 번호를 누르는 것처럼, 버튼 없이 말(명령)로 일을 시킵니다.

오늘 우리가 터미널을 직접 쓸 일은 **없습니다**. Claude가 띄우고, 우리는 두 번만 입력합니다.

11-④ 상자입니다.

크롬을 네가 직접 눌러서 보여줄 수 있게, 터미널 창 띄워줘.

거기서 내가 뭘 하면 되는지도 알려줘.

Claude가 까만 창을 하나 띄웁니다. 그 창에도 Claude Code가 켜져 있습니다. 거기서 우리가 할 일은 둘뿐입니다.

1. `/chrome` 이라고 친다 — 크롬과 연결하는 명령
2. Claude가 만들어 준 **이어가기 문장**을 붙여넣는다

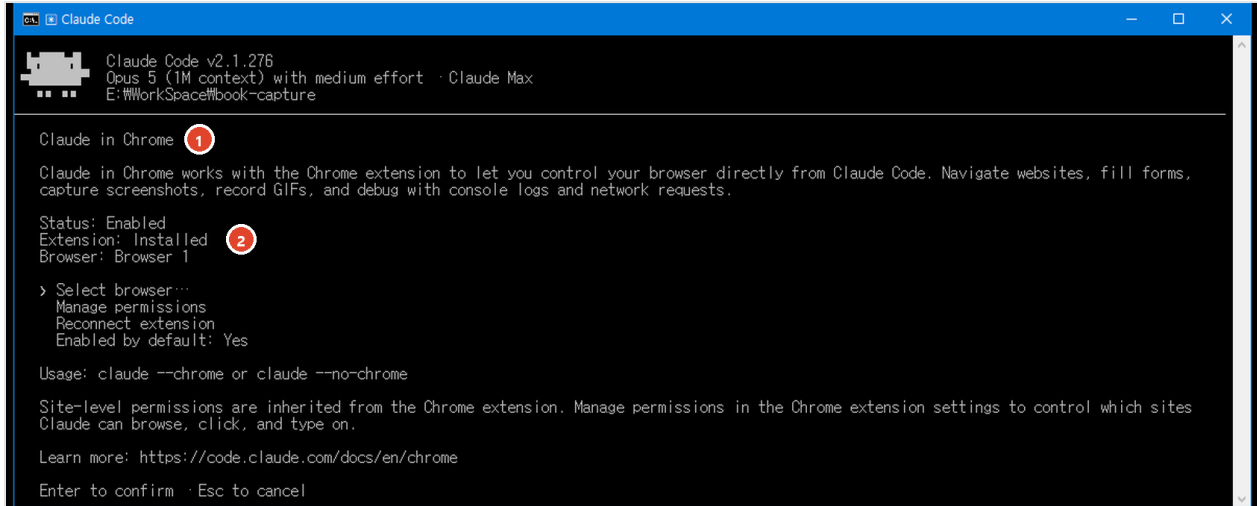


그림 14-4 까만 창에서 `/chrome` 을 치면 뜨는 연결 창. ① Claude in Chrome ② 연결됨(Status: Enabled) · 크롬 확장 설치됨(Extension: Installed). Esc 로 닫고 이어가기 문장을 붙여넣는다

이어가기 문장은 이런 모양입니다. 새 창의 Claude는 ****지금까지의 대화를 모르기 때문에****, 어디까지 왔는지 알려 주는 겁니다(13장의 에이전트와 같은 사정입니다).

2주차 수업 이어서 할게. 2주차-지침서.md 와 docs/20.working/PRD.md 를
먼저 읽어줘. 지금 11단계까지 끝났어. 웹은 localhost:3000 에서 돌고 있어.
크롬 연결부터 커줘.

12단계부터는 그 까만 창에서 합니다. 원래 창은 닫지 않고 그대로 둡니다. 웹이 거기서 돌고 있기 때문입니다.

💡 왜 굳이 옮기나 — 시험은 눌러야 된다

12단계는 바퀴마다 "진짜 되나" 시험으로 끝납니다(15장). 시험은 화면에서 **눌러 봐야** 성립합니다.

누를 수 없는 창에 머물면, Claude는 화면 대신 **코드를 직접 돌려서** 결과표를 채우려 합니다. 강사가 준비하면서 실제로 그랬습니다. 그건 시험이 아니라 다른 일입니다. 그래서 12단계 들어가기 전에 **한 번은 꼭 옮겨** 봅니다.

두 번 해서 안 되면 그냥 진행합니다. 그때는 Claude가 화면을 띄워 주고, 무엇을 눌러 보면 되는지 알려 줍니다. "검색칸에 김 쳐 보세요. 한 줄만 남으면 맞습니다." 우리는 한 번 눌러 보고 맞아 / 이상해 만 답합니다. 이것도 정식 경로입니다.

14.6 v0.4 — 「무엇으로 만드나」 장이 붙는다

11단계 끝에 PRD 3장 뒤로 짧은 장이 하나 붙습니다. 표 하나면 됩니다.

자리	쓰는 것	폴더
웹 (주인이 보는 쪽)	Next.js (수강생이 고름)	web/
앱 (신청하는 쪽)	Expo	app/
데이터	Supabase	supabase/

그 아래에 세 줄이 붙습니다.

- 웹은 지금 내 컴퓨터에서만 돈다 (localhost:3000). 인터넷에 올리는 건 3주차
- 앱은 내 폰에서 Expo Go로 연다. 스토어에 올리는 건 3주차
- 작업 폴더는 C:\workspace\month-claude-2-week1 — 동기화 폴더 밖에 둔다

변경 이력에 v0.4 한 줄. Claude가 이렇게 말하고 멈춥니다.

"만들 자리가 생겼습니다. 웹과 앱 폴더가 하나씩, 그리고 빈 화면이 하나 뒀어요. 이제 지난주에 고른 화면을 하나씩 만듭니다."

14.7 정리

- 11단계는 화면을 만들지 않는다. 만들 자리만 만든다
- 웹 도구만 고른다 — 모르면 추천대로(Next.js). 앱과 데이터는 정해져 있다
- 개발 환경 = Node(실행기) · npm(재료 배달부) · 실행 정책(윈도우 잠금장치). PC에 한 번만
- localhost:3000 은 내 컴퓨터에서만 도는 웹. 인터넷에는 아직 없다
- 12단계 전에 까만 창으로 옮겨 Claude가 크롬을 직접 누르게 한다

이 장에서 나온 말

말	쉬운 뜻
프레임워크	부품과 조립 규칙이 갖춰진 만들기 도구 세트. Next.js 등
개발 환경	프로그램을 만들 수 있게 PC에 갖춰 둔 도구와 설정. 부엌
npm · 패키지	남이 만든 부품(패키지)을 받아 오는 배달부
node_modules	npm이 받아 온 재료가 쌓이는 폴더. 파일이 수만 개
실행 정책	모르는 프로그램이 못 돌게 한 윈도우의 잠금장치
winget	윈도우에 들어 있는 설치 심부름꾼
localhost · 포트	"바로 이 컴퓨터" 라는 주소 · 그 안의 호수(3000호)
터미널	글자로 명령하는 까만 창
/chrome	까만 창의 Claude를 크롬에 연결하는 명령

14.8 안 될 때

증상	이렇게
npm ... cannot be loaded because running scripts is disabled	실행 정책이 막혀 있습니다. 11-② 상자 를 그대로 다시 붙여넣으세요
회사 PC라 실행 정책이 안 풀린다고 한다	회사가 잠가 둔 것입니다. Claude가 다른 이름으로 불러서 진행합니다. 그대로 두세요
Node가 없어서 깐다는데 창이 뜬다	「계속」이나 「예」를 누르세요. 3분쯤 걸립니다
설치 중에 EPERM · EBADF 가 나온다	폴더가 동기화 폴더 안인지 봐줘
화면이 안 뜬다	화면이 안 떠 — 주소가 3001일 수도 있습니다. Claude가 실제 주소를 알려줍니다
까만 창에서 /chrome 이 안 된다	두 번 해 보고 안 되면 그냥 진행하자
설치가 너무 오래 걸린다	지금 뭐 하는 중이야? 얼마나 걸려?

다음 장이 2주차의 본체입니다. 웹 첫 화면을 네 바퀴에 걸쳐 완성합니다.

15장 — 웹 첫 화면, 네 바퀴 — 문서 먼저 화면 나중

초고 v0.1 (2026-09-18) · 2주차 12단계

15.1 이 단계가 2주차의 본체다

드디어 화면을 만듭니다. 지난주에 고른 **첫 번째 화면**입니다. 공방 예시라면 웹 「오늘 클래스」 — 오늘 누가 오는지 한눈에 보는 명단입니다.

이 화면을 한 번에 만들지 않습니다. **네 바퀴**에 걸쳐 완성합니다. 그리고 바퀴마다 상자가 세 개입니다.

	상자	하는 일
㉑	문서	설계 문서가 먼저 바뀐다. 화면은 그대로
㉒	화면	바뀐 문서대로 화면이 따라간다
㉓	시험	문서 9장 확인 기준으로 눌러 보고 표로 남긴다

네 바퀴는 이렇게 돕니다.

바퀴	㉠ 문서	㉡ 화면	㉢ 시험
1 만들기	확인만 한다 3장 그림·동작 표.5.9 데이터를 보여준다	제일 큰 덩어리 + 가짜 데이터	9장 #1 · #2 를 눌러 본다
2 검색 붙이기	있는 문서를 넓힌다 무엇으로 찾아 · 못 찾으면 문구 → v0.4.1	검색칸을 붙이고 직접 쳐 본다	없는 값 · 뒷자리 일부까지 여기서 버그가 나온다
3 자세히 보기	없던 기능을 새로 그린다 6장 확인 → 그림 + 표 + 동작 표 · 누가 보나	이름을 누르면 오른쪽 칸	닫기 · 기록이 없는 사람
4 고치기	마음에 안 드는 곳을 문서부터 + 왜 바꿨는지 한 줄	시킨 것만 고친다	9장을 처음부터 끝까지 → ✓

바퀴마다 ㉠ → ㉡ → ㉢ · ㉠가 끝나면 화면은 아직 그대로

그림 15-1 12단계 네 바퀴. 바퀴마다 ㉠ 문서 → ㉡ 화면 → ㉢ 시험 순서로 돈다. 1바퀴는 문서를 확인만 하고, 2바퀴는 있는 문서를 넓히고, 3바퀴는 없던 기능을 문서에 새로 그리고, 4바퀴는 마음에 안 드는 곳을 문서부터 고친다

이 30분 동안 우리가 보는 것은 하나입니다. **문서가 먼저 바뀌고, 화면이 그걸 따라가는 것.** 네 번 봅니다. 지침서는 이걸 "오늘 가져갈 전부" 라고 적어 두었습니다.

💡 왜 네 번이나 도나

한 번 보면 "그렇구나" 로 끝납니다. 네 번 돌면 **손이 먼저 움직입니다**. 다음 주에, 혼자 할 때, 무언가 마음에 안 들면 "화면 고쳐줘" 가 아니라 "**문서부터 고쳐줘**" 가 먼저 나옵니다.

그리고 네 바퀴가 각각 다른 상황입니다 — 문서대로 만들기, 있는 기능 넓히기, **없던 기능** 넣기, 마음에 안 드는 곳 고치기. 앞으로 만나는 요청은 거의 이 넷 중 하나입니다.

15.2 1바퀴 · 만들기

12-1㉠ 문서 확인

첫 번째 화면 만들기 전에, 설게 문서에 그 화면이 어떻게 적혀 있는지 보여줘.

1바퀴의 ㉠은 **고치지 않습니다**. 10단계에서 채운 3장 그림·동작 표.5.9 넣어둘 데이터를 **보여주기만** 합니다. 지금부터 만들 화면이 문서에 어떻게 적혀 있는지 눈에 담아 두는 자리입니다.

12-1㉞ 화면 만들기

설계 문서대로 첫 화면 만들어줘. 한 번에 다 말고 제일 큰 덩어리만.
설계 문서에 적은 가짜 데이터로 채워서 화면 띄워 보여줘.

제일 큰 덩어리만 만듭니다. 12장의 "한 상자 = 한 기능" 입니다.

가짜 데이터는 **파일 하나에** 모아 둡니다. 화면마다 흩어 적지 않습니다. 이 약속이 나중에 효과를 봅니다. 데이터 창고를 붙일 때(22장) **그 파일 하나만** **같이 끼우면** 진짜 데이터로 바뀝니다.

Claude가 크롬에서 화면을 띄웁니다.

손끝 공방	오늘 클래스	9월 18일 금요일	① 14	10	2	3
오늘 클래스	오늘 오는 사람	남은 자리	출석 체크함	오늘 클래스 수		
주간	출석	이름	연락처	클래스	시간	②
알림판	③	✓	김하나	010-0000-0001	도자기 물레	10:00
	✓	이두리	010-0000-0002	도자기 물레	10:00	
	□	박세라	010-0000-0003	가족 카드지갑	15:00	
	□	최민준	010-0000-0004	가족 카드지갑	15:00	
	□	정유리	010-0000-0005	가족 카드지갑	15:00	④ 취소됨
	□	김하나	010-0000-0001	캔들 만들기	19:30	
	□	이두리	010-0000-0002	캔들 만들기	19:30	
	□	박세라	010-0000-0003	캔들 만들기	19:30	
	□	최민준	010-0000-0004	캔들 만들기	19:30	
	□	정유리	010-0000-0005	캔들 만들기	19:30	
	□	한서윤	010-0000-0006	캔들 만들기	19:30	
	□	오지후	010-0000-0007	캔들 만들기	19:30	
	□	배수린	010-0000-0008	캔들 만들기	19:30	

그림 15-2 1바퀴가 끝난 「오늘 클래스」. 문서 5.9에 적은 가짜 데이터로 채워졌다. ① 숫자 네 칸 ② 시간 순 명단 ③ 출석 ✓ 가 찍힌 줄 ④ 벽돌색으로 칠해진 취소된 줄

> "지금은 ****가짜 데이터****로 채워졌습니다. 화면 모양부터 보시라고요. 이따가 진짜로 바꿉니다."

문서에 그린 그림과 화면을 나란히 비교해 봅니다. 같게 나왔으면 문서가 잘 된 겁니다. 10단계에서 쓴 30분이 여기서 돌아옵니다 — 칸 순서도, 숫자 네 칸의 이름도, 취소된 줄의 색도 **묻지 않고 한 번에** 나옵니다.

12-1㉔ 시험

설계 문서 9장에 적은 확인 기준으로 이 화면을 시험해줘.
크롬에서 직접 눌러서 해줘. 못 누르면 터미널 창부터 띄우고.
입력한 것, 나온 결과, 9장 몇 번인지 표로 보여줘.

㉔는 「만들었다」를 「진짜 된다」로 바꾸는 자리입니다.

📖 시험(테스트, test)

만든 것이 정한 대로 움직이는지 직접 해 보는 일. 새 차를 받으면 시동을 걸고 창문을 올려 보고 깜빡이를 켜 보는 것과 같습니다. "만들었다"는 말만으로는 모릅니다.

무엇을 해 볼지는 미리 적어 둔 목록을 따릅니다. 우리 목록은 PRD 9장 확인 기준입니다.

Claude가 크롬에서 직접 들어가 보고, 출석칸을 눌러 보고, 결과를 표로 남깁니다. 이 표가 ㉔의 결과물입니다.

입력한 것	결과	9장
들어온다	오늘 날짜 · 숫자 네 칸 · 시간 순 명단 · 정유리 줄 취소됨	#1 ✓
박세라 출석칸 누름 → 다시 누름	✓ + 「출석 체크함」 +1 → ✓ 빠지고 -1	#2 ✓

무엇을 시험할지 Claude가 지어내지 않습니다. 9장에서 그 바퀴에 해당하는 줄만 가져옵니다. 9장에 없는 걸 시험해야겠으면 9장에 먼저 한 줄 씩입니다.

💡 코드를 돌려 본 것은 시험이 아니다

크롬을 못 누르는 창에서는, Claude가 화면 대신 화면이 쓰는 코드를 직접 돌려서 표를 채우고 싶어집니다. 결과는 비슷하게 나옵니다. 그런데 그건 시험이 아닙니다.

9장에는 「누른다」 「친다」라고 적혀 있습니다. 손님과 사장님은 코드를 돌리지 않고 화면을 누릅니다. 화면에서 버튼이 가려져 있거나, 글씨가 안 보이거나, 눌러도 반응이 없으면 — 코드는 멀쩡해도 손님에게는 고장입니다.

그래서 표에는 화면에서 나온 것만 적습니다. 14장에서 까만 창으로 옮긴 이유가 이것입니다. 누를 수 없으면 Claude가 무엇을 눌러 보면 되는지 알려 주고, 우리가 한 번 눌러 봅니다.

15.3 2바퀴 · 기능 붙이기 — 검색

명단이 생겼습니다. 이제 **전화가 오면 그 사람을 찾아야** 합니다. 검색을 붙입니다.

12-2㉓ 문서에 적기

이 화면에 이름으로 찾는 검색을 붙이고 싶어.
화면 말고 설계 문서에 먼저 적어줘. 무엇으로 찾는지, 못 찾으면 뭐라고 뜨는지.
변경 이력에도 남겨줘.
모르는 건 나한테 물어봐.

아직 화면은 안 바뀝니다. 문서만 바뀝니다. 그게 맞습니다.

Claude가 묻습니다 — 이름으로만 찾을지, 전화번호로도 찾을지. 공방 예시에서는 **"이름 또는 연락처 뒷자리"** 를 골랐습니다. 전화가 오면 번호가 먼저 보이기 때문입니다.

Claude는 PRD 3-2의 사양 표와 동작 표에 줄을 더하고, 9장에 확인 기준을 더하고, 변경 이력에 한 줄을 씁니다. 그리고 이렇게 말하고 **멈춥니다**.

"설계 문서 3-2 에 검색을 더했습니다. 변경 이력에도 v0.4.1 로 남겼고, 아직 화면은 그대로예요. 문서가 먼저입니다. 다음 상자에서 이대로 만듭니다."

지침서는 이 장면을 **"이 수업의 핵심 장면"** 이라고 부릅니다. 문서가 먼저 바뀌는 순간을 보는 것입니다.

버전 번호가 한 자리 늘었다

v0.4 다음이 v0.5가 아니라 **v0.4.1** 입니다.

언제	버전	예
단계가 끝날 때	둘째 자리를 올린다	v0.3 → v0.4
그 사이의 모든 변경	셋째 자리를 올린다	v0.4 → v0.4.1 → v0.4.2

오늘은 문서가 자주 바뀝니다. 바뀔 때마다 둘째 자리를 올리면 하루 만에 v0.9가 됩니다. 그래서 단계는 둘째 자리, 그 사이 손질은 셋째 자리로 나눕니다.

12-2㉞ 화면에 붙이기

설계 문서에 적은 대로 검색 붙여줘.
되는지 보여주고, 내가 뭘 쳐보면 되는지도 알려줘.

Claude가 검색칸을 붙이고, 크롬에서 직접 쳐 봅니다.

"크롬 보세요. 설계 문서에 적은 대로 만들었고, 제가 직접 쳐보겠습니다. ... 됩니다."

12-2㉟ 시험 — 여기서 버그가 나온다

검색을 9장 기준으로 크롬에서 직접 쳐서 시험해줘.
없는 값이랑 뒷자리 일부도 쳐봐.
결과를 표로 보여주고, 어긋난 게 있으면 고치고 다시 시험해줘.

상자에 "없는 값이랑 뒷자리 일부도" 가 들어 있습니다. 이게 중요합니다.

☐ 경계값(edge case, 엣지 케이스)

평소에는 잘 안 들어오는 끝자락의 입력. 없는 이름, 빈 칸, 일부만 맞는 숫자 같은 것입니다.
엘리베이터로 치면 정원 딱 그 인원이 탔을 때, 아무도 안 탔을 때입니다.

버그는 대개 가운데가 아니라 가장자리에서 나옵니다. 그래서 시험할 때 꼭 한 줄씩 넣습니다
— 없는 것 · 일부만 맞는 것 · 빈 칸.

공방 예시에서 실제로 이렇게 나왔습니다.

입력한 것	결과	9장
김	김하나만 남는다	#3 ✓
없는사람	찾는 분이 없습니다 · 칸 이름 줄은 그대로	#3 ✓
0001	김하나(010-0000-0001) 와 신도윤(010-0000-0010**) ** 이 같이 나옴	#3 X

0001 을 쳤는데 010-0000-0010 인 사람까지 딸려 나왔습니다. 번호 어딘가에 0001 이 들어 있기만 하면 찾도록 만들었기 때문입니다. 0000-0010 을 붙여 읽으면 0001 이 들어 있습니다.

전화를 받으면서 뒷자리로 찾았는데 엉뚱한 사람이 뜹니다. 가운데 값(「김」)만 쳐 봤다면 모르고 지나갔을 버그입니다.

고칠 때도 문서가 먼저

Claude는 화면만 고치고 넘어가지 않습니다. "뒷자리로 찾는다" 는 건 PRD에 없던 결정이기 때문입니다. 순서는 이렇습니다.

1. PRD 동작 표 — 연락처 검색을 「들어간다」 에서 「그 숫자로 끝난다」 로 고친다
2. 9장에 확인 한 줄 — 0001 → 김하나만. 신도윤은 안 나온다
3. 변경 이력 v0.4.2 — 왜: 0001 을 쳤더니 010-0000-0010 이 같이 나왔다
4. 화면을 고치고 다시 친다

버전	무엇	왜
v0.4.2	연락처 검색을 「들어간다」 에서 「그 숫자로 끝난다」 로	0001 을 쳤더니 010-0000-0010 이 같이 나왔다 — 뒷자리로 찾는 게 맞다
v0.4.1	검색을 이름 또는 연락처 뒷자리로, 못 찾으면 찾는 분이 없습니다	전화가 오면 번호로 먼저 찾게 된다

「왜」 칸을 보세요. "수강생 요청" 이라고 쓰지 않습니다. 무엇을 보고 왜 그렇게 느꼈는지를 씁니다. 이유가 없으면 석 달 뒤에 누군가 "번호 가운데로도 찾게 해 주세요" 라고 할 때 같은 버그를 다시 만듭니다.

💡 버그를 고친 것도 설계 문서에 남긴다

버그를 고치는 건 화면 일 같지만, 대개는 문서가 정하지 않은 빈칸이 버그로 드러난 겁니다. "뒷자리로 찾는다" 가 문서에 있었다면 Claude는 처음부터 그렇게 만들었습니다.

그래서 빈칸을 채우듯 문서에 먼저 적습니다. 그러면 오늘 밤 새 대화에서 이 화면을 다시 만들어도 버그가 다시 태어나지 않습니다.

공방 예시에서는 여기서 하나가 더 붙었습니다. 시험하는 걸 보다가 "도자기 클래스 명단만 따로 보고 싶다" 가 나와서, 검색에 클래스 이름까지 넣었습니다(v0.4.3). 역시 문서부터 적고, 9장에 한 줄 더하고, 화면을 고치고, 다시 쳤습니다.

15.4 3바퀴 · 없던 기능 — 누르면 자세히

12-3㉓ 문서에 그리기

명단에서 이름을 누르면 그 사람 정보를 보고 싶어.
설계 문서에 없는 기능이지? 먼저 문서에 그림이랑 표로 그리고,
변경 이력에도 남겨줘.

처음 생각난 기능은 늘 설계 문서에 없습니다. 그래서 문서부터 그림니다.

Claude는 순서대로 합니다.

1. 6장 「안 만드는 것」에 걸리나 본다. 걸리면 그 이유를 읽어 주고 "그래도 넣을까요?" 라고 묻는다. 넣기로 하면 6장에서 빼고 왜 뺐는지 남긴다
2. 3장에 그림 + 사양 표 + 동작 표를 새로 그린다 — 어디를 누르면, 무엇이 뜨고, 무엇을 보여주고, 어떻게 닫고, 비었을 때
3. 4장 「누가 보나」를 본다 — 연락처처럼 가려야 할 게 뜨면 적는다
4. 7장 만드는 순서에 한 줄, 9장 확인 기준에 한 줄
5. 그림을 보여주고 멈춘다

공방 예시에서 Claude는 두 가지를 물었습니다 — 어디에 뜯지(가운데 창 / 오른쪽 칸), 무엇을 보여줄지(오늘 신청만 / 지난 기록까지). **오른쪽 칸, 지난 기록까지**를 골랐습니다. 전화가 왔을 때 "전에 오셨던 분인가" 를 그 자리에서 보려고요.

그리고 1번에서 하나가 짚었습니다. **연락처가 보이는 칸**이라는 점입니다. 4장 「누가 보나」대로 **주인(웹)만** 봅니다. 앱에는 이 칸을 만들지 않는다고 문서에 적었습니다.

⚠ 큰 기능이면 3바퀴에서 만들지 않는다

3바퀴는 이미 있는 화면에 붙는 작은 기능이어야 합니다. 탭이 하나 늘어날 만큼 큰 것이면 여기서 만들지 않고, 16장의 간판처럼 **시안부터** 봅니다.

12-3㉔ 화면에 만들기

그린 대로 만들어줘. 되는지 보여주고, 내가 뭘 눌러보면 되는지 알려줘.

손끝 공방

오늘 클래스

주간

알림판

오늘 클래스

9월 18일 금요일

14
오늘 오는 사람

10
남은 자리

2
출석 체크함

3
오늘 클래스 수

이름 · 연락처 뒷자리 · 클래스

출석	이름	연락처	클래스	시간
☒	김하나	010-0000-0001	도자기 물레	10:00
☒	이두리	010-0000-0002	도자기 물레	10:00
☐	박세라	010-0000-0003	가족 카드지갑	15:00
☐	최민준	010-0000-0004	가족 카드지갑	15:00
☐	정유리	010-0000-0005	가족 카드지갑	15:00
☐	김하나	010-0000-0001	캔들 만들기	19:30
☐	이두리	010-0000-0002	캔들 만들기	19:30
☐	박세라	010-0000-0003	캔들 만들기	19:30
☐	최민준	010-0000-0004	캔들 만들기	19:30
☐	정유리	010-0000-0005	캔들 만들기	19:30
☐	한서윤	010-0000-0006	캔들 만들기	19:30

박세라

010-0000-0003

오늘

15:00 가족 카드지갑

19:30 캔들 만들기

지난 기록 - 3번 왔고 2번 출석

9월 6일 캔들 만들기

8월 30일 도자기 물레

8월 16일 가족 카드지갑

받기

신청

신청

참석함

참석함

안음

취소됨

그림 15-3 3바퀴가 끝난 「오늘 클래스」. 명단에서 박세라를 누르자 ① 오른쪽 칸이 열리고 ② 오늘 신청 두 줄과 ③ 지난 기록이 보인다. ④ 누른 줄에는 강조색 테두리가 남는다

12-3㉔ 시험

누르면 자세히 뜨는 걸 9장 기준대로 크롬에서 직접 눌러서 시험해줘.
닫기랑 비었을 때도 같이 봐줘. 표로 보여줘.

여기서도 경계값을 봅시다 — [닫기] 를 눌렀을 때, **지난 기록이 하나도 없는 사람**을 눌렀을 때(처음 오시는 분입니다).

15.5 4바퀴 · 마음에 안 드는 곳 고치기

이제 화면을 한번 쭉 봅시다. 그리고 **마음에 안 드는 걸 말로** 합니다.

12-4㉕ 문서 고치기

(화면을 보시고 - 마음에 안 드는 걸 말로)

이름 칸이 좁고, 위쪽 숫자가 너무 작아.

화면 말고 설계 문서부터 고쳐줘. 왜 바꾸는지 변경 이력에도 남겨줘.

위는 예시입니다. 내 화면을 보고 내 말로 합니다. **고칠 게 없으면 그대로 가자** 도 좋은 답입니다.

Claude는 PRD 3.0 공통(글자 크기)과 3-2(칸 폭)를 **먼저** 고치고, 변경 이력에 남깁니다.

버전	무엇	왜
v0.4.5	3-2 이름 칸 넓힘, 위 숫자 32 → 40	4바퀴 — 크롬에서 보니 숫자가 안 읽혔다

12-4㉞ 화면 고치기

고친 문서대로 화면 고쳐줘.

시킨 것만 바꿉니다. 하나만 바뀌어야 **뭐가 달라졌는지 눈에 보입니다**. 이왕 하는 김에 색도 손보고 간격도 바꾸면, 무엇이 좋아졌고 무엇이 나빠졌는지 모릅니다.

제작prompt는 4바퀴 끝에 이렇게 적어 두었습니다.

이 두 상자가 앞으로 제일 많이 하실 일입니다. 마음에 안 들면 → 문서 먼저 → 화면. 다음 주에도, 혼자 하실 때도 똑같습니다.

화면만 고치면 어떻게 될까요? 오늘은 멀쩡합니다. 그런데 다음에 **설계 문서로 다시 만들면** 숫자가 다시 32로 작게 나옵니다. 고친 게 문서에 없으니까요. 작가의 말의 "처음 만든 이유를 모르고 다르게 고친다" 가 그대로 일어납니다.

