

주소로 복습하는 홈페이지의 구조

chemion.kr

제 홈페이지의 도메인

chemion-lab.chemsty-c23.workers.dev

원래 주소

Cloudflare가 제공한 기본 주소

te.chemion.kr

하위 도메인(ai수업용)

서버 Server	요청한 페이지·데이터를 보내 주는 컴퓨터·프로그램
호스팅 Hosting	파일이나 앱을 인터넷에서 제공하는 서비스
도메인 Domain	홈페이지 주소에 사용하는 이름
URL	접속 방식·도메인·경로 등을 포함한 위치 주소

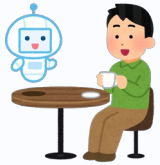
URL 예: https://te.chemion.kr/p?day=3#slide=1

2일차(3,4차시)

원하는 화면을 설명하고
실무 용어로 완성합니다

레퍼런스와 화면 설계부터 구현 · 최적화 · 배포까지





제작 흐름과 진행 보고

기획 Planning	무엇을 누구를 위해 만들지 정하는 일
구현 Implement	정한 화면·기능을 코드로 만드는 일
테스트 Test	기대 동작과 실제 결과를 비교하는 확인
진행 보고 Status	완료·진행 중·막힌 일을 구분해 알리는 것

AI에게 이렇게 부탁합니다

“먼저 계획을 세우고, 승인 후 한 메뉴씩 구현해 줘.”

“진행은 완료·진행 중·확인 필요로 나눠 알려 줘.”

“완료했다고 할 때는 실제 확인한 결과도 보여 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

진행 보고 예시

완료: 메뉴 구조

진행 중: 프로그램 목록

확인 필요: 기관 운영자료

예시 보고 · 단계별로 구분

실제 사례 · 남해타임의 구현·검증·공개 구분 ↗

확인: 계획·구현·검증·공개를 구분해서 보고하는가?



조사·분석·검토·검증 요청법

조사 Research	공식 자료·사례를 찾아 출처와 사실을 모으기
분석 Analysis	모은 자료의 차이·관계·원인을 해석하기
검토 Review	현재 결과의 문제·영향·우선순위를 살피기
검증 Verify	기준과 실제 자료·동작을 비교해 확인하기

AI에게 이렇게 부탁합니다

“수업 자료에 쓸 공식 출처와 확인 날짜를 조사해 줘.”

“평가 기준과 문항이 맞는지 검토해 줘.”

“사실 오류와 학생 개인정보가 없는지 검증해 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

검토 결과 예시

검토: “더보기”가 모호함

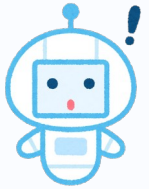
기준: 목적이 보이는 이름

프로그램 보기

검증: 실제 이동 확인 전

실제 사례 · 재단 홈페이지의 “확인 중” 안내 ↗

확인: 자료 수집·판단·수정·확인을 각각 요청했는가?



버전 번호와 알파·베타·RC

버전 번호 1.2.3	팀 규칙 예: 큰 변경·기능 추가·오류 수정
알파 Alpha / 베타 Beta	초기 내부 시험 / 기능을 갖춘 넓은 시험 단계
RC / 정식 Release	출시 후보 / 검수·승인 후 공개하는 기준 버전
커밋 Commit / 태그 Tag	변경 기록 / 특정 버전에 붙이는 기준 이름

AI에게 이렇게 부탁합니다

“0.1.0-alpha.1, 0.9.0-beta.1, 1.0.0-rc.1 순서로 기록해 줘.”

“정식 1.0.0 뒤 오타 수정은 1.0.1, 검색 추가는 1.1.0으로 해 줘.”

“큰 변경은 2.0.0 후보로 검토하고, 수정 이력과 승인 기록을 남겨 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

릴리스 기록 예시

- 0.1.0-alpha.1 · 내부 시험
- 0.9.0-beta.1 · 확대 시험
- 1.0.0-rc.1 · 공개 전 검수
- 1.0.0 · 승인 후 정식판

각 기록을 눌러 보세요.

실제 사례 · 남해타임의 앱 버전·웹 배포 기록 ↗

확인: 화면의 버전·태그·배포 기록이 같은 결과물을 가리키는가?



프로젝트 폴더와 파일 관리

프로젝트 루트 Root	한 홈페이지 작업의 기준이 되는 폴더
src/ · public/	소스 코드 / 공개 파일. 프로젝트 방식에 따라 다름
assets/ · docs/	그림 등 자료 / 기획·운영·인수인계 문서
dist/ · 별도 백업	빌드 결과 / 프로젝트 밖에 보관하는 복구 사본

AI에게 이렇게 부탁합니다

“남해고등학교-홈페이지/ 안에서만 작업하고 기존 구조부터 알려 줘.”

