

내 파일과 인터넷 주소를 열기

시연 1분 → 직접 실습 2분 · 완료 증거를 기록합니다.

어디에서 무료 배포 시작 키트 · 브라우저

- 1 ZIP을 풀고 public/index.html을 엽니다.
- 2 같은 예제를 공개 HTTPS 주소로 엽니다.
- 3 주소창에서 file과 https의 차이를 찾습니다.

막히면 파일을 열기 어렵다면 공개 예제로 관찰하고 로컬 준비는 미완료로 기록합니다.

완료 증거

내 파일과 공개 예제

완료하면 채팅에 ✓

화면 뒤의 요청 세 가지 찾기

시연 2분 → 직접 실습 5분 · 완료 증거를 기록합니다.

어디에서 공개 예제 · Network 탭

- 1 개발자 도구의 Network를 열고 새로고침합니다.
- 2 문서·styles.css·app.js의 요청을 찾습니다.
- 3 쿼즈를 누르고 추가 서버 요청이 있는지 봅니다.

첫 운영 작업 · 10분

HTML·CSS·JS 요청

프롬프트 01 · 복사해서 시작

막히면 파일이 캐시에서 오면 표시를 기록합니다. 파일을 더블클릭한 창 대신 HTTPS 예제를 씁니다.

웹은 요청과 응답으로 화면을 만듭니다

브라우저는 주소의 서버에 요청하고, 받은 HTML·CSS·JS를 해석해 화면과 동작을 만듭니다.



실전 요청·적용

DNS로 도메인의 연결 정보를 찾고 HTTPS로 통신합니다. 쿼즈 채점은 받은 JS가 브라우저에서 합니다.

잠깐 확인

버튼이 작동하면 반드시 서버 앱일까요?

실습으로 →

내 파일·localhost·공개 주소를 구분합니다.

file·localhost·HTTPS 주소 구분

시연 2분 → 직접 실습 4분 · 완료 증거를 기록합니다.

어디에서 주소창 · 웹 동작 매뉴얼

- 1 file은 내 파일, localhost는 이 기기라 적습니다.
- 2 공개 URL의 프로토콜·호스트·경로를 나눕니다.
- 3 짝의 휴대폰에 전달할 주소를 고릅니다.

막히면 localhost를 휴대폰에 입력하면 PC가 아니라 휴대폰 자신을 가리킵니다.

완료 증거

세 주소의 뜻

프롬프트 01 · 복사해서 시작

HTML·CSS·JS로 내 안내 만들기

시연 2분 → 직접 실습 10분 · 완료 증거를 기록합니다.

어디에서 텍스트 편집기 · public 폴더

- 1 HTML 구조·CSS 표현·JS 동작을 찾아봅니다.
- 2 제목·소개 문장만 바꾸고 저장합니다.
- 3 파일을 다시 열어 퀴즈와 자료 링크를 봅니다.

막히면 편집이 막히면 원본으로 배포합니다. .html.txt로 저장하지 않았는지 확인합니다.

완료 증거

내 수업명 v1 화면

프롬프트 02 · 복사해서 시작

배포·호스팅·도메인은 다릅니다

배포는 특정 버전의 파일·프로그램을 서버에 반영해 다른 기기에서 접근할 수 있게 하는 과정입니다.

01

배포

버전을 올리고 확인

02

호스팅

파일 제공·프로그램 실행

03

도메인

서비스를 찾는 이름

실전 요청·적용

이 강의의 `snu-ai-workshop.pages.dev`는 Cloudflare 기본 주소입니다. `.dev` 자체는 배포 서비스가 아닙니다.

잠깐 확인

배포하려면 도메인을 꼭 사야 하나요?

실습으로 →

사례에 맞는 무료 서비스 하나를 고릅니다.

무료 서비스 다섯 가지 비교

시연 2분 → 직접 실습 6분 · 완료 증거를 기록합니다.