12-4㉟ 시험 — 이 화면 마무리

설계 문서 9장 확인 기준을 처음부터 끝까지 크롬에서 다시 시험해줘.
표로 보여주고, 다 맞으면 9장에 체크 남겨줘.

마지막 시험은 **처음부터 끝까지**입니다. 4바퀴에서 칸 폭을 바꾸다가 2바퀴의 검색이 망가지지 않았는지까지 봅니다. 다 맞으면 9장 해당 줄에 ✓ 가 찍힙니다.

9장에 ✓ 가 찍힌 만큼이 「다 만들었다」의 근거입니다. 여기까지 오면 첫 화면은 끝입니다.

15.6 변경 이력이 자란다

네 바퀴를 돌고 나서 PRD 맨 아래를 보면, 오늘 문서가 어떻게 자라왔는지가 한 줄씩 남아 있습니다.

변경 이력

버전	무엇	왜
v0.4.5	3-2 이름 칸 넓힘, 위 숫자 32 → 40	4바퀴 — 크롬에서 보니 숫자가 안 읽혔다
v0.4.4	3-2-1 자세히 보기 — 이름을 누르면 오른 쪽 칸에 오늘 신청 · 지난 기록	전화가 오면 그 사람을 명단에서 찾아 바로 확인해야 한다
v0.4.3	검색에 클래스 이름을 더함	한 클래스 명단만 따로 보고 싶다
v0.4.2	연락처 검색을 「들어간다」에서 「그 숫자로 끝난다」로	0001 을 쳤더니 010-0000-0010 이 같이 나왔다 — 뒷자리로 찾는 게 맞다
v0.4.1	검색을 이름 또는 연락처 뒷자리로, 못 찾으면 「찾는 분이 없습니다」	전화가 오면 번호로 먼저 찾게 된다
v0.4	3.5장 「무엇으로 만드나」 추가	만들 자리를 정했다

최신이 위 · 줄마다 「왜」가 있다 · 셋째 자리는 단계 사이의 손질

그림 15-4 네 바퀴를 돈 뒤의 변경 이력. v0.4(만들 자리)에서 시작해 검색 추가(v0.4.1) → 뒷자리 버그 수정(v0.4.2) → 클래스 이름 검색(v0.4.3) → 자세히 보기(v0.4.4) → 글씨·칸 고침(v0.4.5)이 쌓였다. 줄마다 「왜」가 있다

규칙은 하나입니다. **PRD를 한 글자라도 고쳤으면, 그 상자 안에서 한 줄 쓴다.** 나중에 모아서 쓰지 않습니다. "나중에 모아서 문서에 반영"은 반영되지 않기 때문입니다. 그리고 맨 위 **버전 줄**도 같이 바꿉니다. 둘이 어긋나면 어느 게 최신인지 모릅니다.

15.7 네 바퀴 밖에서 "이것도" 가 나오면

네 바퀴를 돌아 보면 자꾸 더 붙이고 싶어집니다. "여기에 메모도 달 수 있으면", "출석률 그래프도 있으면".

이렇게 말하면	Claude는
글씨가 작아 · 색이 달라	4바퀴처럼 — 문서를 먼저 고치고 화면. 되묻지 않는다
설계 문서랑 달라	문서 그림과 대조해 화면 을 맞춘다
이것도 있으면 좋겠는데	PRD 8장에 적어만 두고 넘어간다. 오늘 안 만든다

네 바퀴에서 멈춥니다. 더 붙이기 시작하면 오늘 못 끝냅니다. 8장에 적어 두면 사라지지 않습니다. 23장에서 다시 모읍니다.

15.8 정리

- 첫 화면을 네 바퀴로 완성한다 — 만들기 · 기능 붙이기 · 없던 기능 · 고치기
- 바퀴마다 ㉠ 문서 → ㉢ 화면 → ㉣ 시험. ㉠이 끝나면 화면은 **아직 그대로**다
- 시험은 9장 확인 기준으로 **화면에서 눌러 본 것만** 적는다. 경계값을 꼭 넣는다
- 버그를 고쳐도, 새 기능을 넣어도 **문서가 먼저**. 변경 이력에 **v0.4.0** 로 한 줄
- 9장에 ✓ 가 찍힌 만큼이 「다 만들었다」의 근거

이 장에서 나온 말

말	쉬운 뜻
바퀴	문서 → 화면 → 시험을 한 번 도는 것. 12단계는 네 바퀴
시험(테스트)	정한 대로 움직이는지 직접 해 보는 일. 9장 목록을 따른다
경계값	없는 값 · 빈 칸 · 일부만 맞는 값처럼 끝자락의 입력. 버그가 잘 나온다
버그(bug)	만든 것이 정한 대로 움직이지 않는 곳. 원래 뜻은 벌레
셋째 자리 버전	v0.4.1 · v0.4.2 — 단계 사이의 작은 손질에 붙이는 번호
✓ (체크)	9장 확인 기준을 화면에서 통과한 줄에 찍는 표시

15.9 안 될 때

증상	이렇게 치세요
화면이 설계 문서랑 다르다	설계 문서랑 다른데 문서대로 해줘
한 번에 다 만들어 버렸다	방금 건 취소하고 제일 큰 덩어리만 다시 하자
문서는 안 고치고 화면만 고쳤다	설계 문서에도 반영하고 변경 이력에 남겨줘
변경 이력에 「왜」가 비었다	변경 이력 왜 칸 채워줘. 뭘 보고 바꿨는지로
시험 표를 코드로 채운 것 같다	크롬에서 직접 눌러서 다시 해줘
크롬을 못 누른다고 한다	알려 주는 것을 한 번 눌러 보고 맞아 / 이상해
마음에 안 든다	oo가 마음에 안 들어. 설계 문서부터 고치고, 화면도 고쳐줘.

첫 화면이 끝났습니다. 다음 장에서 **둘째 화면**을 만들고, 앱에 들어가기 전에 **간판 시안**을 고릅니다.

16장 — 웹 둘째 화면과 간판 시안

초고 v0.1 (2026-09-18) · 2주차 13단계

16.1 둘째 화면은 바로 만든다

PRD 7장 「만드는 순서」의 2번 화면입니다. 공방 예시라면 웹 「주간」 — 한 주를 시간 × 요일 표로 보고, 빈 칸을 눌러 클래스를 여는 화면입니다.

첫 화면과 같은 고리입니다. 다만 ㉠ 문서 상자가 없습니다. 10단계에서 이 화면의 그림·동작 표까지 이미 채워 두었기 때문입니다. 바로 만듭니다. 상자는 셋입니다 — 틀 · 눌러지게 · 점검.

13-① 틀

설계 문서 순서대로 다음 화면 만들자. 한 번에 다 말고 틀만. 화면으로 보여줘.

손끝 공방 오늘 클래스 주간 알림판	주간 9월 14일 — 9월 20일							클래스 열기
		월 14	화 15	수 16	목 17	금 18 · 오늘	토 19	일 20
	10:00					① 도자기 물레 2/8	도자기 물레 5/8	가족 카드지갑 3/6
	14:00						② 캔들 만들기 마감	
	15:00					가족 카드지갑 2/6		
	19:30	캔들 만들기 4/10		도자기 물레 1/8		캔들 만들기 마감		

그림 16-1 13-① 이 끝난 「주간」. 이번 주 월~일 칸과 시간 줄, 문서 5.9대로 넣어 둔 가짜 클래스들. ① 열린 칸(이름 + 신청 수/정원) ② 꽉 찬 칸은 회색 마감. 목요일은 비워 두었다

13-② 눌러지게

설계 문서대로 빈 칸을 누르면 만들어지게 해줘. 되는지 보여줘.

Claude가 크롬에서 **빈 칸을 직접 누르고**, 창에 클래스 이름을 적고, [열기] 를 누릅니다. 칸에 새 클래스가 생깁니다. 그리고 한 가지를 더 보여줍니다. **새로고침을 누르면 방금 만든 클래스가 사라집니다.**

"지금은 누르면 화면에 생기지만, **새로고침하면 사라집니다.** 저장할 곳이 아직 없거든요. 이따가 진짜로 저장되게 만듭니다."

💡 왜 새로고침하면 사라지나 — 화면은 기억하지 못한다

지금 화면은 가짜 데이터 파일을 읽어서 그림니다(15장). 빈 칸을 누르면 화면이 **그 자리에서만** 클래스를 하나 더 그려 둡니다. 칠판에 분필로 적은 것과 같습니다.

새로고침은 화면을 **처음부터 다시 그리는** 일입니다. 다시 그릴 때는 원래 파일을 읽으니, 칠판에 적었던 건 지워지고 처음 모습으로 돌아갑니다.

칠판이 아니라 **공책**에 적으려면, 화면 바깥에 **따로 기록해 두는** 곳이 있어야 합니다. 그게 21~22장의 데이터 창고입니다.

이 말을 미리 해 두지 않으면 수강생이 "없어졌어요!" 하고 당황합니다. 그래서 지침서는 Claude에게 **숨기지 말고 먼저 보여주라고** 합니다. 크롬에서 새로고침까지 눌러서요.

마음에 안 드는 데가 있으면 4바퀴처럼 합니다.

설계 문서부터 고치고, 화면도 고쳐줘.

16.2 13-③ 앱으로 넘어가기 전에 웹을 확인받는다

앱 만들기 전에 웹 두 화면을 설계 문서의 확인 기준으로 점검해줘.

설계 문서랑 화면이 다른 데 있으면 알려줘.

Claude가 9장 확인 기준을 한 줄씩 크롬에서 눌러 보고 ✓ 를 찍은 다음, 이렇게 묻습니다.

"앱으로 넘어가기 전에 **웹부터 한 번 보고** 갈게요.

① 「오늘 클래스」 화면 — 지난주에 고르신 모양이 맞나요? ② 「주간」 화면 — 여기서 하나 만들어 보세요. 생기나요?

둘 다 괜찮으면 좋아 라고만 해주세요. 이상한 데가 있으면 지금 고치고 갑니다."

이상한 데가 나오면 **지금** 고칩니다. 앱까지 갔다가 돌아오지 않습니다.

그리고 문서와 화면이 다를 때는 Claude가 **어느 쪽이 맞는지** 묻습니다.

이럴 때	이렇게
문서가 맞다	화면을 문서에 맞춘다
화면이 맞다 (만들어 보니 이게 낫다)	문서를 고친다. 변경 이력에 한 줄

⚠ 여기서 웹을 끝냅니다

앱은 오늘 **제일 어려운 구간**입니다(17장). 거기서 막혔는데 웹까지 덜 되어 있으면 되돌아올 시간이 없습니다. 웹이 확실히 끝나 있어야, 앱이 안 풀리더라도 **오늘 만든 것이 남습니다.**

16.3 없는 화면은 시안부터 — 간판

앱에 들어가기 전에 하나를 더 합니다. 앱 첫 화면 맨 위의 **간판**입니다.

지난주 시안에서 앱 「둘러보기」를 골랐을 때, 맨 위 큰 자리는 **색만 칠해 두고** 넘어갔습니다. PRD에도 이렇게 남아 있습니다.

간판 자리는 지금 **큰 색면**이다. `images/`에 사진을 넣으면 그 자리에 들어간다.

그러니까 간판은 **아직 모양이 정해지지 않은 화면**입니다. 이걸 앱을 만들면서 그냥 만들면 어떻게 될까요? Claude가 그럴듯하게 하나 만듭니다. 13장의 "추측으로 채운다"가 또 일어납니다.

그래서 지난주에 배운 순서를 **그대로** 한 번 더 갑니다.

시안을 본다 → 고른다 → 문서에 적는다 → 그다음에 만든다

💡 새 화면이 필요하면 매번 시안부터

지난주 8단계는 한 번 하고 끝나는 행사가 아니었습니다. **없는 화면이 생길 때마다 하는 일**입니다. 간판이 그걸 한 번 더 겪게 해 줍니다.

그리고 **자리**가 중요합니다. 웹이 끝난 직후, 앱을 띄우기 전입니다. 앱을 폰에 띄우고 나서 시안을 보여주면, 이미 화면을 보고 있어서 **고를 마음이 없습니다**. 눈앞에 있는 게 답처럼 보이기 때문입니다(6장의 "그림이 정답 행세").

13-④ 시안 만들기

앱 첫 화면 맨 위에 들어갈 간판, 지난주 시안에 없었잖아.
만들기 전에 시안부터 보고 고를게. 몇 가지 만들어서 보여줘.

Claude가 먼저 **사진**을 묻습니다.

"앱 첫 화면 맨 위에 **간판**을 만들 거예요. 사진이 한 장 들어갑니다. 쓰실 사진이 있으면 **이 대화창에 그냥 끌어다 놓으세요**. 없으셔도 됩니다. 색으로 채운 걸로 보여드릴게요."

사진이 있으면 시안에 **실제로 넣어서** 보여줍니다. 여러 장을 주면 한 장만 씁니다. 넘기는 슬라이드는 오늘 만들지 않습니다. 사진이 없으면 간판 자리를 **색으로 채워서** 시안을 만듭니다. 공방 예시가 그랬습니다.

시안이 뜬니다. 지난주처럼 **한 페이지에 세로로 셋**, 폰 폭 틀 안에 그렸습니다.

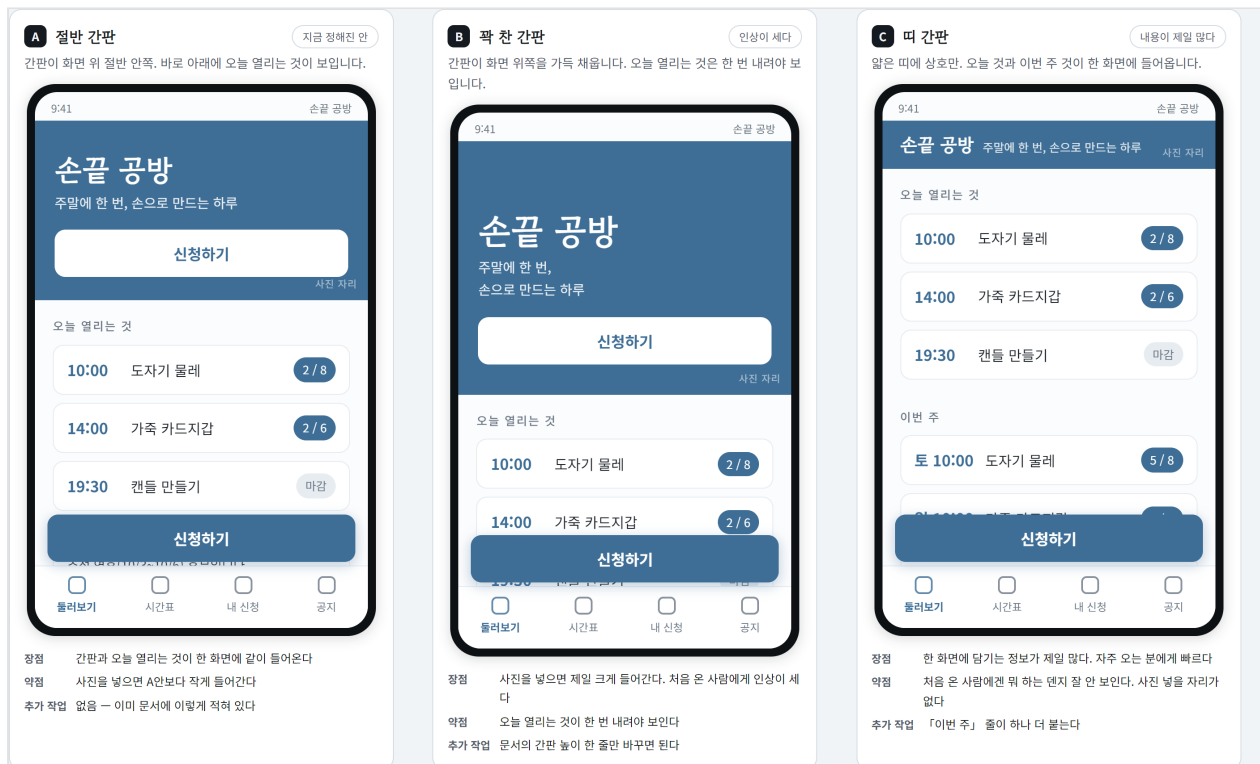


그림 16-2 간판 시안 세 안. A 절반 간판(지금 정해진 안) · B 꼭 찬 간판 · C 떠 간판. 사진이 없어 간판 자리를 강조색으로 채웠고, 안마다 장점 · 약점 · 추가 작업이 붙어 있다

| 안 | 어떻게 다른가 | 대가 ||---|---|---| | ****A**** 절반 간판 | 위 절반만 간판. 바로 아래에 오늘 열리는 것이 보인다 | 사진이 작게 들어간다 || ****B**** 짝 찬 간판 | 간판이 화면 위쪽을 ****가득**** 채운다. 인상이 제일 세다 | 아래 내용이 한 번 내려야 보인다 || ****C**** 띠 간판 | 얇은 띠에 상호만. 내용이 제일 많이 보인다 | 처음 온 사람에게 뭐 하는 덴지 안 보인다 |

A 자리에는 **지난주에 이미 정해 둔 것**(「표지를 반만」)이 들어갔습니다. 지난주 규칙 그대로입니다 — **현재 정해진 것도 한 안으로 넣는다.** 그래야 바꿀지 말지를 나란히 놓고 볼 수 있습니다.

13-⑤ 고르기

화면을 위에서 아래로 다 보고, 대화창으로 돌아와서칩니다.

B로 할게.

B인데 글씨가 좀 작아 처럼 말로 해도 됩니다. 그러면 B로 정하고 **그 하나만** 고쳐서 다시 보여줍니다. 모르겠어 도 됩니다.

공방 예시에서는 **B 짝 찬 간판**을 골랐습니다. 지난주에 정해 둔 A를 **뒤집은** 겁니다. 그리고 하나를 더 정했습니다. 시안에는 간판 안에 [신청하기] 버튼이 있었는데, 간판이 화면을 채우니 아래 떠 있는 신청 버튼과 **위아래로 붙어 보였습니다.** 그래서 간판 안 버튼은 빼기로 했습니다.

간판 사진 고르기

모양이 정해지니 이번엔 **어느 사진을 넣을지**가 남았습니다. 짝 찬 간판은 사진이 크게 들어가서 사진이 곧 첫인상입니다. 쓸 사진이 없어서 **A로 세 장을 만들고,** 같은 방식 으로 나란히 놓고 골랐습니다. 셋 다 같은 자리에 같은 글씨를 엮고, 사진 위에는 글씨가 읽히도록 **어두운 막**을 한 겹 덮었습니다.

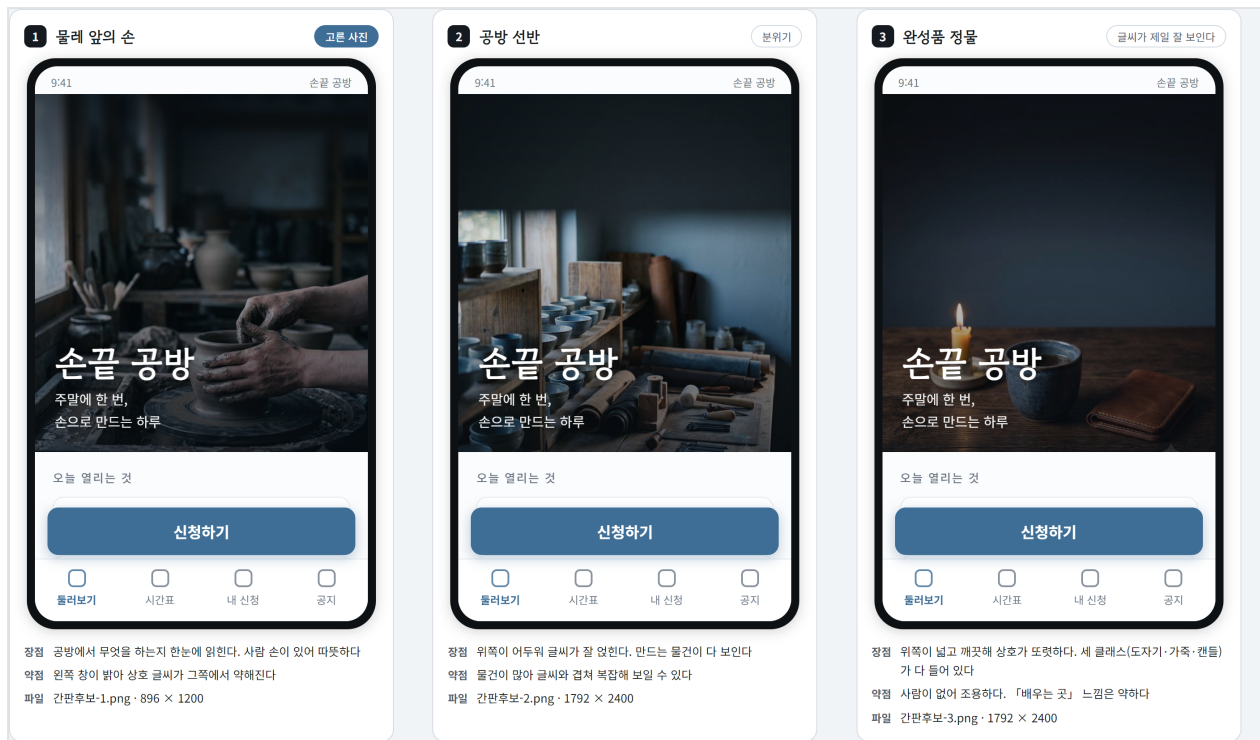


그림 16-3 간판 사진 후보 세 장. 1 물레 앞의 손 · 2 공방 전반 · 3 완성품 정물. 같은 자리에 같은 글씨를 얹고 장점 · 약점을 붙였다. 1번이 골라져 images/간판.jpg 로 들어갔다

1번 **물레 앞의 손**을 골랐습니다. 공방에서 무엇을 하는지 한눈에 읽히고, 사람 손이 있어 따뜻합니다. 약점도 적혀 있습니다 — 왼쪽 창이 밝아 상호 글씨가 그쪽에서 약해진다. 그래서 그 사진만 위쪽 막을 **한 단계 더 진하게** 넣었습니다.

13-⑥ 문서에 그리기

고른 거 설계 문서에 그림이랑 표로 적어줘. 안 고른 것도 왜 안 골랐는지 같이.

13장의 양식 그대로입니다. PRD 3장에 **3-5-1 간판**이 새로 생깁니다 — 배치 그림 + 사양 표 + 안 고른 안.

항목	사양
높이	화면 위쪽을 가득 . 처음 보이는 화면의 3분의 2쯤
바탕	사진 한 장 — images/간판.jpg
글자	상호(아주 크게 · 흰색) · 한 줄 소개. 간판 안에 버튼은 두지 않는다
사진 위 글씨	사진 위에 어두운 막을 한 겹 덮는다

안 고른 안 — A 절반 간판(지난주에 정했던 것): 간판과 오늘 것이 한 화면에 들어오지만 사진이 작게 들어간다 / C 띠 간판: 처음 온 사람에겐 뭐 하는 덴지 안 보이고 사진 자리가 없다

지난주에 정한 A를 뒤집었지만 **A를 지우지 않습니다**. 「안 고른 안」으로 내려 적고 이유를 남깁니다 (11장). 그리고 8장 「아직 안 정한 것」에 있던 「간판 자리에 넣을 사진」은 풀렸으니 지웁니다.

Claude가 이렇게 말하고 멈춥니다.

"간판 모양이 정해졌습니다. **아직 안 만들었어요. 도면만 그린 겁니다**. 이게 순서예요 — **시안 보고 · 고르고 · 적고 · 그다음에 만든다**. 지난주에 하신 거랑 똑같죠. 앞으로도 새 화면이 필요하면 매번 이렇게 갑니다."

만드는 건 18장, 앱 화면을 만들 때입니다. 그때는 시안을 다시 열지 않고 **문서에 그린 대로** 만듭니다.

16.4 정리

- 둘째 화면은 문서가 이미 채워져 있어 **㉠ 없이** 바로 만든다 — 틀 · 눌러지게 · 점검
- 지금은 **새로고침하면 사라진다**. 저장할 곳이 없어서다. 숨기지 않고 먼저 보여준다
- 앱으로 가기 전에 **웹을 확인받고 끝낸다**. 앱이 막혀도 웹은 남는다
- 없던 화면(간판)은 **시안 → 고르기 → 문서 → 만들기**. 지난주 순서를 매번 되풀이한다
- 정해 둔 안을 뒤집어도 **지우지 않고** 안 고른 안으로 이유와 함께 남긴다

이 장에서 나온 말

말	쉬운 뜻
새로고침(F5)	브라우저가 화면을 처음부터 다시 그리는 것
저장	화면 바깥에 따로 적어 두어, 다시 그려도 남게 하는 것. 첩판이 아니라 공책
간판	앱 첫 화면 맨 위 큰 자리. 지난주의 히어로 영역
점검	다음으로 넘어가기 전에 확인 기준으로 한 번 훑는 것

16.5 안 될 때

증상	이렇게 치세요
만든 클래스가 새로고침하면 없어졌다	정상입니다. 저장할 곳을 아직 안 만들었습니다
화면이 설계 문서랑 다르다	설계 문서랑 다른데 문서대로 해줘
점검 없이 앱으로 넘어가려 한다	앱 가기 전에 웹 두 화면 점검부터 해줘
간판을 바로 만들려 한다	만들기 전에 시안부터 보고 고를게
시안이 대화창에 안 뜬다	기다리세요. Claude가 파일로 만들어 크롬으로 엽니다
사진을 바꾸고 싶다	간판 사진 바꿔줘 (새 사진을 끌어다 놓으면서)

다음 장이 **오늘의 고비**입니다. 앱을 내 폰에 띄웁니다.

17장 — 앱을 내 폰에 띄우기

초고 v0.1 (2026-09-18) · 2주차 14단계

17.1 오늘의 고비

제작prompt는 14단계를 이렇게 소개합니다.

오늘의 고비입니다. **여기만 넘으면 됩니다.**

웹은 내 컴퓨터 안에서 돌아갑니다. 앱은 다릅니다. 코드는 컴퓨터에 있는데, 화면은 **폰에** 떠야 합니다. 컴퓨터와 폰 사이에 길이 하나 놓여야 하고, 그 길이 오늘 제일 자주 막힙니다.

그래서 이 장은 **되는 순서와 안 될 때 순서**를 같이 봅니다. 20분 잡혀 있습니다.

폰에 어떻게 앱이 뜨나

보통 앱은 스토어에서 받아 설치합니다. 그런데 스토어에 올리려면 설치 파일을 만들고 심사를 받아야 합니다(3주차). 오늘은 그럴 시간이 없습니다. 그래서 **남이 만든 앱 하나를 빌려서** 그 안에서 내 앱을 띄웁니다.

📖 Expo(엑스포)와 Expo Go

Expo 는 앱을 만드는 도구 세트입니다(14장의 프레임워크). 하나의 코드로 안드로이드와 아이폰 앱을 같이 만들 수 있습니다. 11단계에서 `app/` 폴더를 이걸로 만들었습니다.

Expo Go 는 폰에 까는 **빈 액자 같은 앱**입니다. 스토어에서 받습니다. 혼자서는 아무것도 안 보이지만, 컴퓨터에서 만들고 있는 앱을 **불러와서 그 안에 띄워** 줍니다. 스토어에 올리진 않아도 내 폰에서 내 앱을 바로 볼 수 있는 이유가 이것입니다.



그림 17-1 폰에 앱이 뜨는 길. 컴퓨터의 app 폴더에서 개발 서버가 돌고, 폰의 Expo Go 가 QR 코드로 주소를 받아 같은 와이파이를 타고 앱을 불러온다. 컴퓨터에서 코드를 고치면 같은 길로 폰이 따라 바뀐다

17.2 14-① 준비 확인

이제 앱 차례야. 시작하기 전에 내가 폰에서 뭘 준비해야 하는지 알려줘.

폰에서 할 일은 **우리가** 합니다. 이건 Claude가 대신 못 합니다. 그래서 말이 아주 짧습니다.

"폰에서 **플레이스토어**(아이폰은 앱스토어) 열고 Expo Go 검색해서 설치해 주세요. 다 되면 **깔았어** 라고 알려주세요."