“원본·코드·공개 파일·산출물·백업을 구분해 줘.”

“.env·토큰·신청자 자료는 공개·업로드·Git 추적에서 제외해 줘.”

실제 사례 · AI 연수의 public·작업 폴더 분리 ↗

확인: 작업 폴더와 공개할 폴더, 복구용 사본을 구별하는가?

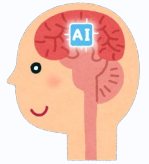
교육용 예시 · 눌러서 직접 확인 ↗

폴더 구성 예시

꿈나눔센터–홈페이지/

```
├ src/      소스
├ public/   공개 파일
├ assets/   원본 자료
└ docs/     작업 안내
```

백업은 프로젝트 밖에 따로



Plus 사용량을 아끼는 수업 준비 습관

Plus 구독	모델명이 아닌 이용 요금제. API 비용과 별도로 봅니다
토큰 Token	요청·파일·대화 이력·응답을 처리하는 정보 단위
문맥 Context	한 번에 참고하는 정보. 남은 구독 한도와 구별
사용량·회복 시각	계정 화면의 한도와 리셋 시각을 확인하는 기준

교사의 실무 사용 습관

① 목표·수정 범위·완료 기준을 묶고, 필요한 파일과 화면만 제공합니다.
② 초안 하나를 확인한 뒤 고칩니다. 전체 재생성과 긴 출력 반복을 줄입니다.

③ 단순 수정은 가벼운 선택지로, 복잡한 판단은 필요한 만큼 추론합니다.
④ 새 대화에는 요약을 넘깁니다. 새 대화가 사용량 한도를 리셋하지는 않습니다.

⑤ 남은 한도·회복 시각을 확인합니다. 추가 결제와 API 전환은 직접 결정합니다.

실제 사례 · 남해타임의 작은 범위 수정 요청 ↗

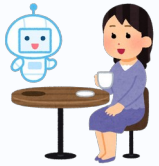
확인: 한도는 계정 화면 기준. 고정 월 토큰량·새 대화 무제한으로 설명하지 않습니다.

교육용 예시 · 눌러서 직접 확인 ↗

범위를 좁힌 요청 예시

대상: 프로그램 카드 1개
수정: 버튼 이름만 변경
유지: 다른 화면·데이터
확인: 휴대폰 화면
답변: 변경점 3줄

계정 한도는 본인 화면에서 확인



GDWEB에서 레퍼런스 가져오기

레퍼런스 Reference	원하는 분위기와 구성을 설명할 참고 사례
벤치마킹 Benchmarking	좋은 구성의 이유를 비교해 우리 업무에 적용
와이어프레임	색·사진보다 정보 배치에 집중한 화면 초안

AI에게 이렇게 부탁합니다

“GDWEB에서 학교·교육 분야의 참고 사이트를 골라 줘.”

“URL과 마음에 드는 부분, 적용할 곳을 정리해 줘.”

“그대로 복제하지 말고 학교 와이어프레임으로 바꿔 줘.”

실제 사례 · 이라스토야 자료실의 출처·해시 기록 ↗

확인: 참고 URL과 적용할 요소, 제외할 요소가 기록됐는가?

교육용 예시 · 눌러서 직접 확인 ↗

레퍼런스 기록 예시

GDWEB 원문 열기 ↗

적용: 메뉴·정보 배치

제외: 사진·로고 복제

URL + 적용할 부분을 기록



수업·학급 도구의 메뉴와 페이지

정보 구조 IA 센터 정보를 목적별로 묶는 설계

사이트맵 Sitemap 메뉴와 하위 페이지를 정리한 지도

GNB 사이트 전체에서 쓰는 주요 메뉴 묶음

섹션 Section 한 페이지에서 주제별로 나눈 구역

AI에게 이렇게 부탁합니다

“소개·시설·프로그램·소식·오시는 길로 묶어 줘.”

“상단 GNB와 페이지 안의 섹션을 구분해 줘.”

“학생이 동아리 신청 정보를 찾는 동선부터 확인해 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

메뉴와 섹션 예시

센터소개 | 프로그램 | 소식

센터소개 섹션

공간 소개와 이용 안내

실제 사례 · 재단 홈페이지의 상위·하위 메뉴 ↗

확인: 같은 안내를 여러 메뉴에 중복해서 넣지 않았는가?