서비스	이런 사례에	무료 조건 · 먼저 확인
Cloudflare Pages	강의 자료실·시각화	월 500 Git 빌드 · 함수 별도
Netlify	정적 도구·서버 함수	월 300 credits · 배포도 차감
Vercel	개인 Next.js 데모	Hobby: 개인·비상업 용도
GitHub Pages	공개 수업 문서	Free: public 저장소 · 정적만
Render	Node·Python 서버 데모	15분 휴면 · production용 아님

확인일 2026-10-27 · AI API·DB·도메인 비용 별도

실습으로 → Cloudflare Pages 직접 업로드 · 전체 조건·공식 링크는 매뉴얼

내 강의 자료를 웹 주소로 공유

시연 2분 → 직접 실습 9분 · 완료 증거를 기록합니다.

어디에서

Cloudflare Pages · public 폴더 · 짝의 브라우저

완료 화면

짝이 여는 HTTPS 주소

01

public 파일을 Pages에 직접 업로드합니다.

02

제공된 HTTPS 주소에서 제목·퀴즈를 봅니다.

03

짝에게 URL을 보내 자료 열기·다운로드를 봅니다.

막히면 계정·배포가 막히면 제공 공개 예제에서 검사하고 본인 배포는 미완료로 남깁니다.

404·깨진 화면·이전 버전 진단

시연 0분 → 직접 실습 5분 · 완료 증거를 기록합니다.

어디에서

주소창 · Network · 배포 대시보드

- 1 실제 URL과 기대·실제 화면을 기록합니다.
- 2 경로·대소문자·업로드 폴더를 확인합니다.
- 3 한 가지를 고쳐 같은 주소에서 재검사합니다.

막히면 오류가 없으면 없는 파일 주소에서 404를 확인하고 원래 페이지로 돌아옵니다.

완료 증거

**원인과 재검사
기록**

프롬프트 06 · 복사해서 시작

주소와 다음 행동을 남기고 휴식

작업 상태를 파일로 남기고 70분에 다시 시작합니다.

어디에서

DEPLOYMENT.md · 작업 폴더

10

분 쉬어가기

1. 현재 파일과 배포 URL을 저장합니다.
2. 배포 성공·미완료와 다음 행동을 적습니다.
3. 작업 창을 남겨두고 70분에 돌아옵니다.

돌아오면: 현재 상태와 다음 행동

막히면 사이트가 안 열리면 마지막 성공 단계와 오류를 남깁니다.

프로그램 배포와 데이터 공유는 별개

같은 화면을 함께 열어도 개인 메모는 각 브라우저에 저장될 수 있습니다. 저장 위치를 따로 설계합니다.

01

정적 파일

모두 같은 화면·퀴즈

02

브라우저 저장

출처·브라우저별 메모

03

공유 저장소

서버·DB·접근 권한

실전 요청·적용

강의 안내는 정적 파일, 개인 메모는 localStorage, 공동 응답 수집은 공유 저장소가 필요합니다.

잠깐 확인

같은 URL인데 짝의 메모가 안 보이면 오류인가요?

실습으로 →

같은 HTTPS 주소를 일반·시크릿 창에서 봅니다.

같은 웹 도구의 저장 범위 확인

시연 1분 → 직접 실습 7분 · 완료 증거를 기록합니다.

어디에서 같은 운영 HTTPS 주소 · 일반 창과 시크릿 창

- 1 일반 창에서 메모 저장·새로고침을 합니다.
- 2 시크릿 창에서 같은 주소의 메모를 비교합니다.
- 3 메모 내보내기 파일과 원문을 대조합니다.

완료 증거

**같은 화면·다른
메모**

프롬프트 10 · 복사해서 시작

막히면 본인 배포가 없으면 제공 공개 HTTPS 예제로 검증하고 저장 차단 시 파일로 내보냅니다.

다음 수정의 배포 경로 설계

시연 1분 → 직접 실습 11분 · 완료 증거를 기록합니다.

어디에서 내 파일 · 직접 업로드와 Git 연결

- 1 HTML 파일·빌드 결과·서버 실행을 구분합니다.
- 2 직접 업로드와 Git 자동 배포를 비교합니다.
- 3 내 프로젝트의 수정→확인→배포 순서를 적습니다.