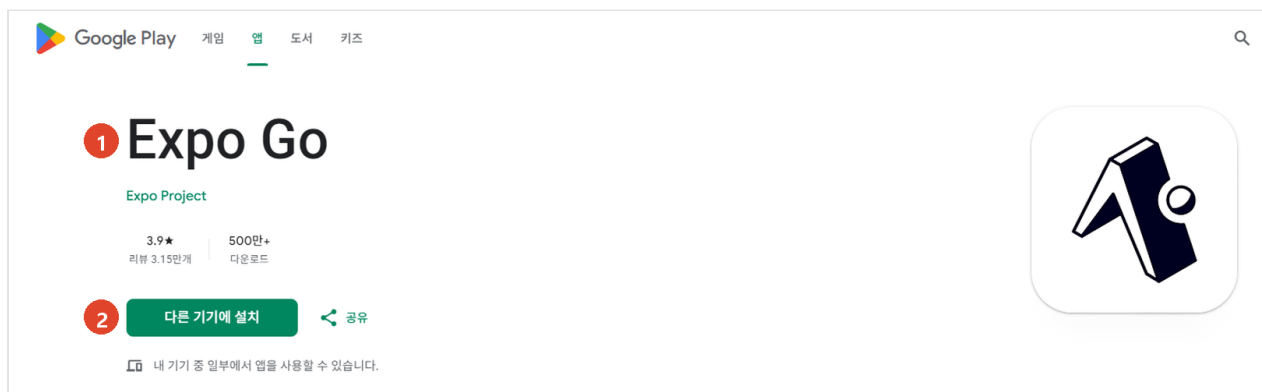


그림 17-2 플레이스토어의 Expo Go. ① 앱 이름 ② 설치 버튼 — 폰에서 열면 「설치」로 보인다

그리고 하나를 확인합니다. **폰과 컴퓨터가 같은 와이파이**에 있어야 합니다. 폰이 LTE로 붙어 있거나, 컴퓨터가 유선으로만 붙어 있으면 길이 안 놓입니다.
안드로이드든 아이폰이든 둘 다 됩니다. 안내 문구만 조금 다릅니다.

17.3 버전이 안 맞으면 앱이 아예 안 열린다

QR을 찍기 **전에** Claude가 먼저 하는 확인이 있습니다. 지침서가 별표 셋을 붙여 둔 자리입니다.
"가장 크고, 가장 흔하고, 모르면 절대 못 고치는 사고" 라고요.

📦 SDK 버전 — 부품 세트의 판 번호

SDK(Software Development Kit, 개발 도구 모음) 는 앱을 만드는 데 쓰는 부품 세트입니다.
Expo의 부품 세트도 판이 계속 올라갑니다 — 54판, 55판... 57판.

문제는 **폰의 Expo Go가 읽을 수 있는 판이 정해져 있다**는 것입니다. 옛날 DVD 플레이어는 블루레이를 못 읽는 것처럼, 폰의 Expo Go보다 새 판으로 만든 앱은 **아예 열리지 않습니다**.

11단계에서 `app/` 폴더를 만들 때, 도구는 **가장 새 판**으로 만듭니다. 그런데 스토어에 올라온 Expo Go는 그보다 **낮은 판**일 수 있습니다. 그러면 QR을 찍어도 이런 글씨만 뜹니다.

Project is incompatible with this version of Expo Go

강사가 준비할 때 실제로 이랬습니다. 프로젝트는 57판, 스토어의 Expo Go는 54판. **프로젝트를 54판으로 내려서** 해결했습니다. 코드는 한 줄도 고치지 않았습니니다.

💡 폰을 올리는 게 아니라, 프로젝트를 내린다

"폰 앱을 최신으로 업데이트하면 되지 않나?" 싶습니다. 안 됩니다. 폰의 Expo Go를 최신으로 올려도 **스토어에 아직 안 나온 판은 못 엽니다**.

그래서 순서가 정해져 있습니다. **폰의 Expo Go 버전을 먼저 보고, 프로젝트를 거기에 맞춰 내린다**. 반대로 하지 않습니다.

그리고 이 확인을 **QR을 찍기 전에** 합니다. 찍고 나서 빨간 글씨가 뜬 뒤에 고치면, 수강생은 이미 "내가 뭘 잘못했나" 하고 있습니다.

Claude는 버전 이야기를 길게 하지 않습니다. 한 줄이면 됩니다.

"폰에 깔린 버전이랑 맞춰야 해서 버전을 하나 낮췄습니다. 화면은 그대로예요."

17.4 14-② 띄우기 — 빈 화면으로 띄우지 않는다

준비 다 했어. 폰에 띄워줘.

띄우기 전에 내 가게 이름 한 줄만 넣어줘. 빈 화면 말고.

상자에 "내 가게 이름 한 줄만" 이 들어 있습니다. 이유가 있습니다.

새로 만든 앱을 그대로 띄우면 폰에 이런 영어가 뜹니다. Open up App.js to start working on your app! 설치하고, QR 찍고, 와이파이 맞추고, 고생 끝에 뜬 게 남의 영어 안내문이면 보람이 없습니다. 13장의 "빈 화면이 최악이다" 가 여기에도 적용됩니다.

그래서 띄우기 전에 PRD 1장의 가게 이름을 화면 가운데에 크게 한 줄 넣습니다. 화면을 만드는 게 아닙니다. 목록도 버튼도 없습니다. 그건 15단계(18장) 차례입니다. 이름 한 줄뿐입니다.

그다음 Claude가 app/ 에서 개발 서버를 띄웁니다.

📄 개발 서버(development server)

만들고 있는 앱을 폰이나 브라우저에 실시간으로 보내 주는 프로그램. 방송국의 생방송 송출기와 같습니다. 켜져 있는 동안은 코드가 바뀔 때마다 새 화면을 내보냅니다.

웹의 localhost:3000 도 개발 서버가 띄운 것입니다. 끄면 화면도 끊깁니다.

컴퓨터 화면에 QR 코드가 뜹니다.

"컴퓨터 화면에 QR 코드가 떴습니다. 안드로이드 — Expo Go 앱을 열고 Scan QR code 를 눌러 찍으세요. 아이폰 — 기본 카메라로 QR을 비추면 알림이 뜹니다. 그걸 누르세요.

처음엔 10초쯤 하얀 화면이 보입니다. 정상입니다. 기다려 주세요."



그림 17-3 개발 서버를 띄우면 컴퓨터에 뜨는 QR 코드. ① 폰의 Expo Go 로 이 QR을 찍는다 ② QR 안에 든 주소 — 컴퓨터의 와이파이 주소라서 폰이 같은 와이파이에서 찾아온다

QR 코드에는 *****컴퓨터의 이 주소로 와서 앱을 받아 가라***** 는 주소가 들어 있습니다. 폰이 그 주소를 읽고, 같은 와이파이를 타고 컴퓨터에 찾아와 앱을 받아 갑니다(그림 17-1). 10초쯤 하얀 화면이 지나고, **내 가게 이름**이 폰에 뜹니다.

손끝 공방

그림 17-4 가게 이름 한 줄만 넣고 띄운 첫 화면. 목록도 버튼도 아직 없다

> "폰에 뜬 그 이름, **○○님 가게 이름**이죠. 지금 폰에 뜬 건 **제 컴퓨터가 아니라 > ○○님 폰에서 도는 앱**입니다. 화면은 다음 단계에서 만듭니다. > 지금은 **뒀다는 것**이 중요해요."

17.5 14-③ 따라 바뀌는지 보기

폰에 떴어. 그 이름 글씨 크기나 색 바꿔봐. 폰이 따라 바뀌는지 볼게.

Claude가 컴퓨터에서 가게 이름의 글씨를 키우거나 색을 바꿉니다. 그리고 **폰을 봅니다**. 몇 초 안에 폰의 글씨가 따라 바뀝니다. 앱을 다시 켜지 않았는데도요.

📖 바로 반영(Fast Refresh, 패스트 리프레시)

코드를 고치면 **앱을 껐다 켜지 않아도** 화면이 곧바로 바뀌는 기능. 개발 서버가 켜져 있는 동안 코드가 바뀐 부분만 폰에 다시 보내 줍니다.

생방송 중에 자막을 고치면 화면에 바로 나가는 것과 같습니다.

아무 글씨나 고치지 않습니다. **내 가게 이름이 눈앞에서 바뀌는 것**을 보는 게 오늘 앱 구간의 첫 성취입니다. 지침서는 이 시연을 **건너뛰지 말라고** 적어 두었습니다.

"폰에 뜬 게 ○○님이 만든 앱입니다. 지금 컴퓨터에서 글씨를 바꾸면 **폰이 바로 따라 바뀝니다**."

앞으로 앱 화면을 만들 때는 늘 이 상태입니다. Claude가 코드를 고치면 폰이 따라 바뀌고, 우리는 **폰을 보고** 말합니다.

17.6 안 될 때 — 위에서부터 차례로

앱을 폰에 띄우는 건 막히는 곳이 정해져 있습니다. Claude는 원인을 묻지 않고 **이 표를 위에서부터** 시도합니다.

증상	먼저	그래도 안 되면
incompatible with this version of Expo Go	프로젝트 판을 내린다 (17.3). 폰을 탓하지 않는다	—
QR을 찍어도 아무 일 없다	Expo Go를 열고 앱 안에서 찍는다	같은 와이파이인지
계속 하얀 화면	30초 더 기다린다	앱을 껐다 켜다
Something went wrong	개발 서버를 껐다 다시 띄운다	아래
폰이 컴퓨터를 못 찾는다	같은 와이파이인지 — 컴퓨터가 유선이거나 폰이 LTE면 안 된다	터널 모드
회사·학교 와이파이	기기끼리 통신이 막혀 있다	터널 모드

터널 모드 — 와이파이가 달라도 되는 길

터널 모드(tunnel)

폰과 컴퓨터가 같은 와이파이로 못 만날 때, **인터넷 너머의 중간 지점을 거쳐** 만나게 하는 방법. 같은 동네에서 직접 못 만나니 서울역에서 만나는 것과 같습니다.

돌아가는 길이라 **조금 느립니다**. 대신 와이파이가 달라도, 회사 와이파이여도 됩니다.

와이파이 문제가 의심되면 Claude는 붙잡지 않고 **바로 터널로** 갑니다.

"와이파이 때문인 것 같습니다. **다른 방식으로 다시 띄울게요**. 조금 느릴 수 있어요."

5분이 넘으면 — 브라우저로 먼저

5분 넘게 안 되면 붙잡지 않습니다. 앱을 **크롬에서 먼저** 봅니다. Expo는 앱을 웹 화면으로도 띄울 수 있습니다. 단, 크롬 창을 **폰 폭(390 × 844)으로 좁혀서** 봅니다. 넓은 창에서는 틀린 모양도 그럴듯해 보이기 때문입니다.

그런데 브라우저로는 **못 잡는 것**이 있습니다 — 폰 위쪽 시계 자리, 아래쪽 버튼 자리, 실제 글자 크기, 손가락으로 누르는 느낌. 이걸 **실제 폰에서만** 보입니다. 그래서 폰에 띄우는 건 건너뛰지 않고, 쉬는 시간이나 수업 끝나고 따로 잡습니다. 한 사람 때문에 전체가 멈추지 않게 하는 겁니다.

오늘 안 하는 것

안 한다	왜
Android Studio · Xcode	개발자용 큰 프로그램. 오늘 안 쓴다. 이름만 들어도 겁난다
에뮬레이터(컴퓨터 안의 가짜 폰)	10GB를 깔아야 한다. 진짜 폰이 있다
앱 설치 파일(.aab) 만들기	3주차
화면 만들기	14단계는 띄우는 것까지 . 화면은 다음 단계

17.7 정리

- 폰의 **Expo Go**(빈 액자 앱)가 컴퓨터의 앱을 불러와 띄운다. 스토어는 3주차
- 판 번호(SDK)가 안 맞으면 **아예 안 열린다**. 폰에 맞춰 **프로젝트를 내린다**. QR 찍기 전에
- 빈 화면 말고 **내 가게 이름 한 줄부터** 띄운다
- 컴퓨터에서 고치면 **폰이 바로 따라 바뀐다**
- 폰이 못 찾으면 → 같은 와이파이 → **터널 모드** → 5분 넘으면 브라우저로 먼저

이 장에서 나온 말

말	쉬운 뜻
Expo	한 코드로 안드로이드·아이폰 앱을 만드는 도구 세트
Expo Go	폰에 까는 빈 액자 앱. 컴퓨터의 앱을 불러와 띄운다
SDK · 판 번호	앱을 만드는 부품 세트와 그 판. 폰이 읽을 수 있는 판보다 새면 안 열린다
개발 서버	만들고 있는 앱을 폰·브라우저로 실시간 내보내는 송출기
QR 코드	폰 카메라로 찍으면 주소가 읽히는 네모 무늬
바로 반영	코드를 고치면 앱을 다시 켜지 않아도 화면이 바뀌는 것
터널 모드	같은 와이파이가 아닐 때 인터넷 너머 중간 지점을 거쳐 잇는 길
에뮬레이터	컴퓨터 안에 띄우는 가짜 폰. 오늘은 안 쓴다

17.8 안 될 때 — 이렇게 치세요

증상	이렇게 치세요
폰에 앱이 안 뜬다	폰에 안 떠
폰에서 열었는데 하얗다	폰이 하얀 화면이야 (30초는 기다려 보세요)
빨간 화면에 영어가 떴다	폰에 빨간 글씨 나왔어
QR을 어디서 찍는지 모르겠다	QR 어디서 찍는지 그림으로 알려줘
회사 와이파이다	회사 와이파이야 — 터널 모드로 다시 띄워 줍니다

폰에 앱이 떴습니다. 오늘 제일 어려운 데를 넘었습니다. 다음 장에서 여기에 화면을 넣습니다.

18장 — 앱 화면과 간판, 문서대로 하나씩

초고 v0.1 (2026-09-18) · 2주차 15단계

18.1 이번엔 폰을 보며 만든다

PRD 7장 「만드는 순서」의 **3번** 화면입니다. 공방 예시라면 앱 「시간표」 — 위에 달력, 아래 그날 열리는 클래스 목록. 수강생이 여기서 신청합니다.

웹에서 한 것과 똑같습니다. **문서에 그린 대로, 한 덩어리씩**. 다른 점은 하나입니다. **컴퓨터 화면이 아니라 폰을 봅니다**. Claude가 코드를 고치면 폰이 따라 바뀌니까요 (17장의 바로 반영).

상자는 여섯입니다. 한 상자에 한 덩어리입니다.

상자	만드는 것
15-①	맨 아래 탭만 . 안은 비어 있어도 된다
15-②	문서 그림에서 제일 위에 있는 것부터
15-③	그 아래 목록 — 가짜 데이터로
15-④	신청 버튼 — 규칙대로 막히게
15-⑤	간판 — 16장에서 그려 둔 대로
15-⑥	폰에서 점검

앱 화면 만들자. 설계 문서대로 맨 아래 탭만 먼저 만들어줘. 안은 비어 있어도 돼.

설계 문서에 그린 화면에서 제일 위에 있는 것부터 만들어줘.

그 아래 목록 붙여줘. 설계 문서에 적은 가짜 데이터로.

목록에는 **웹과 같은 가짜 데이터**를 씁니다. 아직 웹과 앱이 연결되지 않았으니, 같은 내용을 앱 폴더에도 한 벌 둡니다. 지금은 두 벌이지만, 데이터 창고를 붙이면(22장) 한 벌이 됩니다.

그리고 로그인 만들지 않습니다. 로그인 3주차입니다. 앱은 "나는 김하나" 로 고정합니다. PRD 5.9에 적어 둔 가짜 수강생 한 명입니다. 오늘 앱에서 누르는 신청은 전부 김하나의 신청이 됩니다.

18.2 15-④ 신청 버튼 — 누르기 전에 막는다

신청 버튼 붙여줘. 설계 문서 규칙대로 - 자리가 다 찼으면 누르기 전에 막히게.

상자에 "누르기 전에 막히게" 가 들어 있습니다. 10단계에서 3.0 공통에 적어 둔 약속 입니다.

공통 상태	웹	앱
규칙에 막힘	버튼을 누르기 전부터 흐리게 + 이유 한 줄	같다

자리가 다 찬 클래스에 **눌리는 버튼**이 남아 있으면 안 됩니다. 눌러 놓고 "마감입니다" 를 보여주는 것보다, 처음부터 회색 **마감** 으로 두는 게 손님에게 친절합니다. 이미 시작한 클래스도 같습니다 — 흐리게 **지남** .

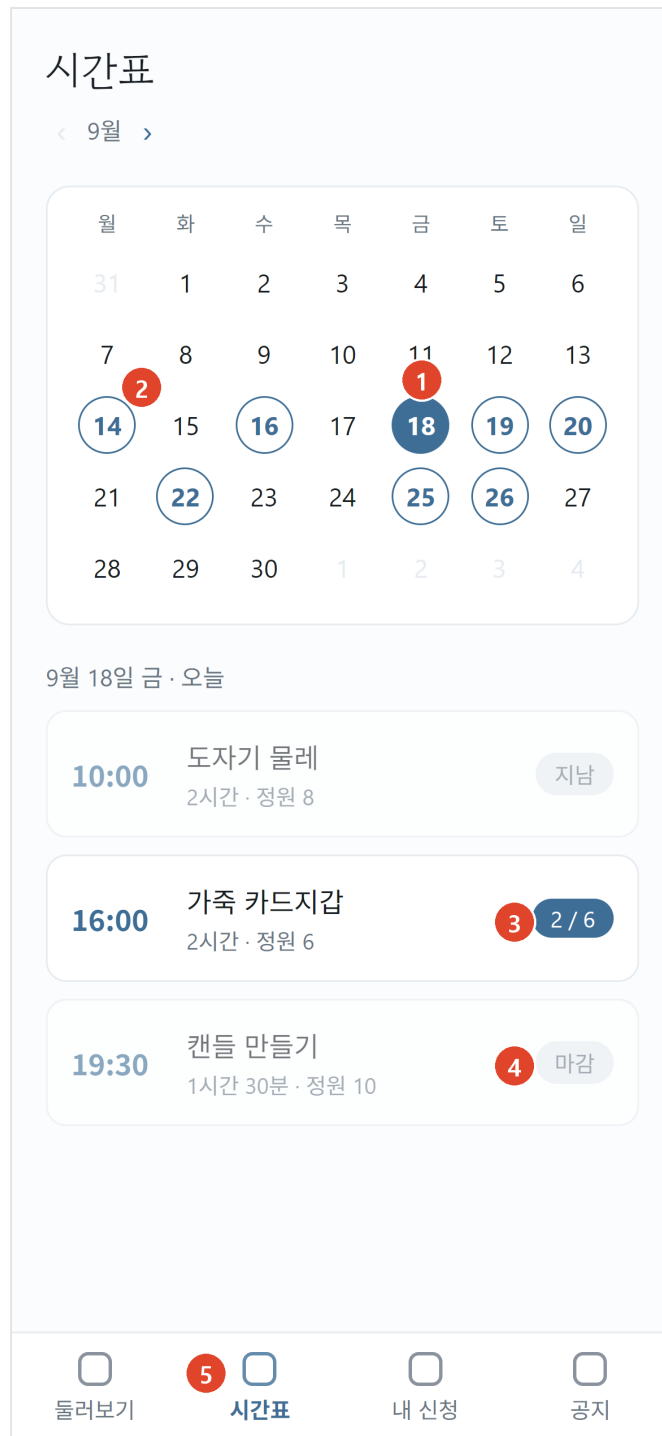


그림 18-1 15-④ 가 끝난 앱 「시간표」. ① 오늘(18일)은 채운 동그라미 ② 클래스가 있는 날은 테두리 ③ 자리 있는 카드는 신청 수/정원 알약 ④ 짝 찬 캔들 만들기는 누르기 전부터 회색 마감 ⑤ 맨 아래 탭 네 개. 이미 시작한 10:00 클래스는 흐리게 「지남」

아직 **저장은 안 됩니다.** 신청을 누르면 알약 숫자가 하나 올라가고 '신청함' 이 붙지만, 앱을 다시 열면 처음으로 돌아갑니다. 16장의 "칠판에 적은 것" 과 같은 사정입니다.

18.3 안드로이드에서 반드시 지키는 세 가지

수강생 대부분이 안드로이드 폰을 씁니다. 그리고 강사가 준비하면서 **안드로이드에서만 터진** 함정이 세 개 있었습니다. 10단계에서 PRD 10장 「만들 때 지킬 것」에 Claude가 미리 적어 둔 것들입니다. 나중에 고치면 두 배로 걸리니 **처음부터** 이렇게 만듭니다.

① 맨 아래 탭이 폰 버튼과 겹친다

갤럭시 폰 맨 아래에는 **뒤로 · 홈 · 최근 앱** 버튼이 있습니다. 앱의 탭을 화면 맨 아래에 그대로 붙이면 이 버튼들과 **같은 줄에 겹칩니다**. 강사 폰에서 실제로 탭을 누르면 **홈 버튼이 눌렸습니다**.

안전 영역(safe area)

폰 화면 중에 **앱이 마음 놓고 그려도 되는 부분**. 맨 위 시계·배터리 줄과 맨 아래 버튼 줄을 뺀 가운데입니다. 주차장의 "주차 금지선" 안쪽이라고 보면 됩니다.

문제는 이 금지선의 두께가 **폰마다 다르다**는 것입니다. 버튼이 세 개 있는 폰은 두껍고, 손가락으로 밀어 움직이는 폰은 얇습니다.

그래서 여백을 **숫자로 박아 두지 않고** 폰에게 물어서 받아 옵니다. "너는 아래에 얼마나 비워야 하니?" 그 값만큼 탭을 올립니다. 어떤 폰에서 열어도 겹치지 않습니다.

② 알림은 화면 안에 그린다

"신청했습니다" 같은 알림을 **팝업 창**(화면 가운데 툭 튀어나오는 작은 창)으로 띄우면, 폰에서는 뜨는데 **크롬 미리보기에서는 아무 일도 안 일어납니다**. 조용히 사라집니다. "신청이 안 돼요"의 흔한 정체가 이것입니다. 실제로는 뜨고 있었는데 안 보였을 뿐입니다.

그래서 알림은 **화면 안에 남는 한 줄로 만듭니다**. 신청할지 묻는 창도 팝업이 아니라 화면 안에 그립니다.

③ 사진이 화면을 밀어낸다

사진을 넣을 때 **크기를 같이 주지 않으면** 사진이 원래 크기(예: 가로 900 × 세로 1300)로 박혀서 화면이 폰 폭을 벗어납니다. 화면이 옆으로 밀리고 글씨가 잘립니다.

고약한 건 화면이 **"그려지긴"** 한다는 점입니다. 에러가 안 나니 늦게 발견됩니다. 그래서 사진에는 처음부터 **"가로·세로 100%로 채워라"** 를 같이 줍니다. 간판을 만들 때(18.4절) 바로 이 자리입니다.

컴퓨터 화면으로는 못 잡는 것이 있다

앱도 크롬에서 미리 볼 수 있습니다. 편하지만, 넓은 창에서 보면 **틀린 모양도 그럴듯해 보입니다**. 그래서 크롬에서 볼 때는 창을 폰 폭(390 × 844)으로 좁힙니다.

그래도 못 잡는 게 남습니다. 위 세 가지가 전부 그렇습니다 — 폰 버튼과 겹치는 것, 팝업이 뜨는지, 실제 글자 크기. **진짜 폰에서만** 보입니다. 14단계에서 고생해서 폰에 띄운 이유가 이것입니다. 마지막 확인은 늘 **내 손에 든 폰**으로 합니다.

18.4 15-⑤ 간판 만들기

아까 설계 문서에 그려둔 간판대로 만들어줘.

16장에서 **도면만 그려 둔 것**을 이제 진짜로 만듭니다. Claude는 시안을 다시 열지 않습니다. **여기서 새로 정하지도, 물어보지도 않습니다**. PRD 3-5-1에 다 적혀 있기 때문입니다. 10단계에 쓴 시간이 또 한 번 돌아오는 자리입니다.

간판을 만들 때 챙기는 것이 세 가지 있습니다.

	이렇게	안 그러면
사진 크기	가로·세로 100% 로 채운다	사진이 폰 폭을 벗어난다 (18.3 ③)
위쪽 시계 줄	사진은 시계 줄 밑까지 채우고, 글자만 안전 영역만큼 내린다	상호가 시계·배터리와 겹친다
글자가 안 읽히면	사진 위에 어두운 막을 한 겹 덮는다	밝은 사진 위 흰 글씨가 사라진다

사진을 바꾸라고 하지 않습니다. 막을 한 겹 깔면 됩니다. 공방 예시의 사진은 왼쪽 창이 밝아서 막을 **한 단계 진하게** 넣었습니다. 16장에서 사진을 고를 때 이미 문서에 적어 둔 것입니다.

그리고 여기서 하나가 실제로 터졌습니다. 사진 파일 이름이 `간판.jpg` 였는데, 안드로이드는 그림 파일 이름에 **한글이 들어가면 말뼉**을 부립니다. 앱 폴더 안에서는 `sign.jpg` 로 이름을 바꿨고, 그 사실을 PRD에 적었습니다. **다시 만들 때 같은 데서 안 걸리게**.

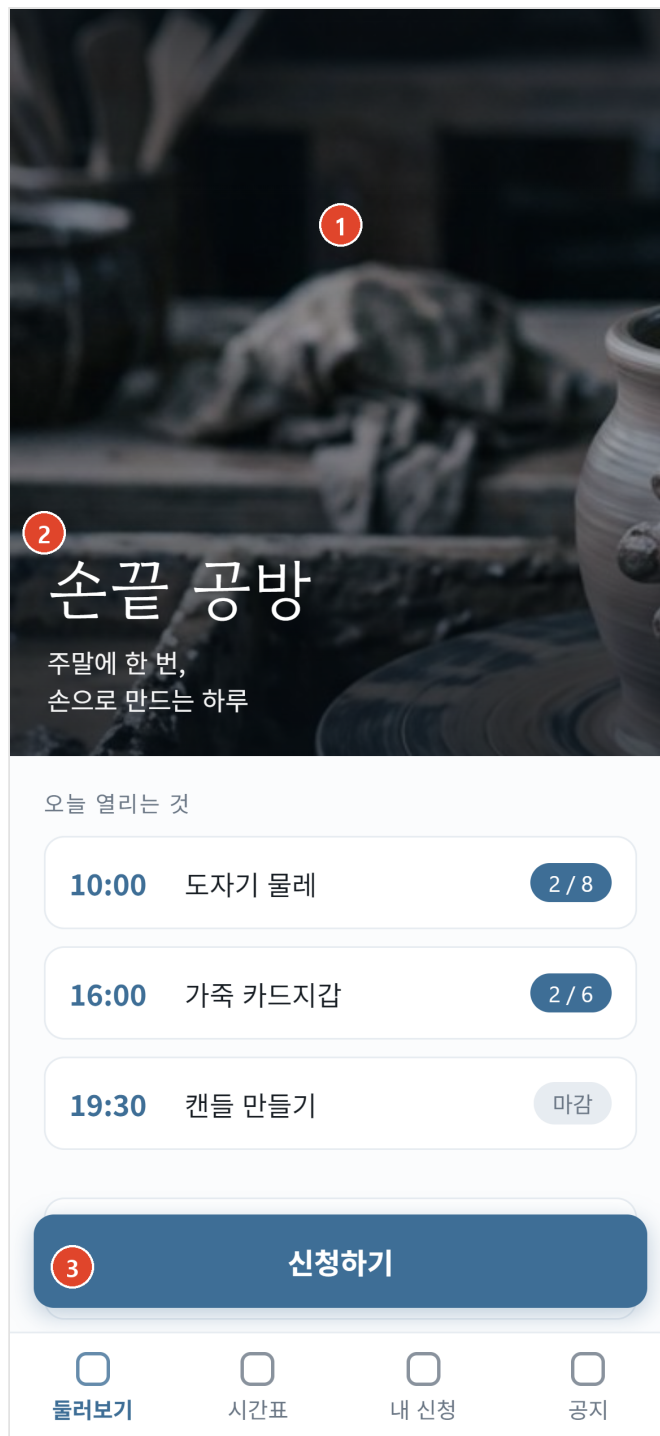


그림 18-2 앱 첫 화면의 짝 찬 간판. ① 사진이 화면 위쪽을 가득 채운다 ② 사진 위 어두운 막 덕분에 흰 상호가 읽힌다 ③ 탭 바로 위에 떠 있는 신청하기 버튼

> "폰 보세요. **아까 고르신 간판이 그대로 들어갔습니다.** > 지난주엔 색칠만 해줬던 자리예요. > 사진 바꾸고 싶으시면 언제든지 말씀하세요. **한 줄이면 바꿉니다.**"

18.5 15-⑥ 폰에서 점검

폰에서 볼게. 설계 문서 확인 기준으로 점검해주고, 글씨가 잘 읽히는지도 봐줘.

9장 확인 기준에서 폰 줄을 하나씩 봅니다.

#	어디서	이렇게 하면	이게 보여야 한다
10	폰 시간표	들어온다	오늘이 채운 동그라미 · 오늘 카드 3개
11	폰 시간표	마감 카드를 누른다	회색 마감, 아무 일 없음
12	폰 시간표	자리 있는 카드 → [신청]	신청했습니다 · 알약 +1 · 신청함
14	폰 (안드로이드)	맨 아래 탭을 누른다	폰 버튼과 안 겹친다

여기서도 하나가 잡혔습니다. 첫 화면을 맨 아래까지 내렸더니, 떠 있는 [신청하기] 버튼이 **마지막 글을 가리고** 있었습니다. PRD 3-5에 "마지막 글까지 버튼 위로 올라와야 한다 — 목록 아래에 버튼 높이만큼 빈자리를 둔다" 를 적고, 화면을 고쳤습니다. 순서는 역시 **문서 먼저**입니다.