메인 화면과 바로가기

히어로 Hero	처음 방문했을 때 보이는 큰 소개 영역
배너 Banner	모집·행사 등 소식을 강조하는 영역
CTA	프로그램 보기처럼 다음 행동을 안내하는 요소

AI에게 이렇게 부탁합니다

“히어로에는 학교의 역할을 한 문장으로 넣어 줘.”

“프로그램 보기와 시설 안내를 바로가기로 만들어 줘.”

“배너는 확인된 모집 내용만 보여 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · 학교 활동 홈페이지의 첫 화면·바로가기 ↗

확인: 처음 온 사람이 원하는 정보를 바로 찾을 수 있는가?



“고유의 색을 넣어줘”

브랜드 컬러	학교의 정체성을 나타내는 기준 색
컬러 팔레트	화면에서 함께 사용할 색의 조합
포인트 컬러	중요 버튼과 강조에 제한해서 쓰는 색
대비 Contrast	글자와 배경 등을 구별하는 밝기·색 차이

AI에게 이렇게 부탁합니다

“센터의 고유의 색을 넣어줘. 먼저 색상안을 보여 줘.”

“주색·보조색·포인트색의 역할을 정해 줘.”

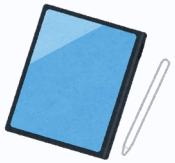
“모집 상태는 색과 글자를 함께 써 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · 재단 홈페이지의 메뉴별 고유색 ↗

확인: 같은 의미에 같은 색을 쓰고 글자가 잘 읽히는가?



배치·여백·글자 크기

그리드 Grid	카드를 행과 열로 정돈하는 배치
패딩 Padding	카드 테두리와 내용 사이의 안쪽 여백
갭 Gap	카드나 버튼 등 항목 사이의 간격
타이포그래피	제목·본문·보조 글자의 크기와 굵기 체계

AI에게 이렇게 부탁합니다

“PC 두 열·휴대폰 한 열로 카드 그리드를 만들어 줘.”

“카드 안쪽 패딩과 카드 사이 갭을 넓혀 줘.”

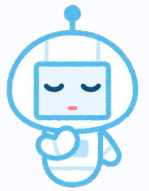
“제목·본문 크기와 행간을 전체에 맞춰 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · 학교 활동 홈페이지의 모바일 한 열 배치 ↗

확인: 긴 제목과 글자 확대에서도 겹침이나 잘림이 없는가?



“칩으로 만들어줘”

칩 Chip	짧은 항목을 선택·필터 등에 쓰는 작은 요소
배지 Badge	모집 중 같은 상태·개수를 알려 주는 표시
카드 Card	프로그램 사진·설명·행동을 묶은 콘텐츠 단위

AI에게 이렇게 부탁합니다

“프로그램 대상 필터를 칩으로 만들어줘.”

“선택된 칩은 색과 체크 표시로 구분해 줘.”

“모집 상태는 클릭하지 않는 배지로 보여 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · 남해타임의 과목 칩·학점 표시 ↗

확인: 칩을 눌렀을 때 할 일과 선택 해제 방법이 분명한가?



드롭다운·단추·선택 메뉴

드롭다운 Dropdown	눌러서 아래로 펼쳐 보이는 목록 형태
선택 메뉴 Select	항목 중 값을 고르는 입력 요소
단추 메뉴 Menu button	버튼을 눌러 동작·이동 목록을 여는 구성
탭 Tab	같은 영역에서 관련 내용을 바꿔 보는 구성

AI에게 이렇게 부탁합니다

“대상이 많으면 드롭다운 선택 메뉴로 보여 줘.”

“공유·인쇄는 더보기 단추 메뉴에 모아 줘.”

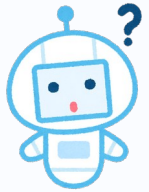
“소개·이용 안내는 탭으로 나눠 보여 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · 재단 홈페이지의 하위 메뉴 열기 ↗

확인: 값 선택·동작 실행·내용 전환을 서로 구별했는가?



하나 선택·여러 선택·켜고 끄기

라디오 Radio	그룹 안에서 한 가지 값을 고르는 입력
체크박스 Checkbox	여러 항목을 각각 선택하거나 해제하는 입력
토글 Toggle	하나의 설정을 켜고 끄는 전환

AI에게 이렇게 부탁합니다

“참가 구분은 하나만 고르는 라디오로 만들어 줘.”

“관심 분야는 여러 개 고르는 체크박스로 해 줘.”

“알림 설정을 켜고 끌 때는 토글을 써 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · AI 연수 설문의 라디오·체크박스 ↗

확인: 몇 개를 고르는지와 현재 상태를 사용자가 아는가?