완료 증거

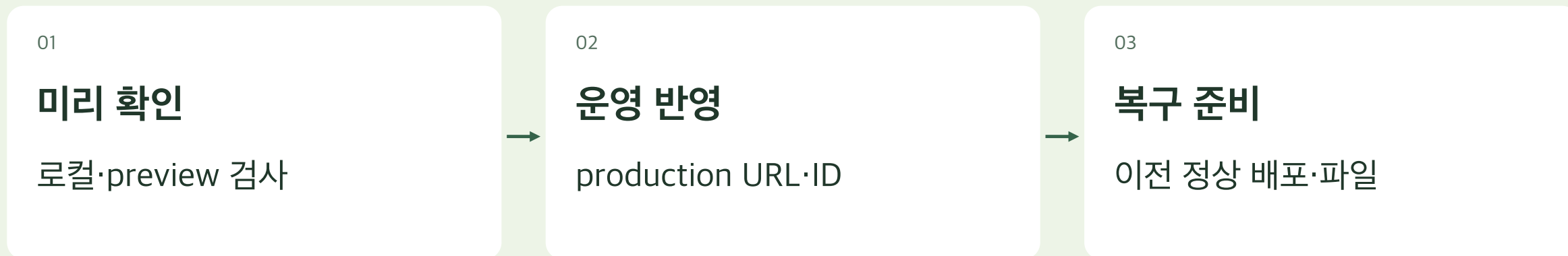
**다시 올릴 대상
폴더**

프롬프트 05 · 복사해서 시작

막히면 이번 시작 키트는 빌드 없이 public 내용을 올립니다. Git 연결은 선택 후속 활동입니다.

수정·확인·배포·복구를 한 묶음으로

로컬 수정만으로 공개 사이트가 바뀌지 않습니다. 어떤 버전을 어디에 올렸는지 기록하고 확인합니다.



실전 요청·적용

직접 업로드는 수정 파일을 다시 올리고, Git 연결은 push로 빌드·배포합니다. 운영 URL에서 v2를 확인합니다.

잠깐 확인

이전 배포로 돌아가면 메모도 돌아갈까요?

실습으로 →

제목만 v2로 바꾸고 재배포·복구를 기록합니다.

수업 도구 v2 배포·복구

시연 2분 → 직접 실습 4분 · 완료 증거를 기록합니다.

어디에서 동일 Pages 프로젝트 · Deployments · 운영 URL

- 1 정상 배포를 적고 화면 제목만 v2로 고칩니다.
- 2 같은 프로젝트에 올려 운영 URL을 확인합니다.
- 3 이전 성공 배포의 복구 경로를 기록합니다.

완료 증거

v2 확인과 복구 기록

프롬프트 11 · 복사해서 시작

막히면 권한·시간이 부족하면 로컬 v2와 복구 경로만 기록합니다. 보관한 정상 public을 재업로드할 수도 있습니다.

링크·사용량·검증 결과 기록

시연 1분 → 직접 실습 7분 · 완료 증거를 기록합니다.

어디에서

운영 URL · DEPLOYMENT.md

완료 화면

배포와 검사 근거

01

제목·퀴즈·자료 다운로드를
다시 확인합니다.

02

플랜·사용량·공개 범위를
대시보드에서 봅니다.

03

실제 검사·미검증·복구
방법을 저장합니다.

막히면

호스팅 무료와 AI API·DB·도메인 비용은 별개입니다. 미검증을 통과로 쓰지 않습니다.

짹의 휴대폰에서 최종 확인

시연 0분 → 직접 실습 5분 · 완료 증거를 기록합니다.

어디에서 휴대폰 브라우저 · 운영 HTTPS 주소

- 1 짹이 로그인 없이 운영 URL을 열어봅니다.
- 2 좁은 화면의 퀴즈·자료 링크를 확인합니다.
- 3 선택: 홈 화면 바로가기로 다시 엽니다.

막히면 접근이 막히면 공개 설정을 봅니다. 홈 화면 아이콘은 네이티브 앱·동기화가 아닙니다.

완료 증거

**다른 기기의 열람
증거**

완료하면 채팅에 ✓

결과를 저장하고 다음 도구 선택

시연 0분 → 직접 실습 3분 · 완료 증거를 기록합니다.

어디에서

결과물 · 인계 파일 · 추천 도구

완료 화면

검토한 결과물 + 다음 도구 하나

01

Claude 웹: 변경 범위를
명확히 요청

02

Chrome DevTools: 오류를
좋은 질문으로 변환

03

VS Code 비교: AI 수정 전후
대조

마치면

결과물·검토·다음 행동을 먼저 저장합니다. 도구 설치·가입은 선택입니다.