마음에 안 들면 웹과 똑같습니다.

oo가 마음에 안 들어. 설계 문서부터 고치고, 화면도 고쳐줘.

18.6 1부 끝

여기서 1부가 끝납니다. Claude가 상태를 정리해 줍니다.

"여기까지가 1부입니다. 웹 두 화면, 앱 한 화면이 생겼어요. 폰에서 돌아가고 있습니다."



그림 18-3 1부가 끝났을 때. 웹에는 「오늘 클래스」(네 바퀴)와 「주간」, 폰에는 간판이 들어간 첫 화면과 「시간표」. 둘은 아직 각자 가짜 데이터를 들고 있어 서로를 모른다

그림 오른쪽 아래를 보세요. ****웹과 앱이 아직 서로 모릅니다.**** 폰에서 신청해도 웹 명단에 안 뜹니다. 둘이 각자 가짜 데이터를 한 벌씩 들고 있기 때문입니다. 이 둘을 잇는 게 다음 장의 갈림길 A입니다.

18.7 정리

- 앱 화면도 **문서대로 한 덩어리씩** — 탭 → 위쪽 → 목록 → 신청 버튼 → 간판 → 점검
- 로그인은 3주차. 오늘 앱은 **"나는 김하나"** 로 고정한다
- 규칙에 막히는 버튼은 **누르기 전부터** 흐리게
- 안드로이드 세 가지 — **안전 영역**만큼 여백 · 알림은 **화면 안에** · 사진은 **100%**
- 간판은 16장에서 그려 둔 **도면대로**. 시안을 다시 열지 않고, 묻지도 않는다
- 마지막 확인은 **실제 폰으로**

이 장에서 나온 말

말	쉬운 뜻
탭(tab) · 탭바	앱 맨 아래 화면을 바꾸는 버튼 줄. 둘러보기 · 시간표 · 내 신청 · 공지
안전 영역	시계 줄과 폰 버튼 줄을 뺀, 앱이 마음 놓고 그려도 되는 부분
팝업(Alert)	화면 가운데 툭 튀어나오는 작은 창. 오늘은 안 쓴다
어두운 막	사진 위에 덮는 반투명한 검은 층. 흰 글씨를 읽히게 한다
떠 있는 버튼	화면을 내려도 같은 자리에 붙어 따라오는 버튼

18.8 안 될 때

증상	이렇게 치세요
탭이 폰 아래 버튼과 겹친다	탭이 폰 버튼이랑 겹쳐
폰에서 글씨가 안 읽힌다	폰에서 글씨가 안 보여
간판 사진이 화면 밖으로 빠져나갔다	간판 사진이 화면을 벗어나
간판 위 글씨가 사진에 묻힌다	간판 글씨가 안 읽혀 — 어두운 막을 덮어 줍니다
신청을 눌렀는데 아무 일이 없다 (크롬에서)	신청 눌렀는데 아무것도 안 떠
앱을 다시 켜더니 신청이 없어졌다	정상입니다. 저장할 곳을 아직 안 만들었습니다
로그인은 언제 만드냐고 묻고 싶다	다음 주입니다. 인터넷에 올리기 직전에 붙입니다

19장 — 갈림길, 저장되게 만들기 또는 화면 더

초고 v0.1 (2026-09-18) · 2주차 16단계 갈림길 · B 경로 16~17단계

19.1 여기서 길이 갈린다

1부가 끝났습니다. 웹 두 화면, 앱 한 화면. 폰에서 돌아갑니다. 그런데 18장 끝에서 봤듯이 **웹과 앱이 아직 서로 모릅니다**.

2주차의 원래 목표는 여기서 한 걸음 더 가는 것입니다. 둘을 **같은 데이터에 연결**해서, 컴퓨터에서 누르면 폰에 뜨고 폰에서 누르면 컴퓨터에 뜨게 하는 것. 제작prompt 첫머리의 약속 — "**컴퓨터에서 누르면 폰 화면이 따라 바뀝니다**" 가 바로 이것입니다.

그런데 수업 시간은 두 시간입니다. 앞에서 누가 어디서 얼마나 막혔느냐에 따라 **남은 시간이 사람마다 다릅니다**. 그래서 여기서 길이 둘로 갈립니다.



그림 19-1 15단계 뒤의 갈림길. 남은 시간이 50분 이상이면 A 저장되게 만들기(16~19단계 · 가입 → 표 → 웹 연결 → 앱 연결), 50분 미만이면 B 화면 더 만들기(16~17단계). 어느 길이든 20단계 설계 문서 고쳐 쓰기로 모인다

| 길 | 무엇 | 이 책 | |---|---|---| | ****A**** 저장되게 만들기 (기본) | 데이터 창고를 만들어 웹과 앱을 잇는다 | 20~22장 | | ****B**** 화면 더 만들기 | PRD 순서의 다음 화면을 만든다. 참고는 다음 주 첫 순서 | 이 장 19.4절 |

둘 다 하지 않습니다. 한 길로만 갑니다.

19.2 16-① — Claude가 시계를 보고 정한다

먼저 이 상자를 붙여넣습니다.

여기까지 됐어. 다음은 뭐 할 차례야?

이 질문에 **Claude가** 답합니다. 우리에게 고르라고 하지 않습니다. 시계를 보고 정합니다.

남은 시간	간다
50분 이상	A 저장되게 만들기
50분 미만	B 화면 더 만들기

Claude가 「저장되게 만들기」라고 하면 제작prompt의 **A** 쪽으로, 「화면 더 만들기」라고 하면 **B** 쪽으로 갑니다.

왜 50분인가

A의 네 단계에 걸리는 시간을 더하면 이렇습니다.

단계	시간
16 참고 가입	15분
17 답을 칸 만들기	12분
18 웹 연결	10분
19 앱 연결	15분
합	52분

💡 중간에 끊기는 게 제일 나쁘다

40분 남았는데 A로 가면 어떻게 될까요? 가입하고, 칸을 만들고, 웹을 붙이다가 시간이 끝납니다. 웹은 참고에 반쯤 붙어 있고, 앱은 아직 가짜 데이터를 보고 있습니다. 집에 가서 열면 **둘 다 이상하게** 돌아갑니다.

손에는 아무것도 안 남고, "뭔가 복잡한 걸 하다 말았다" 는 기억만 남습니다.

그래서 욕심내지 않습니다. 40분이면 **화면을 하나 더** 만드는 쪽이 확실하게 남습니다. 12장의 "어설프게 걸치는 것보다 끝난 게 낫다" 가 여기서 가장 크게 쓰입니다.

Claude가 하는 말

A로 갈 때는 갈림길을 티 내지 않습니다. 그냥 다음 단계처럼 말합니다.

"이제 웹이랑 앱이 같은 데이터를 보게 만들 거예요. 저장되게 만들기로 갈게요.
제작prompt의 A 쪽, 16-① 상자를 붙여넣어 주세요."

B로 갈 때는 "시간이 없어서 못 한다" 고 말하지 않습니다. 다음 주 첫 순서라고 말합니다.

"여기까지 잘 오셨습니다. 저장되게 만드는 건 다음 주 첫 순서로 잡을게요. 오늘 남은 시간엔
화면을 더 만들겠습니다. 제작prompt의 B 쪽, 16-① 상자를 붙여넣어 주세요."

B로 가더라도 20~22장은 읽어 두세요. 다음 주 첫 시간에 그대로 하게 됩니다.

19.3 A로 갔는데 시간이 모자라면

A를 시작했는데 생각보다 오래 걸릴 수 있습니다. 가입 메일이 늦게 온다든지요. 그럴 때도 끊기는 자리를 정해 둡니다.

어디까지 갔나	이렇게
가입까지 (16단계)	칸까지는 만들어 둔다. 연결은 다음 주
칸까지 (17단계)	웹만 연결한다. 새로고침해도 안 사라지는 것까지 본다
웹까지 (18단계)	앱 연결은 다음 주. 웹만으로도 "저장된다" 는 건 보인다

어디서 멈추든 20단계(설계 문서 고쳐 쓰기)는 합니다. 5분이면 됩니다. 그게 다음 주의 출발점입니다(23장).

19.4 B · 화면 더 만들기

A로 가신 분은 이 절을 건너뛰고 20장으로 가세요.

16-① · 16-② 다음 화면

설계 문서 순서에서 다음 거 만들자.

문서에 빠진 게 있으면 문서부터 채우고, 화면은 큰 덩어리부터.

다음 기능 하나 붙여줘. 화면으로 보여줘.

새 화면을 고르지 않습니다. PRD 7장 「만드는 순서」의 **다음 번호**를 그대로 따릅니다. 공방 예시라면 4번 「출석 체크 붙이기」, 그다음 6번 「알림판 + 공지」입니다. 지난주에 순서를 매겨 둔 이유가 이것입니다. 지금 고르면 또 고민이 시작됩니다.

만드는 방식은 12~15단계와 똑같습니다.

1. PRD에 그 화면의 **동작 표**가 없으면 **먼저** 채운다 — 13장처럼, 질문은 두 개까지
2. 큰 덩어리부터 한 상자씩
3. 크롬(또는 폰)에서 보여준다

두 번째 상자(16-②)를 **몇 번 반복**하게 됩니다. 조금씩 쌓이는 게 보입니다. 마음에 안 들면 여기서도 — 설계 문서부터 고치고, 화면도 고쳐줘.

몇 개까지 하나

미리 "세 개 하겠다" 고 정하지 않습니다. 시간을 보고 **하나씩** 늘립니다.

남은 시간	이렇게
25분 이상	다음 번호 하나 + 그다음 하나
15분쯤	다음 번호 하나만
10분 미만	시작하지 않는다. 바로 17단계로

짜이 있는 화면은 같이 만든다

웹 「알림판」(사장님이 공지를 쓰는 곳)과 앱 「공지」(손님이 읽는 곳)처럼 **짜이 있는 화면**은 붙여서 만듭니다. 따로 만들면 왜 두 번 만드는지 모릅니다.

다만 **아직 연결은 안 됩니다.** Claude가 미리 말해 둡니다.

"웹에서 쓰면 앱에 뜨게 하고 싶은데, 그건 다음 주에 창고를 만들고 나서 됩니다. 오늘은 양쪽 화면만 만들어 둘게요."

17-① 오늘 만든 것 훑어보기

오늘 만든 거 쪽 보여줘. 웹부터 보고 폰도 볼게.

오늘 만든 게 여러 화면에 흩어져 있습니다. 한 번 **쪽** 봅니다. 웹은 만든 순서대로 하나씩, 앱은 **폰으로** 봅니다.

그리고 Claude가 한 가지를 **숨기지 않고** 먼저 말합니다.

"하나만 미리 말씀드릴게요. 지금 만드신 건 **새로고침하면 사라집니다**. 고장 난 게 아니에요. **저장할 창고를 아직 안 만들었거든요**. 그게 다음 주 첫 순서입니다.

화면은 다 살아 있습니다. 집에서 다시 열어보셔도 화면은 그대로예요."

이걸 숨기면 집에 가서 열어 보고 **"다 없어졌다"** 고 놀랍니다. 그래서 미리 말합니다. 그리고 오늘 만든 것을 못 박습니다.

"오늘 만드신 걸 정리하면요. · 웹 화면 ○장 — 내 컴퓨터에서 돌아갑니다 · 앱 화면 ○장 — 내 폰에서 돌아갑니다 지난주엔 종이에 그린 그림뿐이었는데, **지금 폰에서 돌아가고 있습니다.**"

B는 여기까지입니다. 23장(설계 문서 고쳐 쓰기)으로 갑니다.

19.5 정리

- 15단계 뒤 **갈림길**. 남은 시간 **50분 이상** → A 저장되게 만들기, **미만** → B 화면 더
- 고르는 건 **Claude**가 시계를 보고. 우리는 **다음은 뭐 할 차례야?** 만 묻는다
- 50분인 이유 — A는 52분. **중간에 끊기는 게 제일 나쁘다**
- B는 PRD 7장 **다음 번호**를 한 덩어리씩. 새로고침하면 사라진다는 걸 숨기지 않는다
- 어느 길이든 **20단계(23장)** 는 한다

이 장에서 나온 말

말	쉬운 뜻
A 경로	데이터 창고를 만들어 웹과 앱을 잇는 길. 2주차의 원래 목표
B 경로	시간이 빠듯할 때 화면을 더 만드는 길. 창고는 다음 주 첫 순서
짜 화면	웹 알림판과 앱 공지처럼, 한쪽에서 쓰고 다른 쪽에서 읽는 화면 한 쌍

19.6 안 될 때

증상	이렇게 치세요
A인지 B인지 헷갈린다	다음은 뭐 할 차례야? 를 한 번 더
B인데 새 화면을 고르라고 한다	설계 문서 7장 순서대로 다음 거 하자
화면 하나를 한 번에 다 만들어 버렸다	방금 건 취소하고 큰 덩어리부터 다시 하자
짜 화면인데 연결이 안 된다	정상입니다. 연결은 창고를 만든 뒤(다음 주)
시간이 다 됐다	여기까지 하고 설계 문서 정리하자 — 23장으로

20장 — GitHub 가입, 열쇠부터 만든다

초고 v0.1 (2026-09-22) · 2주차 A 경로 16단계 앞 · Supabase(21장)와 Vercel(24장)에 들어갈 계정

20.1 왜 GitHub이 먼저인가

다음 장에서 데이터 창고(Supabase)를 빌립시다. 다음 주에는 웹을 인터넷에 올리는 서비스(Vercel)에도 가입합니다. 두 곳 다 **GitHub 계정으로** 들어갑니다. 그래서 GitHub 계정이 **먼저** 있어야 합니다.

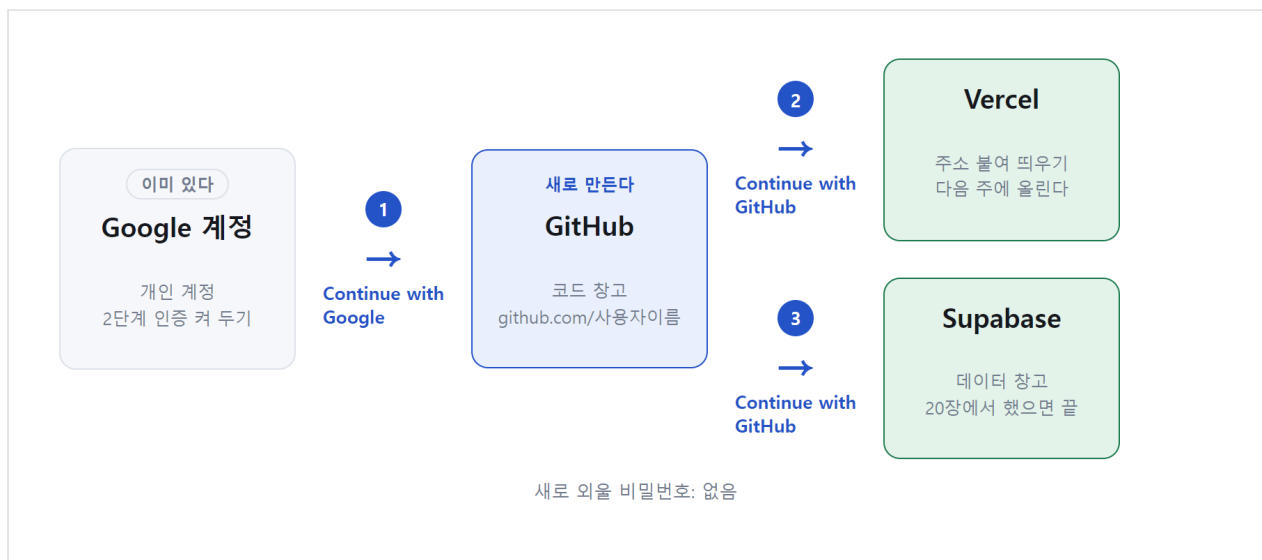


그림 20-1 가입 순서. ① Google 계정으로 GitHub에 가입한다 ② 그 GitHub 계정으로 Vercel에 가입한다(24장) ③ Supabase도 같은 GitHub 계정으로 들어간다(21장). 새로 외울 비밀번호가 없다

| 순서 | 어디 | 무엇으로 들어가나 | 언제 | |---|---|---|---| | 1 | **GitHub** | Google 계정 | 지금 (이 장) | | 2 | Supabase | GitHub 계정 | 바로 다음 (21장) | | 3 | Vercel | GitHub 계정 | 다음 주 전에 (24장) |

GitHub 계정이 이미 있으면 이 장은 건너뛰고 21장으로 갑니다.

📖 GitHub(깃허브)

만든 코드를 인터넷에 보관해 두는 곳. 코드의 클라우드 창고입니다. 다음 주에 웹을 인터넷에 올릴 때, 올리는 서비스(Vercel)가 여기서 코드를 가져갑니다.

오늘은 로그인 열쇠로만 씁니다. GitHub 계정 하나로 Supabase와 Vercel에 들어갑니다.

가입할 곳이 셋인데, 아이디와 비밀번호는 하나도 새로 만들지 않습니다. 이미 가진 Google 계정 하나에서 시작해 열쇠를 사슬처럼 잇습니다.

📖 소셜 로그인 — 「○○로 계속하기」

가입 화면에 있는 **Continue with Google, Continue with GitHub** 같은 버튼. 새 아이디를 만드는 대신, 이미 가진 계정으로 나를 확인시켜 주는 방식입니다.

건물 출입증을 새로 만들지 않고 주민등록증을 보여 주고 들어가는 것과 같습니다. 비밀번호는 Google이나 GitHub이 확인하고, 새 서비스는 "이 사람 맞다" 는 답만 받습니다.

20.2 Google 계정으로 가입하기

Chrome을 열고 `github.com` 으로 갑니다. 영어 화면입니다. 버튼 글씨를 그대로 적어 둡니다.

	할 일	됐으면
1	오른쪽 위 Sign up 을 누른다	가입 화면이 뜬다
2	맨 위 Continue with Google 을 누른다. 아래 이메일·비밀번호 칸은 쓰지 않는다	Google 계정 고르는 창이 뜬다
3	내 Google 계정을 고른다. 여러 개면 앞으로 계속 쓸 계정 하나	GitHub으로 돌아온다
4	Username (사용자 이름)을 정한다. 메일 주소로 만든 이름이 미리 들어와 있으면 고쳐 쓴다	초록 체크가 뜬다
5	나라(Country/Region)가 Korea, South 인지 보고 Create account 를 누른다	GitHub 첫 화면이 뜬다

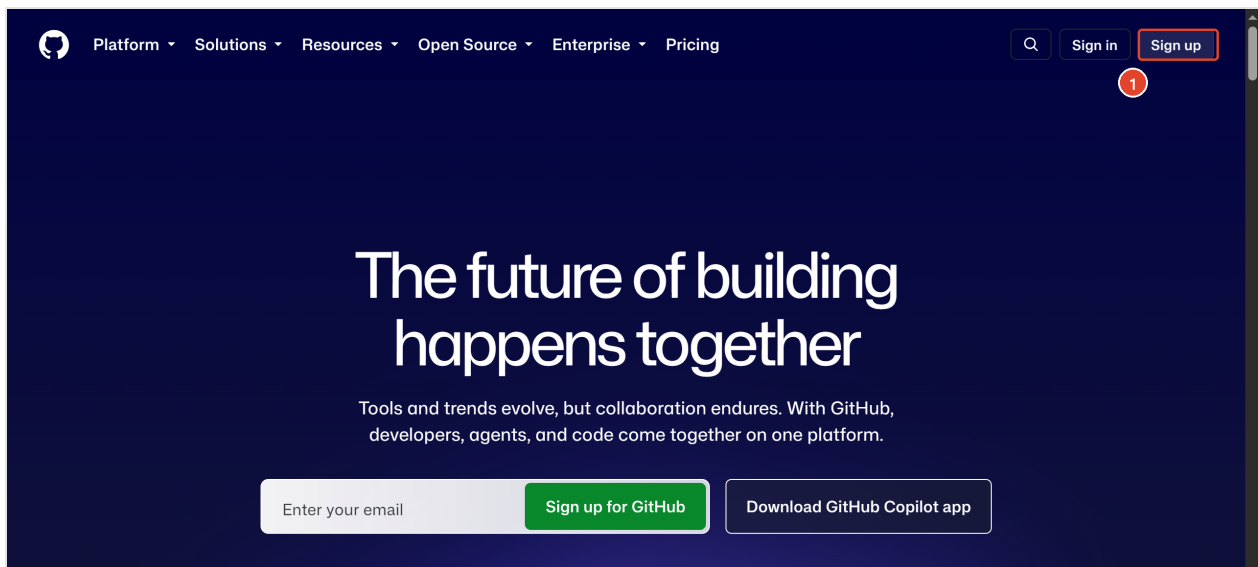


그림 20-2 github.com 첫 화면. ① 오른쪽 위 Sign up. 가운데 이메일 칸에는 적지 않는다

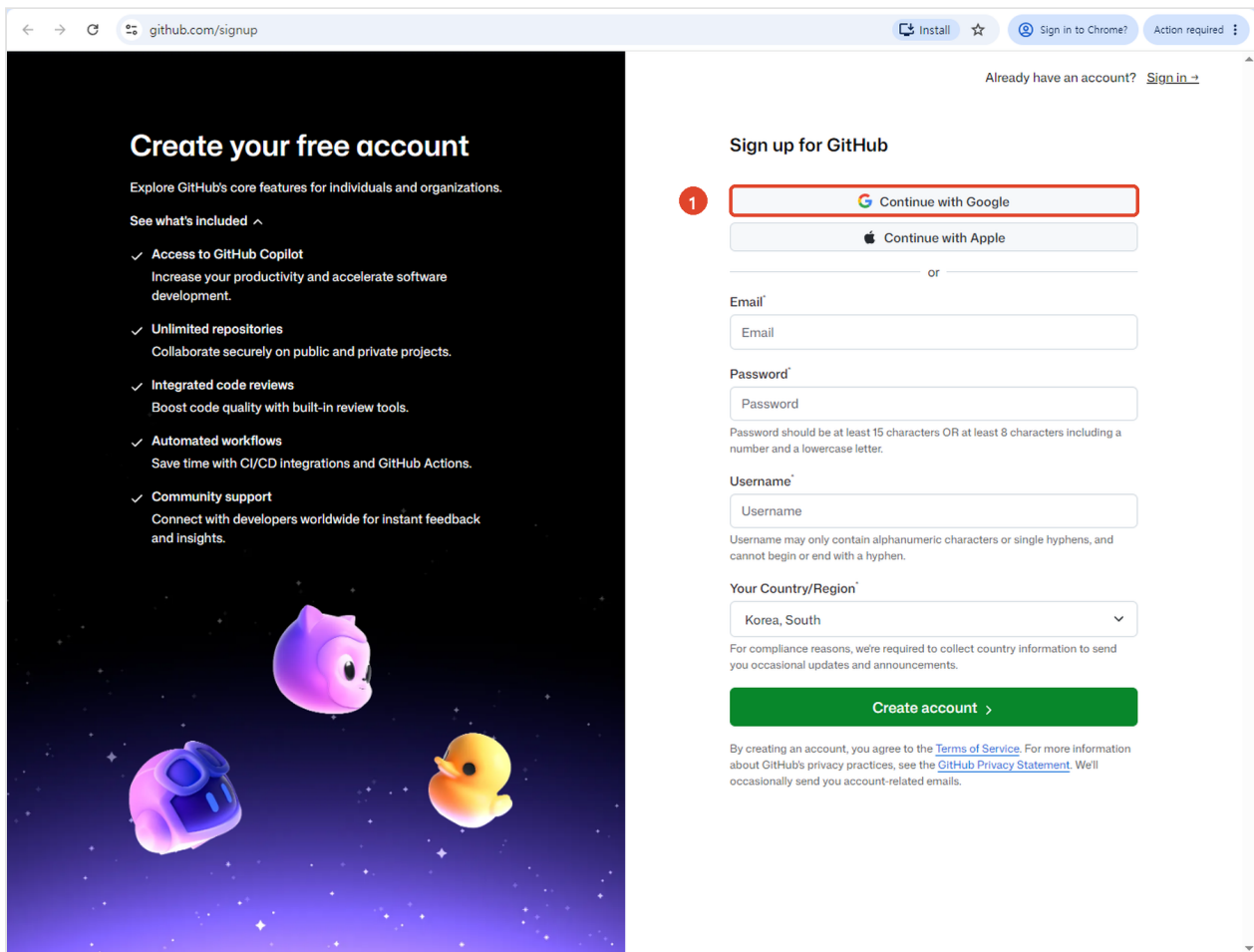


그림 20-3 GitHub 가입 화면. ① 맨 위 Continue with Google — 여기를 누른다. 그 아래 Email · Password · Username 칸은 쓰지 않는다

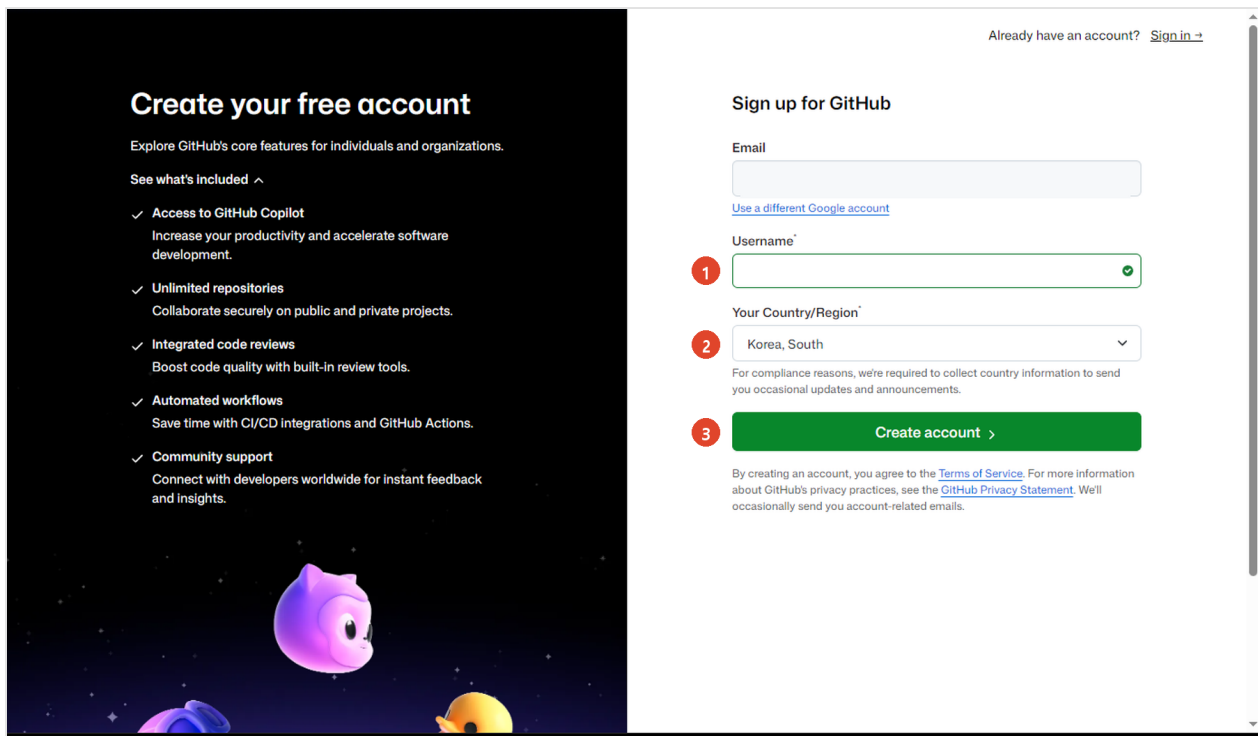


그림 20-4 Google 계정을 고르고 돌아온 화면. Email 칸에는 고른 Google 메일이 들어와 있다. ① Username — 초록 체크가 뜨면 쓸 수 있는 이름 ② Country/Region — Korea, South ③ Create account. 메일 주소와 이름은 가렸다

2번 — 왜 이메일 칸을 안 쓰나

이메일 칸으로 가입해도 계정은 만들어집니다. 그런데 그러면 **비밀번호를 하나 더** 정하고 외워야 하고, 확인 메일을 기다려야 합니다. Google로 들어가면 둘 다 없습니다. 로그인할 때도 **Continue with Google** 한 번이면 됩니다.

4번 — 사용자 이름은 주소가 된다

Username은 그냥 별명이 아닙니다. **내 창고의 주소**가 됩니다.