학교·교과·교실 안내 화면

콘텐츠 Content	학교 소개·이용 안내에 쓸 글과 사진
상세 페이지	시설 한 곳의 정보를 자세히 보여 주는 화면
외부 링크	공식 지도 등 다른 사이트로 이동하는 연결

AI에게 이렇게 부탁합니다

“학교 소개와 이용 안내를 섹션으로 묶어 줘.”

“시설 카드는 상세 페이지로 연결해 줘.”

“오시는 길은 확인된 주소와 공식 링크만 써 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

시설 상세 예시

다목적실 (예시)

운영시간: 운영자료 확인 필요

대관 여부: 운영자료 확인 필요

시설 상세 보기

주소·이용 규칙은 확인 후 표시

실제 사례 · NAMGO 360의 공간 이동·안내 ↗

확인: 시설 소개와 교실 예약 가능 여부를 구분했는가?



동아리 목록과 신청 조건

카테고리 Category 대상·활동 분야 등 프로그램의 분류

필터 Filter 조건에 맞는 프로그램만 골라 보기

정렬 Sort 마감일 등 기준으로 목록 순서를 바꾸기

정원 Capacity 학교가 정한 참가 가능 인원

AI에게 이렇게 부탁합니다

“교과·동아리별로 분류하고 학년 칩으로 필터해 줘.”

“신청 마감일순 정렬과 결과 없음 안내를 넣어 줘.”

“신청 기간·활동일·정원·대상 학년을 구분해 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · 남해타임 과목 보관함의 검색·필터 ↗

확인: 조건을 바꾸거나 초기화해도 올바른 목록이 보이는가?

공지·갤러리·자주 묻는 질문

게시판 Board	공지의 목록과 상세 글을 관리하는 구조
썸네일 Thumbnail	사진 목록에서 보이는 작은 대표 이미지
아코디언 Accordion	질문 제목을 눌러 답을 펼치고 접는 구성

AI에게 이렇게 부탁합니다

“공지는 검색·상단 고정·페이지 이동을 넣어 줘.”

“갤러리는 썸네일과 촬영 시점을 함께 보여 줘.”

“FAQ는 아코디언으로 묶고 문의 동선을 남겨 줘.”

실제 사례 · 재단 홈페이지의 FAQ 아코디언 ↗

확인: 지난 공지와 사진의 공개 범위를 확인했는가?

교육용 예시 · 눌러서 직접 확인 ↗





동아리 신청폼과 특별실 예약

필드 Field	대상·시간처럼 이름을 붙인 정보 항목
유효성 검사	필수 항목 누락·형식 오류를 확인하는 처리
시간 슬롯 Slot	시설을 선택할 수 있도록 나눈 시간 단위
승인 대기	담당자가 검토하기 전의 신청 상태

AI에게 이렇게 부탁합니다

“신청폼은 승인할 항목을 정한 뒤 만들어 줘.”

“시설·시간 선택과 검토 화면을 순서대로 보여 줘.”

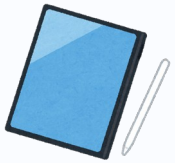
“신청 완료와 예약 확정을 구분해 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · 센터 신청 시제품의 필수값·서버 검증 ↗

확인: 시연 결과를 실제 신청·예약으로 오해하지 않는가?



관리자 화면과 데이터 구조

CMS	공지·프로그램 등 내용을 관리하는 시스템
데이터베이스 DB	공지·신청 등 구조화된 자료의 저장소
API	화면이 서버와 요청·응답을 주고받는 연결 규칙
임시 저장 Draft	아직 공개하지 않은 편집 상태

AI에게 이렇게 부탁합니다

“공지 작성·임시 저장·미리보기·게시를 나눠 줘.”

“화면과 서버, 데이터 저장 역할을 설명해 줘.”

“운영 DB 대신 가상 데이터로 시연해 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

CMS 임시 저장 예시

공지 제목

센터 소식 (예시)

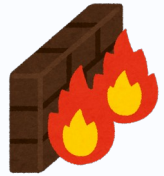
임시 저장 | 미리보기 | 게시 시연

상태: draft · 공개 전

메모리에서만 시연 · DB 연결 없음

실제 사례 · 센터 관리자 시제품의 처리 상태 ↗

확인: 저장·게시·배포가 서로 다른 처리임을 구분했는가?



학생 기록을 보호하는 권한 설계

인증 Authentication	접속한 사람이 누구인지 확인
인가 Authorization	그 작업을 할 권한이 있는지 확인
역할 Role	공지 담당·신청 담당처럼 권한을 묶은 구분
보유 기간	학교가 목적과 기준에 따라 정한 정보 보관 기간

AI에게 이렇게 부탁합니다

“공지 게시와 신청 정보 조회 권한을 나눠 줘.”

“관리자 버튼 숨김 외에 서버에서도 권한을 확인해 줘.”

“수집 항목과 보관·삭제 기준은 학교 승인을 받아 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

역할에 따른 권한 예시

공지 담당 | 신청 담당

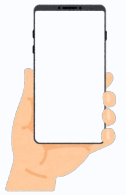
공지 편집: 허용

신청 명단: 접근 불가

표시 예시일 뿐, 실제 서버 인가 필요

실제 사례 · 센터 시제품의 서버 관리자 검사 ↗

확인: 공개 페이지·파일·API에 신청자 정보가 없는가?



모바일 대응과 최적화

반응형 Responsive	화면 폭에 맞춰 배치를 바꾸는 설계
최적화 Optimization	동작을 유지하며 속도·용량 등을 개선하는 일
이미지 압축	필요한 화질을 유지하며 파일 용량 줄이기
지연 로딩 Lazy loading	바로 필요하지 않은 자원을 나중에 불러오기

AI에게 이렇게 부탁합니다

“휴대폰에서 메뉴와 동아리 목록이 잘 보이게 해 줘.”

“사진 용량과 첫 화면 로딩을 측정하고 최적화해 줘.”

“수정 전후의 차이와 유지한 기능을 알려 줘.”

교육용 예시 · 눌러서 직접 확인 ↗



실제 사례 · NAMGO 360의 최대 4096px 사진 변환 ↗

확인: 같은 조건에서 개선을 확인하고 화면·동작을 유지했는가?



미리보기·빌드·배포

미리보기 Preview	변경한 결과를 확인하는 화면이나 환경
빌드 Build	소스를 배포용 결과물로 만드는 과정
배포 Deploy	확인한 결과물을 서비스 환경에 반영하기
도메인 Domain	홈페이지 주소에 사용하는 이름

AI에게 이렇게 부탁합니다

“미리보기에서 확인하고 승인 후 배포해 줘.”

“계정·프로젝트·환경·업로드 폴더를 먼저 알려 줘.”

“배포 성공과 실제 공개 주소 확인을 구분해 줘.”

실제 사례 · AI 연수의 PPTX·PDF·공개 파일 확인 ↗

확인: 승인한 주소에서 다른 기기로도 최신 화면이 열리는가?

교육용 예시 · 눌러서 직접 확인 ↗

공개 주소 확인 예시

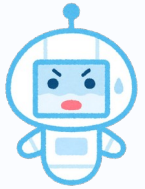
미리보기: 내 작업 환경

빌드: 공개할 결과물

배포: 서비스에 반영

공개: ai.chemion.kr

공개 강의 열기 ↗



오류 수정과 회귀 검수

캐시 Cache	전에 받은 파일을 다시 쓰는 임시 저장
콘솔 Console	코드의 메시지와 오류를 확인하는 도구
회귀 Regression	수정 뒤 기존 기능이 다시 깨지는 현상

AI에게 이렇게 부탁합니다

“기대 결과·실제 결과·발생 단계를 적어서 확인해 줘.”

“주소와 파일부터 비교하고 최소 범위로 고쳐 줘.”

“메뉴·칩·선택 메뉴·모바일이 유지되는지 검수해 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

오류와 수정 결과 비교

오류 상태 | 수정 상태

기준: 문화 목록 1개

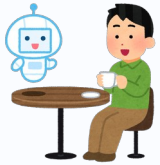
문화 체험

체육 활동

실패: 2개가 보임

실제 사례 · 남해타임의 오류 재시도·입력 보존 ↗

확인: 통과·실패·미확인과 변경 파일을 구분해 보고하는가?



인수인계와 다음 진행

README	파일 역할·실행·배포 방법을 적은 안내 문서
백업 Backup	복구를 위해 따로 보관하는 사본
롤백 Rollback	문제가 생겼을 때 이전 정상 배포로 되돌리기

AI에게 이렇게 부탁합니다

“수정 이력과 실제 되는 기능·시제품 기능을 남겨 줘.”

“백업 위치와 롤백 방법을 확인해 줘.”

“다음 진행 범위와 승인할 일을 정리해 줘.”

교육용 예시 · 눌러서 직접 확인 ↗

인수인계 README 예시

README.md

수정: 프로그램 목록

확인: 칩·선택·모바일

시제품: 신청·예약

백업: 별도 위치 기록

다음: 기관 자료 승인

다음 담당자가 실행할 수 있게 기록

실제 사례 · NAMGO 360의 ZIP 백업·복원 안내 ↗

확인: 다음 담당자가 구조와 미완료 작업을 이해하는가?