`https://github.com/사용자이름`

이렇게	이러지 않게
영어 소문자·숫자·불임표(-)	한글, 띄어쓰기 — 쓸 수 없다
가게 이름이나 내 이름 영어로 (hana-pottery)	남에게 보이기 민망한 장난 이름
짧고 읽기 쉬운 것	뜻 없는 숫자 나열 (user38271)

GitHub이 메일 주소를 따서 이름을 미리 넣어 두기도 합니다. 그대로 두면 메일 주소가 주소창에 드러나니, **가게 이름으로 고쳐 씁니다**. 이미 누가 쓰는 이름이면 빨간 글씨가 뜹니다. 뒤에 숫자나 -

kr 을 붙여 봅니다. 나중에 바꿀 수는 있지만 **주소가 같이 바뀝니다**. 처음에 정해 두는 게 편합니다.

5번 뒤에 — 설문이 나오면

만든 직후에 "무엇에 쓰세요?" 같은 설문이 나올 수 있습니다. 답해도 되고, 아래쪽 **Skip** 으로 넘겨도 됩니다. 요금제를 고르라는 화면이 나오면 **Free** 입니다. 우리가 쓰는 기능은 전부 무료입니다.

20.3 창고를 가진다는 것

💡 공개 창고는 누구나 본다

GitHub 창고에는 **공개(Public)** 와 **비공개(Private)** 가 있습니다. 공개 창고는 인터넷의 누구나 안의 파일을 열어 볼 수 있습니다.

그래서 **비밀번호 · 계좌번호 · 열쇠** 는 창고에 넣으면 안 됩니다. 한 번 올라가면 지워도 **변경 이력에 남습니다**. 다음 장에서 데이터 창고의 주소와 열쇠를 받아 오면, Claude는 그걸 코드가 아니라 **따로 떼어 둔 쪽지 파일**에 적습니다. 이유가 여기 있습니다. 다음 주에 창고에 올릴 때도 그 쪽지는 **빼고** 올립니다.

오늘은 계정만 만듭니다. 창고는 다음 주에 Claude와 같이 만듭니다.

💡 열쇠를 하나로 묶으면 맨 위 열쇠가 제일 중요해진다

이제 GitHub · Supabase · Vercel 세 곳을 **Google 열쇠 하나로** 묶습니다. 기억할 게 한 별이라 편합니다. 그 대신 맨 위 열쇠를 잃으면 **아래가 전부 같이 잠깁니다**. Google 계정을 잃으면 GitHub에 못 들어가고, GitHub에 못 들어가면 나머지 둘에도 못 들어갑니다.

그래서 맨 위 **Google 계정에 2단계 인증**(비밀번호 + 폰 확인)이 켜져 있는지 한 번 봐 둡니다. 그리고 **회사 계정이 아닌 개인 계정**으로 묶습니다. 회사를 옮기면 회사 계정은 사라지고, 그 아래 가게 사이트까지 같이 잃습니다.

GitHub 계정이 생겼습니다. 이제 이 계정으로 **데이터 창고**를 빌리러 갑니다(21장).

20.4 정리

- Supabase(21장)도 Vercel(24장)도 **GitHub 계정으로** 들어간다. 그래서 GitHub이 먼저

- GitHub은 **Google 계정으로** 가입한다. 비밀번호를 새로 만들지 않는다
- **Username**은 주소가 된다. 영어 소문자로, 가게나 내 이름
- 공개 창고는 누구나 본다. 비밀번호·열쇠는 올리지 않는다
- 맨 위 열쇠(Google)에 **2단계 인증**. 개인 계정으로

이 장에서 나온 말

말	쉬운 뜻
GitHub	코드를 인터넷에 보관하는 곳. 오늘은 로그인 열쇠로 쓴다
소셜 로그인	Continue with Google 처럼, 이미 가진 계정으로 나를 확인시키는 가입·로그인 방식
Username	GitHub에서 쓰는 내 이름. <code>github.com/이름</code> 주소가 된다
공개 · 비공개 창고	누구나 볼 수 있는 창고와 나만 보는 창고
2단계 인증	비밀번호에 폰 확인을 하나 더 거는 것. 맨 위 열쇠에 건다

20.5 안 될 때

증상	이렇게
GitHub 가입 화면에 Continue with Google 이 안 보인다	오른쪽 위 Sign in → Continue with Google 로 들어가도 됩니다. 계정이 없으면 만드는 화면으로 이어집니다
Username 화면 없이 바로 GitHub 첫 화면이 났다	그 Google 계정으로 이미 가입돼 있던 겁니다. 새로 만들 필요 없습니다
사용자 이름이 계속 빨갛다	이미 누가 쓰는 이름입니다. 뒤에 숫자나 <code>-kr</code> 을 붙여 보세요
퍼즐이나 그림 맞추기가 나온다	사람인지 확인하는 겁니다. 풀면 넘어갑니다
메일로 숫자(코드)를 넣으라고 한다	Google 메일함을 보세요. 없으면 스팸함. 그 숫자를 넣습니다
회사 Google 계정으로 가입해 버렸다	지금 바로 말하세요. 계정이 적을 때 바꾸는 게 제일 쉽습니다
영어라 뭘 눌러야 할지 모르겠다	Claude에게 화면을 캡처해서 붙이고 <code>어디를 눌러야 돼?</code>

21장 — Supabase, 데이터를 둘 창고 만들기

초고 v0.1 (2026-09-18) · 2주차 A 경로 16단계

21.1 웹과 앱이 서로 모르는 이유

지금 웹과 앱은 각자 **가짜 데이터 파일**을 한 벌씩 들고 있습니다(15장·18장).

- 웹 폴더 안에 한 벌 — 사장님 화면이 읽는다
- 앱 폴더 안에 한 벌 — 손님 폰이 읽는다

손님이 폰에서 신청을 누르면 **폰 안의 파일**에만 적힙니다. 사장님 웹은 자기 파일을 읽으니 그 신청을 모릅니다. 그리고 둘 다 **새로고침하면 처음으로 돌아갑니다**. 칠판에 적은 것이니까요(16장).

둘이 같은 걸 보려면 방법은 하나입니다. 둘 다 들여다볼 수 있는 **공책**을 하나 두고, 거기에 적고 거기서 읽는 것.



그림 21-1 지금과 연결한 뒤. 지금은 웹과 앱이 각자 가짜 데이터 파일을 들고 있어 서로 모른다. 연결하면 둘 다 인터넷에 있는 창고 하나(Supabase)를 읽고 쓴다. 폰에서 신청하면 창고에 적히고, 웹은 그 창고를 읽어 명단에 보여준다

> 📖 ****데이터베이스(database) — 데이터 창고**** > > 데이터를 ****정리해서 오래 보관해 두는 곳**** 가게 장부나 도서관 서가와 같습니다. 칸이 > 정해진 표에 한 줄씩 적어 두고, 필요할 때 꺼내 봅니다. 화면을 꺾다 쳐도, 폰을 바꿔도 > 장부는 그대로 남아 있습니다. > > 이 책에서는 쉽게 ****창고****라고 부릅니다. 9장에서 본 데이터(화면 뒤에서 기억해 두는 > 내용 전부)가 여기에 담깁니다.

21.2 Supabase는 무엇인가

창고를 내 컴퓨터에 둘 수도 있습니다. 그런데 그러면 **폰이 내 컴퓨터를 찾아와야** 합니다. 17장에서 겪은 와이파이 문제가 또 생깁니다. 그리고 내 컴퓨터를 끄면 창고도 닫힙니다.

그래서 창고를 **인터넷에** 하나 빌립니다. 인터넷에 있으면 폰이든 컴퓨터든 **그냥 붙습니다**. 그 창고를 빌려주는 서비스가 Supabase입니다.

Supabase(수파베이스)

인터넷에 **데이터 창고를 빌려주는 서비스**. 가입하면 내 창고가 하나 생기고, 웹과 앱이 주소와 열쇠만 있으면 거기에 붙어서 읽고 씁니다. 무료로 시작할 수 있습니다.

창고를 직접 짓고 관리하는 대신 **창고 임대 회사**를 쓰는 것과 같습니다. 청소·경비·전기는 임대 회사가 맡고, 우리는 물건만 넣고 꺼냅니다.

수업에서 Claude는 이렇게 설명합니다.

"웹이랑 앱이 **같은 데이터를 보려면** 둘 다 들여다볼 공용 창고가 하나 있어야 합니다. 그 창고를 인터넷에 하나 빌리는 거예요. **공짜입니다**."

왜 가입까지 시키나

가입이 번거로워 보이지만 오히려 **쉬운 길**입니다. 내 컴퓨터에 창고를 두면 와이파이·주소·방화벽 같은 문제가 줄줄이 따라옵니다. 인터넷 창고는 가입 10분이면 끝납니다.

들어갈 때는 20장에서 만든 **GitHub 계정**을 씁니다. 새 아이디와 비밀번호를 만들지 않습니다. GitHub 계정이 아직 없으면 **20장부터** 하고 오세요.

오늘 쓰지 않는 기능

Supabase에는 기능이 많습니다. 오늘은 **창고에 표를 만들고, 읽고, 쓰는 것만** 씁니다.

안 쓴다	왜
실시간 알림(구독)	실패하는 곳이 많다. 화면에 들어올 때 다시 읽는 것으로 충분하다
창고 안의 계산 기능	오늘 수준엔 과하다
로그인 기능	로그인이 3주차다
별도 서버	웹도 앱도 창고에 직접 붙는다

21.3 가입하기 — 한 번에 하나씩

☐ 16-① 상자부터 붙여넣습니다.

지금은 웹이랑 앱이 각자 가짜 데이터를 들고 있는 것 같아.
둘이 같은 데이터를 보게 하고 싶어. 어떻게 하면 되는지 알려줘.

Claude가 21.1~21.2절의 이야기를 짧게 하고, 가입하자고 합니다. ☐ 16-② 상자입니다.

가입할게. 뭘 눌러야 하는지 하나씩 알려줘. 한 번에 하나씩만.

Claude는 절차를 **한꺼번에 늘어놓지 않습니다**. 한 줄 말하고, 됐다는 답을 받고, 다음 줄로 갑니다. 영어 화면이라 버튼 글씨를 그대로 불러 줍니다. 10분쯤 걸립니다.

	Claude가 말하는 것	우리 답
1	<code>supabase.com</code> 을 열고 오른쪽 위 Start your project 를 눌러 주세요	열었어
2	Continue with GitHub 를 눌러 주세요	눌렀어
3	GitHub 로그인 화면이 뜨면 Continue with Google 로 로그인해 주세요(20장). 이미 로그인돼 있으면 이 줄은 건너뛰니다	했어
4	Authorize supabase 가 나오면 눌러 주세요. Supabase가 GitHub 계정으로 들어가고 있는지 묻는 겁니다	됐어
5	New project 를 누르고, 이름은 아무거나 넣어 주세요	했어
6	Database password 칸 아래 Generate a password 를 누르고, 옆의 Copy 로 복사해 꼭 어디 적어 두세요	적었어
7	★ Region 을 열고 목록을 아래로 내려 Northeast Asia (Seoul) 을 꼭 골라 주세요. 맨 위 Asia-Pacific 이 아닙니다	골랐어
—	그 아래 Security 칸은 그대로 둡니다	
8	맨 아래 Create new project 를 누르면 2분쯤 걸립니다. 기다려 주세요	됐어

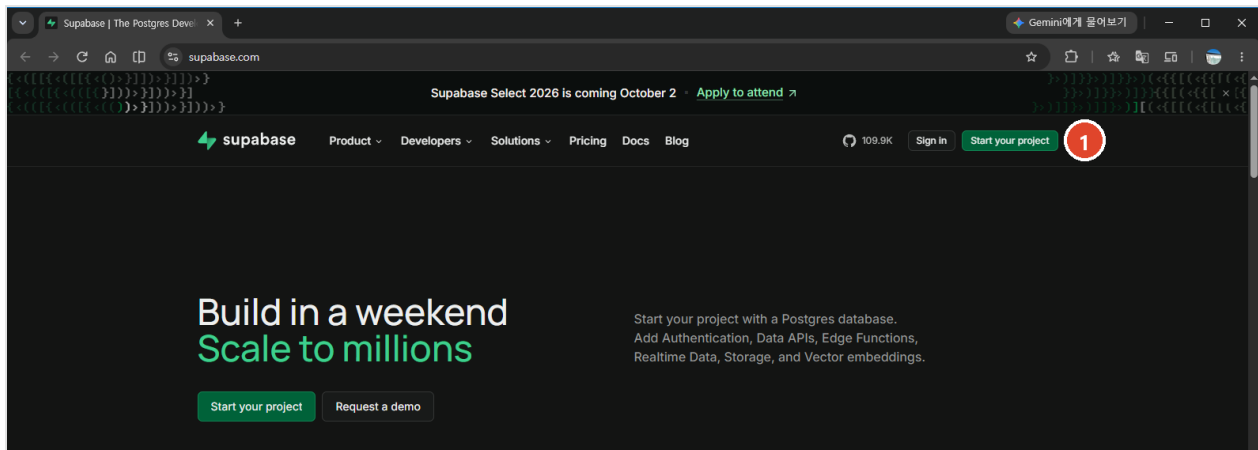


그림 21-2 supabase.com 첫 화면. ① 오른쪽 위 Start your project

Get started

Create a new account

1



Continue with GitHub

LAST USED



Continue with ChatGPT



Continue with SSO

or

Email

2

you@example.com

Password

••••••••



3

Sign up

Have an account? [Sign in](#)

By continuing, you agree to Supabase's [Terms of Service](#) and [Privacy Policy](#), and to receive periodic emails with updates.

그림 21-3 가입 화면. ① Continue with GitHub — 여기를 누른다. 아래 이메일 칸은 쓰지 않는다

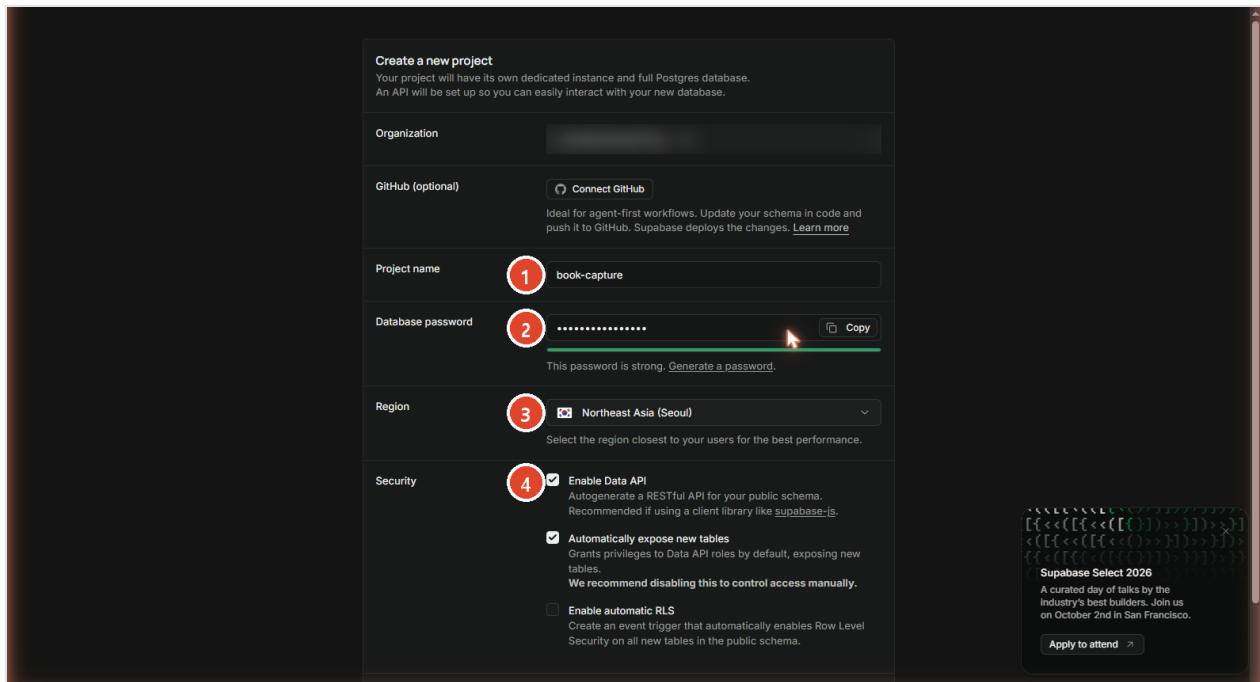


그림 21-4 새 프로젝트 만들기. ① Project name ② Database password — Generate a password 로 만들어 적어 둔다 ③ Region — 목록 아래쪽에서 Northeast Asia (Seoul) ④ Security — 기본값 그대로. 맨 아래 Create new project 를 누른다

2~4번 — 왜 GitHub으로 들어가나

이메일로 따로 가입해도 됩니다. 그런데 그러면 다음 주에 GitHub과 Vercel에 또 가입하고, 아이디와 비밀번호가 **세 벌**이 됩니다. GitHub 계정 하나로 세 곳에 들어가면 기억할 게 **한 벌**입니다. 3주차에 Vercel이 GitHub에서 코드를 가져가고 Supabase 주소를 읽을 때도, 같은 계정끼리라 잊기가 쉽습니다.

4번의 **Authorize** 는 처음 한 번만 나옵니다. "Supabase가 내 GitHub 이름과 이메일을 봐도 되나요?" 라는 확인입니다. 코드를 가져가는 권한이 아닙니다.

6번 — 비밀번호는 적어 두기만

여기서 정하는 비밀번호는 **참고 관리자용**입니다. 오늘은 쓸 일이 없습니다. 그래도 나중에 필요할 수 있으니 **어딘가 적어 둡니다**. 까먹어도 괜찮습니다. 설정 화면에서 다시 정할 수 있습니다.

7번 — 지역은 되돌릴 수 없다

⚠ 지역은 만든 뒤에 못 바꿉니다

참고가 어느 나라의 컴퓨터에 놓일지 고르는 칸입니다. 한 번 만들면 **바꿀 수 없습니다**. 바꾸려면 창고를 새로 만들어야 합니다.

목록 맨 위에는 **Asia-Pacific** 처럼 넓은 지역이 먼저 나오고 「추천」 딱지가 붙어 있습니다. 그걸 고르면 서울이 아닌 곳에 놓일 수 있습니다. 목록을 내려서 **Northeast Asia (Seoul)** 을 직접 고릅니다.

멀리 잡으면(미국 등) 폰에서 신청을 누르고 웹에 뜨기까지 **눈에 띄게 느려집니다**. 22장의 "누르면 바로 뜬다" 가 살지 않습니다. 그래서 여기만은 Claude가 **확인하고** 넘어갑니다 — **Northeast Asia (Seoul)**.

8번 뒤에 — 창고는 일주일 안 쓰면 잠든다

만들기를 누르고 기다리는 동안 Claude가 한 가지를 말해 둡니다.

"이 창고는 **일주일 넘게 안 들어가면 잠자기로 들어갑니다**. 다음 주 수업 오시기 전에 한 번 열어보시면 됩니다."

무료 창고는 **7일 동안 아무도 안 쓰면 스스로 멈춥니다**. 멈춘 창고는 깨우는 데 몇 분이 걸립니다. 다음 주 수업에 와서 "어제까지 됐는데 안 돼요" 가 되지 않게, 수업 전에 supabase.com 에 한 번 들어가 보면 됩니다.

💡 되돌릴 수 없는 것과 조용히 멈추는 것

가입 절차 중 대부분은 틀려도 나중에 고칠 수 있습니다. 이름은 바꾸면 되고, 비밀번호는 다시 정하면 됩니다. 그런데 두 가지는 다릅니다.

지역은 되돌릴 수 없습니다. **잠자기**는 아무 경고 없이 조용히 일어납니다. 강사가 준비하면서 따로 적어 둔 두 가지이고, 그래서 PRD 10장 「만들 때 지킬 것」 에도 들어가 있습니다. **틀리면 비싼 것만 확인하고, 나머지는 빨리 지나갑니다**.

21.4 주소와 열쇠 받아 오기

창고가 만들어졌습니다. 이제 웹과 앱이 이 창고를 **찾아오고 들어올 수 있게** 두 가지를 받아 옵니다.
☐ 16-③ 상자입니다.

가입 끝났어. 이제 뭘 복사해서 주면 돼?

"왼쪽 아래 톱니바퀴(Project Settings) → API Keys 를 눌러 주세요. Publishable key 옆 복사 버튼을 눌러 저한테 붙여 주세요. 맨 위 초록 **Connect** 를 누르면 나오는 `https://....supabase.co` 주소도 같이요."

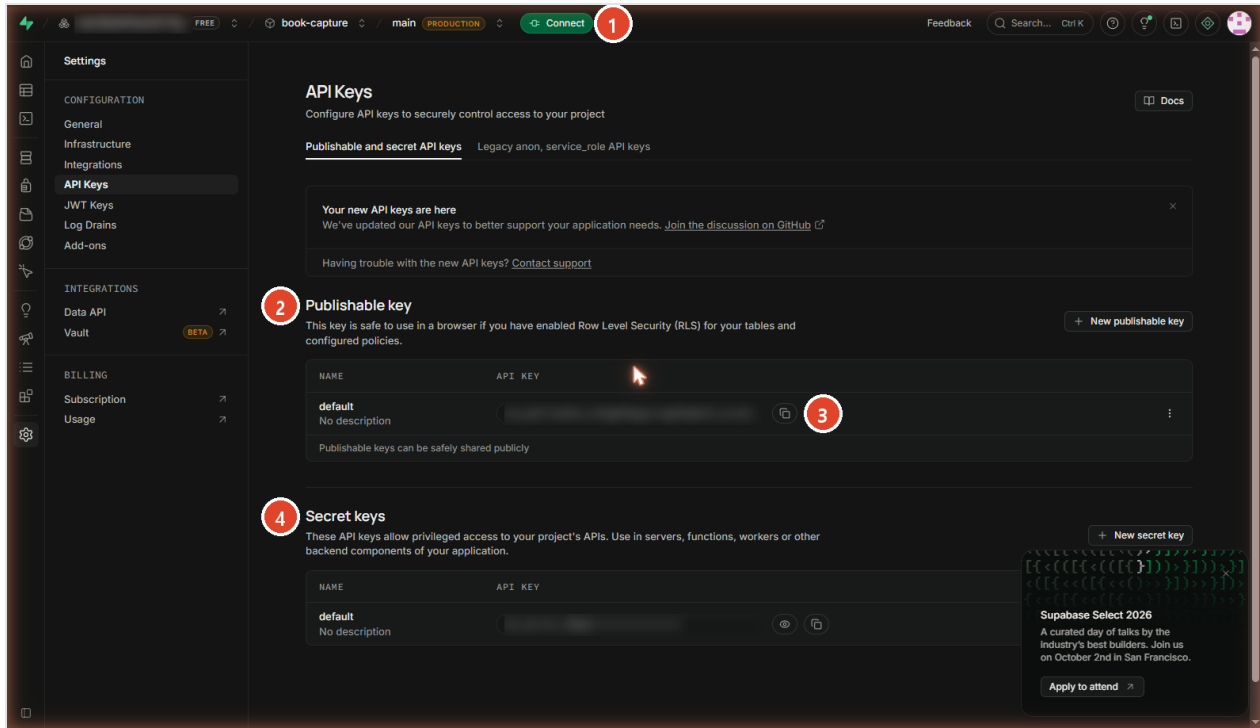


그림 21-5 Project Settings → API Keys. ① Connect — 누르면 Project URL 이 보인다 ② Publishable key ③ 복사 버튼 ④ Secret keys — 쓰지 않는다. 키 글자는 가렸다

| 받아 오는 것 | 무엇 | 비유 | |---|---|---| | ****Project URL**** | 내 창고의 인터넷 주소 | 창고의 번지수 |
| ****Publishable key**** | 웹과 앱이 창고에 말을 거는 데 쓰는 키. 예전 이름은 `anon public` | 창고 ****초인종**** |

📖 API 키(key) — 초인종과 진짜 열쇠

프로그램이 창고에 들어갈 때 내미는 **긴 암호 글씨**. Supabase는 두 종류를 줍니다.

Publishable key — 누구나 누를 수 있는 **초인종**입니다. 앱에 넣어 손님 폰에 들어가도 괜찮게 만들어져 있습니다. 초인종을 눌러도 문을 여는 건 창고의 규칙이 정합니다. 예전에는 `anon public` 이라고 불렀습니다.

Secret key — 모든 문을 여는 **진짜 열쇠**입니다(예전 이름 `service_role`). 이걸 앱에 넣으면 안 됩니다. 오늘은 쓰지 않습니다.

"이거 남한테 보여도 돼요?" 라고 물으면 Claude는 이렇게 답합니다.

"이건 보여도 되는 열쇠예요. 문을 여는 열쇠가 아니라 **초인종**에 가깝습니다. 진짜 열쇠(Secret key)는 안 씁니다."

붙여 주면 **Claude**가 웹 폴더와 앱 폴더에 적어 둡니다. 우리에게 파일을 만들라고 하지 않습니다.

📄 .env(닷엔브) 파일 — 설정 쪽지

주소와 열쇠처럼 **프로그램 바깥에서 넣어 주는 설정값**을 모아 두는 작은 파일. 이름이 점(.)으로 시작해서 탐색기에서는 잘 안 보입니다.

코드 안에 주소를 직접 박지 않고 이 쪽지에 따로 적어 두면, 나중에 창고가 바뀌어도 **쪽지만 바꾸면** 됩니다.

Claude가 이렇게 말하고 멈춥니다.

"창고가 생겼습니다. 이제 **답을 칸**을 만듭니다. **17단계 상자**를 붙여넣어 주세요."

21.5 가입 중에 하지 않는 것

안 한다	왜
이메일로 따로 가입	다음 주 GitHub · Vercel 까지 계정이 세 벌이 된다. GitHub 하나로 묶는다
Secret key 복사	모든 문을 여는 열쇠. 앱에 들어가면 안 된다
요금제 고르기	무료로 충분하다. 요금 이야기는 꺼내지 않는다
화면 캡처를 보고 알아서 찾기	Claude가 한 줄씩 말로 안내한다

21.6 정리

- 웹과 앱이 같은 걸 보려면 **둘 다 들여다볼 창고 하나**가 필요하다
- 창고를 인터넷에 빌려주는 서비스가 **Supabase**. 무료로 시작한다

- **GitHub 계정으로** 들어간다(다음 주 Vercel도 같은 계정). 한 번에 한 줄씩. **지역은 Seoul** — 되돌릴 수 없다
- 무료 창고는 **7일 안 쓰면 잠든다**. 다음 주 수업 전에 한 번 열어 본다
- 받아 오는 건 둘 — **Project URL**(번지수) · **Publishable key**(초인증). 진짜 열쇠(Secret key)는 안 쓴다

이 장에서 나온 말

말	쉬운 뜻
데이터베이스 · 창고	데이터를 표로 정리해 오래 보관하는 곳. 가게 장부
Supabase	인터넷에 데이터 창고를 빌려주는 서비스
Authorize	다른 서비스가 내 GitHub 계정으로 들어가도 되는지 묻는 확인
프로젝트(Supabase)	Supabase 안에 만든 내 창고 하나
지역(Region)	창고가 놓일 나라. 만든 뒤 못 바꾼다. 우리는 Seoul
일시정지(잠자기)	무료 창고가 7일 동안 안 쓰이면 스스로 멈추는 것
Project URL	내 창고의 인터넷 주소
API 키 · Publishable key	프로그램이 창고에 말을 걸 때 내미는 암호. 초인증. 예전 이름 anon public
Secret key	모든 문을 여는 진짜 열쇠(예전 이름 service_role). 오늘은 안 쓴다
.env 파일	주소·열쇠 같은 설정값을 모아 두는 쪽지 파일

21.7 안 될 때

증상	이렇게
GitHub 계정이 없다	20장으로 가서 Google 계정으로 먼저 만듭니다. 5분이면 됩니다
GitHub 로그인 화면에서 막힌다	비밀번호 칸이 아니라 아래 Continue with Google 을 누릅니다
영어라 뭘 눌러야 할지 모르겠다	어디를 눌러야 되는지 그림으로 알려줘
창고 비밀번호를 까먹었다	괜찮습니다. 오늘은 쓸 일이 없습니다
지역을 Seoul로 안 골랐다	지역을 잘못 골랐어 — 창고를 새로 만들어야 합니다. 지금 바로 말하세요
5분 넘게 막힌다	강사·조교에게 손을 드세요. 전체가 멈추지 않게 따로 봅니다
다음 주에 "어제까지 됐는데 안 돼요"	창고가 잠든 겁니다. supabase.com 에 들어가 깨우면 됩니다

22장 — 웹과 앱을 참고에 잇기

초고 v0.1 (2026-09-18) · 2주차 A 경로 17~19단계

22.1 참고에 칸부터

참고가 생겼습니다(21장). 그런데 지금은 **빈 참고**입니다. 선반도 칸도 없습니다. 물건을 넣으려면 먼저 **칸을 짜야** 합니다.

어떤 칸을 짤지는 새로 정하지 않습니다. **PRD 4장 「답을 것」**에 이미 있습니다. 10단계에서 덩어리마다 칸과 예시, 상태 흐름까지 채워 두었습니다(13장). 그걸 **그대로** 참고의 표로 옮깁니다.

📄 표(테이블, table)

참고 안에서 데이터를 담는 **칸이 정해진 표**. 엑셀 시트 한 장과 같습니다. 맨 윗줄에 칸 이름(이름 · 연락처 · 상태...)이 있고, 아래로 한 줄씩 쌓입니다.

공방 예시라면 표가 넷입니다 — 수강생 · 클래스 · 신청 · 공지. PRD 4장의 덩어리 하나가 표 하나가 됩니다.



그림 22-1 PRD 4장이 참고의 표가 된다. 「신청」 덩어리의 칸(누가 · 어느 클래스 · 상태 · 신청한 때)이 그대로 신청 표의 칸 이름 줄이 되고, 가짜 데이터가 한 줄씩 들어간다

22.2 17단계 · 답을 칸 만들기

17-① 상자입니다.

설계 문서의 답을 그대로 데이터 답을 자리를 만들어줘.
내가 뭘 복사해서 어디에 붙여야 하는지 알려줘.

Claude가 파일 두 개를 씁니다. 11단계에서 이름만 만들어 둔 `supabase/` 폴더 안입니다.

파일	무엇
<code>schema.sql</code>	칸을 짜는 지시서 — 표 네 개와 칸 이름
<code>seed.sql</code>	처음 넣어 둘 데이터 — 1부의 가짜 데이터를 그대로 옮긴 것

SQL(에스큐엘)

참고에게 일을 시키는 **전용 말**. "이런 칸으로 표를 만들어라", "이 줄을 넣어라", "상태가 신청인 줄만 꺼내 와라" 같은 지시를 이 말로 씁니다.

우리가 쓸 일은 없습니다. Claude가 써 주고, 우리는 **통째로 붙여넣고 실행 버튼만** 누릅니다.

`seed.sql` 에 넣는 데이터는 1부의 가짜 데이터와 **같은 규칙**을 지킵니다. **오늘 날짜**로, 몇 시간 뒤 하나, 짝 찬 것 하나, 빈 날 하나(13장). **빈 화면이 최악**이라는 참고로 옮겨도 그대로입니다.

SQL Editor에 붙여넣기

이 두 파일을 참고에 넣는 건 **우리 손**으로 합니다. Supabase 화면에서 해야 하기 때문입니다.

"왼쪽 메뉴에서 **SQL Editor** 를 눌러 주세요. 제가 드리는 글을 통째로 붙여넣고 오른쪽 위 초록 **Run** 을 누르시면 됩니다."

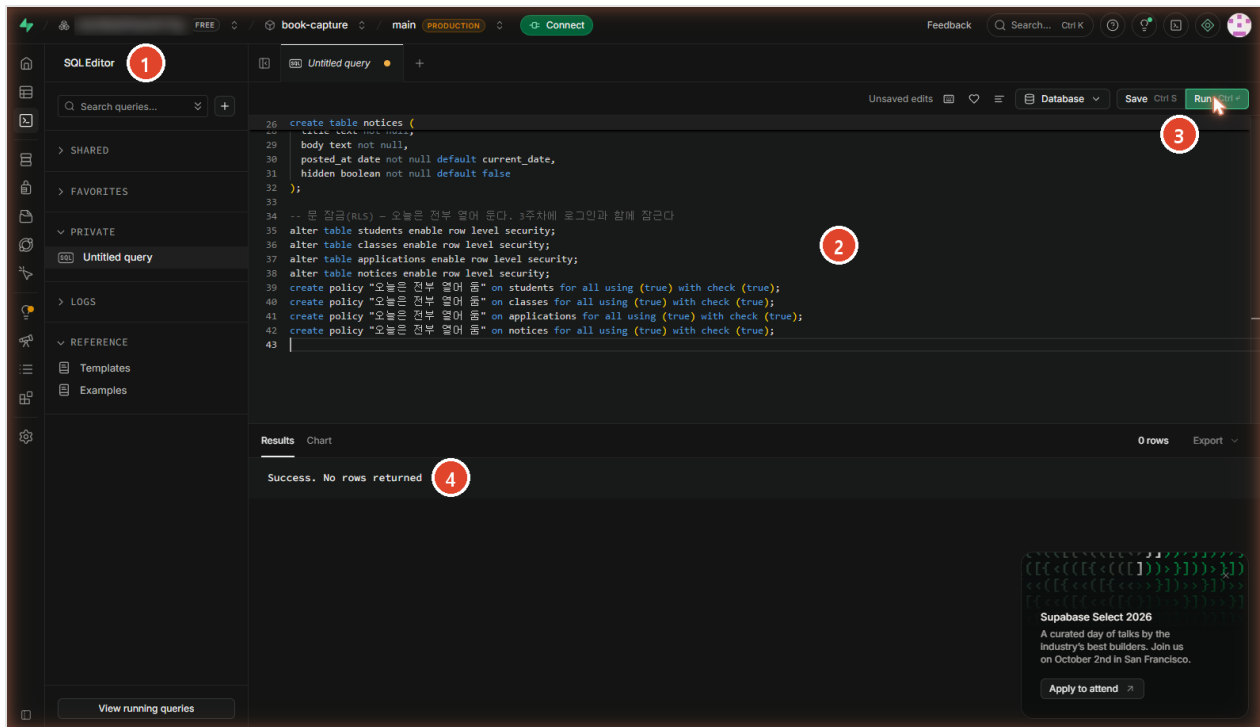


그림 22-2 Supabase의 SQL Editor. ① SQL Editor ② schema.sql 을 붙여넣은 칸 ③ Run ④ 아래에 뜨는 Success. No rows returned

****한 번에 하나씩**** 줍니다. `schema.sql` 을 먼저 붙여넣고 Run, 아래에 ****Success**** 가 뜨는 걸 확인하고, 그다음 `seed.sql`. 제작prompt에 "붙여넣을 게 ****두 번**** 옵니다" 라고 적힌 이유입니다. 두 번째를 붙여넣을 때는 칸에 남아 있는 첫 번째 글을 **먼저 지웁니다**(Ctrl + A 로 전부 고른 뒤 붙여넣기). 앞의 글이 남아 있으면 표를 또 만들려다 `already exists` 오류가 납니다. Run 을 눌렀을 때 **Potential issue detected** 창이 뜨기도 하는데, 표는 앞에서 이미 잠가 두었으니 **Run without RLS** 를 누르면 됩니다.

했어. 다음 거 줘.

둘 다 끝나면 **17-③ 상자**로 확인합니다.

들어갔는지 보여줘.

"칸이 생겼고 데이터도 들어갔습니다. 왼쪽 **Table Editor** 를 눌러 보시면 보여요."

The screenshot shows the Supabase Table Editor for the 'public:applications' table. The table has columns: id (int4), student_id (int4), class_id (int4), status (text), and created_at (timestampz). The table contains 20 rows of data. The sidebar on the left shows the database structure with 'applications' highlighted. The right-hand panel shows the RLS policy 'postgres' and a SQL editor with a 'Supabase Select 2026' event.

id	student_id	class_id	status	created_at
1	1	1	출석	2026-09-18 05:32:22.04477
2	2	1	출석	2026-09-18 05:32:22.04477
3	3	2	신청	2026-09-18 05:32:22.04477
4	4	2	신청	2026-09-18 05:32:22.04477
5	5	2	취소	2026-09-18 05:32:22.04477
6	1	3	신청	2026-09-18 05:32:22.04477
7	2	3	신청	2026-09-18 05:32:22.04477
8	3	3	신청	2026-09-18 05:32:22.04477
9	4	3	신청	2026-09-18 05:32:22.04477
10	5	3	신청	2026-09-18 05:32:22.04477
11	6	3	신청	2026-09-18 05:32:22.04477
12	7	3	신청	2026-09-18 05:32:22.04477
13	8	3	신청	2026-09-18 05:32:22.04477
14	9	3	신청	2026-09-18 05:32:22.04477
15	10	3	신청	2026-09-18 05:32:22.04477
16	1	4	신청	2026-09-18 05:32:22.04477
17	2	4	신청	2026-09-18 05:32:22.04477
18	3	4	신청	2026-09-18 05:32:22.04477
19	4	4	신청	2026-09-18 05:32:22.04477
20	6	5	신청	2026-09-18 05:32:22.04477

그림 22-3 Table Editor 에서 본 신청 표(applications). ① 표 이름 넷 ② 칸 이름 줄 ③ seed.sql 로 들어간 데이터 — 출석 · 신청 · 취소가 한 번씩 보인다

22.3 문 잠금 — 오늘은 다 열어 둔다

표를 만들면 Supabase는 표를 잠가 둡니다. 안 풀면 웹과 앱이 초인종(21장)을 눌러도 아무도 문을 안 열어 줍니다. 화면이 텅 빕니다.

📖 RLS(Row Level Security, 줄 단위 문 잠금)

창고의 표마다 누가 어떤 줄을 보고 쓸 수 있는지 정해 두는 규칙. 아파트 우편함과 같습니다. 건물에는 누구나 들어와도, 자기 우편함만 열 수 있게 하는 것입니다.

"수강생은 자기 신청만 본다" 같은 규칙이 여기에 들어갑니다.

오늘은 전부 열어 두는 규칙 하나만 겁니다. `schema.sql` 안에 같이 넣어 둡니다.

"이게 뭐예요?" 라고 물으면 Claude는 이렇게 답합니다.

"창고 문을 누가 열 수 있는지 정하는 겁니다. 오늘은 다 열어놨어요. 로그인을 다음 주에 만들면서 제대로 잠급니다."

💡 오늘 열어 둔 문은 인터넷에 올리기 전에 반드시 잠근다

지금은 내 컴퓨터와 내 폰에서만 보니 문이 열려 있어도 괜찮습니다. 그런데 3주차에 웹을 인터넷에 올리는 순간, **주소만 알면 누구나** 사장님 화면을 열 수 있습니다. 수강생 이름과 연락처가 그대로 보입니다.

그래서 이 사실을 **PRD에 적어 둡니다**. 3주차의 Claude가 PRD를 읽고 "문이 열려 있다 — 올리기 전에 로그인과 잠금부터" 를 먼저 봅니다.

그리고 잠글 때 따를 표는 **이미 있습니다**. 10단계에서 적은 4장 「누가 무엇을 볼 수 있나」입니다(13장). 그때 "오늘은 잠그지 않아도 지금 적는다" 고 한 이유가 이것입니다.

v0.5 — PRD 4장에 영어 칸 이름

창고의 표에는 영어 이름이 붙습니다. 17단계 끝에 PRD 4장도 **실제 표 구조**로 올립니다.

```
### 클래스 `classes`  
| 칸 | 무엇 | 비고 |  
|-----|-----|-----|  
| id | 번호 | 자동 |  
| 날짜 | 언제 열리나 |  
| 정원 | 몇 명까지 | 5장 - 6~10명 |  
  
> 문 잠금 - 오늘은 전부 열어둬. 3주차에 로그인 붙이면서 잠근다.
```

변경 이력에 **v0.5** — 4장을 실제 표 구조로 올림, 문 잠금 상태 기록. 왜: 데이터 둘 곳을 만들었다.

22.4 18단계 · 웹을 잇기

📦 18-① 상자입니다.

웹을 방금 만든 곳에 연결해줘. 가짜 데이터 말고 진짜를 보게.

Claude가 하는 일은 넷입니다.

1. 웹 폴더에 **창고에 붙는 부품**을 받아 온다 (npm, 14장)
2. 주소와 초인종을 웹 폴더의 쪽지(`.env`)에 적는다 — **Claude가**

3. **가짜 데이터 파일을 걷어내고**, 화면이 창고를 읽게 바꾼다
4. 크롬을 새로고침해서 **아까와 똑같이 보이는지** 본다

3번이 15장에서 한 약속입니다. 가짜 데이터를 **파일 하나에** 모아 두었기 때문에, 그 파일 하나만 갈아 끼우면 됩니다. 화면 코드는 거의 그대로입니다.

18-② 화면이 똑같은데 진짜 바뀐 거 맞아?

화면이 똑같아 보이는데 진짜 바뀐 거 맞아? 어떻게 확인해?

겉모습이 안 바뀌어서 **뭘 한 건지 모릅니다**. 당연한 질문입니다. Claude가 이렇게 답합니다.

"화면은 아까랑 똑같아 보이죠. 근데 **속이 바뀌었습니다**. 아까는 화면 안에 적어둔 가짜였고, 지금은 **창고에서 꺼내온 진짜**예요.

확인해볼까요? **새로고침(F5)** 해보세요. 안 사라지죠? 아까 주간 화면에서 만든 게 새로고침하면 사라졌던 거, 기억나시죠. 이제 안 사라집니다."

「주간」에서 클래스를 하나 열고, **F5** 를 누릅니다. **안 사라집니다**. 16장에서 칠판에 적던 것이 이제 **공책**에 적힙니다. 이 새로고침이 18단계의 **눈에 보이는 유일한 증거**라서, 지침서는 이 시연을 꼭 하라고 적어 두었습니다.



그림 22-4 같은 동작, 다른 결과. 18단계 전에는 주간에서 연 클래스가 새로고침하면 사라졌다. 18단계 뒤에는 창고에 적히므로 새로고침해도 남는다

> "웹이 창고에 붙었습니다. **이제 앱만 붙이면 오늘 끝입니다.**"

22.5 19단계 · 앱을 잇기 — 오늘의 결승선

앱도 연결해줘.

웹과 같은 일을 앱 폴더에 합니다. 여기에 두 가지가 더 붙습니다.

- **화면에 들어올 때마다 다시 읽는다.** 탭을 갔다 오면 새 내용이 보인다
- **당겨서 새로고침.** 화면을 아래로 쪽 당겼다 놓으면 다시 읽는다

창고가 바뀔 때마다 폰에 **즉시 알려 주는 기능**(실시간 구독)도 있지만 쓰지 않습니다 (21장). 탭을 갔다 오는 것만으로 충분히 보이고, 실패하는 곳이 줄어듭니다.

시연 — 눈앞에서 두 번

이 단계는 **시연이 본체**입니다. 말로 설명하지 않고 눈앞에서 두 번 보여줍니다.

① 웹 → 앱 — 19-② 상자

웹에서 하나 만들어볼게. 폰에 뜨는지 같이 확인하자.

"제가 지금 웹에서 **새 클래스를 하나 열겠습니다.** (실제로 연다) 자, 이제 **폰에서 다른 탭** 갔다가 다시 오세요. ...땀쥌? 컴퓨터에서 만든 게 폰에 온 겁니다."

② 앱 → 웹 — 19-③ 상자

이번엔 반대로 폰에서 신청해볼게. 웹에 뜨는지 보자.

"이번엔 반대로요. **폰에서 신청을 눌러보세요.** ...누르셨죠? 이제 **컴퓨터 화면을 보세요.** 새로고침(F5). ...명단에 이름이 들어왔습니다."

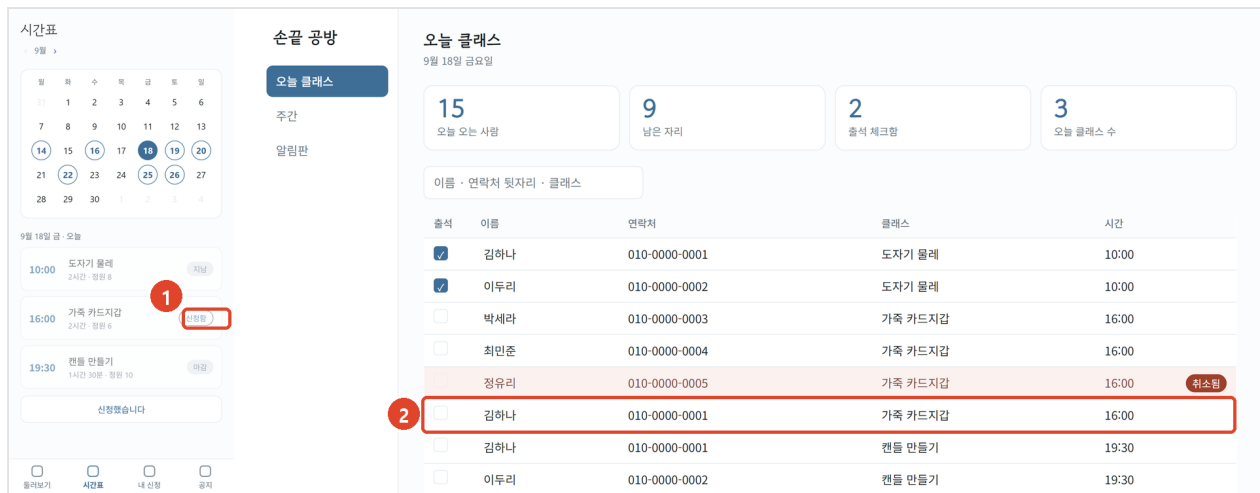


그림 22-5 오늘의 결승선. ① 앱 시간표에서 가족 카드지갑을 신청하자 「신청함」이 붙는다 ② 웹 「오늘 클래스」를 새로고침하니 가족 카드지갑 명단에 김하나가 들어와 있다

> "이게 오늘 만든 겁니다. > **웹과 앱이 같은 데이터를 보고 있어요.** 한쪽에서 누르면 다른 쪽에 뜹니다. > 지난주에 종이에 그렸던 게 지금 폰에서 돌아갑니다."

12장 첫머리의 약속 — "컴퓨터에서 누르면 폰 화면이 따라 바뀝니다" — 가 여기서 이루어집니다.

💡 왜 창고에 붙여도 화면은 그대로인가

웹과 앱을 창고에 이을 때 **화면은 거의 손대지 않았습니**다. 가짜 데이터 파일 하나만 창고로 바꿨습니다. 그게 가능했던 건 처음부터 **데이터와 화면을 나눠 두었기** 때문 입니다 — 가짜 데이터는 파일 하나에, 칸 이름과 상태는 PRD 4장에.

9장에서 "화면은 보여주는 곳, 답을 것은 기억해 두는 것" 이라고 했습니다. 이 둘을 섞지 않고 설계해 두면, 기억하는 곳이 칠판에서 공책으로 바뀌어도 **보여주는 쪽은 모릅니다**. 그대로 돌아갑니다. 설계가 먼저인 이유가 여기서 한 번 더 보입니다.

22.6 정리

- PRD 4장 「답을 것」이 그대로 창고의 표가 된다. 새로 정하지 않는다
- `schema.sql` (칸) → `seed.sql` (가짜 데이터) 순서로, **SQL Editor**에 붙여넣고 **Run**
- 문 잠금(RLS)은 **오늘 다 열어 둔다**. 인터넷에 올리기 전에 4장 「누가 보나」대로 잠근다
- 웹은 가짜 데이터 파일만 걸어내면 창고에 붙는다. 증거는 **새로고침해도 안 사라진다**
- 앱까지 붙으면 **웹에서 만든 게 폰에, 폰에서 누른 게 웹에 뜬다** — 오늘의 결승선

이 장에서 나온 말

말	쉬운 뜻
표(테이블)	창고 안의 칸이 정해진 표. 엑셀 시트 한 장
SQL	창고에게 일을 시키는 전용 말. Claude가 쓴다
schema.sql	창고에 표와 칸을 짜는 지시서
seed.sql	창고에 처음 넣어 둘 데이터
SQL Editor · Run	Supabase 안에서 SQL을 붙여넣는 칸 · 실행 버튼
Table Editor	Supabase 안에서 표의 내용을 눈으로 보는 화면
RLS(문 잠금)	누가 어떤 줄을 보고 쓸 수 있는지 정하는 규칙. 아파트 우편함
당겨서 새로고침	폰 화면을 아래로 당겼다 놓으면 내용을 다시 읽는 것

22.7 안 될 때

증상	이렇게
웹 화면이 텅 비었다	문 잠금이 안 풀렸을 수 있습니다. 화면이 텅 비었어
<code>Invalid API key</code> 가 나온다	초인종을 잘못 붙였습니다. 열쇠 다시 줄게 하고 Project Settings → API Keys 에서 다시 복사
아무것도 안 나온다	<code>seed.sql</code> 이 안 돌았을 수 있습니다. Table Editor 를 열어 보세요
「오늘」 화면인데 비어 있다	넣어 둔 데이터 날짜가 오늘이 아닙니다. 오늘 날짜로 다시 넣어줘
웹은 되는데 앱만 텅 비었다	앱 쪽에 주소나 초인종이 빠졌습니다. 앱만 비어 있어
폰에서 신청이 안 놀린다	문 잠금이 읽기만 열려 있을 수 있습니다. 폰에서 신청이 안 돼
시간이 없다	①(웹 → 앱)만 보고 23장으로. ②는 남는 시간에

23장 — 설계 문서 고쳐 쓰기, 누가 만들어도 똑같이

초고 v0.1 (2026-09-18) · 2주차 20단계 · 2주차 마지막

23.1 만들고 나면 또 어긋난다

지난주 마지막 9단계에서 한 일을 기억하나요? 화면 시안을 보고 나니 PRD가 안 맞아서 **고쳐 썼습니다**(11장).

오늘도 똑같은 일이 생겼습니다. 이번엔 시안이 아니라 **진짜로 만들어 보고** 알게 된 것들입니다. 공방 예시만 봐도 이만큼입니다.

- 검색은 이름만이 아니라 **연락처 뒷자리 · 클래스 이름**으로도 찾는다
- 뒷자리는 「**그 숫자로 끝난다**」로 찾는다 — 0001 버그
- 이름을 누르면 **오른쪽 칸**에 지난 기록까지 보인다
- 간판은 절반이 아니라 **딱 찬 간판**, 간판 안에 버튼은 없다
- 안드로이드는 그림 파일 이름에 **한글**을 쓰면 말썽이다
- 떠 있는 신청 버튼 밑에 **빈자리**를 뒤야 마지막 글이 안 가려진다

대부분은 이미 PRD에 들어가 있습니다. 오늘 내내 "문서 먼저"로 고쳐 왔으니깐요. 20단계는 그걸 **확인하고, 빠진 걸 채우고, 다음 주로 넘길 것을 적는** 10분입니다.

어디서 멈췄든 20단계는 합니다. A로 갔든 B로 갔든, 시간이 모자라 중간에 끊겼든. 5분이면 됩니다. 이게 다음 주의 출발점입니다.

23.2 20-① 달라진 것 반영하기

오늘 만들면서 지난주에 정한 거랑 달라진 게 있을 것 같아.

설계 문서 다시 봐주고, 달라진 거랑 이번에 새로 정해진 걸 반영해줘.

뭐가 왜 바뀌었는지도 남겨줘.

Claude가 PRD를 이 순서로 훑습니다.

어디	무엇을 보나
9장 확인 기준	통과한 줄에 ✓. 못 한 줄은 왜 못 했는지 한 줄
변경 이력	오늘 PRD가 바뀐 곳마다 줄이 있나 · 「왜」가 비지 않았나 · 맨 위 버전 줄과 맞나
8장 아직 안 정한 것	오늘 풀린 것을 지우고 확정 장으로 올린다
3.5장 무엇으로 만드나	내가 고른 웹 도구가 적혀 있나
5장 규칙	만들다 새로 정해진 규칙이 들어갔나
3장 화면	화면과 문서가 다른 곳이 있나 — 있으면 어느 쪽이 맞는지 묻고 맞춘다
7장 만드는 순서	오늘 한 것에 ✓, 다음 주로 넘긴 것 표시

3장에서 "화면과 문서가 다른 곳" 이 나오면 어떻게 될까요? 지침서의 답은 이렇습니다. **"고리를 제대로 돌았다면 없어야 한다."** 오늘 한 번이라도 화면만 고치고 문서를 안 고쳤다면 여기서 드러납니다.

8장에서 풀린 것 찾기

지난주에 「아직 안 정한 것」으로 남겨 둔 것 중에, **오늘 만들면서 저절로 정해진 것** 이 있습니다. 하나씩 짚어서 옮깁니다.

8장에 있던 것	오늘	어디로
간판 자리에 넣을 사진	정해졌다 — <code>images/간판.jpg</code>	3-5-1 간판
공방 상호와 한 줄 소개	간판에 넣으며 정해졌다	1장
웹을 무엇으로 만드나	11단계에서 Next.js	3.5장
로그인	아직 — 3주차, 올리기 전에	8장에 남긴다

남아 있는 것은 지우지 않습니다. 지난주 규칙 그대로입니다. 풀린 것은 줄을 긋고 「정해졌다 — 어디 참고」를 적어 두기도 합니다. 언제 풀렸는지가 보이게요.

오늘 나온 「다음에」를 모은다

네 바퀴를 돌다가, B 경로에서 화면을 더 만들다가 **"이것도 있으면 좋겠는데"**가 나왔을 겁니다. 그때 오늘 안 만들고 넘어갔습니다(15장). 그 한마디들을 **8장에 한 줄씩** 넣습니다. 오늘 안 만들었다고 사라지면 안 됩니다.

23.3 20-② 완성 검사, 한 번 더

제작prompt는 이 상자 앞에 한 줄을 붙여 두었습니다.

검사 전에 **변경 이력**을 한번 보세요. 오늘 문서가 어떻게 자라왔는지가 한 줄씩 남아 있습니다.

PRD 맨 아래를 열어 봅니다.

PRD.md - 맨 아래 「변경 이력」 (최신이 위)

버전	무엇	왜
v0.6 (A) v0.5 (B)	완성 검사 다시 · 9장 확인 결과 · 8장에서 풀린 것 정리 · 「다음 주에 할 것」	이 문서만으로 오늘 만든 걸 다시 만들 수 있게
v0.4.5	이름 칸 넓힘 · 위 숫자 32 → 40	크롬에서 보니 숫자가 안 읽혔다
v0.4.4	이름을 누르면 오른쪽 칸 — 자세히 보기	전화가 오면 그 사람을 바로 봐야 한다
v0.4.3	검색에 클래스 이름 추가	한 클래스 명단만 따로 보고 싶다
v0.4.2	연락처 검색을 「그 숫자로 끝난다」로	0001 을 쳤더니 010-0000-0010 이 같이 나왔다
v0.4.1	이름 · 연락처 뒷자리 검색	전화가 오면 번호로 먼저 찾게 된다
v0.4	「무엇으로 만드나」 — 웹 Next.js · 앱 Expo	만들 자리를 정했다
v0.3	화면마다 그림 · 동작 표 · 3.0 공통 · 5.9 · 9장 · 10장 · 완성 검사 18개 채움	이 문서만 보고 만들 수 있어야 한다
v0.2	고른 시안으로 화면 교체	화면을 실제로 보고 정했다
v0.1	처음 씀	7단계

회색 | **파랑** | 이번 주 — 하루 동안 쌓인 줄

그림 23-1 2주차가 끝난 PRD의 변경 이력. 지난주 v0.1 · v0.2 에서 시작해 10단계 v0.3(그림·동작 표·완성 검사), 11단계 v0.4(무엇으로 만드나), 네 바퀴의 v0.4.1~v0.4.5 를 지나 20단계 v0.5(B) 또는 v0.6(A)에 이른다. 줄마다 「왜」가 있다

아침에 v0.2였던 문서가 하루 동안 열 줄 넘게 자랐습니다. 그리고 **줄마다 「왜」가 있습니다.** 0001 버그를 왜 고쳤는지, 간판을 왜 뒤집었는지, 오른쪽 칸을 왜 열게 했는지. 이 줄들이 다음 주의 Claude에게, 석 달 뒤의 나에게 **만든 사람의 생각**을 전합니다. 그리고 검사합니다.

오늘 만든 웹이랑 앱을, 이 설계 문서 하나만 보고 똑같이 다시 만들 수 있는지
검사해줘. 빠진 게 있으면 채워줘.

13장의 완성 검사를 **한 번 더** 합니다. 대화를 모르는 새 Claude에게 `PRD.md` 한 파일만 주고, 만들다
멈출 곳을 찾게 합니다. 이번 목표는 더 구체적입니다.

오늘 만든 웹·앱을, 이 문서만으로 다시 만들 수 있다.

💡 빠진 게 거의 없다면, 그게 오늘 배운 것의 증거

제작prompt는 이 상자 아래에 이렇게 적어 두었습니다. "오늘 **문서 먼저** 고쳐 왔다면 빠진 게
거의 없어야 합니다."

아침 10단계의 첫 검사에서는 공방 예시에 **18개**가 나왔습니다. 오늘 네 바퀴를 돌며 검색을
넣고, 버그를 고치고, 자세히 보기를 붙이고, 간판을 뒤집었습니다. 그런데도 저녁 검사에서
구멍이 거의 없다면 — **바꾼 것마다 문서에 먼저 적었다**는 뜻입니다.

반대로 여기서 구멍이 많이 나오면, 어딘가에서 **화면만 고치고 넘어간** 겁니다. 그 구멍이 바로
다음 주에 "수정해줘" 를 했을 때 AI가 **자기 생각대로** 채울 자리입니다.

23.4 다음 주에 할 것 — 인계 메모

PRD 맨 뒤에 표 하나를 붙입니다. 3주차의 Claude가 이걸 보고 시작합니다. **간 길에 따라 1~2번이
다릅니다.**

	무엇	왜
1	(B) 데이터 창고 만들기 / (A) 문 잠그기	B — 새로그침하면 사라진다 / A — 오늘은 전부 열어 뒀다 ★
2	(B) 웹과 앱을 창고에 연결 / (A) 없음	
3	로그인	인터넷에 올리면 사장님 화면이 누구에게나 열린다. 올리기 전에 ★
4	(B) GitHub 계정 만들기 / (A) 이미 있음	올리려면 필요하다. Supabase · Vercel 도 이 계정 하나로 들어간다
5	웹을 인터넷에 올리기	
6	앱 설치 파일 만들기	시간이 걸리니 먼저 걸어 둔다
7	스토어 등록	아이콘 · 설명 · 캡처가 필요하다

★ 두 개를 보세요. 둘 다 "올리기 전에" 입니다. 오늘 로그인을 안 만들고 문을 열어 둔 채로 끝났습니다. 그건 괜찮습니다. 내 컴퓨터와 내 폰에서만 보니까요. 그런데 이 사실이 문서에 없으면, 다음 주에 "올려줘" 한마디에 열린 문 그대로 인터넷에 올라갑니다.

그래서 적어 둡니다. 3주차의 Claude는 PRD를 먼저 읽고, "올리기 전에 로그인과 잠금 부터" 를 봅니다. 기록이 사고를 막는 자리입니다.

버전

마지막 변경 이력 한 줄로 PRD 버전이 올라갑니다.

간 길	버전	왜
A	v0.6	17단계에서 v0.5(창고의 표 구조)를 이미 올렸다
B	v0.5	창고를 안 만들었으니 한 칸만 오른다

23.5 20-③ 2주차 끝

오늘 뭐 만들었는지 정리해서 보여줘. 다음 주에 뭐 하는지도.

Claude가 이렇게 말하고 멈춥니다. 3주차 내용을 미리 하지 않습니다.

" PRD.md 가 v0.0 가 됐습니다. 이 문서만 있으면 오늘 만든 걸 누가 다시 만들어도 똑같이 나옵니다.

오늘 만든 것 · 웹 화면 ○장 — 내 컴퓨터에서 돌아갑니다 · 앱 화면 ○장 — 내 폰에서 돌아갑니다 · 설계 문서 — 이게 오늘 진짜 만드신 겁니다. 화면은 이걸 확인한 거예요

지난주엔 코드가 한 줄도 없었는데, 오늘 진짜로 도는 것이 생겼습니다. 다음 주엔 저장되게 만들고, 인터넷에 올립니다."

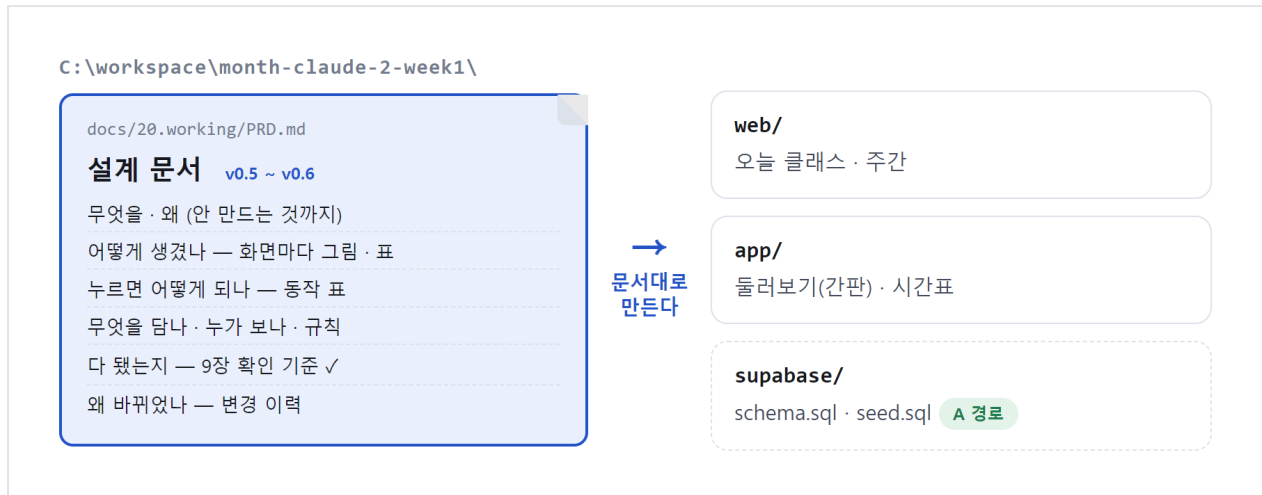


그림 23-2 2주차가 끝난 작업 폴더. docs 에는 v0.5~0.6 이 된 PRD(무엇을·왜·어떻게 생겼나·누르면 어떻게 되나·다 됐는지), web 과 app 에는 그 문서대로 만든 화면, supabase 에는 창고의 칸(A 경로). 폴더의 중심은 여전히 PRD다

"설계 문서 — 이게 오늘 진짜 만드신 겁니다" 가 이 장의, 그리고 2주차의 요지입니다.

23.6 이것이 "고치는 법" 이다

작가의 말은 이렇게 끝났습니다.

만든 사람의 생각을 기록해 두고, 그 기록을 지키면서 고친다.

지난주는 기록하는 법이었습니다. 무엇을 만들지, 무엇을 안 만들지, 어떻게 생겼는지. 오늘은 그 기록을 지키면서 고치는 법이었습니다. 네 바퀴 동안 같은 순서를 네 번 돌았습니다.



그림 23-3 오늘 몸매 익힌 순서. ① 지금 뭐가 있는지 본다 ② 어디를 바꿀지 정한다 ③ 문서를 먼저 고치고 변경 이력에 왜를 남긴다 ④ 그것만 화면에 반영한다 ⑤ 화면에서 눌러 본다. 다음 주에도, 혼자 할 때도 같은 순서다

> 💡 **"수정해줘"** 대신 **"문서부터 고쳐줘"** > > 작가의 말에서 **"수정해줘"** 만 하면 AI가 처음 만든 이유를 모르고 **"자기 생각대로"** > 고친다고 했습니다. 오늘은 그 반대를 네 번 했습니다. > > **"이름 칸이 좁아. 설계 문서부터 고쳐줘."** — 이 한마디가 들어가면, AI는 고치기 > 전에 문서를 **"읽습니다."** 왜 이 칸이 여기 있는지, 무엇을 안 만들기로 했는지, 지난번에 > 왜 바꿨는지를 봅니다. 그리고 **"그 생각 안에서"** 고칩니다. 고친 뒤에는 이유를 한 줄 > 남깁니다. 다음번 **"수정해줘"** 는 그 줄까지 읽고 시작합니다. > > 이게 **"유지보수"**입니다(11장). 만드는 건 오늘 하루에 끝났지만, 고치는 건 이제부터 > 계속합니다. 그리고 이제 고치는 법을 압니다.

23.7 다음 주 전에

- **창고를 한 번 열어 보기** (A로 간 분) — supabase.com 에 들어가기만 하면 됩니다. 7일 넘게 안 들어가면 잠듭니다(21장)
- **PRD 변경 이력 읽어 보기** — 오늘 무엇이 왜 바뀌었는지 한 줄씩. 다음 주의 Claude도 이걸 읽고 시작합니다
- **Vercel 가입해 오기** — GitHub 계정으로. GitHub이 아직 없으면 20장부터. 올리는 건 다음 주, 가입까지만(24장)
- 수업에서 20단계까지 못 갔다면 **20-① · 20-② 만이라도** 해 오기

🔄 혼자 해보기

답을 알려주지 않습니다.

1. 웹 「오늘 클래스」에서 마음에 안 드는 곳을 **하나** 찾으세요. 그리고 **oo**가 마음에 안 들어. 설계 문서부터 고치고, 화면도 고쳐줘. 라고 쳐 보세요. 변경 이력에 몇 번 버전으로 남았나요? 「왜」 칸에 뭐라고 적었나요?
2. 새 대화를 열고 PRD.md 만 읽고, 오늘 클래스 화면에서 검색이 어떻게 동작하는지 설명해줘 라고 물어보세요. **0001** 을 치면 누가 나오는지까지 답하나요?

3. 이번엔 **화면만** 고쳐 달라고 해 보세요(문서는 두고 화면만 고쳐줘). 그리고 새 대화에서 PRD대로 오늘 클래스 화면 다시 만들어줘 라고 하면 어떻게 될까요? 방금 고친 게 살아 있나요?

23.8 막혔을 때 — 2주차 전체

그대로 치시면 됩니다.

이런 일이 생기면	이렇게 치세요
뭘 기다리는지 모르겠다	지금 뭐 하는 중이야? 얼마나 걸려?
빨간 글씨가 잔뜩 나왔다	빨간 글씨 나왔어 (그대로 복사해서 붙여도 됩니다)
화면이 안 뜬다	화면이 안 떠
폰에 앱이 안 뜬다	폰에 안 떠
폰에서 열었는데 하얗다	폰이 하얀 화면이야
어디를 눌러야 할지 모르겠다	어디를 눌러야 되는지 그림으로 알려줘
아까로 돌아가고 싶다	방금 건 취소하고 다시 하자
화면이 설계 문서랑 다르다	설계 문서랑 다른데 문서대로 해줘
마음에 안 든다	oo가 마음에 안 들어. 설계 문서부터 고치고, 화면도 고쳐줘.
너무 빨리 지나갔다	방금 뭐 한 거야? 쉽게 설명해줘
답이 너무 길다	짧게
간판 사진을 바꾸고 싶다	간판 사진 바꿔줘 (새 사진을 끌어다 놓으면서)
폰에서 글씨가 안 읽힌다	폰에서 글씨가 안 보여

여기 없는 문제는 그냥 물어보세요. 이거 왜 이래? 도 충분한 질문입니다.

23.9 정리

- 만들어 보면 또 어긋난다. 20단계는 확인하고 · 채우고 · 넘길 것을 적는 10분
- 8장에서 풀린 것은 확정 장으로, 오늘 나온 「다음에」는 8장으로

- **완성 검사 한 번 더** — 오늘 만든 웹.앱을 이 문서만으로 다시 만들 수 있나
- PRD 맨 뒤에 **다음 주에 할 것**. 로그인과 문 잠금은 **올리기 전에** ★
- 오늘 진짜로 만든 것은 **설계 문서**. 화면은 그 문서를 확인한 것

이 장에서 나온 말

말	쉬운 뜻
인계 메모	다음 사람(다음 주의 Claude)이 이어받을 것을 적어 둔 표
확정 장으로 올리기	8장 「아직 안 정한 것」에서 풀린 것을 해당 장으로 옮기는 일
완성 검사(다시)	오늘 만든 것까지 넣고, 대화를 모르는 Claude에게 PRD만 읽혀 보는 일
고치는 순서	본다 → 정한다 → 문서 먼저 → 그것만 화면에 → 눌러 본다

2주차가 끝났습니다. 다음 주에는 이 문서를 들고 **인터넷에 올립니다**.

24장 — 올리기 전에, Vercel 가입

초고 v0.1 (2026-09-22) · 2주차 과제 「3주차 전에 가입해 오기」 · 인터넷에 올리기 직전 단계

24.1 다음 주에 세 곳을 만난다

지금 만든 웹은 내 컴퓨터 안에만 있습니다. 주소창에 `localhost` 라고 떠 있던 것을 기억하나요 (14장)? 내 컴퓨터를 끄면 아무도 볼 수 없습니다.

다음 주에는 이 웹을 인터넷에 올립니다. 올린다는 건, 항상 켜져 있는 남의 컴퓨터에 파일을 옮겨 두고 그 자리에 주소를 붙이는 일입니다. 예전에는 그 컴퓨터를 직접 빌리고 설정해야 했지만, 지금은 서비스들이 대신해 줍니다.

서비스	하는 일	비유	가입
GitHub	만든 코드를 인터넷에 보관한다	코드 창고	20장
Supabase	신청 같은 데이터를 보관한다	데이터 장부	21장
Vercel	창고의 코드를 가져가 주소를 붙여 띄운다	창고 물건을 진열하는 가게	이 장

수업 시간에 가입부터 하면 한 시간이 금방 갑니다. 영어 화면, 확인 문자, 비밀번호 찾기에서 한 명씩 멈추기 때문입니다. 그래서 가입은 수업 전에 해 옵니다. 올리는 것 자체는 하지 않습니다. 올리기 바로 앞 단계까지만 갑니다.

Vercel(버셀)

만든 웹사이트에 주소를 붙여 인터넷에 띄워 주는 서비스. 무료로 시작할 수 있습니다.

GitHub 창고를 연결해 두면, 창고의 코드가 바뀔 때마다 알아서 새 버전을 띄웁니다. 우리가 할 일은 창고에 넣는 것뿐입니다. 이런 일을 배포(deploy) 라고 하고, 다음 주에 합니다.

먼저 확인 — GitHub 계정이 있나

Vercel은 GitHub 계정으로 들어갑니다(그림 20-1). 19장 갈림길에서 A 로 간 분은 20장에서 이미 만들었습니다. B 로 간 분은 20장부터 하고 오세요. 5분이면 됩니다.

24.2 GitHub 계정으로 가입하기

GitHub에 로그인된 상태 그대로, 새 탭에서 `vercel.com` 을 엽니다.

	할 일	됐으면
1	오른쪽 위 Sign Up 을 누른다	버튼 네 개짜리 가입 화면이 뜬다
2	★ Continue with GitHub 을 누른다. Google이 아니다	GitHub 확인 창이 뜬다
3	「Vercel by Vercel wants access...」 창에서 초록 Authorize 를 누른다	Vercel로 돌아온다
4	Verification 화면에서 휴대폰 번호를 넣고 Continue	문자로 숫자(코드)가 온다
5	문자로 받은 숫자를 넣는다	다음 화면으로 넘어간다
6	이름이나 쓰임새를 물으면 내 이름을 넣고 Hobby(개인) 를 고른다	Vercel 첫 화면이 뜬다

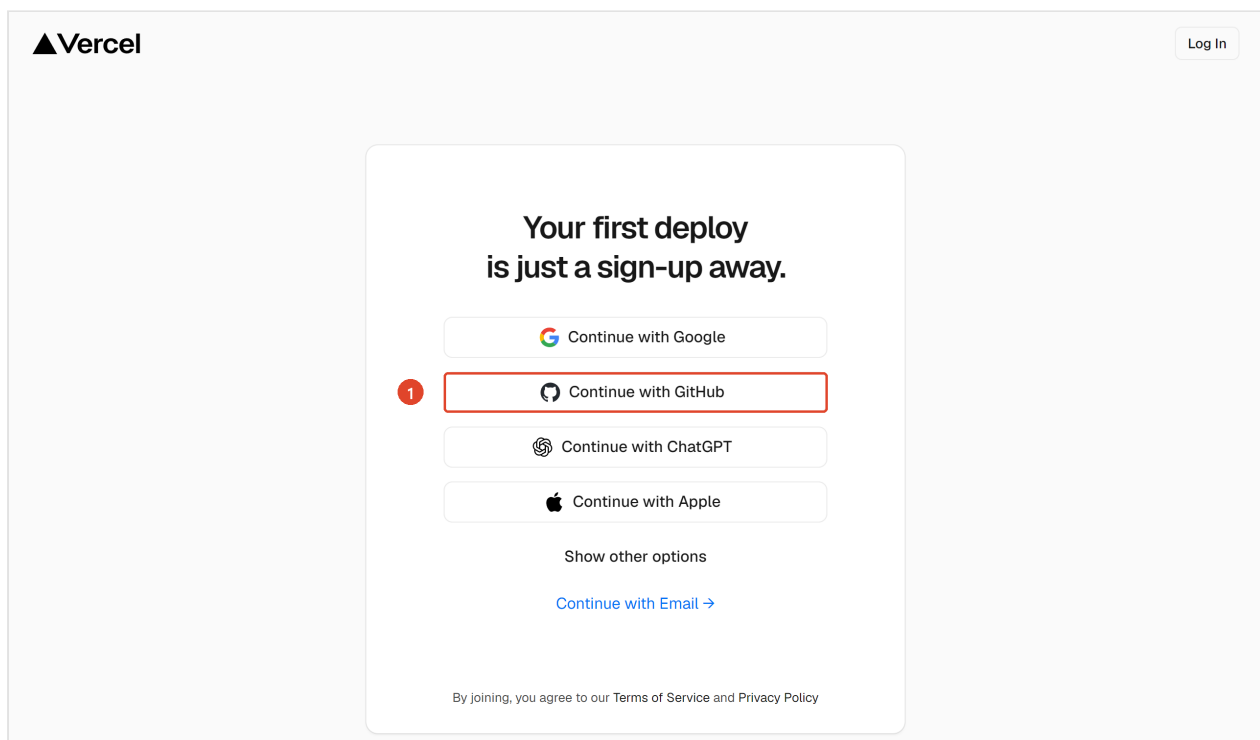


그림 24-1 Vercel 가입 화면. ① Continue with GitHub — 여기를 누른다. 맨 위 Continue with Google 이 아니다

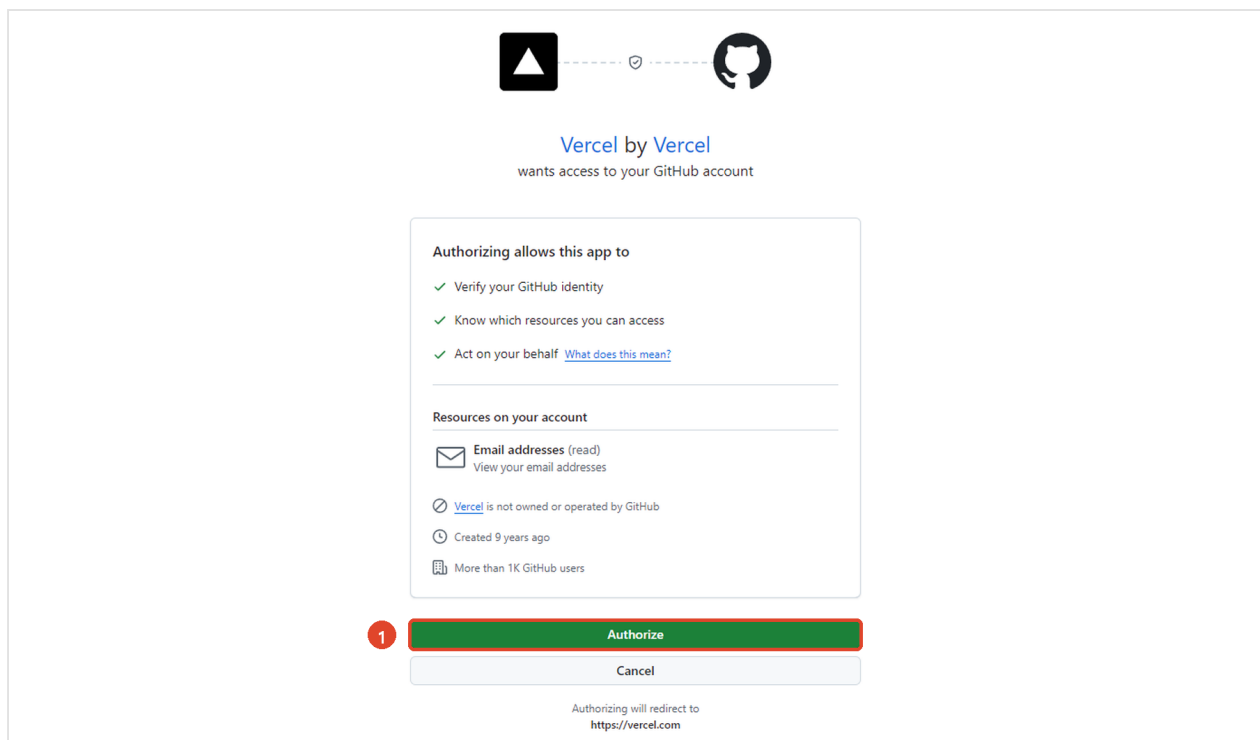


그림 24-2 GitHub이 묻는 확인 창. Vercel이 내 GitHub 계정을 확인하고 메일 주소를 읽어도 되는지 묻는다. ① Authorize. 사용자 이름은 가렸다

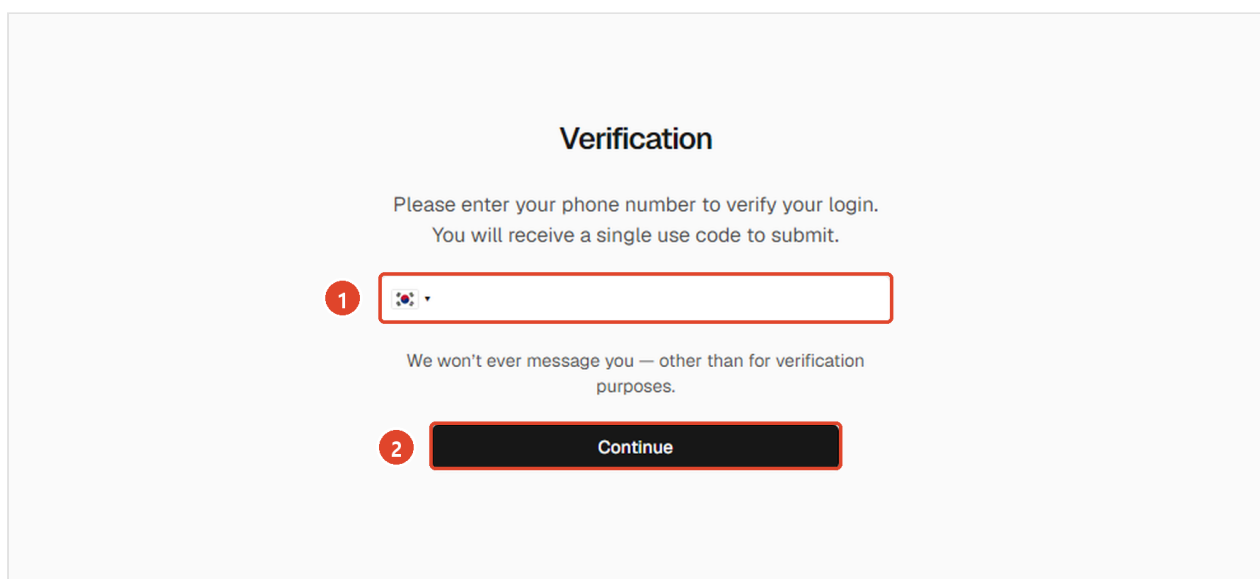


그림 24-3 휴대폰 확인. ① 나라는 태극기(+82) 그대로, 번호는 010의 맨 앞 0을 빼고 10부터 ② Continue — 문자로 온 숫자를 다음 화면에 넣는다. 번호는 가렸다

2번 — 버튼이 네 개인데 왜 GitHub인가

Vercel 가입 화면에는 **Continue with Google** 이 맨 위에 있습니다. 여기서 Google을 누르고 싶어집니다. GitHub을 Google로 만들었으니까요.

그런데 Vercel은 **Google이 아니라 GitHub과 이어져야** 합니다. 다음 주에 Vercel이 하는 일은 **GitHub 창고에서 코드를 가져가는** 것입니다. GitHub으로 들어가 있으면 창고가 바로 보입니다. Google로 들어가면 나중에 GitHub을 따로 또 이어야 합니다.

무엇으로 들어가나	들어가는 곳	왜
Google	GitHub	비밀번호를 새로 안 만들려고 (20장)
GitHub	Supabase · Vercel	셋이 같은 계정이어야 다음 주에 서로 잇기 쉽다

그리고 Vercel은 **휴대폰 번호 하나에 계정 하나만** 만들어 줍니다(4~5번). Google로 가입을 끝내 버리면, 같은 번호로 GitHub 가입을 **다시 할 수 없습니다**. 그때는 새로 만들지 말고 그 계정 설정에서 GitHub을 이어야 합니다. 처음에 GitHub을 누르는 게 제일 쉽습니다.

3번 — Authorize 는 무엇을 허락하나

"Vercel이 내 GitHub 계정으로 들어가도 되나요?" 라는 확인입니다(21장 Supabase 때와 같습니다). 창에 적힌 허락은 **내가 누구인지 확인하고, 메일 주소를 읽는 것**입니다.

Act on your behalf (나 대신 움직인다) 라는 줄이 무섭게 보일 수 있습니다. 그래도 **창고의 코드를 가져가는 허락은 여기 없습니다**. 그건 다음 주에 창고를 이을 때 따로 묻고, 그때도 우리가 고른 창고 하나만 허락합니다.

4~5번 — 왜 휴대폰 번호를 묻나

Authorize를 누르면 Vercel이 **휴대폰 번호**를 묻습니다. 로그인한 사람이 진짜 사람인지 확인하는 절차입니다. 번호를 넣으면 **한 번만 쓰는 숫자**가 문자로 오고, 그걸 넣으면 넘어갑니다. 화면에도 "확인 말고는 문자를 보내지 않는다" 고 적혀 있습니다.

번호는 이렇게 넣습니다.

칸	넣는 것
나라	태극기(+82) 그대로
번호	010-1234-5678 이면 맨 앞 0을 빼고 10 1234 5678

문자가 1~2분 안에 안 오면 번호를 다시 봅니다. 맨 앞 0을 넣었거나, 나라가 한국이 아닌 경우가 대부분입니다.

6번 뒤에 — 여기서 멈춘다

첫 화면에 **Import** 나 **Deploy** 같은 버튼이 보입니다. **누르지 않습니다**. 아직 올릴 창고가 없고, 문도 열려 있습니다(22장). 로그인과 문 잠그기를 먼저 하고 올린다는 것이 PRD의 「다음 주에 할 것」에 적혀 있습니다(23장). 오늘은 **가입까지**.

24.3 Supabase를 아직 안 했다면

19장 갈림길에서 **B(화면 더)** 로 간 분은 아직 Supabase가 없습니다. GitHub 계정이 생겼으니 21.3절 순서 그대로 하면 됩니다. **Continue with GitHub** 로 들어가고, 지역은 **Northeast Asia (Seoul)**.

A로 간 분은 이미 했습니다. 대신 **다음 주 수업 전에 supabase.com 에 한 번 들어가 보세요. 7일 넘게 안 들어가면 창고가 잠듭니다(21장).**

24.4 다 됐는지 확인

세 곳에 각각 들어가서 **오른쪽 위 내 프로필 사진**을 눌러 봅니다. 셋 다 **같은 사람**으로 보이면 끝입니다.

확인	됐으면
github.com 에 들어가면 로그인돼 있다	☐
내 Username 을 적어 두었다 (github.com/ooo)	☐
supabase.com 에 들어가면 로그인돼 있다	☐
vercel.com 에 들어가면 로그인돼 있고, 계정에 GitHub이 이어져 있다	☐

Vercel에 GitHub이 이어졌는지 모르겠으면 Claude에게 물어봐도 됩니다.

Vercel에 GitHub 계정이 제대로 연결됐는지 확인하는 방법 알려줘.
어디를 눌러야 하는지 하나씩.

24.5 정리

- 다음 주에 웹을 인터넷에 올린다. **GitHub**(코드 창고) → **Vercel**(주소 붙여 띄우기)
- 가입은 **수업 전에** 해 온다. 올리는 건 다음 주, 오늘은 **가입까지**
- Vercel은 **Continue with GitHub** — 맨 위 Google 이 아니다
- Vercel은 **휴대폰 번호 하나에 계정 하나**. 잘못 가입하면 다시 가입이 안 된다
- 번호는 +82 다음에 **010의 0**을 빼고
- B로 간 분은 GitHub(20장) → Supabase(21장)도 같이

이 장에서 나온 말

말	쉬운 뜻
Vercel	웹사이트에 주소를 붙여 인터넷에 띄워 주는 서비스
배포(deploy)	만든 것을 올려서 실제로 남이 볼 수 있게 하는 일. 다음 주
Authorize	다른 서비스가 내 계정으로 들어가도 되는지 묻는 확인
휴대폰 확인(Verification)	문자로 온 한 번짜리 숫자를 넣어 진짜 사람인지 확인하는 것
Hobby	Vercel의 개인용 무료 요금제

다음 주에는 세 곳을 **잇습니다**. 코드를 GitHub 창고에 넣고, Vercel이 그걸 가져가 주소를 붙입니다. 그 주소를 폰으로 열면, 내 가게 사이트가 인터넷에 있습니다.

24.6 안 될 때

증상	이렇게
GitHub 계정이 없다	20장부터 하고 오세요. Google 계정으로 5분이면 됩니다
Vercel에서 실수로 Google로 가입했다	다시 가입하지 마세요 — 같은 번호로는 안 됩니다. Vercel에 GitHub 연결하는 방법 알려줘 — 설정에서 GitHub을 이어 줍니다
This phone number is already associated with a Vercel account	그 번호로 만든 Vercel 계정이 이미 있습니다 . 번호 하나에 계정 하나입니다. 새로 가입하지 말고 Log In 으로 예전 계정에 들어간 뒤 Vercel에 GitHub 연결하는 방법 알려줘
휴대폰 확인 문자가 안 온다	번호 맨 앞 0을 뺐는지, 나라가 +82(태극기)인지 봅니다. 그래도 안 오면 1~2분 뒤 다시
Authorize 창이 안 뜨고 멈췄다	팝업이 막힌 겁니다. 주소창 오른쪽의 팝업 차단 표시를 눌러 허용하세요
영어라 뭘 눌러야 할지 모르겠다	Claude에게 화면을 캡처해서 붙이고 어디를 눌러야 돼?
수업 전까지 못 했다	괜찮습니다. 수업 첫 시간에 같이 합니다. 대신 조금 일찍 와 주세요

25장 — 공공데이터포털, 가입하고 쓸 데이터 고르기

초고 v0.1 (2026-09-23) · 3주차 과제 「다음 주에 쓸 데이터 정하고 키 신청하기」

25.1 지금 화면은 내가 적은 것만 보여 준다

지금까지 만든 웹과 앱을 떠올려 봅시다. 클래스 이름, 시간, 정원, 가게 소개. 전부 **내가 적어 넣은** 것입니다. 내가 안 고치면 화면도 그대로입니다.

다음 주에는 여기에 **밖에서 흘러 들어오는 데이터**를 붙입니다. 오늘 미세먼지, 내일 비 소식, 이번 주말 동네 축제 같은 것들입니다. 내가 적지 않아도 **알아서 바뀝니다**.

이런 데이터를 정부와 공공기관이 **무료로** 내놓는 곳이 있습니다.

📖 공공데이터포털(data.go.kr)

정부·공공기관이 가진 데이터를 **국민 누구나 무료로** 쓸 수 있게 모아 둔 공식 창구. 날씨, 대기질, 교통, 축제, 관광 정보 같은 것이 들어 있습니다.

들어가서, 쓰고 싶은 데이터에 **활용신청**을 넣으면 **인증키(열쇠)**를 줍니다. 그 열쇠로 데이터를 불러옵니다.

붙이는 건 다음 주입니다. 그런데 **신청은 이번 주에** 해야 합니다. 신청하자마자 되는 데이터도 있지만, **심사에 1~2일** 걸리는 데이터도 있기 때문입니다. 다음 주 수업 당일에 신청하면 늦습니다.

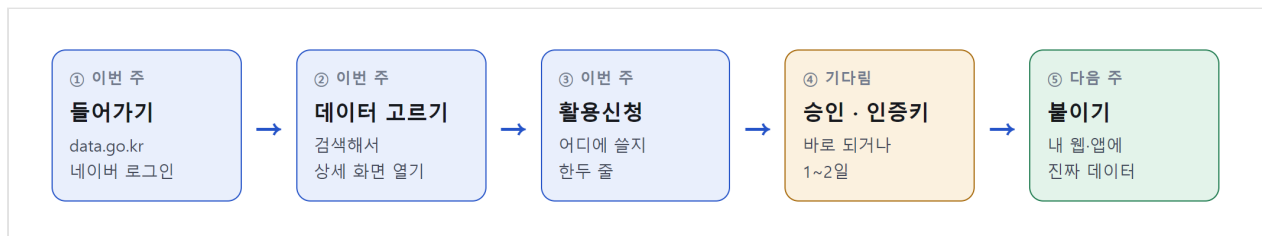


그림 25-1 공공데이터를 붙이기까지. ①~③ 네이버로 들어가기 · 데이터 고르기 · 활용신청은 이번 주에 한다 ④ 승인과 인증키는 바로 나오기도 하고 1~2일 걸리기도 한다 ⑤ 다음 주에 내 웹·앱에 붙인다

1장에서 설치 순서를 *****얼마나 기다려야 하나*****로 정했던 것과 같은 이유입니다. 기다리는 일은 Claude가 앞당길 수 없습니다(4장).

25.2 API — 주방에 들어가지 않고 주문서를 낸다

공공데이터를 받아 오는 방법에는 이름이 있습니다.

📖 API(에이피아이)

다른 곳이 가진 데이터를 **정해진 방식으로 받아오는 창구**.

식당에 비유하면, 주방에 직접 들어가지 않고 **주문서를 내밀면** 음식이 나옵니다. 그 주문서 양식이 API입니다. 정식 이름은 Application Programming Interface.

우리 쪽 일은 단순합니다. 주소 하나에 **열쇠와 원하는 것을 붙여 부르면**, 데이터가 **글자 덩치**로 돌아옵니다. 그걸 화면에 예쁘게 뿌립니다.



그림 25-2 API로 데이터를 받는 길. 내 사이트가 주소와 인증키, 원하는 조건을 담아 주문서를 내면(요청) 창구가 열쇠를 보고 데이터를 꺼내 글자 덩치로 돌려준다(응답). 그걸 화면에 「미세먼지 보통」처럼 보여 준다

돌아오는 글자 덩치에도 이름이 있습니다.

📖 JSON(제이슨)

데이터를 주고받을 때 쓰는 **표준 글자 형식**. "이름": "값" 을 중괄호로 묶어 늘어놓습니다. 사람도 읽을 수 있게 생겼습니다.

쓰는 법은 몰라도 됩니다. Claude가 읽고 화면으로 바꿉니다. 우리는 **읽는 법만** 알면 됩니다.

그림 25-2의 JSON은 "종로구 측정소, 미세먼지 등급 2" 라는 뜻입니다. 등급 2가 「보통」이라는 건 데이터 설명서에 적혀 있고, 그걸 화면 글자로 바꾸는 게 다음 주에 Claude가 할 일입니다.

25.3 가입하기 — 네이버 로그인으로 들어간다

Chrome에서 `data.go.kr` 을 엽니다. 한국어 화면입니다. 오른쪽 위 **로그인**을 누르면 「공공데이터포털 로그인」 화면이 나옵니다. 왼쪽은 아이디 칸, 오른쪽은 **간편 로그인**입니다.

공공데이터포털 로그인

※ Any-ID 사용을 위해 최초 1회 아이디/비밀번호 로그인이 필요합니다.
 ※ 공공데이터포털의 회원으로 등록되지 않으셨다면, 회원가입을 해주셔야 합니다.

그림 25-3 공공데이터포털 로그인 화면. ① 간편 로그인의 네이버 로그인 — 여기를 누른다 ② 왼쪽 아이디 · 비밀번호 칸은 쓰지 않는다

|| 할 일 | 됐으면 | ---|---|---| | 1 | 오른쪽 위 ****로그인****을 누른다 | 「공공데이터포털 로그인」 화면이 뜬다 || 2 | ★ 오른쪽 ****간편 로그인**** 아래 ****네이버 로그인****을 누른다 | 네이버 로그인 창이 뜬다 || 3 | 평소 쓰는 네이버 계정으로 로그인한다 | 정보 제공에 동의하는지 묻는다 || 4 | 동의하고 넘어간다 | 공공데이터포털로 돌아온다 || 5 | 오른쪽 위에 내 이름이 보이는지 본다 | 들어왔다 |

20장의 **소셜 로그인**과 같은 방식입니다. GitHub에 「Continue with Google」로 들어갔듯이, 공공데이터포털에는 **네이버**로 들어갑니다. 새 아이디도, 새 비밀번호도 만들지 않습니다.

왜 왼쪽 아이디 칸을 안 쓰나

왼쪽 칸을 쓰려면 먼저 **회원가입**으로 아이디를 만들어야 합니다. 회원구분 → 약관동의 → 정보입력 → 본인인증 → 가입완료, 다섯 단계입니다. 비밀번호도 하나 더 외워야 하고, 들어갈 때마다 **자동등록방지 글자**(찌그러진 글자)까지 따라 적어야 합니다.

들어가는 법	새로 만들 것	들어갈 때마다
네이버 로그인	없음	네이버로 확인
회원가입 → 아이디 로그인	아이디 · 비밀번호	아이디 · 비밀번호 · 찌그러진 글자

예전에 아이디로 가입해 둔 분(1장 준비물)은 왼쪽 칸으로 들어가도 됩니다. 둘 중 편한 쪽 하나면 됩니다.

이번엔 네이버다 — 열쇠가 하나 더 생긴다

GitHub · Supabase · Vercel은 Google 열쇠 하나로 묶었습니다(20장). 공공데이터포털은 그 사슬 밖이고, **네이버 열쇠**로 엮입니다. 네이버를 안 쓰면 바로 아래 **카카오 로그인**도 같은 순서입니다.

20장에서 한 말이 여기도 그대로입니다. 맨 위 열쇠(네이버)를 잃으면 이 문도 같이 잠깁니다. 네이버 계정에도 **2단계 인증**이 켜져 있는지 한 번 봐 둡니다.

25.4 쓸 데이터 고르기

로그인한 뒤 첫 화면 가운데 검색칸에 찾는 것을 적습니다. **미세먼지** , **단기예보** , **문화축제** 처럼 짧게 적으면 됩니다. 결과는 종류별로 나뉘어 나옵니다.

종류	무엇	우리가 쓰나
파일 데이터	엑셀·CSV(표 파일)를 통째로 내려받는다. 갱신이 느리다	쓰지 않는다
오픈API	주소를 부르면 그때그때 최신 데이터 가 온다	이것

「오늘 미세먼지」를 보여 주려면 오늘 것이 와야 합니다. 그래서 **오픈API** 칸에서 고릅니다. 결과에서 이름을 누르면 상세 화면이 열립니다.

신청 가능 트래픽** | 하루에 몇 번 부를 수 있나 | 에어코리아는 하루 500번, 기상청 단기에보는 10,000번 |

심의유형에 **개발단계**와 **운영단계**가 따로 적혀 있습니다. 처음 신청하면 받는 것은 **개발계정**입니다. 만들어 보고 시험하는 용도라 대부분 **자동승인**이고, 우리 가게 사이트에 붙이는 데도 이걸로 충분합니다. 손님이 많아져 하루 한도가 모자라면 그때 운영계정을 신청합니다.

📖 트래픽(호출 한도)

하루에 그 창구를 몇 번 부를 수 있는지의 한도. 식당으로 치면 하루에 받아 주는 주문서 장수입니다.

만드는 동안 새로고침을 수십 번 하면 그때마다 한 번씩 썩니다. 한도를 다 쓰면 그날은 데이터가 안 옵니다. 다음 날 다시 채워집니다.

「활용신청」이 아니라 「제공처 바로가기」라면

검색 결과를 보면 어떤 데이터는 오른쪽에 **활용신청** 버튼이 있고, 어떤 데이터는 **제공처 바로가기** 버튼이 있습니다.

교통물류

공공기관

미리보기

XML

JSON

서울시설공단_공공자전거 대여소 현황 >

서울특별시 공공자전거 대여소(따릉이) 현황정보입니다. 대여소의 이름, 관리번호, 위치정보와, 거치대수 정보를 제공합니다.

제공기관

서울시설공단

수정일

2025-06-05

조회수

6,105회

활용신청

1,372건

키워드

공공자전거 대여소,따릉이 대여소,서울시 대여소

1

제공처 바로가기

그림 25-6 따릉이 대여소 현황 검색 결과. ① 활용신청이 아니라 「제공처 바로가기」다. 이 데이터는 공공데이터포털이 아니라 제공한 기관의 사이트에서 따로 신청한다

> 💡 ****버튼 하나가 가입을 한 번 더 부른다**** > > 공공데이터포털은 여러 기관의 데이터를 ****한곳에 모아 보여 주는**** 창구입니다. 그런데 > 모든 데이터의 열쇠를 여기서 주지는 않습니다. > > ****활용신청**** 버튼이 있으면 이 포털의 열쇠 하나로 됩니다. ****제공처 바로가기****만 있으면 > 그 기관 사이트(예: 서울 열린데이터광장)에 ****따로 가입하고 따로 열쇠를**** 받아야 > 합니다. 계정이 하나 늘고, 기다림도 따로 생깁니다. > > 그래서 처음 하나는 ****활용신청** 버튼이 있고, 자동승인인 것 ****으로 고릅니다.** 버튼 > 하나만 봐도 일이 얼마나 늘지 알 수 있습니다.

25.5 활용신청하고 인증키 받기

상세 화면의 **활용신청**을 누르면 「**오픈 API 개발계정 신청**」 화면이 나옵니다. 위에서 아래로 내려가며 채웁니다. 우리가 손대는 곳은 **네 군데**입니다.

오픈 API 개발계정 신청

기본 정보

데이터명	JSON+XML 한국환경공단_에어코리아_대기오염정보
제공기관	한국환경공단
서비스 유형	REST
심의여부	자동승인 ①
신청유형	개발계정 활용신청
처리상태	신청
활용기간	승인일로부터 24개월 간 활용가능

활용목적 *

② ☒ 웹 사이트 개발

☐ 앱개발 (모바일,솔루션등)

☐ 기타

☐ 참고자료

☐ 연구(논문 등)

③ 활용목적에 대한 설명을 적어주세요. *

0 / 4000 bytes

그림 25-7 활용신청서 앞부분. ① 심의여부 — 자동승인이면 신청하자마자 승인된다 ② 활용목적 — 웹 사이트 개발을 고른다 ③ 활용목적에 대한 설명 — 한두 줄을 적는다(필수)

|| 칸 | 할 일 ||---|---|---|| — | **기본 정보** | 읽기만 한다. 심의여부(자동승인) · 활용기간 (승인일로부터 24개월)을 본다 || — | 위치기반서비스사업자 신고 안내 | 위치정보로 사업을 하는 사업자 이야기다. 가게 사이트에 미세먼지를 띄우는 데는 해당 없다 || 1 | **활용목적** | **웹 사이트 개발**을 고른다. 앱에 붙일 거면 앱개발 || 2 | **활용목적에 대한 설명** | 무엇에 쓰는지 한두 줄. **비우면 신청이 안 된다** || — | 첨부파일 | 비워 둔다 || — | **상세기능정보 선택** | 이미 전부 체크돼 있다. 그대로 둔다 || 3 | **라이선스 표시** | 이용허락범위 **동의합니다**에 체크. 안 하면 신청이 안 된다 || 4 | **활용 신청하기** | 누른다 |

상세기능정보 선택

1

상세기능	설명	일일 트래픽
☒ 대기질 예보통보 조회	통보코드와 통보시간으로 예보정보와 발생 원인 정보를 조회하는 대기질(미세먼지/오존) 예보통보 조회	500
☒ 초미세먼지 주간예보 조회	통보코드와 통보시간으로 대기질 전망과 주간예보 정보를 조회하는 초미세먼지 주간예보통보 조회	500
☒ 측정소별 실시간 측정정보 조회	측정소명과 측정데이터 기간(일,한달,3개월)으로 해당 측정소의 일반항목 측정정보를 제공하는 측정소별 실시간 측정정보조회	500
☒ 통합대기환경지수 나쁨 이상 측정소 목록조회	통합대기환경지수가 나쁨 등급 이상인 측정소명과 주소 목록 정보를 제공하는 통합대기환경지수 나쁨 이상 측정소 목록조회	500
☒ 시도별 실시간 측정정보	시도명을 검색조건으로 하여 시도별 측정소목록에 대한 일반 항목과 CAI최중 실시간 측정값과 지수	500

라이선스 표시

저작자표시-변경금지

이용허락범위

2

☐ 동의합니다

※ 이용허락범위에 동의하셔야 신청이 가능합니다.

취소하기

3

활용 신청하기

그림 25-8 활용신청서 뒷부분. ① 상세기능정보 — 전부 체크된 채로 둔다. 기능마다 일일 트래픽 500 ② 이용허락범위 동의합니다 — 체크해야 신청된다 ③ 활용 신청하기

2번 — 설명은 한두 줄이면 된다

거창하게 쓸 필요는 없습니다. **내 가게 사이트에 무엇을 보여 줄지** 한두 줄이면 됩니다. 뭐라고 적을지 막히면 Claude에게 PRD를 보여 주고 물어봅니다.

공공데이터포털에 「에어코리아 대기오염정보」 활용신청을 하려고 해.
docs/20.working/PRD.md 를 읽고, 우리 가게 사이트에 이 데이터를
어디에 쓰는지 활용목적을 두 줄로 써줘. 과장하지 말고.

상세기능 — 창구 하나에 주문서가 여러 장

상세기능정보 표를 보면 에어코리아 하나에 **기능이 다섯 개** 들어 있습니다. 대기질 예보, 초미세먼지 주간예보, 측정소별 실시간 측정정보... 창구는 하나인데 **주문서 양식이 여러 장**인 셈입니다. 어느 걸 쓸지는 다음 주에 Claude가 정합니다. 지금 빼 두면 그때 다시 신청해야 하니, **전부 체크된 그대로** 신청합니다.

일일 트래픽 500도 **기능마다 따로** 셉니다. 25.4절에서 본 「하루 500번」이 이것입니다.

신청한 뒤 — 승인 확인하고 인증키 보기

활용 신청하기를 누른 뒤 오른쪽 위 마이페이지로 가면 「활용신청 현황」이 나옵니다. 자동승인 데이터는 이미 승인으로 넘어가 있습니다.



그림 25-9 마이페이지의 활용신청 현황. ① 활용신청 현황 ② 활용 중 — 승인 1. 자동승인이라 신청하자마자 여기로 온다 ③ 신청한 데이터 — 이름 앞에 [승인] 이 붙었다. 만료예정일은 신청일로부터 2년 뒤

| 칸 | 뜻 | |---|---| **활용 신청** — 신청 · 보류 · 반려 | 아직 심사 중이거나 돌려받은 것. 심의승인 데이터는 1~2일 여기 머문다 | | **활용 중** — 승인 | 열쇠가 나왔다. **여기 1이 뜨면 끝** | | **만료 · 중지** | 기간(2년)이 지났거나 멈춘 것 |

인증키는 옆 칸 「인증키 발급현황」에 있습니다. 마이페이지 왼쪽 메뉴에서 데이터 활용 → Open API → 인증키 발급현황 순서로 들어갑니다.



그림 25-10 인증키 발급현황. ① 왼쪽 메뉴의 인증키 발급현황 ② 인증키 — 긴 영어·숫자 한 줄이 여기 나온다. 키 값은 가렸다 ③ 일반 인증키 재발급하기 — 키가 새어 나갔을 때 새로 받는 버튼

인증키는 **한 사람에게 하나**입니다. 데이터마다 따로 받는 게 아닙니다. 나중에 기상청 단기에보를 더 신청해도, 승인만 나면 **같은 키**로 부릅니다. 그래서 이 한 줄이 새어 나가면 내가 신청한 **모든 데이터의 문**이 같이 열립니다.

승인 뒤에 막히는 세 곳

막히는 곳	왜	이렇게
승인됐는데 데이터가 안 온다	승인 직후에는 최대 1시간쯤 키가 동작하지 않을 수 있다	기다린다. 당황하지 않는다
키를 넣었는데 틀렸다고 한다	옳게 적다가 한 글자가 빠지거나 앞뒤에 빈칸이 붙었다. 긴 영어·숫자라 눈으로는 못 찾는다	손으로 옳게 적지 않는다. 화면에서 복사해 붙인다
오후에 갑자기 안 온다	그날 호출 한도를 다 썼다	다음 날 다시. 만드는 동안 새로그침을 줄인다

💡 인증키는 내 이름으로 된 열쇠다

인증키는 **나에게 발급된** 열쇠입니다. 남이 가져다 쓰면 **내 한도가** 줄고, 문제가 생기면 책임도 내게 옵니다.

그래서 20장의 이야기가 여기서 또 나옵니다. 열쇠를 코드에 그대로 적어 GitHub에 올리면 **누구나 봅니다**. 공개 창고를 자동으로 훑어 열쇠를 모으는 프로그램도 있습니다. 다음 주에 Claude는 이 열쇠를 코드가 아니라 **.env 쪽지**(21장)에 적고, 인터넷에 올린 사이트에는 **올리는 서비스의 설정 칸**에 따로 넣습니다.

실수로 올렸다면 파일을 지우는 것으로는 모자랍니다. 이력에 남습니다. 인증키 발급현황의 **일반 인증키 재발급하기로 새 키를 받고**, 옛 키는 버립니다.

키를 받았으면 **메모장이나 카톡 나와의 채팅에 붙여 두지 않습니다**. 다음 주에 인증키 발급현황에서 다시 복사하면 됩니다. 열쇠가 머무는 곳은 적을수록 좋습니다.

25.6 무엇을 붙일까 — 활용 아이디어

이번 주 과제의 절반은 **무엇을 고르느냐**입니다. 수업에서 추천하는 데이터는 이렇습니다. 열쇠가 빨리 나오고, 돌아오는 글자 문치가 단순한 것 위주입니다.

추천	데이터	왜 좋은가	받는 곳
	미세먼지 (에어코리아)	바로 발급 · 지역별로 바로 와닿는다	공공데이터포털
	박스오피스 (영화진흥위원회 KOBIS)	바로 발급 · 응답이 단순하고 결과가 눈에 잘 보인다	KOBIS 자체 사이트
	기상청 단기예보	친숙하다. 다만 넣어야 할 조건이 좀 많다	공공데이터포털
	전국 문화축제	목록이라 화면 짜기 연습에 좋다	공공데이터포털
	서울 따릉이	실시간으로 바뀌는 게 크게 느껴진다	서울 열린데이터광장 (「제공처 바로가기」)

수업에서는 **하나를 다 같이** 붙여 본 뒤, 각자 원하는 데이터로 바꿔 봅니다.

내 가게에 붙이면

데이터는 **내 손님이 궁금해할 것**이어야 붙일 이유가 있습니다. 공방 예시로 한 번 대 봅니다.

데이터	공방 사이트·앱에서는	어느 화면에
미세먼지	「오늘 미세먼지 보통 — 창문 열고 수업합니다」 한 줄	웹 「오늘 클래스」 맨 위
기상청 단기예보	신청한 클래스 날 비 소식이 있으면 「우산 챙기세요」	앱 신청 확인
전국 문화축제	이번 주말 근처 축제 목록 — 축제 보러 온 김에 들르는 손님	웹 새 칸 「이번 주 동네 행사」
서울 따릉이	가게 근처 대여소와 남은 자전거 수	웹 「오시는 길」 (있다면)

업종이 다르면 짝도 다릅니다.

가게	붙일 만한 것
꽃집	단기예보 — 배송 날 비·강풍이면 미리 연락
요가원 · 필라테스	미세먼지 — 야외 수업을 할지 말지
카페	박스오피스 — 「요즘 극장가 1위」 한 줄로 말 걸기
네일 · 미용실	문화축제 — 행사 날 예약이 몰리는 걸 미리 알기

위 표는 **예시**입니다. 내 PRD를 보여 주고 Claude에게 짝을 찾게 하면 더 잘 맞습니다.

docs/20.working/PRD.md 를 읽고, 우리 가게 웹이나 앱에 붙이면 손님한테
도움이 될 공공데이터를 세 개 추천해줘. 공공데이터포털(data.go.kr)에서
「활용신청」 버튼이 있고 자동승인인 것 위주로.
각각 어느 화면에 어떤 한 줄로 보일지도 같이. 아직 만들지는 마.

고르는 기준 세 가지

기준	물어볼 것
손님	내 손님이 이걸 보고 "오, 편하다" 할까?
버튼	상세 화면에 활용신청 버튼이 있고, 심의유형이 자동승인 인가?
자리	화면 어디에 , 어떤 한 줄로 보일지 말할 수 있나?

세 번째가 제일 중요합니다. 자리를 못 대면 붙여 놓고도 **어디에 둘지 몰라 헤맬니다**. 시안 없이 화면을 만들면 두 번 세 번 고친다는 것, 13장에서 이미 겪었습니다.

고른 것은 PRD에 먼저 적는다

정했으면 Claude에게 PRD에 적게 합니다. 만들기는 다음 주입니다. 오늘은 **적기만**.

공공데이터로 「에어코리아 대기오염정보」를 쓰기로 했어. 활용신청도 넣었어.
PRD 8장 「아직 안 정한 것」에 적어줘 - 무슨 데이터, 어느 화면에 어떤 한 줄,
승인 상태. 다음 주에 붙일 거라고. 변경 이력에도 한 줄.

다음 주의 Claude는 PRD를 읽고 시작합니다. 여기 적혀 있으면 "무엇을, 어디에"를 다시 설명하지 않아도 됩니다. 그리고 **왜 이 데이터를 골랐는지**가 남습니다. 몇 달 뒤 누가 "이거 왜 붙였어요?" 하고 물어도 문서가 답합니다.

25.7 다음 주 전에 — 확인

확인	됐으면
data.go.kr 에 네이버 로그인(또는 아이디)으로 들어가진다	☐
쓸 데이터를 하나 정했다 (오픈API · 활용신청 버튼 있음)	☐
활용신청을 넣었다	☐
마이페이지에서 승인 상태를 봤다 (심의 중이면 그대로 두고 온다)	☐
PRD 8장에 무슨 데이터 · 어느 화면 · 어떤 한 줄을 적었다	☐

승인이 수업 전까지 안 나와도 괜찮습니다. 수업 전에 카톡방으로 알려 주세요. 강사가 준비해 둔 예비 키와 예시 데이터로 같이 진행합니다.

25.8 정리

- 지금 화면은 내가 적은 것만 보여 준다. 다음 주에 밖에서 흘러 들어오는 데이터를 붙인다
- 공공데이터포털(data.go.kr) — 공공 데이터를 무료로 받아 쓰는 창구. 가입 → 활용신청 → 인증키
- API 는 주문서, JSON 은 돌아오는 글자 문치. 읽는 법만 알면 된다
- 들어갈 때는 간편 로그인 → 네이버 로그인. 새 아이디·비밀번호를 만들지 않는다
- 고를 때는 오픈API 칸에서. 상세 화면의 비용 · 심의유형 · 트래픽 세 줄을 본다
- 활용신청 버튼이 있는 것부터. 제공처 바로가기는 가입을 한 번 더 부른다
- 심사형은 1~2일. 그래서 이번 주에 신청한다
- 인증키는 내 이름의 열쇠. 코드에 적지 않는다
- 고른 데이터는 PRD에 먼저 적는다 — 무엇을, 어디에, 왜

이 장에서 나온 말

말	쉬운 뜻
공공데이터포털	정부·공공기관 데이터를 무료로 모아 둔 창구. <code>data.go.kr</code>
간편 로그인	네이버·카카오 계정으로 들어가는 방식. 20장의 소셜 로그인과 같다
API	다른 곳의 데이터를 정해진 방식으로 받아오는 창구. 주문서
JSON	데이터를 주고받는 표준 글자 형식. <code>"이름": "값"</code>
오픈API · 파일 데이터	부를 때마다 최신이 오는 것 · 표 파일을 통째로 받는 것
활용신청	이 데이터를 쓰겠다고 신청하는 것. 승인되면 인증키가 나온다
인증키	데이터를 부를 때 내미는 내 이름의 열쇠. 한 사람에 하나, 「인증키 발급현황」에 있다
심의유형	신청하면 바로 되나(자동승인), 심사를 거치나(심의승인)
트래픽	하루에 부를 수 있는 횟수 한도
제공처 바로가기	그 데이터는 이 포털이 아니라 기관 사이트에서 따로 신청하라는 표시

다음 주에는 이 열쇠로 진짜 데이터를 내 화면에 띄우고, 다시 인터넷에 올립니다.

25.9 안 될 때

증상	이렇게
로그인을 눌렀더니 「로그인 방식을 선택해주세요」 카드 화면이 뜬다	카드 중 민간ID (네이버, 카카오, 페이코ID로 로그인)를 누르면 네이버로 들어갈 수 있습니다
네이버 로그인 창이 안 뜬다	팝업이 막힌 겁니다. 주소창 오른쪽의 팝업 차단 표시를 눌러 허용하세요
네이버를 안 쓴다	바로 아래 카카오 로그인 도 같은 순서입니다
예전에 아이디로 가입해 두었다	왼쪽 아이디 칸으로 들어가도 됩니다
검색 결과가 너무 많다	결과에서 오픈API 칸만 봅니다. 제공기관 이름(한국환경공단, 기상청)을 같이 적어 검색합니다
활용신청 버튼이 없다	「제공처 바로가기」 데이터이거나 파일 데이터입니다. 활용신청 버튼이 있는 다른 데이터를 고르세요
심의승인이라 아직 승인 대기다	정상입니다. 1~2일 걸립니다. 수업 전까지 안 나오면 카톡방으로 알려 주세요
월 골라야 할지 모르겠다	 미세먼지(에어코리아) 로 하세요. 추천 1순위이고 공공데이터포털에서 바로 발급됩니다. 그래도 모르겠으면 카톡방에 물어보세요
화면이 뭘 말하는지 모르겠다	Claude에게 화면을 캡처해서 붙이고 이 화면에서 뭘 봐야 돼?